# Machine Learning Empowered Resource Allocation for NOMA Enabled IoT Networks

by

Abdullah Saad Alajmi

A thesis submitted to Queen Mary University of London for the degree of Doctor of Philosophy

School of Electronic Engineering and Computer Science
Queen Mary University of London
United Kingdom

November 2023

# Declaration

I, Abdullah Saad Alajmi, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that Queen Mary University of London has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

The copyright of this thesis rests with the author and no quotation from it or information derived from it may be published without the prior written consent of the author.

Signature: Abdullah Saad Alajmi

Date: 16th June 2023

**Details of collaboration and publications:**

- **Journal Papers:**

  1. **A. Alajmi**, M. Fayaz, W. Ahsan, and A. Nallanathan, "Intelligent Resource Allocation in Backscatter-NOMA Networks: A Soft Actor Critic Framework," *IEEE Transactions on Vehicular Technology*, 2023.

  2. **A. Alajmi**, M. Fayaz, W. Ahsan, and A. Nallanathan, "Semi-Centralized Optimization for Energy Efficiency in IoT Networks with NOMA," *IEEE Wireless Communications Letters*, 2022.

- **Conference Papers:**

  1. **A. Alajmi**, M. Fayaz, W. Ahsan, and A. Nallanathan, "Soft Actor Critic Framework for Resource Allocation in Backscatter-NOMA Networks," *IEEE Latin-American Conference on Communications (LATINCOM)*, 2022.

  2. **A. Alajmi**, and W. Ahsan, "An Efficient Actor Critic DRL Framework for Resource Allocation in Multi-cell Downlink NOMA," *Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, p-77–82, 2022.

**To My Family**

# Acknowledgments

First and foremost I must thank ALLAH, The Most Beneficent and The Most Merciful, for giving me the ability, strength and determination to do a PhD at this prestigious institution. Also, I would like express my sincere gratitude to my king, the custodian of the two holy mosques king Salman bin Abdulaziz and crown prince, prince Mohammed bin Salman and the government of Saudi Arabia for supporting me and my family during my PhD. and during the global pandemic of coronavirus.

I would like to express my sincere gratitude to my supervisor, Prof. Arumugam Nallanathan for his continuous support during my PhD research. Also, I would like to thanks my second supervisor Dr. Yuanwei Liu for all the helps and support during my PhD research. I am grateful to have had the privilege of being their student over the past few years. Amongst the many others, who have helped me throughout this journey and have generously given me a significant amount of their time and advice, I would like to thank Dr. Wenqiang Yi, Dr. Muhammad Fayaz, Dr. Abdulrahman Ghandoura, Dr. Waleed Ahsan, Mr. Edward Hoskins and other friends of the QMUL. I would also like to thank all my collegues from Prince Sattam Bin Abdulaziz University especially Dr. Nasser Al-Kahtani, Prof. Khalid Alotaibi, Eng. Yazeed Almutairi, Dr. Mahmoud Sakhawi, Dr. Faouzi Maddouri, and other friends of the PSAU, who were always available to support me. I extend my heartfelt gratitude to Mr. Bandar AlDabaan for his unwavering support and belief in me, which helped me start my higher educational journey.

Last and most importantly, I would like to express my deepest gratitude to my beloved family, especially my beloved mother Hissah Almania, brothers Hesham and Khalid my sisters Sheikha, Modhi, and Reem who always unconditionally support me at any time. I am blessed and forever grateful to my beloved wife Dr. Manal Aljuhani and my loving son Abdulaziz, who bring immense joy and unwavering support into my life. Finally, thanks

to all my relatives and friends, who always pray for my success. I want to dedicated this thesis to my deeply loved father and grandmother Sara Alsurim, and may their souls rest in peace.

# Abstract

The Internet of things (IoT) is one of the main use cases of ultra massive machine type communications (umMTC), which aims to connect large-scale short packet sensors or devices in sixth-generation (6G) systems. This rapid increase in connected devices requires efficient utilization of limited spectrum resources. To this end, non-orthogonal multiple access (NOMA) is considered a promising solution due to its potential for massive connectivity over the same time/frequency resource block (RB). The IoT users' have the characteristics of different features such as sporadic transmission, high battery life cycle, minimum data rate requirements, and different QoS requirements. Therefore, keeping in view these characteristics, it is necessary for IoT networks with NOMA to allocate resources more appropriately and efficiently. Moreover, due to the absence of 1) learning capabilities, 2) scalability, 3) low complexity, and 4) long-term resource optimization, conventional optimization approaches are not suitable for IoT networks with time-varying communication channels and dynamic network access. This thesis provides machine learning (ML) based resource allocation methods to optimize the long-term resources for IoT users according to their characteristics and dynamic environment.

First, we design a tractable framework based on model-free reinforcement learning (RL) for downlink NOMA IoT networks to allocate resources dynamically. More specifically, we use actor critic deep reinforcement learning (ACDRL) to improve the sum rate of IoT users. This model can optimize the resource allocation for different users in a dynamic and multi-cell scenario. The state space in the proposed framework is based on the three-dimensional association among multiple IoT users, multiple base stations (BSs), and multiple sub-channels. In order to find the optimal resources solution for the maximization of sum rate problem in network and explore the dynamic environment better, this work utilizes the instantaneous data rate as a reward. The proposed ACDRL algorithm is scalable and handles different network loads. The proposed ACDRL-D and

ACDRL-C algorithms outperform DRL and RL in terms of convergence speed and data rate by 23.5% and 30.3%, respectively. Additionally, the proposed scheme provides better sum rate as compare to orthogonal multiple access (OMA).

Second, similar to sum rate maximization problem, energy efficiency (EE) is a key problem, especially for applications where battery replacement is costly or difficult to replace. For example, the sensors with different QoS requirements are deployed in radioactive areas, hidden in walls, and in pressurized pipes. Therefore, for such scenarios, energy cooperation schemes are required. To maximize the EE of different IoT users, i.e., grant-free (GF) and grant-based (GB) in the network with uplink NOMA, we propose an RL based semi-centralized optimization framework. In particular, this work applied proximal policy optimization (PPO) algorithm for GB users and to optimize the EE for GF users, a multi-agent deep Q-network where used with the aid of a relay node. Numerical results demonstrate that the suggested algorithm increases the EE of GB users compared to random and fixed power allocations methods. Moreover, results shows superiority in the EE of GF users over the benchmark scheme (convex optimization). Furthermore, we show that the increase in the number of GB users has a strong correlation with the EE of both types of users.

Third, we develop an efficient model-free backscatter communication (BAC) approach with simultaneously downlink and uplink NOMA system to jointly optimize the transmit power of downlink IoT users and the reflection coefficient of uplink backscatter devices using a reinforcement learning algorithm, namely, soft actor critic (SAC). With the advantage of entropy regularization, the SAC agent learns to explore and exploit the dynamic BAC-NOMA network efficiently. Numerical results unveil the superiority of the proposed algorithm over the conventional optimization approach in terms of the average sum rate of uplink backscatter devices. We show that the network with multiple downlink users obtained a higher reward for a large number of iterations. Moreover, the proposed algorithm outperforms the benchmark scheme and BAC with OMA in terms of sum rate, self-interference coefficients, noise levels, QoS requirements, and cell radii.

# Table of Contents

# List of Figures

# List of Tables

# List of Abbreviations

**1G** first generation.

**2G** second generation.

**3G** third generation partnership project.

**4G** fourth generation.

**5G** fifth generation.

**6G** Sixth Generation.

**ACDRL** actor critic deep reinforcement learning.

**ACDRL-C** actor-critic deep reinforcement learning - continuous action.

**ACDRL-D** actor-critic deep reinforcement learning - discrete action.

**Adam** adaptive moment estimation optimizer.

**AI** artificial intelligence.

**BAC** backscatter communication.

**BPCU** bit per channel use.

**BS** base station.

**CC** computational complexity.

**CDMA** code division multiple access.

**CH** cluster head.

**CSI** channel state information.

**DDPG** deep deterministic policy gradien.

**DL** deep learning.

**DNN** deep neural network.

**DQN** deep Q-network.

**DRL** deep-RL.

**EE** energy efficiency.

**eMBB** enhanced mobile broadband.

**FCNNs** fully connected neural networks.

**FDBS** full-duplex base station.

**FDMA** frequency division multiple access.

**GA** grant acquisition.

**GB** grant-based.

**GF** grant-free.

**IoT** Internet of Things.

**LSTM** long short term memory.

**M2M** machine-to-machine.

**MA-DQN** multi-agent deep Q-network.

**MA-DRL** multi-agent deep reinforcement learning.

**MAB** multi-armed bandits.

**MDP** Markov decision process.

**MIMO** multiple input multiple output.

**ML** Machine learning.

**mMTC** massive machine type communication.

**MUD** multi user detection.

**NN** neural network.

**NOMA** non-orthogonal multiple access.

**NP** non-deterministic polynomial.

**OFDMA** orthogonal frequency division multiple access.

**OMA** orthogonal multiple access.

**PPO** proximal policy optimization.

**PSS** primary synchronization signal.

**QoS** quality of service.

**RA** random access.

**RAR** random access response.

**RB** resource block.

**ReLU** rectified linear unit.

**RF** radio frequency.

**RIS** reconfigurable intelligent surface.

**RL** reinforcement learning.

**RRC** radio resource control.

**SAC** soft-actor critic.

**SC** superposition coding.

**SGD** stochastic gradient descent.

**SGF** Semi-GF.

**SIC** successive interference cancellation.

**SINR** signal-to-interference-plus-noise ratio.

**SSS** secondary synchronization signal.

**SWIPT** simultaneous wireless information and power transfer.

**TDMA** time division multiple access.

**UAVs** unmanned aerial vehicles'.

**URLLC** ultra-reliable low latency communications.

**WPT** wireless power transfer.

# List of Notations

| Notation | Definition |
|---|---|
| $\mathcal{A}$ | Action space |
| $B$ | Total number of BSs |
| $\hat{B}$ | Bandwidth |
| $\mathcal{D}$ | Replay memory |
| $D_i$ | $i$-th Downlink user |
| $e$ | Input layer size |
| $\mathbb{E}$ | Expectation |
| $\hat{E}$ | Real-time computational complexity |
| $g_{U_k}$ | Channel gain between $D_i$ and $U_k$ |
| $h_{D_i}$ | Channel gain (BS to $D_i$) |
| $h_{i_{b,j}}$ | Channel gain for $i_{b,j}$ |
| $h_{SI}$ | Self-interference channel |
| $h_{U_k}$ | Channel gain (BS to $U_k$) |
| $I$ | Total number of IoT users |
| $i_{b,j}$ | $i$-th user connected to the $b$-th BS via $j$-th sub-channel |
| $I_d$ | Interference from other downlink users |
| $I_u$ | Signal reflected by uplink backscatter devices |

| | |
|---|---|
| $J$ | Total number of sub-channels |
| $\mathcal{L}$ | Loss |
| $L$ | Layers |
| $n_{BS}$ | Noise |
| $n_{D_0}$ | Noise |
| $N_e$ | Number of episodes |
| $\mathcal{N}$ | Noise |
| $\mathcal{P}$ | Probabilities |
| $P$ | Transmission power |
| $P_{D_i}$ | Power of downlink user $D_i$ |
| $q_{i_{b,j}}$ | The connectivity for $i_{b,j}$ |
| $\mathcal{R}$ | Reward function |
| $R_{D_i}$ | Data rate for $i$-th downlink user |
| $\hat{R}_{D_i}$ | Target data rate for $i$-th downlink user |
| $R_{i_{b,j}}$ | Data rate for $i_{b,j}$ |
| $R_{\text{sum}}$ | Sum rate (uplink backscatter devices) |
| $\mathcal{S}$ | State space |
| $SINR_{D_i}$ | SINR for the $D_i$-th downlink user |
| $SINR_{i_{b,j}}$ | Signal interference noise ratio for $i_{b,j}$ |
| $T$ | Network time slot |
| $T_e$ | Number of trials |
| $U_k$ | $k$-th Uplink backscatter device |

| | |
|---|---|
| $x_{D_i}$ | Downlink user $D_i$ signal |
| $x_{SI}$ | Self-interference |
| $x_{U_k}$ | Uplink backscatter device signal |
| $\hat{x}_l$ | Neurons in layer $l$ |
| $y_{BS}$ | Signal received by BS |
| $y_D$ | Signal received by downlink user |
| $\hat{\tau}$ | Soft updating parameter |
| $\alpha$ | Learning rate |
| $\gamma$ | Discount factor |
| $\pi$ | Policy pi |
| $\varphi$ | Self-interference coefficient |
| $\sigma^2$ | Noise |
| $\eta_{U_k}$ | BAC reflection coefficient |

# Chapter 1

# Introduction

## 1.1 On the Way to 6G and Beyond

Massive Internet of Things (IoT) is expected to connect billions of devices simultaneously to enable industries or individuals to achieve their full potential. The rise of new applications, such as smart health care systems, self driving, and intelligent home systems, are innovated through massive IoT. In these applications, the massive IoT devices or users connectivity is fully automated without any human involvement [1]. The characteristics of cellular system generations are illustrated in Fig 1.1 to show the improvement of the generations of cellular system through different years where the users capability and services increased. However, the fourth generation (4G) mobile communication system has increasingly struggled to keep up with human expectations due to the growth of the mobile internet, proliferating smart terminals, and massive IoT. In addition to these requirements, the need for higher throughput, wireless cellular technology has always evolved. Data rates have steadily increased from tens of kbit/s to tens of Mbit/s as the world moved from second generation (2G) systems to 4G systems [2, 3]. One key attribute that differentiates the generations of different wireless systems from each other is the multiple access technologies [2, 4]. These are the code division multiple access (CDMA), time division multiple access (TDMA), frequency division multiple access

(FDMA), and orthogonal frequency division multiple access (OFDMA) for first generation (1G), 2G, third generation partnership project (3G), and 4G wireless communication systems, respectively [5, 6]. Moreover, the main focused areas of fifth generation (5G) and beyond includes the ultra-reliable low latency communications (URLLC), enhanced mobile broadband (eMBB), and massive machine type communication (mMTC).

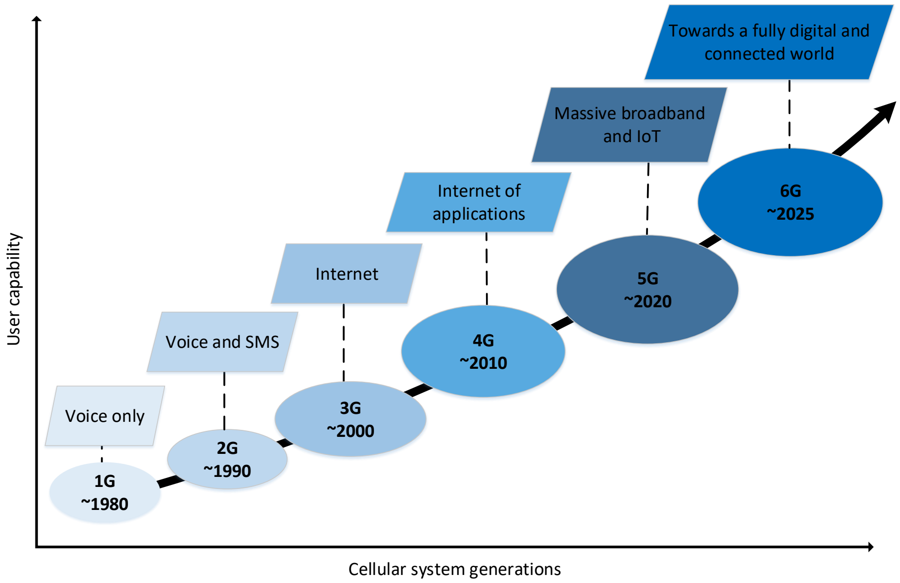


Figure 1.1: The evaluation of cellular technology with corresponding characteristics

The communication requirements for meeting the needs of the industry that URLLC, eMBB, and mMTC help address are briefly summarised as follows: 1) The spectral efficiency is projected to increase by 5 to 15 times compared to 4G, 2) the connectivity density target is 10 times higher than that of 4G, and 3) 5G is expected to meet the demands for low latency (radio latency of $\leq$1 ms), low cost (100 times the cost efficiency of 4G), and the support of various compelling services [7]. Thus, advanced solutions must be created to meet these demanding standards. Compare to 5G cellular system generation, the future of Sixth Generation (6G) technology is expected to be higher achievements such as spectral efficiency and coverage, energy efficiency, Ultra-low latency, and extremely high reliability [8]. Compare to 5G, the latency in 6G is less than 0.1 ms where in 5G is less than 1 ms only. The connectivity of number of devices increased 10 times in 6G where this generation can cover up to 10 million devices/km$^2$. Moreover, the energy efficiency consider on of the important part of improving 6G technology. The achievable energy efficacy improve in the level of the user equipment such as between machines or objects. With strong learning ability by apply artificial intelligence (AI), 6G network can learn and adapt itself where it can support diverse services accordingly without human intervention [9]. Moreover, AI help the wireless network in 6G to improve the reliability within devices connectivity where this network use ultrahigh reliability and ultra low latency communication services. Fig 1.2 shows some of the difference between 5G and 6G in term of capacity, energy efficiency, connectivity, reliability and latency.

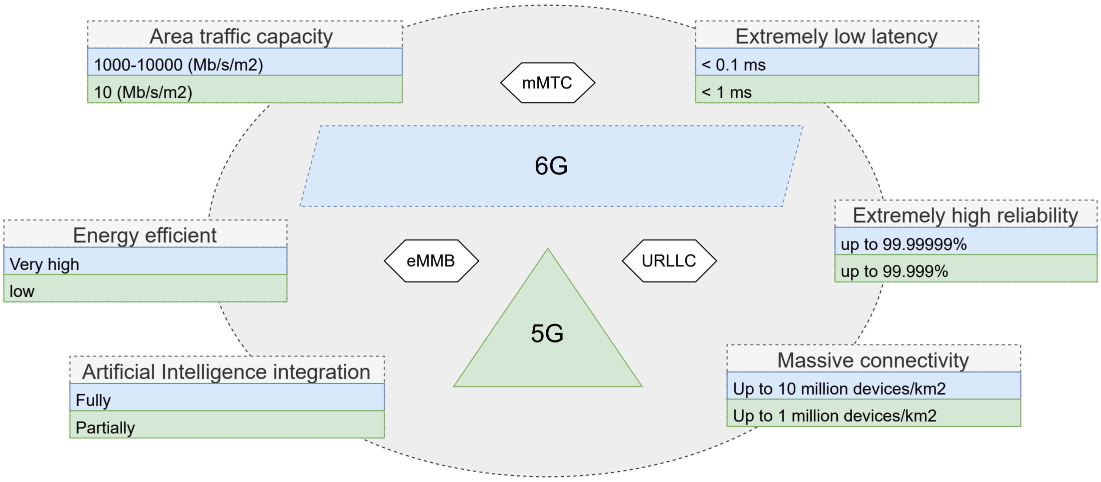


Figure 1.2: 5G and 6G technologies

These multiple access systems fall under the category of orthogonal multiple access (OMA) from the perspective of their design principles, where wireless resources are orthogonally assigned to many users in the time, frequency, and coding domains, or depending on their combinations. This orthogonality has been used suppress cross-user interference [2–4].

By using these approaches, the information signals of users can be separated with low-cost and low-complexity receivers [7]. However, the OMA's availability of orthogonal resources places a cap on the number of supported users. Another issue is that even when orthogonal resources are used in the time, frequency, or code domains, channel-induced impairments almost always degrade their orthogonality. As a result, meeting the extreme spectral efficiency and massive connectivity requirements of 5G remains difficult for OMA.

The power domain non-orthogonal multiple access (NOMA) is considered a potential 5G and beyond multiple access technique with the ability to improve spectrum efficiency and increase connectivity. NOMA can be distinguished into two main categories: code-domain and power-domain. Both approaches have their advantages and disadvantages. Based on the design considerations of our work, we applied power domain which fit the specific requirements of the wireless communication system design of our proposed model.

Therefore, in this thesis, we investigate the resource optimization in power domain NOMA because of several reasons. First, power domain have flexible resource allocation. Based on the channel conditions and quality of service (QoS) of the user, the resource can be dynamically allocated [10]. Also, power domain have better scalability compare to code domain. Large number of users can be applied since power domain relies on adjusting power levels. It can accommodate more users in the resource block with in same time and frequency and it is suitable for massive IoT. Finally, power domain show significant performance gains in terms of spectral efficiency and throughput within different channel conditions [11].

In downlink communication, power domain NOMA multiplexes multiple users signals over a resource block (RB) with the help of superposition coding (SC) at the transmitter, and successive interference cancellation (SIC) is used to separate these signals at the receiver [12]. In uplink power domain NOMA, multiple users transmit their signals over a RB using pre-allocated power levels. These different power levels are used by the base station (BS) for multi user detection (MUD) with the help of SIC. The BS uses the received power difference of multiplex users over a RB and attempts to recover the signals in descending order (based on received power strength). However, to fully exploit the benefits of NOMA, more sophisticated and intelligent algorithms are required to allocate power and sub-channel to uplink/downlink users. Therefore, this thesis explores novel approaches to suggest efficient and effective resource allocation techniques for downlink, uplink and combine downlink and uplink NOMA scenarios to improve system performance.

## 1.2 Artificial Intelligence in Wireless Communication

A minimum use of AI started in 5G. However, the future 6G communication will be fully backed by AI and will achieve the target requirements intelligently. With the help of Machine learning (ML), 6G wireless communication has been significantly improved, especially when it comes to sensing, data mining, and predictive analysis.

Current wireless communication is heavily dependent on mathematical models, and these models are being used for system structure. However, most of these models are based on assumptions that may not be accurate, and for some scenarios, there may be no mathematical model to represent the system. Additionally, such methods may not be able to meet the 6G wireless communication requirements. The rapid increase in connected devices produces a huge data set, which is considered as an enabler for ML, as ML methods are heavily based on large data sets. Therefore, ML is considered as a key enabler technology for 6G wireless communications.

Generally, ML categories are based on the training models, which can be regarded as deep learning (DL) or reinforcement learning (RL). For DL, a large data set is required to find patterns or predictions about the data. DL can be further divided into two categories.

- Supervised Learning:
  - Set of labelled samples are needed to learn or predict a mapping between the input and output spaces.
  - Discrete case, such as classification, and continuous case, such as regression.

Table 1.1: Supervised deep learning vs unsupervised deep learning

| | **Training data** | **Discrete case** | **Continuous case** | **Accuracy of results** | **Number of classes** |
|---|---|---|---|---|---|
| **Supervised** | labelled | classification | regression | high accurate | known |
| **Unsupervised** | unlabelled | clustering | dim-reduction | less accurate | not known |

  - High accuracy.
  - Known number of classes.

- Unsupervised Learning:
  - Classify unlabelled data into different clusters.
  - Discrete case, such as clustering, and continuous case, such as dimensional reduction.
  - Low accuracy.
  - Unknown number of classes.

Table 1.1 shows more details about the difference between the two categories. Moreover, unlike DL, RL works without the requirements of pre-available data sets. An autonomous decision-making entity, known as an agent, interacts with the environment and improves its decision-making policy via trial and error. More details about ML is given in Section 2.5.

## 1.3 Motivation and Contributions

The 5G and beyond wireless networks are expected to connect every object and transform it into an information source. The connected objects are known as the IoT and can be characterized by sporadic transmission, minimum data rate, different QoS requirements, and long battery life. Therefore, considering these characteristics, it is important for NOMA-based IoT networks to allocate resources appropriately and more efficiently. Moreover, using convex optimization in wireless communication comes with many impor-

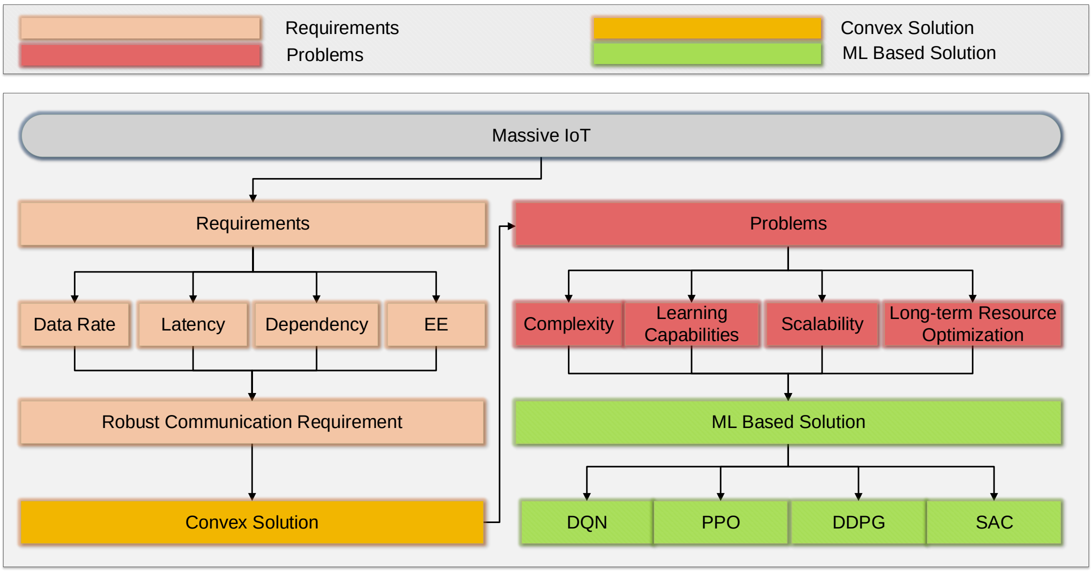


Figure 1.3: Massive IoT requirements, problems, and solutions

tant issues, such as 1) learning capabilities, 2) scalability, 3) increased complexity, and 4) long-term resource optimization (further explained below). Therefore, conventional optimization approaches are not suitable for IoT networks with time-varying communication channels and dynamic network access.

- Learning: Due to the absence of learning ability, the conventional methods for resource optimization problems must be re-run from scratch when there is a small change in the network parameters. Therefore, conventional approaches are not feasible for long-term resource optimization problems.

- Scalability: Scalability is one of the main challenges next-generation cellular networks face. The resource optimization problem in wireless networks is non-deterministic polynomial (NP) hard and combinatorial in nature; therefore, it is mathematically intractable as the size of the network increases.

- Long-term optimization: In wireless communication and network management, the long-term optimization mean the practice of optimizing network parameters, configurations, and policies rather than focusing on short term metrics. By applying long-term optimization, the overall system model will achieve better performance and more efficiency within the wireless networks. For example, long-term optimization can lead to network stability and robustness over time by considering future expected changes within the network conditions. Also, long-term optimization lead to cost saving such as change of the network or upgrades or any reason. Furthermore, sustainability is also another reason why long-term optimization is important since it can help to improve the energy efficiency within the environment.

To provide massive connectivity and fulfill the requirements of IoT users (shown in orange color in Fig. 1.3), such as low data rate, small latency, long-term resource optimization, and energy efficiency (EE), the convex optimization (shown in yellow color in Fig. 1.3) is one solution to fulfill theses requirements. However, convex optimization has some limitations and problems, including complexity, learning capabilities, scalability, and long-term resource optimization (shown in red color in Fig. 1.3). Although ML algorithms are beneficial for wireless communication, they also have several drawbacks. One issue is that they require large, diverse datasets, which can lead to overfitting and increased energy consumption and latency issues. Additionally, wireless environments are dynamic and can challenge the adaptability of ML models. Furthermore, balancing exploration and exploitation and hyperparameters tuning are also challenging. To counter problems with convex optimization, ML is considered as a viable solution. Different ML based algorithms (shown in green color in Fig. 1.3) are available and can be used according to the considered optimization problem. These algorithms are thoroughly explained in Chapter 2.

Resource allocation in 6G networks using convex optimization faces significant challenges owing to the complexity of networks and diverse requirements. As 6G networks are expected to support a vast number of devices with varying bandwidth and latency requirements, scalability is a major concern for convex optimization models [13]. These networks are characterized by a dynamic and heterogeneous environment, encompassing services such as eMBB, URLLC, and mMTC, making the creation of suitable convex optimization models challenging [14]. Energy efficiency is another critical factor that requires a balance between resource allocation and energy consumption owing to the growing number of devices and the demand for high data rates [15]. The high mobility of users results in variable network conditions, and the need for advanced interference management in dense environments adds to the complexity of convex optimization. Moreover, ensuring high QoS and quality of experience (QoE) further complicates optimization models . These issues underline the need for innovative approaches to resource management in order to meet the high demands for speed, reliability, and performance in 6G networks [16]. Therefore, ML is considered a promising alternative to solutions based on convex optimization.

Motivated by the problems with convex optimization and requirements of IoT users, this thesis proposes promising ML-based solutions to enhance network efficiency assisted by NOMA. The main contributions of this thesis are listed below:

- Chapter 3: To dynamically allocate resources in multi-cell downlink NOMA IoT networks and maximize the sum-rate, we designed a tractable framework based on RL. We used actor critic deep reinforcement learning (ACDRL) to particularly optimise the power allocation for various users in a dynamic and multi-cell situation to maximise the sum rate of IoT users. The three-dimensional association between users, sub-channels, and BSs serves as the foundation for the state space in the suggested architecture. This work uses the instantaneous data rate as a reward to find the best response to the sum rate maximisation problem and better explore the dynamic environment. The suggested ACDRL algorithm scales well and can manage various network demands. In terms of the long-term sum rate, the simulation results demonstrate that the suggested solution for a multi-cell network with NOMA is superior to traditional RL, deep-RL (DRL) algorithms, and OMA methods.

- Chapter 4: Due to the exponential growth in the number of connected devices, especially as IoT technology gains widespread adoption, highly efficient energy management is required to ensure network stability and reliability. In addition, the energy efficiency of these networks directly impacts the battery life of mobile devices. Devices that are deployed in remote or inaccessible locations require long-lasting batteries, particularly for 6G networks and beyond. For instance, sensors with various QoS requirements are placed in pressurised pipes, hidden in walls, and placed in radioactive locations. Hence, energy cooperation plans must be made for such circumstances [17]. Therefore, in this chapter, grant-based (GB) and grant-free (GF) IoT users' EE is maximised using a semi-centralized framework for NOMA IoT networks. The EE of GB users is maximised using the proximal policy optimization (PPO) technique, and the resources for GF users are optimised using a multi-agent deep Q-network with the help of a relay node. The suggested framework blends the benefits of centralised and distributed architectures to make up for their drawbacks. The suggested approach improves the EE of GB users compared to the random power allocation and fixed power allocation strategies. Additionally, the numerical results show that GF users' EE is superior to the benchmark scheme. We also demonstrate a significant association between the rise of GB users (relay users) and the EE of GB and GF users.

- Chapter 5: In this chapter, we designed an efficient model-free backscatter communication (BAC) approach to assist the base station with complex resource scheduling tasks (for both uplink and downlink) in dynamic BAC-NOMA IoT networks to enhance the sum rate of uplink backscatter devices. In particular, we jointly optimize the transmit power of downlink IoT users and the reflection coefficient of uplink backscatter devices using a reinforcement learning algorithm, namely soft-actor critic (SAC). The SAC agent learns to explore and exploit the dynamic BAC-NOMA network efficiently due to the advantage of entropy regularization. The suggested approach increases the aggregate rate of uplink backscatter devices while maintaining the QoS needs of downlink users. The suggested algorithm outperforms the standard optimization (benchmark) strategy in terms of the average sum rate of uplink backscatter devices. In terms of the average sum rate with various numbers of backscatter devices, the suggested method performs better than the benchmark system and BAC with orthogonal multiple access. Furthermore, we demonstrate how our suggested technique improves sum rate efficiency with regard to various self-interference coefficients and noise levels. Finally, we compare the proposed algorithm's sum rate efficiency with various QoS criteria and cell radii.

## 1.4 Dissertation Organization

The remainder of the thesis is organised as follows. In Chapter 2, some fundamental concepts are introduced, and related work to this thesis are presented, including the principles of NOMA, downlink and uplink transmission, backscatter communications, GB uplink principles, GF uplink principles, and the principles of ML. Chapter 3 investigates and proposes a novel technique with the effectiveness of actor-critic DRL in resource allocation with multi-cell downlink NOMA and improved the throughput. Chapter 4 proposes also a novel technique in the NOMA network, where we introduce a relay node (GB users) to help the IoT (GF users) to improve the energy efficiency with the help of semi-centralized machine learning design. Chapter 5 proposes a novel backscatter NOMA communication where the SAC framework is adopted to improve users' sum rate. Chapter 6 provides the thesis summary and discusses the future research directions.

# Chapter 2

# Background and Literature Review

This chapter provides a background understanding of wireless communication and machine learning. Part one focuses on the IoT network using NOMA for downlink, uplink, and simultaneous downlink and uplink. The cooperative NOMA and RL design for downlink techniques is discussed in subsection 2.2. Subsection 2.3 describes the design of cooperative NOMA and RL techniques for uplink communication. Section 2 describes backscatter communication using various wireless techniques. Subsection 2.4 describes backscatter communication using various wireless techniques with downlink/uplink OMA and NOMA, while subsection 2.5 focuses on ML in wireless communication.

## 2.1 IoT Networks with NOMA

In 5G, the IoT is anticipated to be one of the largest technological trends in wireless networks. 5G wireless network design is tailored to meet the need of massive IoT devices. Most of the major drivers of 5G and beyond are high data rates, energy efficiency, and massive connectivity for IoT devices with small data. According to authors in [18], the expected number of IoT devices, such as phones, tablets, and sensors, will reach 125 billion in 2030. Thus, an increasing number of IoT devices will lead to massive wireless traffic over the limited spectrum resources. Resource management with OMA will be challenging for a huge number of connected devices. Therefore, NOMA is considered a suitable solution to solve the problem of traffic over the limited spectrum resources in which different users can share the same time/frequency RB at the same time. However, to fully utilize the benefits of NOMA, resource management and optimization is mandatory for IoT networks. NOMA can be used for both downlink and uplink IoT networks, which is explained in the next section.

## 2.2 Downlink NOMA IoT Networks

In downlink NOMA IoT networks, the SC is applied at the BS to multiplex the signal for downlink users on the same RB. The SIC is used on the receiver side to decode and separate its signal from the combined signal. For the decoding order, in most cases where IoT user close to BS, have stronger channel gain use SIC to decode his signal. The user located far from the BS, usually with a weaker channel gain treats the IoT users with higher signals as a noise.

### 2.2.1 Single-cell and multiple-cell Downlink NOMA Networks

NOMA is critical for enhancing spectral efficiency and accommodating a large number of users in 5G and future wireless communication. Single-cell NOMA optimizes resource allocation within the coverage area of a single base station, maximizing spectral efficiency and throughput while effectively managing interference challenges. On the other hand, multi-cell NOMA extends the benefits of NOMA to multiple neighboring cells, addressing inter-cell interference and coordinating resource allocation. However, multi-cell NOMA is more complex due to inter-cell interference and requires efficient signaling and mechanisms for user fairness. In a multiple cell downlink environment (as shown in Fig 2.2, where User 3 is located in the interference area between two BSs. The user receive two signals one from BS 1 and one from BS 2), the composite signal from the b-th BS is denoted as:

$$\begin{aligned} y_{i_{b,j}} =& \underbrace{q_{i_{b,j}} h_{i_{b,j}} \sqrt{P_{i_{b,j}}} x_{i_{b,j}}}_{\text{Desired Signal}} + \underbrace{\sum_{i_{b,j} \neq i'_{b,j}} q_{i'_{b,j}} h_{i'_{b,j}} \sqrt{P_{i'_{b,j}}} x_{i'_{b,j}}}_{\text{Intra-Cell Interference}} \\ &+ \underbrace{\sum_{i' \neq i}^{I} \sum_{b' \neq b}^{B} q_{i'_{b',j}} h_{i'_{b',j}} \sqrt{P_{i'_{b',j}}} x_{i'_{b',j}}}_{\text{Inter-Cell Interference}} + \underbrace{n_0}_{\text{Noise}}, \end{aligned} \tag{2.1}$$

where the first part of the equation shows the desired signal for the $i$-th user connected to $b$-th BS via the $j$-th sub-channel, and the rest of the equation (intra-cell interference, inter-cell interference, and noise) is considered as interference from other users inside the same cell or an interference from outside the cell. Therefor, $q_{i,j}$ denote the $i$-th user connectivity via sub-channel $j$, $h_{i,j}$ channel gain for the $i$-th user connected to the $j$-th sub-channel, $P_{i,j}$ the received power for the BS to the $i$-th user connected to the $j$-th sub-channel, $x_{i,j}$ the $i$-th user information connected to $j$-th sub-channel, and $n_0$ is the noise. The decoding order is based on the statistical channel state information (CSI), where users with strong channel conditions are decoded first, and the last user to decode has the weakest channel condition [19].

The SINR for the $i$-th user connected to the $b$-th BS via the $j$-th sub-channel can be expressed as:

$$SINR_{i_{b,j}} = \frac{q_{i_{b,j}} P_{i_{b,j}} |h_{i_{b,j}}|^2}{\sum_{i'_{b,j} \neq i_{b,j}} q_{i'_{b,j}} P_{i'_{b,j}} |h_{i'_{b,j}}|^2 + \sum_{i' \neq i}^{I} \sum_{b' \neq b} q_{i'_{b',j}} P_{i'_{b',j}} |h_{i'_{b',j}}|^2 + n_{i_{b,j}}^2}. \tag{2.2}$$

Finally, each IoT user calculates their data rate [19], which is shown in the following equation.

$$R_{i_{b,j}} = \hat{B} \log_2(1 + SINR_{i_{b,j}}). \tag{2.3}$$

Table 2.1: Comparison of single cell and multi-cell NOMA network

| **Aspect** | **Single cell** | **Multiple cells** |
|---|---|---|
| Configuration of the network | A single cell operator | Encompasses multiple cells |
| Managing Interference | Limited interference | Higher potential for interference from other cells |
| Resource Allocation | Resources allocated within a single cell | Resources coordinated and allocated across multiple cell |
| Coverage Area | Limited to only one cell area | Extended coverage all around multiple cells |
| Spectral Efficiency | Low spectral efficiency due to limited resources | High spectral efficiency due resource sharing |
| QoS Flexibility | Limited flexible in meeting QoS requirements | Enhanced flexibility in QoS needs |
| Complexity | Low complexity | High complexity due to inter-cell interference |

Both approaches leverage advanced signal processing and machine learning techniques, and the choice between them depends on network requirements and deployment scenarios.

The work on single-cell NOMA is presented in [20–28], while the research on multi-cell NOMA is given in [19, 29–31]. Table 2.1 illustrates a comparison of single-cell and multiple-cell networks from different aspects.

The work mentioned in [20] investigated the user pairing problem and showed that the proposed scheme enhances the sum rate and individual rates compared to the OMA system. The authors in [21] optimized the resource management in two stages. In the first stage, they grouped the IoT users into different clusters, and then, power levels are allocated to these users in the second stage. The authors in [22] improved both capacity and cell-edge users' throughput performance with different channel quality indicators at the BS side. The proposed method improved the data rate for users in the cell-edge area. In [23], the authors investigated the outage performance in terms of QoS of single-cell downlink NOMA, which depends on choices of the users' targeted data rate and allocated

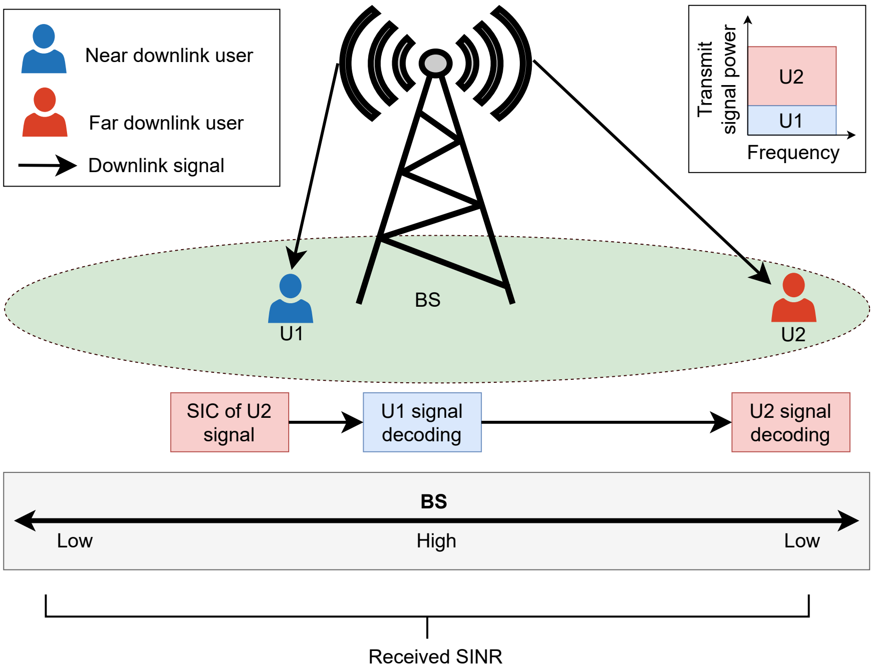


Figure 2.1: Downlink NOMA network with single-cell

power. The fairness of power allocation for downlink users in terms of instantaneous CSI from the BS is studied in [24]. Furthermore, the same study focused on averaging the CSI, which lead to low-complexity. The authors in [25] used the simultaneous wireless information and power transfer (SWIPT) technique, where a near user to the BS can act as energy harvesting to help far NOMA users to gain a better data rate to ensure the QoS requirements. Moreover, the authors in [26] applied Q-learning based on smart antennas to reduce cost, reduce complexity, minimise signal-to-interference-plus-noise ratio (SINR), and enhance the overall sum rate. The authors in [27] used an advance DRL algorithm, namely the actor critic algorithm, to optimize the power allocation coefficient in a single-cell downlink NOMA system. Finally, to maximize the weighted-sum throughput, a joint resource allocation scheme NOMA was proposed in [28]. The simulation outcomes showed that the proposed intelligent scheme is more efficient than benchmark schemes in terms of throughput and suppress interference, especially in a multi-user setting.

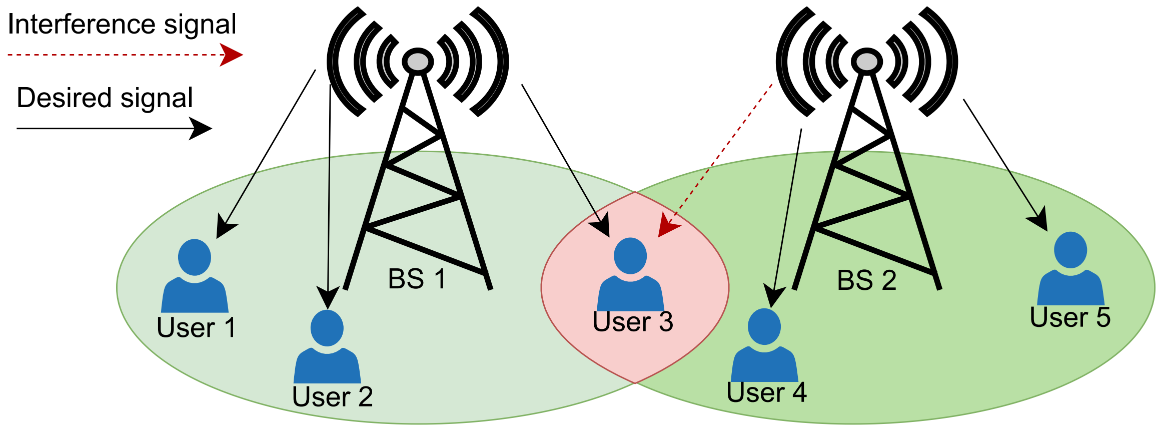


Figure 2.2: Multi-cell downlink NOMA IoT network

Fig 2.2 illustrates a multiple cell downlink NOMA network environment, where there is multiple BSs and multiple IoT users connectivity. User 3 experiencing interference from BS 2. Considering a multi-cell scenario, in [19], the authors enhanced the data rate by optimizing the power of downlink multi-cell NOMA networks. By grouping IoT users into different clusters, different power levels are allocated to different clusters. In [29], the authors improved the spectral efficiency in multi-cell downlink NOMA by deriving expressions for the transmission rate of the strongest IoT user (close to the BS) by using a coordinated superposition coding scheme and getting a better data rate for the IoT users in the cell-edge. The work in [30] evaluate the achievable data rate and outage probability in downlink NOMA system. Based on these conditions (achievable data rate and outage), two different methods were applied to improve NOMA system. To solve the data set problem in wireless communication, RL algorithms were proposed and made positive strides in wireless communication. RL algorithms address the resource allocation problem in the NOMA network, which fulfils the dynamic requirements with different entities. In [31], the authors proposed DRL to improve the sum rate of both orthogonal and non-orthogonal multiple access. The aforementioned research work is summarized in table 2.2.

Table 2.2: Summary of work done on downlink NOMA with single and multiple cell NOMA networks

| **Ref.** | **Objective** | **Solution approach** | **Category** |
|---|---|---|---|
| [20] | Compare NOMA with TDMA statistically | Higher sum rate and individual rates compared to the OMA system | Single cell |
| [21] | Optimize resource management | Proposed two stage method, first grouped IoT into different clusters, second power levels these groups. Reduce system complexity | Single cell |
| [22] | Enhanced the capacity and cell-edge user throughput performance | Investigated NOMA communication baseline receiver scheme for robust multiple access | Single cell |
| [23] | Investigated the outage performance in NOMA communication | The user QoS depends on chose of targeted data rate and allocated power | Single cell |
| [24] | For fairness of power allocation in CSI | Investigated the instantaneous CSI from BS that ensure fairness to the users and average CSI | Single cell |
| [25] | Resource allocation design NOMA with simultaneous wireless information and power transfer | Near NOMA users with high power act as energy harvesting relays to help far NOMA users | Single cell |
| [26–28] | Optimize resource allocation using ML algorithms | Enhanced the overall sum rate, reduce the system complexity, and minimise SINR with different ML algorithms | Single cell |
| [19] | Improved NOMA system by dividing users into different BSs' clusters | enhance the sum rate and outage probability | Multiple cell |
| [29] | Improve the spectral efficiency | Deriving expressions for the transmission rate of the strongest IoT by using SC scheme and getting better data rate for IoT in the cell-edge | multiple cell |
| [30] | Evaluate the achievable data rate and the outage probability | Two different methods were applied to improve NOMA system | multiple cell |
| [31] | Enhanced NOMA system using DRL algorithm | Improved the sum rate and reduce the complexity | multiple cell |

## 2.3 Uplink NOMA IoT Networks

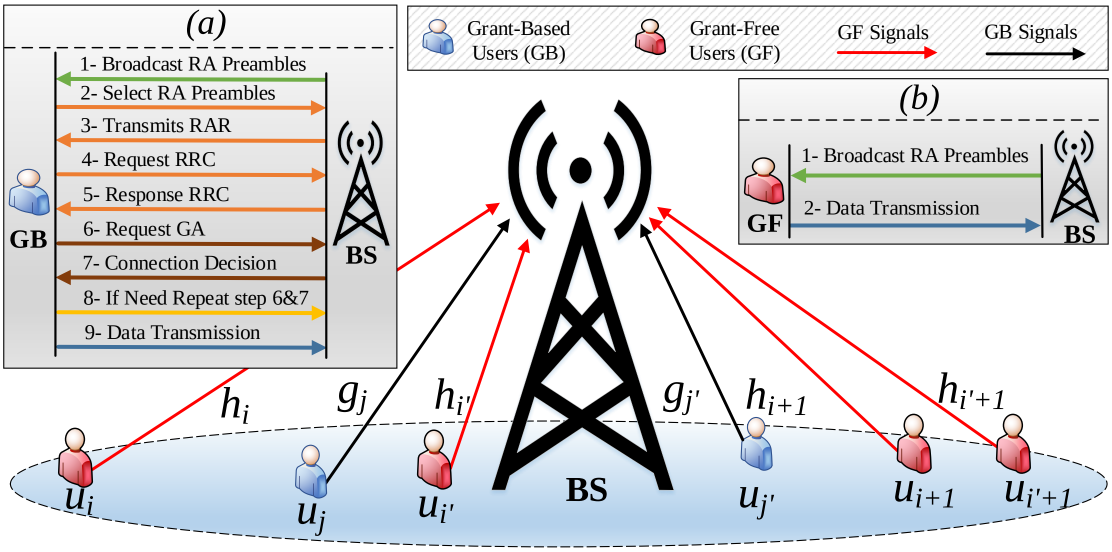


Figure 2.3: An illustration of single-cell uplink IoT NOMA network

In uplink NOMA transmission, the BS receives a combined signal from multiple IoT users in same RB, as shown in Fig. 2.3. The composite signal received at the BS from multiple users can be expressed as

$$y = \sum_{i=1}^{I} \sum_{j=1}^{J} h_{i,j} \sqrt{P_{i,j}} x_{i,j} + n_0, \tag{2.4}$$

where $x_{i,j}$, $h_{i,j}$, and $P_{i,j}$ denote the transmitted signal, channel gain, and transmit power of the $i$-th user on sub-channel $j$, respectively. Here, $n_0$ represents the additive Gaussian noise with variance $(0, \sigma^2)$. The channel decoding order is $P_{i,j}h_{i,j} \geq \cdots \geq P_{I,j}h_{I,j}$. The SINR for user $i \in I$ can be given as follows:

$$SINR_{i,j} = \frac{P_{i,j}|h_{i,j}|^2}{\sum_{i' \neq i}^{I} P_{i',j}|h_{i',j}|^2 + \sigma^2}. \tag{2.5}$$

The data rate of each user is calculated as follows:

$$R_{i,j} = \hat{B} \log_2 \left(1 + SINR_{i,j}\right). \tag{2.6}$$

There are two primary types of users in wireless communication systems: GB and GF users. The GB protocol requires users to obtain permission from the network or base station before transmitting data, which increases signaling overhead but often results in better QoS [32]. A GF user, on the other hand, can transmit data without a prior grant, resulting in lower signaling overhead, but also potentially variable and less predictable QoS. A resource allocation method between GB or GF is determined by the system requirements and the trade-offs between signaling overheads, quality of service, and complexity of the system [33]. Resource allocation is more controlled for GB users, while signaling is reduced for GF users.

### 2.3.1 GB and GF Uplink NOMA

In GB NOMA transmissions, the communication between the sender (uplink NOMA IoT users) and receiver (BS) is based on several handshakes. As shown in the top left sub-figure of 2.3(a), the handshaking steps are as follows: 1) An available random access (random access (RA)) preambles signal is broadcast from the BS to the user. 2) The GB user updates the chosen random access preambles and identifies the occupied channels. 3) random access response (RAR) are sent from the BS, which include several information, such as resource allocation, data rates, and synchronization messages. These messages contain the primary synchronization signal (PSS) and secondary synchronization signal (SSS). 4) The GB user transmits the radio resource control (RRC). 5) The BS arranges target resource blocks. Some time in this phase collision scenario happens, and user needs to wait and retry sending to the channel. If there are no collisions, the GB user occupies the allocated channel, and a connection request well be sent. 6) With the permission accepted from the BS, the GB user transmits the data to the BS. In the top left sub-figure of 2.3(a), the first five steps in the connection between the BS and GB users are combined as the resource allocation process. Steps six to eight are considered the grant acquisition (GA) process. In step nine, the GB user transmits the data to the BS [34].

In GF transmission, the users access the channel in an arrive-and-go manner, that is, the user directly transmits its data without any prior handshake with the BS, as shown in the top right sub-figure of 2.3 (b). The removal of the handshaking process reduces the latency, but, it leads to frequent collisions. The GF transmission is suitable for IoT networks, which need low latency and high energy efficiency [34].

#### 2.3.1.1 Related Work on GB and GF NOMA IoT Networks

Different dynamic types of users in an uplink NOMA network applied based on specific problems or requirements of the environment. Start with the aspects of both GB and GF users, in the mater of resource allocation, the BS controls and schedules resources for GB users, while GF users transmit without BS control. GB users suffer from high overhead signals due to multiple handshakes, while GF users experience low overhead signals due to their technique. With regards to the interference aspect, GF is uncontrolled since GF simultaneously do transmission but GB users schedule their transmission based on the handshake. Moreover, GB users have high QoS control, which increases the complexity. GF users have limited QoS control but with low complexity. In terms of EE, GF users have a higher energy efficiency due to the handshake mechanism compared to GB users. More details about the characteristics of both GB and GF users are shown in Table 2.3.

Table 2.3: Comparison of GB and GF uplink NOMA networks

| **Aspect** | **GB user** | **GF user** |
|---|---|---|
| Resource Allocation | BS control and scheduled the resource | GF user transmit without BS control |
| User Scheduling | Based on multiple handshake | GF user transmit with out handshake |
| Overhead Signaling | Due to multiple handshake, high overhead signals | Low overhead signals due to GF user transmit without handshake |
| Interference | scheduled transmission with controlled interference | Uncontrolled interference because of simultaneously transmission |
| QoS | High QoS control | Limited QoS control |
| Complexity | High complexity | Low complexity |
| Energy Efficiency | Efficient resource usage but with overhead | High energy efficient due to handshake mechanism |

Moreover, by categorizing the type of users (GB and GF), there are a few challenging features and solutions. All GB solution approaches are shown in [35–45] , while the solution approaches with GF users are shown in [46–48] .

A GB-NOMA design was proposed by [35] with performance gains over the OMA scheme in terms of spectral efficiency and fairness. To improve the uplink NOMA system, the same authors in [35] applied multiple user detection in their system model, which enhanced the fairness and spectral efficiency [36]. To reduce the implementation complexity, the authors in [37] proposed a user-pairing policy based on the optimal scheme. This policy was applied to multiple uplink NOMA IoT users.

Moreover, different novel techniques such as power control strategy, were applied in

[38]. This technique investigated the delay-limited sum rate and outage probability. In [39], the authors applied an ML technique to solve the clustering and resource allocation problem for NOMA systems. Moreover, the authors in [40] applied DRL to maximize the computation rate in the multi-access edge area. In [41], the authors improved machine-to-machine (M2M) communications in energy efficiency and considered the QoS using DRL. Furthermore, the authors in [42] applied an ML method based on Q-learning (RL technique) to solve resource allocation problems for the NOMA network based on machine-type communication systems. It is shown in the results section that the proposed schemes are more effective than conventional methods.

To overcome some other problems, such as the maximizing long-term sum energy efficiency, a model-free technique was introduced in [43]. According to [44], the interplay between NOMA and learning-based intelligent algorithms is desirable for the dynamic performance enhancement of NOMA networks. Therefore, on the ML side, a deep deterministic policy gradien (DDPG) strategy recently used an actor-critic approach in which an actor network efficiently samples past memory for an action, and then a critic network maximizes the probability of making the right decision in the action-selection process. The authors in [45] used the DDPG algorithm to reduce the energy consumption and reduce the system computation cost.

Several works investigating GF transmissions are given in [46–50]. In [46], a location-oriented transmit power pool was designed for GF users to reduce the complexity by reducing the information exchanges with the BS. The GF IoT users chose their transmit power from the designed power pool solely according to their communication distances. The authors in [47] improved the GF throughput by reducing the collisions in the system. The authors used DRL to intelligently allocate resources to the GF users. In [48], the authors successfully reduced the impact of the collision for URLLC with GF users where the system needs to have a high success probability within 1 ms.

To find the relationship between optimal resource allocations and dynamic channel conditions, a deep neural network (DNN) was used in [49]. This technique ensures the quality of service and improves the data rate for NOMA users, but with the drawback of the data set requirement (which are not always available). Analyzing the complex of channel characteristics within NOMA networks, the author in [50] applied the deep learning framework to take advantage of the artificial neural network's long short term memory (LSTM). This enhances the system's reliability and lowers the complexity. However, even with low latency and high energy efficiency features, GF transmission suffers from several challenges, such as frequent collision and inability for multiple user detection [34]. The aforementioned research work is summarized in Table 2.4.

Table 2.4: Summary of work done on uplink GB NOMA and uplink GF NOMA networks

| **Ref.** | **Objective** | **Solution approach** | **Category** |
|---|---|---|---|
| [35, 36] | Enhanced the spectral efficiency and fairness | Using detection mechanism to enhance the fairness and spectral efficiency | GB-NOMA |
| [37] | Reduce complexity | Proposed a user pairing policy based on the optimal scheme | GB-NOMA |
| [38] | Investigated delay limited sum rate and outage probability | novel technique applied where the system can do power control strategy | GB-NOMA |
| [39, 42, 44] | Improve system model using ML algorithm | Enhanced resource allocation by using clustering technique with ML algorithm | GB-NOMA |
| [40] | Improve multi-access edge area using DRL algorithm | Build DRL algorithm to maximize the computation rate within edge area | GB-NOMA |
| [41, 43, 45] | Enhance energy efficiency using DRL | Enhanced machine to machine communications energy efficiency | GB-NOMA |

Table 2.4: Summary of work done on uplink GB NOMA and uplink GF NOMA networks

| **Ref.** | **Objective** | **Solution approach** | **Category** |
|---|---|---|---|
| [46] | Improved GF NOMA system by dividing area to multiple power pool | Enhance the sum rate and reduce the complexity | GF-NOMA |
| [47] | Improved system throughput using DRL algorithm | Reduces signaling overhead and access latency effectively | GF-NOMA |
| [48] | Enhanced URLLC | Reduced impact of collision (high success probability within 1ms) | GF-NOMA |
| [49] | Enhanced resource allocations and dynamic channel conditions using DNN | Find the relationship between the channel conditions and the resource allocation which reduce the implementation complexity | GB-NOMA |
| [50] | Adding LSTM to the uplink-NOMA system | Robust and efficient system | GB-NOMA |
| [51] | Enhanced GF-NOMA system by adding DQN instead k-repetition technique | Maximize the long-term average number of successfully served users | GF-NOMA |

### 2.3.2 Semi-GF Uplink NOMA

Conventional GB schemes offer more data rates than the required. These redundant resources might be used to provide connectivity to GF users in the same RB, which forms the Semi-GF (SGF) NOMA scheme. In particular, both users share the same RB for uplink transmission.

#### 2.3.2.1 Related Work on Semi-Grant-Free NOMA IoT Networks

SGF schemes are given in [34, 52–56]. The first work on SGF NOMA was proposed in [52]. The authors in [52] proposed two SGF methods to limit the admitted GF users to the same RB reserved by GB users. The proposed scheme ensures the QoS of GB users, while GF users transmit with fixed power without considering the channel gain or location of the users. A dynamic power allocation for GB users was proposed in [53] using a conventional optimization approach to enhance the outage performance of GF users without considering the impact of path loss. The work given in [54] assumed a homogeneous distribution of users, and only the first two GF users with the largest channel gain were admitted to ensure the GB users' performance. The scheme presented in [55] used stochastic geometry to analyze the ergodic rate and outage performance while considering a dynamic threshold for admitting GF users. Moreover, multi-agent deep reinforcement learning (MA-DRL) based SGF-NOMA was proposed in [56], where only GF users' transmit power is optimized, and GB users transmit with fixed power. The SIC process used in the above-mentioned schemes has a severe effect on the performance of GB and GF users in terms of EE. Because GB and GF users share the same RB simultaneously, this adds to the complexity and energy consumption of these users. In addition, direct access (to the BS) is the focus of existing work due to its simplicity. However, path loss increases with increasing distance, resulting in low energy efficiency and reduced rates. To overcome the effect of distance-dependent path loss, in the existing work, the source node needs to transmit at higher power [57]. However, IoT users have small processing and limited transmit power capability, which makes it impractical to communicate over a long distance.

## 2.4 Backscatter and Multiple Access Communication for IoT Networks

There are different solutions to energies IoT sensors using wireless communication, starting from the general concept such as energy harvesting transmit systems that prioritize

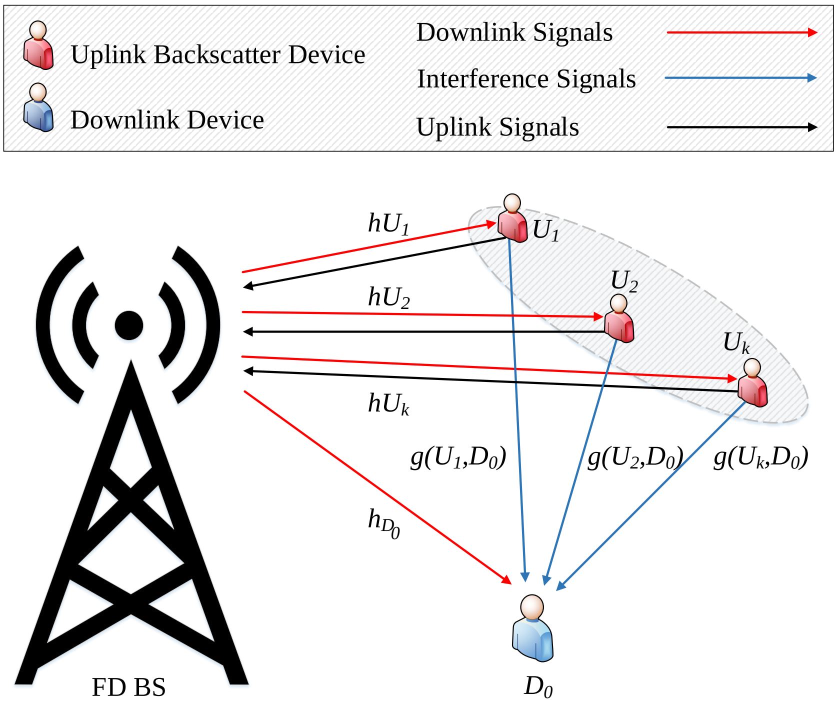


Figure 2.4: An illustration of BAC-NOMA network with downlink IoT and uplink backscatter devices

energy harvesting and autonomy when needed. Also, an energy harvesting transmission system captures energy from the surrounding environment, such as radio frequency (RF) signals, solar energy, and sunlight. Then it stores this energy in a battery. While BAC relays on just the RF signal from the transmitter. This RF can excite the circuit of other IoT devices. Moreover, BAC focuses on low power. By relying on RF, IoT devices receive their energy in a simultaneously way and do not require any batteries. Another technique used to power IoT devices is called wireless power transfer (WPT), which involves transmitting power from the transmitter. The receiver stores this power in a battery. Therefore, the technique is used based on the design requirement.

There are several advantages that determine the use of backscatter, such as low-cost implementation, low power, and reliance on RF signals. However, there are also several technical challenges to overcome, including increased signal interference and collisions (as users rely on reflecting existing signals), low data rates (compared to traditional wireless communication), and the challenge of determining the location of backscatter devices [58–60].

Furthermore, most of the small IoT devices are equipped with small batteries, which are difficult to recharge or replace, such as sensors installed inside a wall [61, 62]. One solution to this problem is WPT or BAC. In WPT, energy is transferred using RF signals [61].

In traditional BAC, one device signal can excite the circuit of other devices [62]. By applying multiple access communication with BAC, many IoT devices work the battery less to improve the energy efficiency of the system. Fig. 2.4 illustrates a communication scenario, where a full-duplex base station (FDBS) sends signals to downlink users and receives signals from uplink backscatter devices simultaneously. The channel gain between BS and all users is denote as $h_{D_0}$ (for downlink) and $h_{U_k}$ (for uplink backscatter device). Moreover, the channel gain between downlink users and uplink backscatter devices is denoted as $g(D_0, U_k)$. The idea behind this communication is to use the downlink signal to excite the circuit of uplink backscatter devices. Different from other techniques, such as power transfer, this technique helps to save energy since part of the power is allocated to different devices without affecting the QoS for the downlink devices.

Backscatter communication operates on a different principle compared to conventional wireless communication. It involves the modulation and reflection of an existing RF signal, which is generated by an external source. Devices that utilize backscatter communication, such as RFID tags, do not generate their own RF signals. Instead, they modify the characteristics of incoming signals (e.g., by altering the phase or amplitude) and relay them back to a receiver. This method consumes far less power than conventional wireless transmission because the backscatter device does not have to produce its own signal [63].

Integrating Backscatter Communication with NOMA is particularly significant in the context of 5G and future wireless networks. NOMA is a crucial technology in 5G that enables multiple users to share the same frequency resources, thereby enhancing the efficiency of spectrum utilization. When combined with the energy efficiency of BAC, it creates new possibilities for massive device connectivity, which is a fundamental aspect of the IoT. The low power requirement of BAC makes it possible to deploy a large number of sensors and devices, which is critical for the IoT paradigm. Furthermore, the cost-effectiveness of BAC devices makes them suitable for widespread implementation, which is a necessary step in order to fully realize the potential of IoT. Although Backscatter Communication has several advantages, its implementation, especially in advanced wireless networks with NOMA, presents several technical challenges. BAC is limited by its relatively short range and low data rate compared with active wireless methods. As a result, long-range communication and high data throughput scenarios pose challenges. Interference management is another challenge in dense network environments, where several backscattering devices coexist with other wireless communications. The detection and decoding of backscattered signals in the presence of this interference is a complex process. To ensure operational efficiency and compatibility, BAC and NOMA must be integrated into existing wireless infrastructure. Finally, developing industry standards for BAC and its integration with technologies like NOMA is necessary to ensure interoperability.

### 2.4.1 Backscatter Communication with OMA

Different studies investigating BAC in orthogonal multiple access are available in the literature. For example, the work in [64] investigates the power allocation problem for cooperative BAC to maximize the system's achievable rate. The authors in [65] provided a closed-form solution for outage probability. The authors in [66] investigated the trade-off between data rate and harvested energy via the power-splitting factor. They also derived a closed-form solution for outage probability over Rayleigh fading channels. In [67], the authors developed a multi-level energy detector and calculated a closed-form expression for the symbol error rate. Meanwhile, the authors in [68] maximized the throughput of BAC-OMA by optimizing the reflection coefficient and showing the trade-off between the sleep and active states. Moreover, in [69], the authors improved the security and reliability of BAC-OMA by calculating the outage and intercept probability of the system. Finally, using backscatter communication with multiple access helps to improve wireless communication. As we start with BAC-OMA, there are some challenges in using BAC-OMA in wireless communication. First, the resource allocation for users is not fairly shared in the network. Moreover, using BAC-OMA, the system model can be costly to scale up to a large number of IoT devices.

### 2.4.2 Backscatter Communication with NOMA

In BAC-NOMA, the downlink users share the same RB with uplink backscatter devices. For example, the downlink user $D_0$ receives the signal from the BS with added interference from the uplink backscatter devices, as the downlink user utilizes the same time slot with the uplink backscatter devices. Equation (2.7) represents the calculated signal $y_{D_0}$ for the downlink user $D_0$

$$y_{D_0} = \underbrace{h_{D_0}\sqrt{P_{D_0}}x_{D_0}}_{\text{Desired Signal}} + \underbrace{\sum_{U_k=1}^{U_K} g_k h_k \sqrt{P_{D_0}\eta_k} x_{D_0} x_{U_k}}_{\text{Intra-Cell } (U_k) \text{ Interference}} + \underbrace{n_{D_0}}_{\text{Noise}} . \tag{2.7}$$

The first part of equation (2.7) is the intended signal for user $D_0$ from the BS, and the second part represents the interference from uplink backscatter devices. The channel gain between the downlink user $D_0$ and BS is denoted as $h_{D_0}$. Moreover, the channel gain between the downlink user $D_0$ and uplink IoT devices $U_k$ is denoted as $g_k$. The noise is denoted by $n_{D_0}$.

The sum rate for uplink backscatter devices that is achievable by BAC-NOMA transmission can be given as:

$$R_{\text{sum}} = \hat{B}\log_2\Big(1 + \frac{\sum_{U_k=1}^{U_K} |h_k|^4 \eta_k P_{D_0} |x_{D_0}|^2}{\varphi P_{D_0}|h_{SI}|^2 + \sigma^2}\Big), \tag{2.8}$$

where, in this system model, we assume that noise for both the BS and downlink user

$D_0$ have the same power; it is denoted as $\sigma^2$. Finally, the data rate for the downlink user is calculated as:

$$R_{D_0} = \hat{B}\log_2\Big(1 + \frac{P_{D_0}|h_{D_0}|^2}{\sum_{U_k=1}^{U_K}|h_k|^2|g_k|^2\eta_k P_{D_0} + \sigma^2}\Big). \tag{2.9}$$

#### 2.4.2.1 Related Work on BAC-NOMA

Recently, NOMA enabled BAC has been investigated in the literature. In [70], a source was equipped with multiple antennae, and a closed-form expression was derived for outage probability. Furthermore, the authors in [71] derived a closed-form expression for ergodic capacity and outage probability in the vehicle to everything network with BAC-NOMA to enhance the sum capacity of the network. Security issues was discussed in [72]. A successful bit rate was maximized by optimizing unmanned aerial vehicles' (UAVs) altitude in [73]. The average successful decoding bits was improved in [63] by optimizing the reflection coefficient selection criteria in BAC-NOMA networks. System minimum throughput was maximized by optimizing the time and reflection coefficient [74]. The outage probability and system throughput were investigated in [75]. The physical layer security of multiple-input single output was studied in [76]. The authors in [77] optimized the transmit power and reflection coefficient to increase the energy efficiency of BAC-NOMA. The reliability and security of BAC-NOMA were investigated in [78]. Finally, to maximize the sum rate of BAC-NOMA with imperfect SIC, the joint power and reflection coefficient optimization problem was investigated in [79].

## 2.5 Artificial Intelligence and Machine Learning for Wireless Communications

Current wireless networks mainly rely on mathematical models to specify the communication system's structure. These mathematical models do not adequately represent the systems. Additionally, some of the structural components of wireless networks and devices do not have mathematical models, making it difficult to represent them. On the

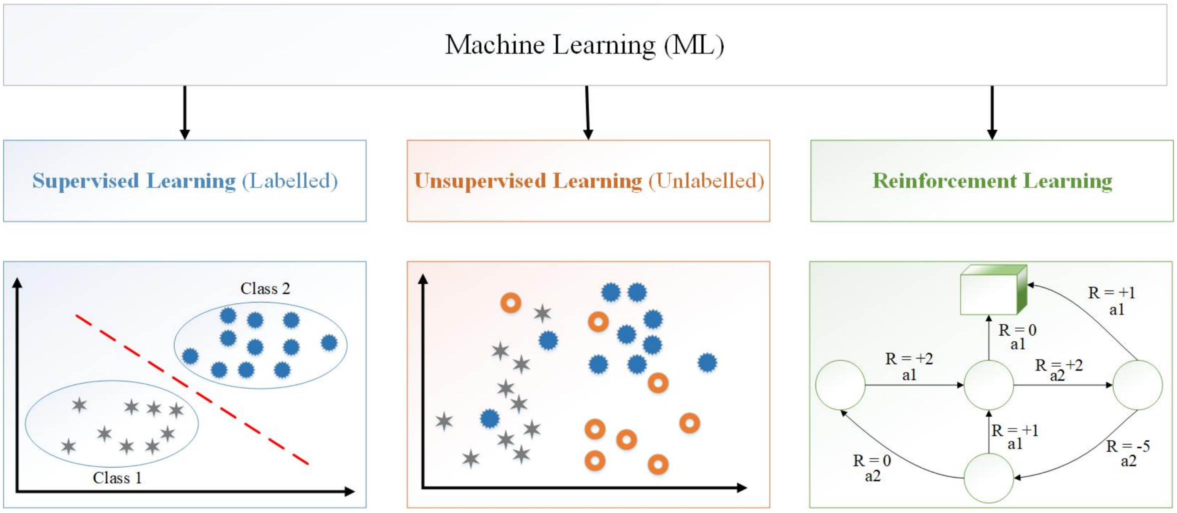


Figure 2.5: An illustration of different ML methods

other hand, the optimization of wireless networks necessitates complicated mathematical solutions that are inefficient in terms of computational complexity and energy efficiency.

Therefore, the existing mathematical model-based solutions are most likely to fall short of the standard requirements set by 5G and beyond applications. Hence, the future 5G and beyond networks will be heavily dependent on AI and ML, as AI and ML can model systems that cannot be represented by mathematical equations [80, 81]. Through AI, human-like behaviour is generally achieved, which enables the machines to make intelligent decisions and achieve specific goals. AI technologies will improve system performance, reliability, and adaptability of communication networks by making real-time robust decisions based on predictions of the networks' and users' behaviours. AI has the potential to minimise manual network development, configuration, and management work and even replace it. ML is considered an application of AI that enables machines to act appropriately by learning from a vast amount of data or by interactions with the environment without being explicitly programmed. ML spans three paradigms [82] (given in Fig. 2.5) discussed in the following sub-sections.

### 2.5.1 Supervised Learning

Supervised learning is an ML-based process that aims to train a model to learn input-to-output mapping functions using a data set with labels [82]. These algorithms are developed for regression and classification problems. Some of the well-developed supervised learning algorithms used in 5G and beyond networks include linear regression, logistic regression, support vector machine, K nearest neighbours, and decision tree. These algorithms should be used in network and physical layers. In the network layer, supervised learning methods can be used for traffic classification, delay mitigation, caching and so on. Moreover, in the physical layer these methods can be used for channel decoding, and channel state estimation etc.

### 2.5.2 Unsupervised Learning

Unsupervised learning is an ML-based method for discovering hidden patterns in unlabeled data sets [82]. Anomaly detection, autoencoders, clustering, and expectation maximisation algorithm are examples of frequently used unsupervised learning methods. Unsupervised learning techniques can be applied to network layer tasks, including parameter prediction, traffic control, and routing, whereas at the physical layer, unsupervised methods can be used for channel-aware feature extraction and optimal modulation [83].

### 2.5.3 Reinforcement Learning

RL is an ML framework to deal with sequential decision-making problems under uncertainty. RL involves self-learning entities, known as agent(s), to maximize long-term system performance by interactions with the RL environment [82], as given in Fig. 2.6. In the RL framework, agents continuously learn the most effective actions to perform in a given state. This learning occurs at discrete time steps, denoted as $(t)$. At each of these steps, the agent observes the current state of the environment, $s^{(t)}$, and based on this information, decides on an action, $a^{(t)}$, to execute. The environment, in turn, evaluates this action and responds with a reward signal, $r^{(t)}$, which serves as feedback to the agent, along with the subsequent state, $s^{(t+1)}$. This reward informs the agent of the

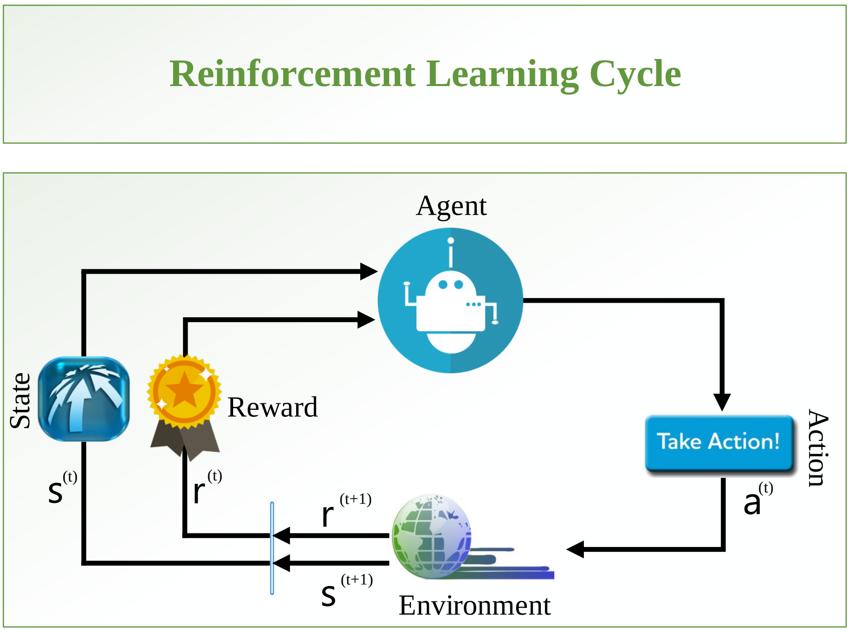


Figure 2.6: An illustration of the reinforcement learning life cycle, where the agent interacts with the environment

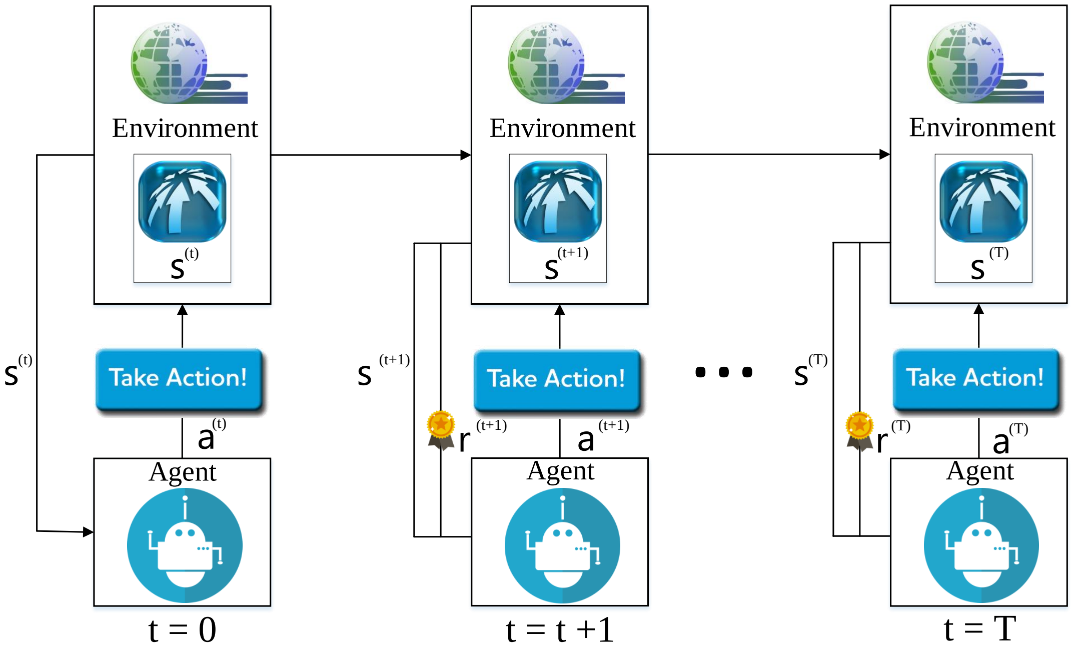


Figure 2.7: The Markov decision process of RL

effectiveness of its actions, guiding it towards strategies that yield the highest reward over time, and ultimately, the best long-term performance.

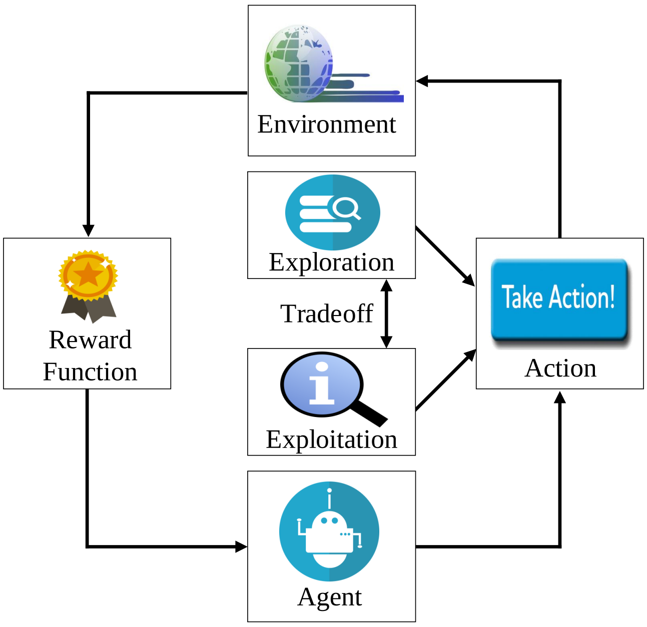


Figure 2.8: Multi-Armed Bandits where the agent takes multi actions to maximise the total reward

#### 2.5.3.1 Fundamentals of RL

To solve the problem using RL, the problem can be mathematically formulated as a Markov decision process (MDP), shown in Fig. 2.7. An MDP consists of a tuple of $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P})$.

- State Space $\mathcal{S}$: This is a finite set of states that an agent can traverse. A state is a piece of relevant information about the environment.
- Action Space $\mathcal{A}$: This is the set of all actions available to the agent.
- Reward $\mathcal{R}$: This is the immediate return (numeric value) after taking action in a given state.
- Transition probability $\mathcal{P}$: The probability of transitioning from the current state to the next state.

Some of the RL algorithms include Q-learning and multi-armed bandits, explained in the next section.

1. Multi-Arm Bandit: A single agent participates in a multi-armed bandits (MAB) model in which each action is followed by a random reward produced by a corresponding distribution with the goal of maximising the total reward. In this model, there is a trade-off between performing the best action right now (exploitation) and learning information to get a bigger payoff later on (exploration). Tuning this parameter is called the temperature [84]. The MAB learning process is given in Fig. 2.8.

2. Q-Learning:

   To solve the formulated MDP using Q-learning, the agent learns Q-values based on the agents' actions. The Q-learning procedure is given in Fig. 2.9(a). At each time step $(t)$ during the learning process, the agent observes the current state and selects the action following its policy $\pi$, upon which it get a reward $r^{(t)}$. Subsequently, the agent moves to the next $s^{(t+1)}$, and the agent recursively uses the policy to take an action given the current state until the maximum sum of rewards are obtained (i.e., to find an optimal $\pi^*$), where a policy can be defined as mapping from state to actions. The goal of an RL agent is to maximize the long-term cumulative discounted reward given below,

   $$r^{(t)} = \sum_{k=0}^{\infty} \gamma^k r^{(t+k+1)}, \tag{2.10}$$

   where $\gamma$ is the discount factor and its value is between zero and one, $k$ represents the number of training episodes, and $t$ is the time step in each episode.

   The classical Q-learning is based on the Q-value function $(Q^{\pi}(s^{(t)}, a^{(t)}))$, which is the expected return in a given learning step, we have

   $$Q^{\pi}(s^{(t)}, a^{(t)}) = \mathbb{E}_{\pi}\left[r^{(t)} | s = s^{(0)}, a = a^{(0)}\right], \tag{2.11}$$

   where values obtained by equation (2.11) are known as Q-values or actions values.

Table 2.5: Q-table [action, state]

| | $a_1$ | $a_2$ |
|---|---|---|
| $s_1$ | $Q^{1,1}(s_1, a_1)$ | $Q^{1,2}(s_1, a_2)$ |
| $s_2$ | $Q^{2,1}(s_2, a_1)$ | $Q^{2,2}(s_2, a_2)$ |
| ... | ... | ... |
| $s_{N_1}$ | $Q^{N1,1}(s_{N_1}, a_1)$ | $Q^{N1,2}(s_{N_1}, a_2)$ |

By solving the formulated MDP, the agent(s) reaches an optimal policy $\pi^*$, which leads the agent to a maximum reward [85]. The associated optimal Q-function $Q^*$ to the optimal policy $\pi^*$ can be given as

$$Q^{\pi^*}(s^{(t)}, a^{(t)}) = \sum_{s^{(t+1)} \in \mathcal{S}} \mathcal{P}^a_{s \to s^{(t+1)}} \big(\mathcal{R}(s^{(t)}, a^{(t)}) + \gamma \max_{a^{(t+1)}} Q^*(s^{(t+1)}, a^{(t+1)})\big). \quad (2.12)$$

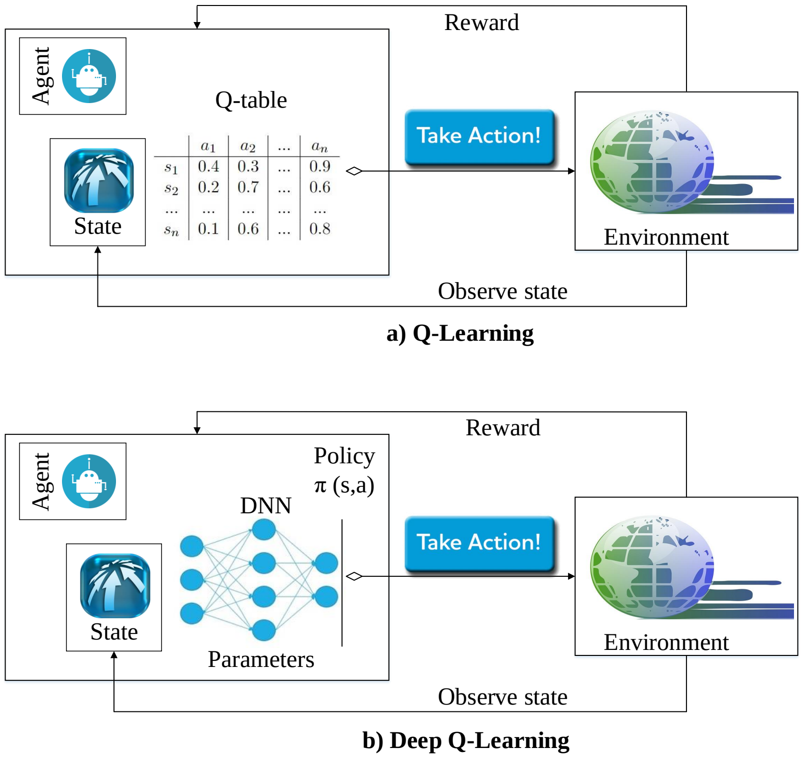
To maximize the reward, the agent takes action based on the following expression:

$$a^{(t)} = \underset{a^{(t)} \in A}{\operatorname{argmax}}\, Q(s^{(t)}, a^{(t)}). \quad (2.13)$$

In classical Q-learning, the agent requires a table known as the Q-table, which includes all state action pairs (given in Table 2.5) from which the agent chooses an action from a given state based on $\epsilon$-greedy policy. The agent takes a random action based on the $\epsilon$ to efficiently explore the environment. However, after sufficient exploration, the agent selects the action in a given state with a maximum Q-value based on 1-$\epsilon$ [86]. After performing action $a^{(t)}$ in a given state $s^{(t)}$, an agent gains a new experience and updates the Q-value in the Q-table.

More specifically, in a training step $(t)$, when an agent performs an action $a^{(t)}$ in a given state $s^{(t)}$, the agent then receives $r^{(t)}$ and goes to the next state $s^{(t+1)}$. Based on this process, the corresponding Q-value can be updated as

$$Q(s^{(t)}, a^{(t)}) \leftarrow r^{(t)} + \gamma \max_{a^{(t)} \in A} Q(s^{(t+1)}, a^{(t)}). \quad (2.14)$$



a) Q-Learning

b) Deep Q-Learning

Figure 2.9: Transfer from Q-learning to deep Q-learning

### 2.5.4 Deep Reinforcement Learning

DRL is the combination of neural network (NN) and RL. In DRL, a NN, known as a deep Q-network, is used to predict the Q-values that enable the agent to learn directly from the data. In particular, the Q-table of Q-learning is replaced (shown in Fig. 2.9) by a replay memory to avoid computational complexity problems. The experiences (state, action, reward, next state) generated during the interaction with the environment are stored in the replay memory and sampled to train the deep Q-network (DQN).

The main components and process of the DQN are given in Fig. 2.10. Conventional Q-learning is expensive for massive IoT scenarios, as the size of the Q-table increases and leads to memory and computational complexity issues. To solve this problem, the authors in [86] introduced DRL, where they replaced the Q-table with a NN. The NN works as a function approximator $Q(s^{(t)}, a^{(t)}; \theta)$ with weights $\theta$. An agent with a DQN uses two NNs, a primary DQN and a target DQN. During the interaction with the

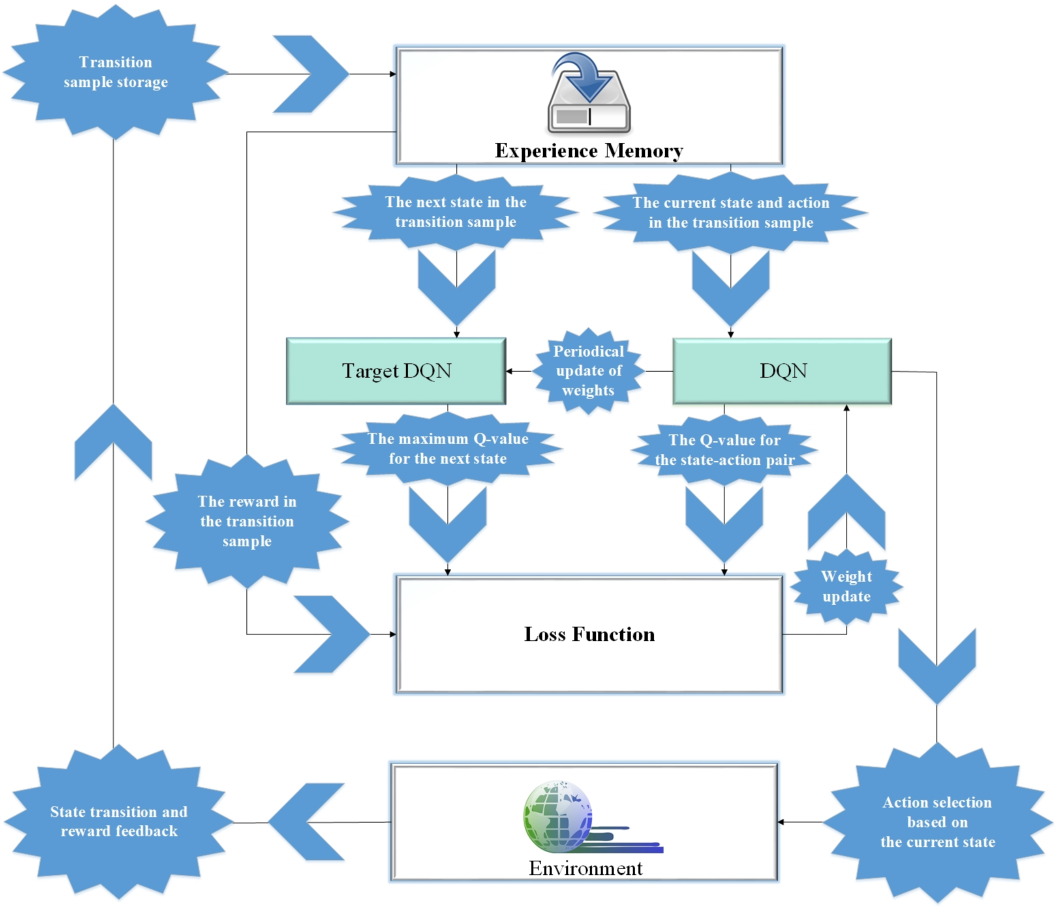

Transition sample storage
Experience Memory
The next state in the transition sample
The current state and action in the transition sample
Target DQN
Periodical update of weights
DQN
The maximum Q-value for the next state
The Q-value for the state-action pair
The reward in the transition sample
Weight update
Loss Function
State transition and reward feedback
Environment
Action selection based on the current state

Figure 2.10: The main components and process of the DQN

environment, the agent forms a tuple of $s^{(t)}$, action $a^{(t)}$, reward $r^{(t)}$, and next state $s^{(t+1)}$ and stores it to its replay memory. To train the DQN and update its weights, the agent samples mini batches randomly from its memory and minimizes the loss between the actual Q-value and target Q-value. The target value produced by the target network can be expressed as

$$y^{(t)} = r^{(t)} + \gamma \operatorname*{argmax}_{a^{(t+1)} \in A} Q(s^{(t+1)}, a^{(t+1)}; \hat{\theta}), \tag{2.15}$$

where $\hat{\theta}$ represents the target Q-network weights. To train the primary Q-network, the loss between the Q-network and target network can be expressed as

$$\mathcal{L}(\theta) = \left(y^{(t)} - Q^{(t)}(s^{(t)}, a^{(t+1)}; \theta)\right)^2. \tag{2.16}$$

### 2.5.5 Proximal Policy Optimization Learning

PPO is considered as one of the simplest in the RL algorithm family. Moreover, PPO is known for minimal hyperparameter tuning; therefore, it is easy to tune the parameter in a wireless network. PPO starts with one agent and is updated with multiple agents in future research [87, 88]. PPO can maintain smooth gradual gradient updates in the neural network, which leads to continuous improvement and avoids unrecoverable crashes in learning. In PPO, the algorithm looks within two different policies: the current policy, which the agent learns, and the baseline policy, which is an earlier version of the policy. The current policy is represented as $\pi_\theta(a^{(t)}|s^{(t)})$, and the old policy established after experience is represented as $\pi_{\theta_i}(a^{(t)}|s^{(t)})$. This early policy is obtained from some previous experience in the past during the training time. Therefore, PPO uses these two different policies and makes a comparison where the agent uses the ratio between them to reach the optimization for better performance. In other words, PPO estimates how good an action made is compared to the average action from state space. The ratio between these policies can be defined as $r^{(t)}(\theta) = \pi_\theta(a^{(t)}|s^{(t)}/\pi_{\theta_i}(a^{(t)}|s^{(t)})$. Furthermore, it estimates a trust region where an agent can safely take reasonable steps in the right direction without falling off the learning cliff. Therefore, the agent's steps depend on whether the step is large or small based on hazards nearby or cliffs. The new objective function where PPO optimize whether the new policy is far from the old policy is represented as

$$\mathcal{L}\theta_i^{CLIP}(\theta) = \mathbb{E}_{\tau\approx\pi_i}\left[\sum_{t=0}^{T}\left[\min\left(r^{(t)}(\theta)\hat{A}^{(t)\pi_i}, \text{clip}\left(r^{(t)}(\theta)1-\epsilon, 1+\epsilon\right)\hat{A}^{(t)\pi_i}\right)\right]\right], \qquad (2.17)$$

where $i = 0$ is the initial policy, and $i = 1$ is the next policy. If the probability ration between the new policy and the old policy is outside the range (which is 1-$\epsilon$ and 1+$\epsilon$), the advantage function will be clipped. Therefore, the agent can take a huge step if the current policy is not different from the old policy. In order to encourages stable and controlled the updated policy while doing the training, the minimum value between the original value and the clipped value is taken into account by the objective function of

PPO. In other word, if the probability ratio between two policy (the old one and the new one) get out side the the range (1-$\epsilon$ and 1+$\epsilon$) the advantage function will be clipped. [87].

### 2.5.6 Actor Critic Deep Reinforcement Learning

DDPG is also one of the RL algorithms family, where deterministic is contrasted with stochastic. This algorithm can handle continuous action space. The DDPG has two different neural networks where the first neural network is called an actor and the second is called a critic. Traditional RL algorithms are based on simple Q-table and epsilon-based simple exploration/exploitation methods (greedy approaches) and are therefore prone to poor policy learning. However, the ACDRL has an exploration/exploitation feature and can handle continuous action space, which further enhances the learning process. Fig 2.11 shows that the actor network takes the state as an input and optimal action as an output. This optimal action is considered the same as the optimal policy, which is $a^*(s) = \underset{a}{\operatorname{argmax}}\ Q(s, a)$. Where the critic network validates and criticises stat-action tuple. This Q-network takes state and action as the inputs and outputs the corresponding Q-value. This is to measure how good the action is from the actor network. Therefore, the equation of the $Q$-value can be used ($Q$ = reward + discount value . $Q$ next). Thus,

$$\begin{aligned} \nabla_{\theta^\mu} J \approx & \mathbb{E}_{s^{(t+1)}} \left[ \nabla_{\theta^\mu} Q(s^{(t)}, a^{(t)} | \theta^Q)|_{s^{(t)}=s^{(t+1)}, a^t=\mu(s^{(t+1)}|\theta^\mu)} \right] \\ = & \mathbb{E}_{s^{(t+1)}} \left[ \nabla_a Q(s^{(t)}, a^{(t)} | \theta^Q)|_{s^{(t)}=s^{(t+1)}, a^{(t)}=\mu(s^{t+1})} \nabla_{\theta_\mu} \mu(s^{(t)} | \theta^\mu)|_{s^{(t)}=s^{(t+1)}} \right]. \end{aligned} \tag{2.18}$$

- **Initialization:**

  To begin the optimization processes, initialized the actor and critic network as $\mu(s|\theta^\mu)$ and $Q(s, a|\theta^Q)$, with weights as $\theta^\mu$ and $\theta^Q$. We also initialized the target networks $\mu'$ and $Q'$ with their weights $\theta^{\mu'} \leftarrow \theta^\mu$ and $\theta^{Q'} \leftarrow \theta^Q$. Finally, we initialized the replay buffer as $\mathcal{D}$.

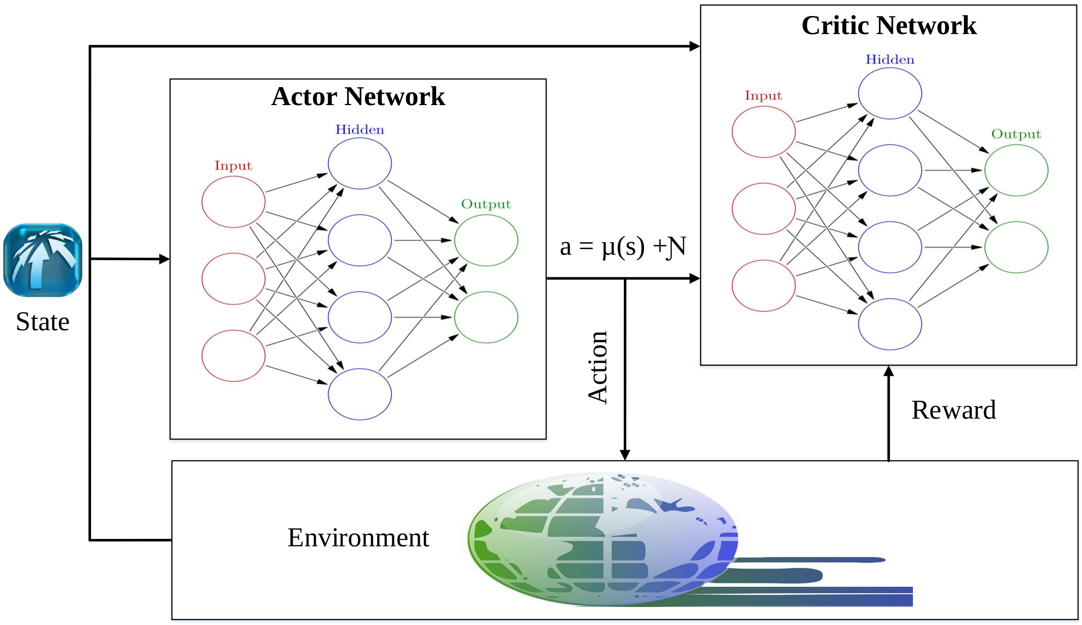


Figure 2.11: An illustration of the actor and critic model

- **Learning Architecture Process:**

For each episode, the agent initializes a random process $N$ for action exploration. Next, the agent receives the state. For each iteration in the episode, the agent receive a state as input and take the action as $a^*(s) = \underset{a}{\text{argmax}}\ Q(s,a) + N$. Next, the same state is considered as an input for the critic network. Also, the output action from the actor added as input for the critic network. The output of critic network is known as Q-value which is the expected total reward for the current state and action pair. This tuple (action $a^{(t)}$, stat $s^{(t)}$, reward $r^{(t)}$, and next state $s^{(t+1)}$) are stored in $\mathcal{D}$. After that, the agent use this data (random mini batch sample) from $\mathcal{D}$ to update both actor and critic networks. Next, the agent update the target networks wights and minimize the loss. The loss function is expressed as $\mathcal{L} = \frac{1}{N}\sum_i \big(y_i - Q(s_i, a_i|\theta^Q)\big)^2$, where $y_i = r_i + \gamma Q'\big(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}\big)$. In addition, the agent updates the actor policy, which is expressed as

$$\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_i \nabla_a Q(s,a|\theta^Q)|_{s=s_i,a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i}. \tag{2.19}$$

The updated target networks shows for the actor and critic as $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$ and $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$.

Therefore, the actor-critic algorithm can learn the dynamic environment of the actor and critic networks.

### 2.5.7 Soft Actor-Critic Learning

SAC is a variation of the actor-critic algorithm that uses a soft value function instead of a hard one. Rather than maximising the expected reward, SAC maximizes the expected entropy-weighted reward. Exploration is encouraged by the entropy term, so the agent will not be stuck in a local optimum. In SAC, there are three neural networks: the actor network, the critic network, and the temperature parameter network. Actor networks output actions based on the current environment state. Taking into account the actor's action and the current state of the environment, the critic network outputs a Q-value. A temperature parameter network scales the entropy term based on the current environment state. The detailed process of SAC is given below.

- **Initialization:**

  To begin the optimization processes, network environment parameters and training hyperparameters are initialized. Based on the environment, the maximum episodes and iterations are defined. Moreover, replay memory and batch size are initialized and used by the agent to store and learn from the previous experiences. Finally, the brain of the SAC agent is initialized as three different neural networks (actor, critic, and value) to learn the optimal policy.

- **Brain Architecture:**

  SAC considered fully connected neural networks (FCNNs) architecture for the brain of the proposed agent because FCNNs are considered efficient architecture of artificial neural networks to process the dynamic environment [39, 46, 89]. Additionally, to dynamically tune/adjust the network weights, SAC can be equipped with a for-

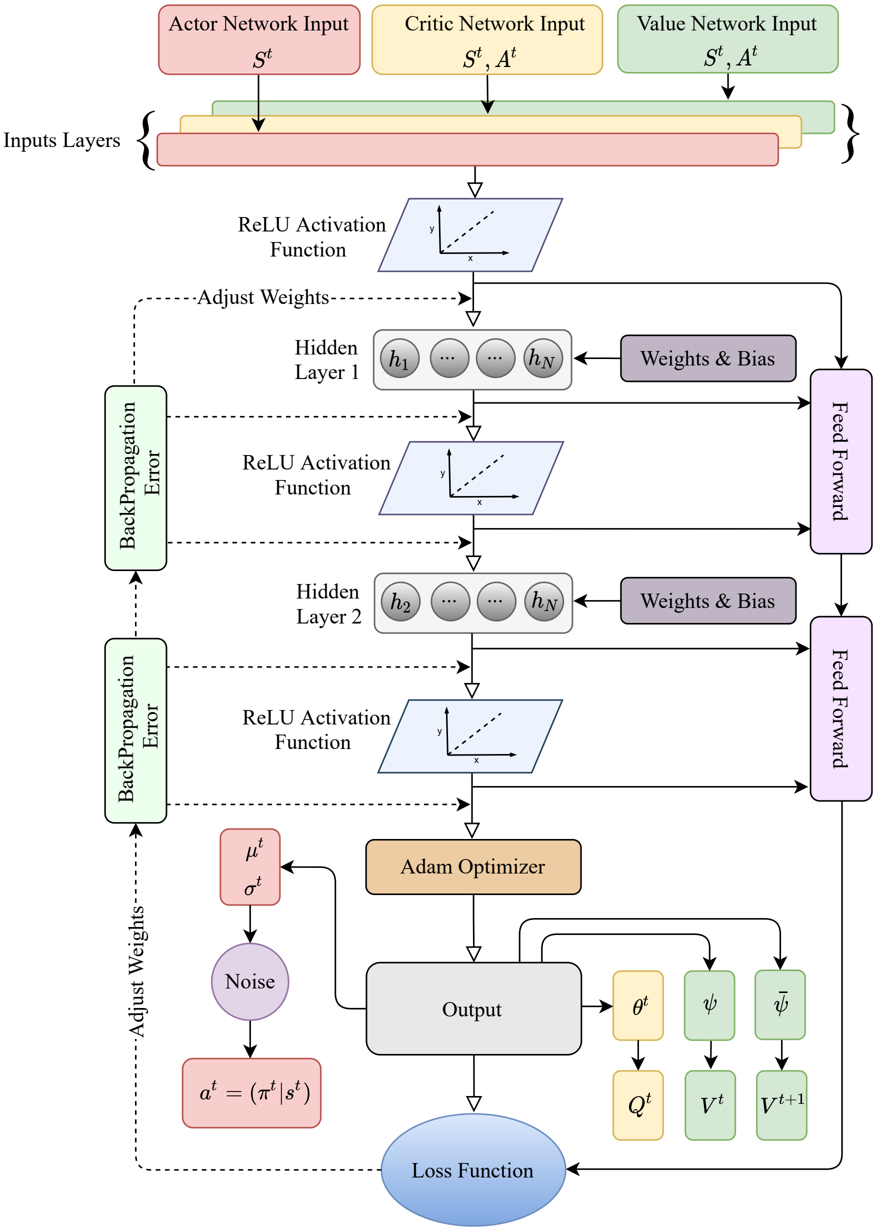


Figure 2.12: An illustration of the actor, critic, and value neural networks model

ward and backward propagation mechanism to the brain of the SAC agent. The feed-forward propagation mainly performs the functions of neuron activation, neuron transfer, and forward propagation. First, the neuron activation computes the weighted sum for the input and the bias. The neuron transfer invokes the activation function, such as the rectified linear unit (ReLU), to activate the neurons. Finally, forward propagation is the process of providing input to the next layer.

This process happens for all the remaining layers.

After doing the feed-forward propagation, the back propagation helps to increase the stability of the weights updated in the neural network. This is based on two main factors: Transfer derivative and error back propagation. Moreover, the optimization function in this model is based on an adaptive moment estimation optimizer (Adam) to optimize the error between the weight and the bias. Finally, to get robust stable learning and optimize the requirement, SAC use the following three neural networks.

– **Actor Network ($\phi$):**

This model is based on the throughput maximization policy $\pi_\phi(s^{(t)}, a^{(t)})$. The architecture of the actor network is shown in Fig. 2.12, where the input and output for the actor network are highlighted within a red coloured box. The architecture of this network consists of one input layer and two hidden layers with ReLU activation functions, feed-forward propagation, backpropagation, loss function, Adam optimizer, and output mechanisms to perform efficient action in the dynamic environment. Starting with the inputs, the actor network receives states as input from the environment. The first hidden layer receives the network environment information, that is, output propagated from the first layer activated by the ReLU activation function. The output of this hidden layer is in the form of weights and bias. The same process continues with the second hidden layer until the final output. We utilize the Adam optimizer to compute the gradients used in updating the weights of the neural networks, thus minimizing the overall loss when predicting the output, which is an action $a^{(t)}$. Generally, this back-propagation process helps the neural network to minimize the weight prediction errors by adjusting neural network weights during the learning process.

The updated parameters of the actor network are:

$$\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi Z_\pi(\phi). \tag{2.20}$$

- **Critic Network ($\theta$):** Similar to the first neural network architecture (actor), the critic network follows the same architectural design. The architecture of the critic network is shown in Fig. 2.12, where the details of the input and output are highlighted within a yellow coloured box. The input of this network is different from that of the actor network, which is based on the state and action at each time slot $t$. The function of the critic network is to learn the current Q-value in the future key value by calculating the Bellman equation. For this reason, the input of the critic network is different from the actor network. As the name suggests, the Bellman equation is updated with soft $Q$ updates. The soft $Q$-function is denoted as $Q_\theta(s^{(t)}, a^{(t)})$. Finally, the $Q$-function update is as follows:

$$\theta \leftarrow \theta - \lambda_Q \hat{\nabla}_\theta Z_Q(\theta). \tag{2.21}$$

- **Value Network and Target Value Network ($\psi, \bar{\psi}$):**

  The value network is denoted by $V^{(t)}(\psi)$, and the target value network is denoted by $V^{(t+1)}$ $(\bar{\psi})$. The architecture of the value network follows the same design as the actor and critic networks. As shown in Fig. 2.12, the details of the input and output are highlighted within a green coloured box. The input of these networks is the state to predict the current and target values for the given state. To learn the efficient requirements via policy $\pi$, the value network output $V^{(t)}$ seeks to minimize the error between the two value networks to assist the agent efficiently. The value network is updated with the help of the following equation:

$$\psi \leftarrow \psi - \lambda_V \hat{\nabla}_{\psi} Z_V(\psi). \tag{2.22}$$

Similarly, the target value network $V^{t+1}$ is updated with the following equation:

$$\bar{\psi} \leftarrow \tau\psi + (1-\tau)(\bar{\psi}), \tag{2.23}$$

where $\tau$ represents the smoothing coefficient of the target value. The function $\tau$ is used to stabilize the training process of the SAC agent. The higher the value of $\tau$, the faster the updating of the value network. Due to this, the learning becomes unstable. However, the smaller target value coefficient leads to slow updates. This helps the SAC agent learn efficiently.

In a variety of continuous control tasks, SAC outperforms other state-of-the-art RL algorithms. However, training and tuning hyperparameters can be computationally expensive.

## 2.6 Chapter Summary

An overview of NOMA, BAC, and ML has been presented in this chapter. In particular, we discussed the NOMA transmission for both downlink and uplink communication, followed by BAC-NOMA communication. Several challenges and motivations have been highlighted for the downlink NOMA, such as enhancing data rate and reducing model complexity. In the case of uplink NOMA, different challenges are encountered, including enhancing the data rate, QoS, EE, reducing interference, especially with different types of users, namely, GB and GF. The simultaneous downlink-uplink NOMA presents challenges related to the increase in complexity, improvement of QoS, and reduction of interference. To address some of these challenges and problems, state-of-the-art optimization solutions have been applied using various ML algorithms, each with its unique features and characteristics tailored for wireless communication. The next chapter delves into the optimization process for downlink NOMA networks using ML techniques.

# Chapter 3

# An Efficient Actor Critic DRL Framework for Resource Allocation in Multi-cell Downlink NOMA

## 3.1 Introduction

One of the key technologies in beyond 5G and 6G wireless networks is NOMA. Unlike OMA networks, NOMA can efficiently share the resource among multiple users thanks to the network's expansion. Resource allocation in NOMA network becomes complex due to the dynamic nature of the communication channels and network access. The research application of NOMA are used in different scenarios such as downlink, uplink and simultaneous downlink and uplink networks. This chapter focuses on the application of downlink NOMA, while subsequent chapters will discuss the uplink network and the simultaneous downlink and uplink NOMA network.

Improving the downlink NOMA network involves adapting to dynamic network environment. This lead to enhance the resource allocation, improved throughput, reduce the complexity, minimize the interference, and increase EE. To address some challenges of dynamic network management, model free techniques have been recommended [39, 49]. Both approaches utilize ML to understand and learn the relationship between optimal resource allocations and dynamic channel conditions, ensuring the QoS for each NOMA user. However, the drawback of adopting some ML algorithms are the requirement of huge trusted training datasets [90, 91] which are not always available. Therefore, applying solution such as RL algorithms considered a suitable to learn the wireless network environment. Interestingly, the new variant of RL algorithms is designed to achieve the human level controls for real-time environment [86]. The extended version of RL algorithms introduces various DNNs to solve complex problems with a large state-action space, which forms different DRL algorithms. Recently, a new DRL approach with two DNNs, i.e., Actor and Critic networks, was proposed to efficiently learn policies in high dimensional, continuous action spaces [43]. The designed of the network needed to be improved such as low the complexity, enhance the network environment and many more.

To further investigate the effect downlink NOMA system with multiple BSs, the proposed model aiming for high data rate, decrease the complexity, improve the QoS and many more, this work proposes a novel state space design for ACDRL algorithms to learn high dimensional or continuous action spaces and achieve faster convergence. The proposed ACDRL algorithms reduce complexity when tasked to learn discrete action spaces (ACDRL-D), and even more so, they enhance performance in continuous action spaces (ACDRL-C).

### 3.1.1 Contributions

In multi-cell downlink NOMA systems, which utilize convex optimization, several inherent challenges exist. These challenges include managing intra- and inter-cell interference, which complicates efficient resource allocation and user fairness. The process of user pairing and clustering, which is crucial for NOMA's performance, presents computational difficulties due to its combinatorial nature and the need to balance efficiency and fairness. Moreover, power allocation, which is a key factor in maximizing system capacity, is impeded by the non-linear nature of the problem and the dynamic wireless environment. This makes it computationally intensive and challenging to achieve a global optimum. Furthermore, the computational requirements for solving these convex optimization problems, especially in real-time situations, raise concerns about scalability and latency, particularly as the user count increases. These complexities highlight the need for advanced solutions to fully harness the potential of NOMA in future wireless networks. Therefore, we proposed the ACDRL system model to enhance throughput and reduce complexity.

The key contributions include;

- We utilize the ACDRL model with two DNNs to handle realistic state-action space for the long-term resource allocations of multi-cell downlink NOMA systems. We designed an efficient action space that helps the agent to perform resource allocation depending on continuous actions to achieve better convergence. Each action represents sub-channel assignment and power allocation operations.

- We developed a state space that represents 3D associations among users, base stations (with varying transmission power levels), and sub-channels.

- Similarly, the proposed continuous actions and the rewarding mechanism are based on data rates of network users to direct the agents for long-term resource allocations.

- With the help of a reward function for the proposed state and action space, the convergence is achieved within less number of episodes. Also, the proposed algorithm outperforms DRL with 30.3% increase in data rate.

## 3.2 System Model and Problem Formulation

This section illustrate the proposed system model in Sub-section 3.2.1 and problem formulation in Sub-section 3.2.2.

### 3.2.1 System Model

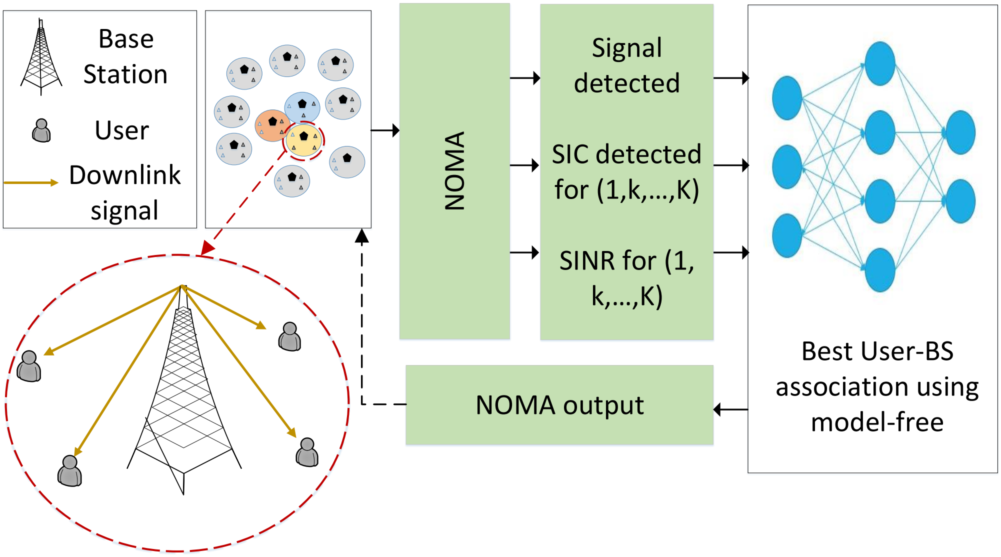


Figure 3.1: Illustrates multi-cell downlink NOMA network by using model-free ACDRL optimization algorithm

This chapter considers a multi-cell downlink NOMA network as shown in Fig. 3.1. $B$, $I$, $J$, denotes the number of BSs, setup users, and number of sub-channels, respectively. All BSs are using single antenna. In the considered region, the locations of BSs are fixed. To enhance the generality, users have random locations across each transmission time slot. We assume that every BS has perfect CSI, transmission power $P$ dBm, bandwidth $\hat{B}$, and $J$ as orthogonal sub-channels, so each sub-channel has $\hat{B}/J$ bandwidth. Each user $i$ communicated to BS $b$ via sub-channel $j$ is denoted by $i_{b,j}$, where $(i \in [1, I], b \in [1, B], j \in [1, J])$. We use $q_{i_{b,j}}$ to indicate the existence of user-BS connection between user $i_{b,j}$ and BS $b$ via sub-channel $j$ . Therefore, a user-BS connection, $q_{i_{b,j}} = 1$ means this connection is active, otherwise $q_{i_{b,j}} = 0$. We assume that the channel gains on sub-channel $j$ of BS $b$ follow the order to $|h_{i_{b,j}}|^2 \geq ... \geq |h_{I_{b,j}}|^2$, where $i_{b,j}$ and $I_{b,j}$ are the users with the strongest and weakest channel conditions respectively [19]. According to

NOMA principles, the user $i_{b,j}$ first cancels the signals of the rest of the users till the last user $I_{b,j}$ via SIC before it decodes its own information. The received signal for the user $i_{b,j}$ is,

$$\begin{aligned} y_{i_{b,j}} = & \underbrace{q_{i_{b,j}} h_{i_{b,j}} \sqrt{P_{i_{b,j}}} x_{i_{b,j}}}_{\text{Desired Signal}} + \underbrace{\sum_{i_{b,j} \neq i'_{b,j}} q_{i'_{b,j}} h_{i'_{b,j}} \sqrt{P_{i'_{b,j}}} x_{i'_{b,j}}}_{\text{Intra-Cell Interference}} \\ & + \underbrace{\sum_{i' \neq i}^{I} \sum_{b' \neq b}^{B} q_{i'_{b',j}} h_{i'_{b',j}} \sqrt{P_{i'_{b',j}}} x_{i'_{b',j}}}_{\text{Inter-Cell Interference}} + \underbrace{n_0}_{\text{Noise}}, \end{aligned} \tag{3.1}$$

where $n_0$ represents noise and $P_{i_{b,j}}$ represents the transmit power for the $i$-th user connected to the $b$-th BS via the $j$-th sub-channel. The decoding order is based on the statistical CSI, where users with strong channel conditions are decoded first, and the last user to decode has the weakest channel condition [19].

The SINR for the $i$-th user connected to the $b$-th BS via sub-channel $j$-th can be expressed as:

$$SINR_{i_{b,j}} = \frac{q_{i_{b,j}} P_{i_{b,j}} |h_{i_{b,j}}|^2}{\sum_{i'_{b,j} \neq i_{b,j}}^{I} q_{i'_{b,j}} P_{i'_{b,j}} |h_{i'_{b,j}}|^2 + \sum_{i' \neq i}^{I} \sum_{b' \neq b}^{B} q_{i'_{b',j}} P_{i'_{b',j}} |h_{i'_{b',j}}|^2 + n_{i_{b,j}}^2}. \tag{3.2}$$

The achievable data rate for user $i_{b,j}$ is given by:

$$R_{i_{b,j}} = \hat{B} \log_2(1 + SINR_{i_{b,j}}). \tag{3.3}$$

### 3.2.2 Problem Formulation

In order to derive an optimal resource allocation strategy, we formulate a problem to maximize the long-term achievable data rate for a period of $T$:

$$\max_{\mathbf{P_t}} \sum_{i=1}^{I}\sum_{b=1}^{B}\sum_{j=1}^{J}\sum_{t=1}^{T} R_{i_{b,j}}(t)/T \tag{3.4a}$$

$$\text{s.t} : |h_{i_{b,j}}|^{2^{(t)}} \geq ... \geq |h_{I_{b,j}}|^{2^{(t)}}, \quad \forall i, b, j, t, \tag{3.4b}$$

$$R_{i_{b,j}} > R_{\gamma}, \quad \forall i, b, j, t, \tag{3.4c}$$

$$2 \leq \sum_{i_{b,j}=1}^{I_{b,j}} q_{i_{b,j}}{}^{(t)} \leq I_{b,j}, \quad \forall b, j, t, \tag{3.4d}$$

$$\sum_{b=1}^{B} q_{i_{b,j}}{}^{(t)} = 1, \quad \forall i, \tag{3.4e}$$

$$\sum_{i_{b,j}=1}^{I_{b,j}} q_{i_{b,j}}^{(t)} P_{i_{b,j}}{}^{(t)} \leq P_{max}, \quad \forall b, t, \tag{3.4f}$$

$$q_{i_{b,j}}^{(t)} \in \{0, 1\}, \quad \forall b, j, t, \tag{3.4g}$$

where $\mathbf{P_t}$ is the transmit power of the BS. $T$ represents the network time slot and for each time slot $t$, (3.4b) represent the decoding order, (3.4c) is to ensure that the rate for the weakest user should never be less than the minimum rate ($R_{\gamma}$=0). (3.4d) implies that the number of users for each sub-channel and each BS is in the range $[2, i_{b,j}]$, (3.4e) ensures that a user can connect to only one BS at a time. (3.4f) limits the power consumption for each BS and the maximal transmit power for the BS is $P_{max}$. (3.4g) shows that the user connectivity is 1 or 0.

Due to multi-cell settings, the optimization of the problem defined in (3.4a) is mixed integer linear programming problem [92]. Therefore, we optimize the resource allocation by using machine learning technique to provide long-term solutions to the formulated problem.

## 3.3 Actor Critic Based Learning Networks

In order to tackle the optimization problem defined above, we propose DDPG actor-critic, that is equipped with the power of two neural networks (actor and critic network) to learn the complex multi-cell downlink NOMA systems. The actor network is responsible to perform optimal actions, while to assist actor network, the estimation of state and action for the Q-value is the responsibility of the critic network [93]. In the proposed design, we have an agent who have to be experienced (knowledgeable enough about the environment) to maximize the resource allocations. In order to strike a balance between exploration and exploitation while learning from the environment, we equip our DDPG with actor and critic networks to learn the efficient long-term allocation policy. Consequently, to drive the long-term policy, the actor network is responsible for selecting and performing optimal actions, while the estimation of the Q-value is done by the critic network for each state and action pair [94].

### 3.3.1 The ACDRL Design Elements

Actor critic networks rely on the MDP to navigate through the environment, which is a downlink NOMA network. At each time step $t$ during the learning process, the agent observes the current state and selects the action following its policy $\pi$, i.e., $Q^{\pi}(s^{(t)}, a^{(t)})$, upon which it receives a reward $r^{(t)}$. Subsequently, the agent transitions to the next state $s^{t+1}$ and the agent recursively uses the policy to take the action in the given state until the maximum sum of rewards are obtained. The MDP for the formulated problem is defined as:

- **Environment:** The multi-cell downlink NOMA network is the learning environment for the proposed ACDRL agent. Using this environment we define the following significant elements for the ACDRL agent to achieve the long-term resource allocation.

- **Agent:** We assume that all BSs are controlled by a central server connected via a high-speed backbone. Therefore, due to the sophisticated design of the proposed state and action, all the BSs jointly optimize long-term average rewards.

- **State space ($\mathcal{S}$):** We assume 3D associations which are a combination of all BSs ($B$), users ($I$), and sub-channels ($J$) associations. Each single state at a time slot $t$ is represented as an association between the three elements of user, BS power level ($P_l$), and sub-channel as in (3.5) and therefore, the entire state space size the agent navigates is the union of all possible states as in (3.6).

$$s^{(t)} = \{i_{b,j}, P_{l_{i_{b,j}}}, h_{i_{b,j}}\}_{i=1}^{I} \tag{3.5}$$

$$S = \bigcup_{b\in[1,B],j\in[1,J],l\in[1,5]} s^{(t)}(\forall i) \tag{3.6}$$

- **Action space ($\mathcal{A}$):** The action is to assign the power to each $i$-th user that is connected to $b$-th BS via the $j$-th sub-channel. Therefore, the action is defined as:

$$\begin{aligned} a^{(t)} =& \{q_{1_{1,1}}{}^{(t)}, \dots, q_{i_{b,j}}{}^{(t)}, \dots, q_{I_{B,J}}{}^{(t)}, \\ & P_{1_{1,1}}{}^{(t)}, \dots, P_{i_{b,j}}{}^{(t)}, \dots, P_{I_{B,J}}{}^{(t)}\}, \end{aligned} \tag{3.7}$$

- **Exploration:** The actor network is a significant component of the ACDRL agent as it guides towards the final solution. Therefore, effective exploration of the environment is the key to find the diverse solution. For this reason, to encourage the agent for more exploration, the $\mathcal{N}^{(t)}$ is a noise which added to the action $a^{(t)}$ to increase the exploration with the agent. The equation to perform the action for the given state is defined as follows:

$$a^{(t)} = \mu(s^{(t)}|\theta^{\mu}) + \mathcal{N}^{(t)}, \tag{3.8}$$

where the output of the actor network is represented by $\mu$ and actor network weights are denoted by $\theta$. We leverage the actor critic design to increase the learning efficiency, and thus treat the problem of exploration independently from the learning algorithm.

- **Reward ($\mathcal{R}$):** The objective of the learning model is to maximize the long-term reward. In the time slot $t$, the agent receives the reward $r^{(t)}$. The reward function used in this model is defined as follows:

$$r^{(t)}(s^{(t)}, a^{(t)}) = \begin{cases} \hat{\mathcal{Z}}^{(t)}, & \text{if } \hat{\mathcal{Z}}^{(t)} \geq \hat{\mathcal{Z}}^{(t-1)} \\ 0, & \text{otherwise.} \end{cases} \tag{3.9}$$

where $\hat{\mathcal{Z}} = \sum_{i=1}^{I}\sum_{b=1}^{B}\sum_{j=1}^{J} R_{i_b,j}^{(t)}$.

- **Learning rate:** $\alpha \in$ (0,1). $\alpha$ is similar to the simple step function. For example, when the step size is larger, it leads the agent towards a random walk. Therefore, due to the large learning rate the agent gains less knowledge from the underline NOMA environment. If the agent take small step size, then the agent moves slowly which results in poor convergence.

- **Discount factor:** The discount factor $\gamma \in$ [0,1] determines the importance of current or the future rewards to the current state. Small discount factor value exhibits high impact towards current rewards (data rate), while the high value of discount factor prioritises the future rewards (data rate).

- **Policy $\pi$:** A policy $\pi$ is used, so that the agent finds the best action $a$ that can be performed for the state $s$, and the equation is defined as follows:

$$\pi(a|s) = \mathcal{P}\left[\mathcal{A}^{(t)} = a|\mathcal{S}^{(t)} = s\right], \tag{3.10}$$

where $\mathcal{P}$ is the probability.

- **Experience replay memory:** Similar to the traditional DRL, ACDRL also uses an experience replay buffer with additional two neural networks to deal with the continuous network environment. The proposed ACDRL based system model use an experience replay memory to store the data from multi-cell NOMA network as $(s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)})$.

The $Q^{\pi}$ function for RL agent is defined as

$$Q^{\pi}(s^{(t)}, a^{(t)}) = \mathbb{E}[r^{(t)}|s^{(t)}, a^{(t)}], \tag{3.11}$$

where $Q^{\pi}(s^{(t)}, a^{(t)})$ is the value function for the policy $\pi$, that is expected return based on initial state $s^{(0)}$ to the final state. A policy $\pi$ is used, so that the agent finds the best action $a^{(t)}$ that can be performed for the state $s^{(t)}$. The optimal policy is given by:

$$\pi^{*} = \operatorname*{argmax}_{\pi} Q(s^{(t)}, a^{(t)}; \theta). \tag{3.12}$$

For each step, the ACDRL algorithm performs interactions with the environment by performing actions to shift from the current state to the next suitable state according to the selected actions. In this way, the agent begins to learn by updating two neural networks. The detailed discussion is provided in the following subsection.

### 3.3.2 Actor Critic Network Details

The Q-value for a state and action in Bellman equation can be updated using the following equation:

$$Q^{\pi}(s^{(t)}, a^{(t)}) = \mathbb{E}_{r^{(t)},s^{(t+1)}}\left[r(s^{(t)}, a^{(t)}) + \gamma \mathbb{E}_{a^{(t+1)}\sim\pi}[Q^{\pi}(s^{(t+1)}, a^{(t+1)})]\right]. \tag{3.13}$$

With the deterministic target policy, we can defined it as a function $\mu : \mathcal{S} \leftarrow \mathcal{A}$

$$Q^{\mu}(s^{(t)}, a^{(t)}) = \mathbb{E}_{r^{(t)},s^{(t+1)}}[r(s^{(t)}, a^{(t)}) + \gamma Q^{\mu}(s^{(t+1)}, \mu(s^{(t+1)}))]. \tag{3.14}$$

**Algorithm 1** ACDRL for Downlink NOMA System

```
Initialize $s^{(t)}, a^{(t)}, r^{(t)}, \theta^{(t)}$, replay memory $\mathcal{D}$, and batch-size.
Initialize actor and critic network.
for episode =1, $N_e$ do
    $s^{(t)} \to 0$ and $r^{(t)} \to 0$.
    Initialize a random process $\mathcal{N}^{(t)}$ for action exploration.
    Receive state $s^{(1)}$.
    for iteration = 1: $T_e$ do
        Choose $a^{(t)}$ using equation (3.8).
        Perform $a^{(t)}$ and receive reward $r^{(t)}$ (compute equation (3.9)) and new state $s^{(t+1)}$.
        Save transition $(s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)})$ in $\mathcal{D}$.
        Sample mini-batch $(s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)})$ from $\mathcal{D}$.
        Update the actor using the sampled gradient:
        $\nabla_{\theta^\mu} J$ compute equation (3.17).
        Update the target networks:
        $\theta^{Q'}$ compute equation ($\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$).
        $\theta^{\mu'}$ compute equation ($\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$).
    end for
    Return optimised $\mathbf{P_t}$ (3.4a) under constraints from (3.4b) to (3.4g) .
end for
```

Similarly, the loss function for the DNN is calculated as:

$$\mathcal{L}(\theta^Q) = \mathbb{E}_{s^{(t)},a^{(t)},\xi,r^{(t)}}\left[((Q(s^{(t)}, a^{(t)}|\theta^Q) - y^{(t)})^2\right], \tag{3.15}$$

where $\xi$ is stochastic behaviour policy and target Q-value is $y^{(t)}$ shown in equation (3.16) where the value of $y^{(t)}$ is shown as:

$$y^{(t)} = r(s^{(t)}, a^{(t)}) + \gamma Q(s^{(t+1)}, \mu(s^{(t+1)})|\theta^Q), \tag{3.16}$$

the parameter for actor function $\mu(s^{(t)}|\theta^\mu)$ specifies the current policy where the ACDRL algorithm deterministically maps the states and action pair. Actor network is updated as following:

$$\begin{aligned}\nabla_{\theta^\mu} J \approx & \mathbb{E}_{s^{t+1}}\left[\nabla_{\theta^\mu} Q(s^t, a^t|\theta^Q)|_{s^t=s^{t+1}, a^t=\mu(s^{t+1}|\theta^\mu)}\right] \\ = & \mathbb{E}_{s^{t+1}}\left[\nabla_a Q(s^t, a^t|\theta^Q)|_{s^t=s^{t+1}, a^t=\mu(s^{t+1})} \nabla_{\theta_\mu} \mu(s^t|\theta^\mu)|_{s^t=s^{t+1}}\right].\end{aligned} \tag{3.17}$$

The description of **Algorithm 1** are as follows:

- Line (1-2) Indicates the initialization of the model with a set of inputs that include the state and action space, replay memory $\mathcal{D}$, reward and initialize actor and critic network. The training parameters of the model such as training episodes and batch-size are set.

- Line (3-14) the agent performers episodes to learn the wireless network environment for the long-term and in each episode the state and reward are initialised from zero. However, the short term learning is performed in each trial $(T_e)$. In (7-18) the agent performs trials $(T_e)$. In each trial, the agent takes an action $a^t$ to allocate power to NOMA users. A noise $\mathcal{N}$ is introduced to improve the exploration so the agent could learn efficient actions. Based on these actions, the agent changes from one state $s^t$ to another and computes the reward function as the data rate. Actions in the proposed algorithm represent the power allocation tasks for network users which are performed by the ACDRL agent. For efficient learning, the ACDRL agent use two different neural networks known as actor and critic networks to explore the NOMA networks environment for the long-term basis. After updating the state, the tuple $(s^{(t)}, a^{(t)}, r^{(t)}, s^{(t+1)})$ is saved in the replay memory $\mathcal{D}$. After that, the agent compute the loss function. Therefore, when the replay memory is full, the agent starts learning from that experience. For each episode, a mini batch is sample for the learning process.

- Line (12-13), agent update the neural networks. The actor network performs learning by maximizing the state value function. Target actor and critic networks use $\tau$ as soft updating parameters. For efficient learning, the wights of both target networks are slowly updated.

- In the final line (15), the whole optimization process terminates by providing the resource allocation as an output.

### 3.3.3 The Complexity of ACDRL

When applying ACDRL to resource allocation in wireless communications, two major challenges arise: training and convergence time, as well as the delay in real-time processing. The training process for ACDRL models is intrinsically complicated and demanding of data, often requiring substantial time to attain an optimal policy, particularly in dynamic network environments where continuous learning and adaptation are essential. Moreover, this intricacy can result in substantial latency in decision-making, which is a critical concern for real-time communication applications.

The complexity of the proposed DNN-based ACDRL algorithm, with a neural network having $L$ layers and each layer have number of neuron $\hat{x}$. The input layer size is $e$. Based on these parameters, the computational complexity for a single forward pass and back propagation can be given as $\hat{E} \triangleq e\hat{x}_1 + \sum_{l=1}^{L-1}(\hat{x}_l)\hat{x}_{l+1}$. Therefore the real time complexity of the proposed algorithm for $N_e$ episodes and $T_e$ iterations can be written as $\mathcal{O}(N_e T_e \hat{E})$.

## 3.4 Simulation Results

In this section, we provide simulation results to illustrate the performance efficiency of the proposed ACDRL algorithm. We consider multiple BSs and multiple users. Number of BSs are three and number of users are 12. Different noise were applied such as -154, -165, and -174 decibel-milliwatts (dBm). Number of episodes considered in this simulation is one thousand. We use ReLU as an activation function. The optimiser is adaptive moment estimation optimizer (Adam) and the learning memory is six hundred and fifty [95]. The batch size is one hundred twenty eight. Simulation parameters are listed in Table 3.1. By running different algorithms that shows in all figures, we use computational resources such as CPU, and we test it in small scale models environment where we have only 3 BS and 12 users. We tune the parameters and reduce the size of the neural network to speed the training time without effecting the learning process. Also, using experience replay (past experiences) help to achieve the optimal performance

Table 3.1: Network parameters and ACDRL algorithm parameters

| **Parameter** | **Value** |
|---|---|
| Power levels | $[20 - 40]\ dBm$ |
| Network size | $9, 12$ Max users |
| BSs | 3 |
| Fading | Rayleigh fading |
| Sub-channels | 2 |
| Power of noise | $-[154, 164, 174]\ dBm$ |
| Bandwidth | $30kHz$ |
| Episodes | $1,000$ |
| DNN activation | ReLU |
| DNN layers $(H1, H2)$ | $[300, 400]$ |
| Optimiser | Adam |
| Replay memory | 650 |
| Batch size | 128 |

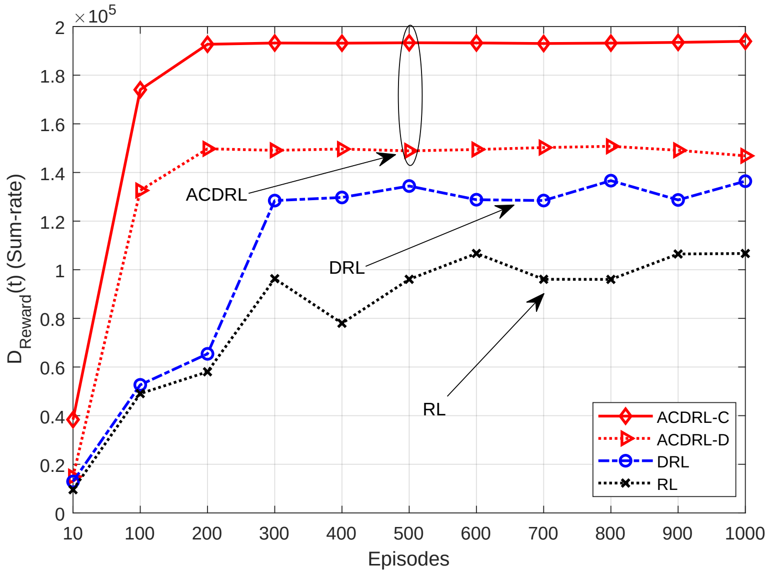


Figure 3.2: Illustrates the reward convergence of ACDRL-C, ACDRL-D conventional DRL, and RL

reduce the delay and time.

The system used to get the simulation results is MacBook Pro macOS system with processor 3.1 GHz Intel Core i5 and memory 8GB (Random Access Memory) 2133 MHz LPDDR3. We use Python version 3.6 in this simulation.

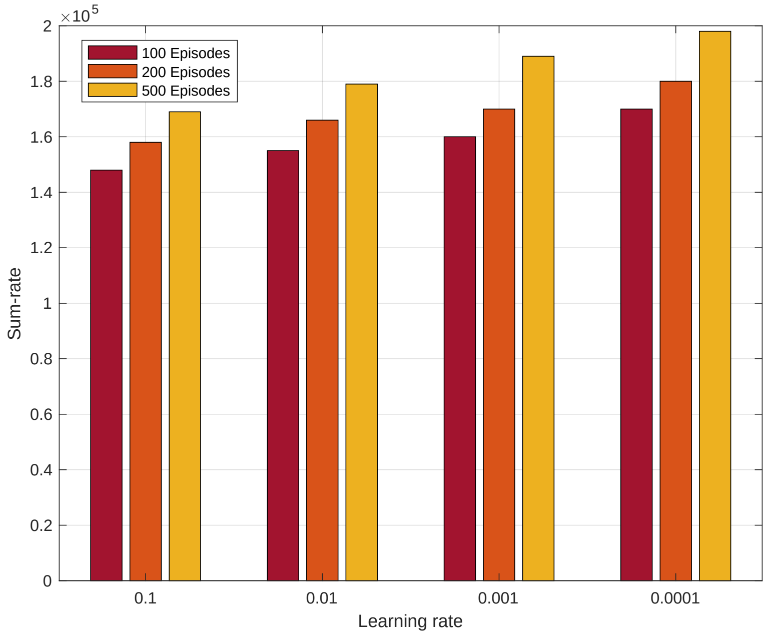


Figure 3.3: An illustration of sum rate with different learning rate in different number of episodes

### 3.4.1 Reward Convergence vs Episodes

Fig. 3.2 illustrates dynamic reward vs. number of episodes with different algorithms (actor-critic deep reinforcement learning - continuous action (ACDRL-C), actor-critic deep reinforcement learning - discrete action (ACDRL-D), DRL, and RL) convergence. The proposed algorithms (ACDRL-C and ACDRL-D) not only receive a higher data rate but also converge within fewer episodes 200 than DRL $\epsilon$ greedy, and traditional RL (Q-learning). Additionally, the ACDRL-C algorithm also performs better than the ACDRL-D algorithm with around increase of around 23.5%. Because ACDRL-C can learn the power allocation efficiently with better rewards in the form of sum rate. Two different network (actor and critic) helped the agent to get better performance in term of sum rate compare to only one NN or traditional RL. This is why the proposed algorithms learned faster and less delay compare to DRL and RL ACDRL-C is around 30.3% better than DRL. Similarly, RL and DRL start converging when the episodes are 300 with less reward performance.

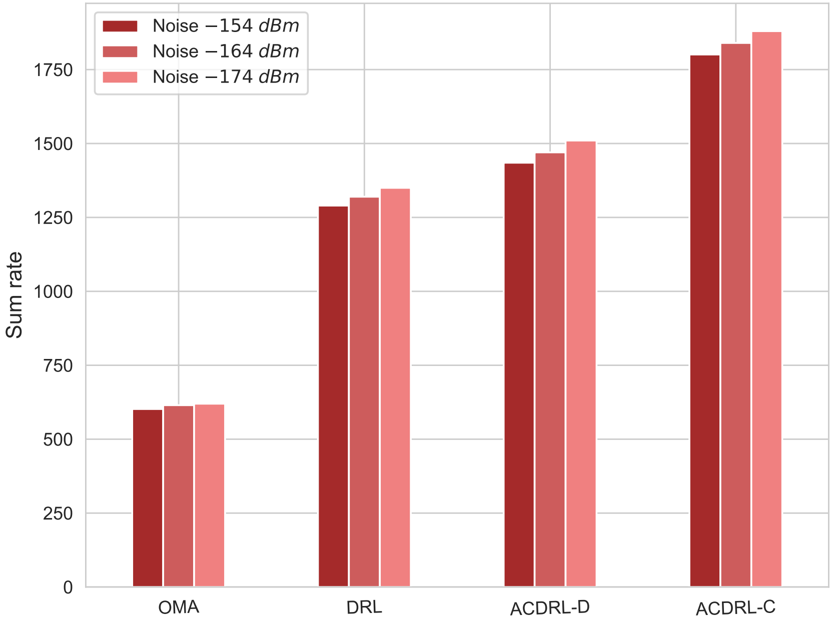


Figure 3.4: Sum rates comparison between OMA, DRL, ACDRL-D and ACDRL-C with different power of noise $n_0$

### 3.4.2 Sum Rate Against Different Learning Rates and Episodes

Fig. 3.3 Shows the sum rate obtained using different learning rates and episodes. It can be seen that learning rate of 0.1 produces small sum rate for all episodes. As we decrease the learning rate from 0.1 to 0.0001, the sum rate increases. Using small learning rate can increase learning time, however, it can find the global optimal solution. On the other hand, using high learning rate can speed up the learning process, but can overshoot the global optimal position or even to diverge. Additionally, it can be observed that increasing the number of episodes enhances the sum rate efficiency. Because, with large number of episodes, agent can efficiently explore the environment for good states and actions.

In rapidly changing network environments, training needs to be fast enough to keep up with these changes. The ACDRL model may lose accuracy and effectiveness if users' channel conditions significantly change without the model being retrained. However, our proposed ACDRL is designed to be continuously learnable, requiring no retraining for updating the model.

### 3.4.3 Sum Rate Comparison with Different Algorithms and Noise

Fig. 3.4 presents the sum rate for DRL, ACDRL-D, ACDRL-C in NOMA, and OMA systems with different power of noise $n_0$. As shown in Fig. 3.4, the sum rate for long-term communications with different power of noise are better for ACDRL algorithm as compared to DRL and traditional OMA systems. From that figure, we can see that when the power of noise parameter decreases, the sum rates increases and vice versa. The general trend from this figure shows that the proposed ACDRL-C with continuous state and actions outperforms conventional DRL and ACDRL-D with discrete state and actions. Which implies that the agent with a continuous action space performs efficient power allocations as compared to an agent with a discrete action space. Lastly, NOMA communication systems outperforms conventional OMA.

### 3.4.4 Sum Rate Comparison with Different Networks Loads

Fig. 3.5 shows several comparisons, i.e., the comparison among dynamic reward, the comparison among different power levels, the comparison between OMA and NOMA, and the comparison among different RL algorithms with two different network loads for discrete action systems. The proposed model ACDRL-D performs better than conventional DRL and RL schemes in all settings. Similarly, the sum rate for 4 users scenarios is higher than 3 users scenarios in all cases. Additionally, the dynamic reward increases (sum-rate) when the transmit power level increases and vice versa across all the setups. Lastly, Fig. 3.5 also shows that NOMA is performing better than OMA for all types of network settings.

## 3.5 Chapter Summary

This chapter has presented the resource allocation based on the ACDRL algorithm with two different designs with continuous and discrete actions. For low complexity, we applied ACDRL-D and for enhanced performances, we utilized ACDRL-C design. In order to improve the sum rate performance of the proposed network, we also provide a dynamic

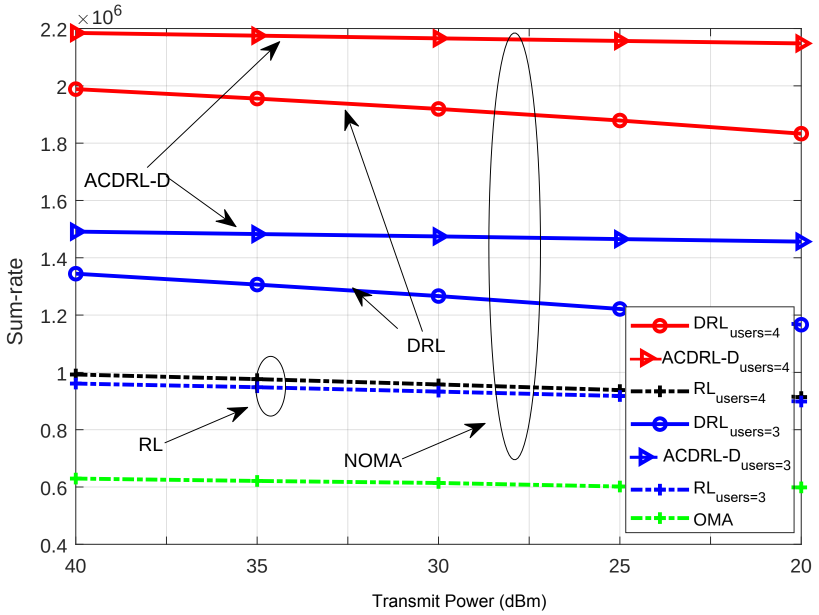


Figure 3.5: Sum-rate comparison of OMA and RL based NOMA techniques with two different network loads

feedback system that is based on the real-time data rate of NOMA users to efficiently guide ACDRL agents. Based on this design, the simulation section has shown that ACDRL-D and ACDRL-C start converging fast within 200 episodes. Also, ACDRL-C outperforms ACDRL-D with 23.5% better in data rate and 30.3% outperforms DRL. Therefore, the dynamical actor critic framework outperforms DRL-NOMA, traditional RL-NOMA, and conventional OMA systems.

# Chapter 4

# Semi-Centralized Optimization for Energy Efficiency in IoT Networks with Uplink NOMA

## 4.1 Introduction

Different from the previous scenario in chapter 3, this chapter focuses only on the scenario of uplink NOMA networks. The application of NOMA network is considered a promising solution for providing massive connectivity to the increasing number of IoT users, which is one of the important use cases of mMTC [39]. In general, IoT users exhibit diverse characteristics, such as a high battery life cycle, sporadic transmission, minimum data rate requirements, and different QoS requirements [61]. Furthermore, with different types of IoT users such as GB and GF, NOMA communication network is different (based on prior handshakes or no handshakes). Many scenarios primarily focus on direct access to the BS due to its simplicity. However, path loss increases with increasing distance, which leads to lower energy efficiency and reduced rates. To overcome the effect of distance-dependent path loss, in the existing work, the source node needs to transmit at a higher power [57]. However, IoT users have small processing and limited transmit power capability, which makes it impractical to communicate over long distances.

In this chapter we focus on enhance characteristics of IoT users (GB and GF). Therefore, we proposed semi-centralized framework where this model solves the problem of distance dependent path loss, improves QoS, and enhances throughput for both GB and GF users.

### 4.1.1 Contributions

The SGF-NOMA method combines aspects of the GB and GF mechanisms to optimize resource allocation efficiency while accommodating a large number of users. The primary challenges involve managing the trade-off between scheduling flexibility and system complexity. Specifically, the SGF-NOMA method must efficiently handle user prioritization, ensuring that critical users have timely access while still maintaining equitable resource distribution among all users. Balancing the dynamic allocation of resources in a rapidly changing environment is crucial. It is necessary to mitigate potential collisions and interference, particularly when users with and without grants coexist. Additionally, maintaining a consistent QoS across varying user demands and network conditions, while minimizing signaling overhead, becomes a significant issue. These challenges require an intelligent and adaptable approach to resource allocation and user management in SGF-NOMA systems. Enhancing the performance of SGF-NOMA in wireless communications can be effectively achieved through a combination of ML techniques and strategic deployment of relay nodes. By utilizing ML algorithms, the system can accurately forecast network conditions and user demands, optimize resource allocation, and effectively manage user prioritization and scheduling. In particular, the agent (BS) use continuous action space (power allocation), PPO algorithm can handle this action space. PPO also known as a stable and robust algorithm. Moreover, within PPO algorithm, a few number of episodes are required in order to achieve a good performance. Therefore, choosing the best action for GB users using PPO within two different networks (actor and critic) lead to improve the learning and get a better policy optimization. This predictive capability is particularly beneficial for balancing the demands of users with grants and those without grants, as well as reducing collisions and interference. Relay nodes can be used to extend coverage, improve signal quality, especially for users at the edge of the network, and EE of small IoT users. The integration of ML enables intelligent control over these relay nodes, optimizing their placement and operation in real time based on dynamic network conditions. This integrated approach leads to a more efficient SGF NOMA system, guaranteeing minimal interference, optimal resource utilization, and consistent quality of service for all users. The main contributions of this chapter are showing as follows:

- We propose a new optimization framework where the GF user transmits its signal to the serving GB user, which is known as a relay node, via the NOMA protocol. Furthermore, we formulate the EE of both the GF and GB users as an optimization problem.

- To jointly optimize the transmit power of GB and GF users, we propose a semi-centralized framework that avoids the disadvantages of fully centralized and fully distributed RL algorithms. In particular, we use the PPO algorithm on the BS side (centralized part) to allocate power level (optimal one) for GB users. However, considering the computational limitations of GF users; a multi-agent deep Q-network (multi-agent deep Q-network (MA-DQN)) algorithm (distributed part) is utilized on the GF user side.

- The experimental results show that our semi-centralized algorithm (the proposed scheme) outperforms the benchmark scheme (random and fixed power allocation) methods and the conventional GF transmission without a relay node in terms of EE. Moreover, we show that the number of GB users has a strong correlation with the EE of both types of users.

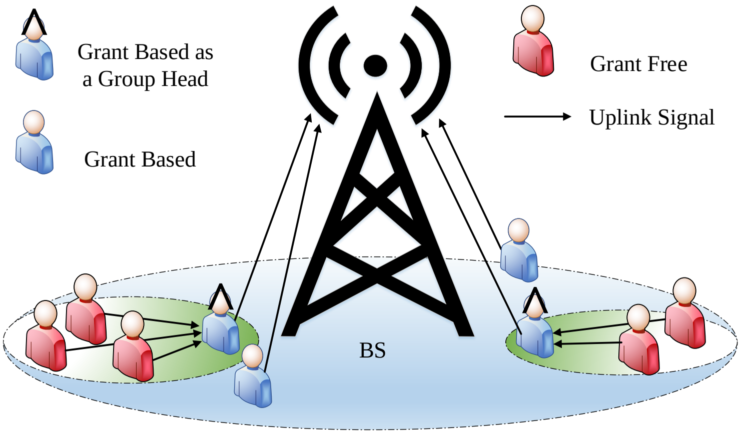


Figure 4.1: The proposed system model with multiple GF users and GB users (including group head)

## 4.2 System Model and Problem Formulation

Unlike downlink NOMA, uplink transmission in SGF NOMA presents new complexities primarily due to its decentralized structure, user-controlled power settings, and the need for efficient simultaneous transmission and interference management. Addressing these challenges requires innovative strategies in resource allocation and signal processing to ensure efficient network operation and user satisfaction. We consider a NOMA IoT network with a single BS located at the center of a circle with a radius $\mathbb{R}$. Two types of users, namely GB (represented by $\mathcal{W}=\{1, 2, \ldots, N_W\}$) and GF (listed as $\mathcal{F}=\{1, 2, \ldots, N_F\}$) transmit their data in an uplink manner, which is given in Fig. 4.1. We assume that GB users are delay sensitive and have enough processing capability to act as cluster head (CH) and the GF users are delay tolerant, e.g, a sensor for temperature monitoring. The GF users send their data to the GB user acting as a CH [96] to reduce the impact of the path loss with the distance $d$, here given by $d^{-\alpha}$ on the energy constrained GF users. The GB users transmit their data to BS via $J$ sub-channels.

### 4.2.1 Signal Model for GB and GF

Both users (GB and GF) transmit their data in a slotted manner. More specifically, the CH $w \in \mathcal{W}$ receives the combined signal from the $N_F$ GF users in time slot $t$, which can be expressed as follows:

$$y_w^{(t)} = \sum_{j=1}^{J} \sum_{i=1}^{N_F} \sqrt{P_{i,j}{}^{(t)}} h_{i,j}^{(t)} x_{i,j}^{(t)} + n_0, \tag{4.1}$$

where $x_{i,j}$, $h_{i,j}$, and $P_{i,j}$ denote the transmitted signal of $i$-th GF user on sub-channel $j$, channel gain of $i$-th GF user on sub-channel $j$, and transmit power of $i$-th GF user on sub-channel $j$, respectively. Here, $n_0$ represents the additive Gaussian noise with variance $(0, \sigma^2)$. The channel decoding order is, $P_{i,j}^{(t)} h_{i,j}^{(t)} \geq \cdots \geq P_{N_F,j}^{(t)} h_{N_F,j}^{(t)}$. The SINR for GF user $i \in \mathcal{F}$ can be given as follows:

$$SINR_{i,j}^{(t)} = \frac{P_{i,j}^{(t)} |h_{i,j}|^{2^{(t)}}}{\sum_{i=i+1}^{N_F} P_{i+1,j}^{(t)} |h_{i+1,j}|^{2^{(t)}} + \sigma^2}. \tag{4.2}$$

The data rate of each GF user is calculated as follows:

$$R_{i,j}^{(t)} = \hat{B} \log \left(1 + SINR_{i,j}^{(t)}\right) \geq \varepsilon_{\mathcal{F}}, \tag{4.3}$$

where the bandwidth of sub-channel $j$ is denoted as $\hat{B}$ and the threshold target data rate for $\mathcal{F}$ users is denoted as $\varepsilon_F$.

The GF user EE is calculated as follows:

$$EE_{\mathcal{F}}^{(t)} \triangleq \frac{\sum_{j=1}^{J} \sum_{i=1}^{N_F} R_{i,j}^{(t)}}{\varsigma^{(t)} + \vartheta_{\mathcal{F}}^{(t)}}, \tag{4.4}$$

where $\varsigma^{(t)} = \sum_{j=1}^{J} \sum_{i=1}^{N_F} p_{i,j}^{(t)}$ and $\vartheta_{\mathcal{F}}^{(t)}$ is the circuit power consumed by $\mathcal{F}$ users similar to [97].

In the next time slot $(t+1)$, the BS receives the combined signal from the CHs and other GB users as follows:

$$y_{BS}^{(t+1)} = \sum_{j=1}^{J} \sum_{w=1}^{N_W} \sqrt{P_{w,j}{}^{(t+1)}} g_{w,j}^{(t+1)} x_{w,j}^{(t+1)} + n_0, \tag{4.5}$$

where $x_{w,j}$, $g_{w,j}$, and $P_{w,j}$ represent the transmitted signal, channel gain, and transmit power of $w$-th GB user, respectively. In this part, the channel gain based decoding order at the BS; that is, the GB users with strong channel gain will be decoded first. Therefore, the first stage of SIC order follows $\mathcal{G} = \{g_{1,j} \geq g_{2,j} \geq \cdots \geq g_{N_W,j}\}$.

Similarly, the SINR for GB users can be shown as follows:

$$SINR_{w,j}^{(t+1)} = \frac{P_{w,j}^{(t+1)} |g_{w,j}|^{2^{(t+1)}}}{\sum_{w=w+1}^{N_W} P_{w+1,j}^{(t+1)} |g_{w+1,j}|^{2^{(t+1)}} + \sigma^2}. \tag{4.6}$$

To calculate the data rate of GB user, we use equation (4.7) which shown as follows:

$$R_{w,j}^{(t+1)} = \hat{B} \log \left(1 + SINR_{w,j}^{(t+1)}\right) \geq \varepsilon_{\mathcal{W}}, \tag{4.7}$$

where the threshold of the target data rate for $\mathcal{W}$ users is denoted by $\varepsilon_{\mathcal{W}}$.

The EE of GB users in time slot $(t+1)$, we have

$$EE_{\mathcal{W}}^{(t+1)} \triangleq \frac{\sum_{j=1}^{J} \sum_{w=1}^{N_W} R_{w,j}^{(t+1)}}{\varrho^{(t+1)} + \vartheta_{\mathcal{W}}^{(t+1)}}, \tag{4.8}$$

where $\varrho = \sum_{j=1}^{J} \sum_{w=1}^{N_W} p_{w,j}$ and $\vartheta_{\mathcal{W}}$ is the circuit power consumed by $\mathcal{W}$ users. Based on equations (4.4) and (4.8) , the EE of the system can be given as follows:

$$EE = EE_{\mathcal{F}}^{(t)} + EE_{\mathcal{W}}^{(t+1)}. \tag{4.9}$$

### 4.2.2 Cluster Head and Sub-channel Selection (GF Users)

In time slot $t$, each GF user is allowed to select at most one GB user as a cluster head and one sub-channel. GF users send their signals to the nearest cluster head, and the cluster head (GB) sends its signal along with the GF users' signals to the BS. By sending the signal to the nearest cluster head, GF users can save energy compared to the traditional method of sending it directly to the BS.

The following variable is used for sub-channel selection:

$$b_{i,j}^{(t)} = \begin{cases} 1, \text{if } i\text{-th GF user selects sub-channel } j \\ 0, \text{otherwise.} \end{cases} \tag{4.10}$$

### 4.2.3 Sub-channel Selection for GB Users

We use the following binary variable for GB users to select sub-channel as follows:

$$m_{w,j}^{(t+1)} = \begin{cases} 1, \text{if } w\text{-th user selects sub-channel } j \\ 0, \text{otherwise.} \end{cases} \tag{4.11}$$

### 4.2.4 Problem Formulation

Our objective is to maximize the EE by optimizing the parameters $m$, $b$, and $P$. The proposed model achieves this by leveraging the advantages of designating the GB user as the cluster head for the nearest GF user. This approach allows us to maximize the EE. Additionally, we employ multiple algorithms, namely DRL and PPO, to reduce system complexity. Specifically, we utilize the DRL algorithm for low hardware IoT users and the PPO algorithm for high hardware equipment, such as BS. Therefore, the optimization problem can be formulated as follows:

$$\underset{m,b,P}{\text{maximize}} \sum_{t=1}^{T}\sum_{j=1}^{J}\sum_{w=1}^{N_W}\sum_{i=1}^{N_F} EE(t) \tag{4.12}$$

$$\text{s.t.} \quad P_{i,j}^{(t)}, P_{w,j}^{(t+1)} \leq P_{max}, \forall w, i, j, t, \tag{4.12a}$$

$$\sum_{j=1}^{J} b_{i,j}^{(t)} \in \{1, 0\}, \quad \forall i, t, \tag{4.12b}$$

$$\sum_{j=1}^{J} m_{w,j}^{(t+1)} \in \{1, 0\}, \quad \forall w, t, \tag{4.12c}$$

$$\sum_{j=1}^{J} R_{w,j}^{(t+1)} \geq \varepsilon_{\mathcal{W}}, \quad \forall w, t, \tag{4.12d}$$

$$\sum_{j=1}^{J} R_{i,j}^{(t)} \geq \varepsilon_{\mathcal{F}}, \quad \forall i, t, \tag{4.12e}$$

where (4.12a) is the maximum transmit power limit of users. Constraints (4.12b) and (4.12c) show that GF and GB users can select only one sub-channel in a given time slot $t$. Constraints (4.12d) and (4.12e) represent the minimum required data rate of GB and GF users for successful SIC, respectively.

## 4.3 Semi-Centralized ML Framework for EE

ML algorithms for resource management are based on a centralized or distributed framework. In particular, in a centralized framework, a central entity (e.g., BS) is responsible for resource allocation, whereas, in the decentralized framework, resource allocation is handled by multiple agents (e.g., IoT users). The downside of the former is increased computational complexity (CC) arising from the overwhelming demand, and the downside of the former is the lengthy learning/training time required to converge to optimality as a result of non-stationarity. To alleviate these challenges, we have designed a semi-centralized framework that minimizes the CC and reduces the learning time. The work flow of the proposed algorithm is given in Fig. 4.2.

This model can be used for OMA-NOMA scenarios for some applications, for example, the GB users (eMBB user) can transmit using OMA, whereas GF users can transmit using NOMA (mMTC user). Next, we formulate the EE problem as MDP problem with the semi-centralized framework.

### 4.3.1 MDP Elements with a Semi-Centralized Framework

An MDP consists of a tuple of $\big(\mathcal{S}, \mathcal{A}, \hat{\mathcal{N}}$, and $\mathcal{R}\big)$, where $\mathcal{S}$ is the set of states, actions are denoted by $\mathcal{A}$, $\hat{\mathcal{N}}$ denote the total number of agent(s)(BS, GF users), and $\mathcal{R}$ is the reward function. To start the learning process, RL agents interact with the environment to maximize the long-term reward following some policy $\pi$.

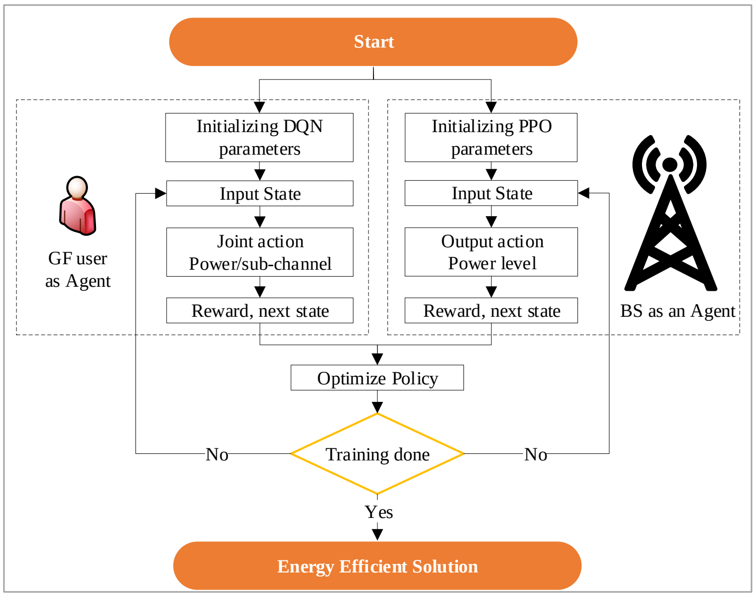

Start
Initializing DQN parameters
Input State
Joint action Power/sub-channel
Reward, next state
GF user as Agent
Initializing PPO parameters
Input State
Output action Power level
Reward, next state
BS as an Agent
Optimize Policy
Training done
No
No
Yes
Energy Efficient Solution

Figure 4.2: An illustration of flow chart for semi-centralized algorithm with two types of agents (BS and GF users)

- **Agent(s):** $\underbrace{\text{The BS}}_{\text{Centralized Part}}$ and $\underbrace{\text{GF users}}_{\text{Decentralized part}}$
- **State:** $\underbrace{\text{Channel gain (GB)}}_{\text{Centralized part}}$ and $\underbrace{\text{data rate (GF)}}_{\text{Decentralized part}}$
- **Action (BS):** $\underbrace{\text{Transmit power}}_{\text{Centralized part}}$
- **Action (GF user):** $\underbrace{\text{Sub-channel, transmit power, and CH}}_{\text{Decentralized part}}$
- **Reward:** The BS as an agent receives the EE of the GB users as a reward, whereas, the $i$-th GF user receives the EE of the GF users as a reward signal, as given below.

$$r_i^{(t)} = \begin{cases} EE_{\mathcal{F}}^{(t)}, & \text{if } EE_{\mathcal{F}}^{(t)} \geq EE_{\mathcal{F}}^{(t-1)} \\ 0, & \text{otherwise,} \end{cases} \tag{4.13}$$

**Algorithm 2** Semi-Centralized Framework for EE NOMA Systems

```
Initialize hyperparameter {MA-DQN}
for Episode = 1: $N_e$ do
    for iteration at time step (t) = 1: $T_e$ do
        for agent = 1: $I$ do
            Input $s_i^{(t)}$, take $a_i^{(t)}$, receive $r_i^{(t)}$ using (4.13) and $s_i^{(t+1)}$
            Store $s_i^{(t)}, a_i^{(t)}, r_i^{(t)}, s_i^{(t+1)}$ to replay memory
        end for
    end for
    From the memory, the agent sample mini-batches and use (4.16) to minimize the loss.
end for
Initialize policy parameters {PPO}
for Episode = 1: $N_e$ do
    for actor = $1, 2, \dots, N_a$ do
        Run policy $\pi_{\theta_{old}}$ for $T_e$ time steps
        Calculate advantage estimates $\hat{A}_1, \dots, \hat{A}_T$
    end for
    Optimize $\mathcal{L}$ (4.17) w.r.t $\theta$
end for
```

$$r_{BS}^{(t+1)} = \begin{cases} EE_{\mathcal{W}}^{(t+1)}, \text{ if } EE_{\mathcal{W}}^{(t+1)} \geq EE_{\mathcal{W}}^{(t)} \\ 0, \qquad\qquad \text{otherwise.} \end{cases} \tag{4.14}$$

The PPO uses two DNN and handles a continuous action space, which increases the complexity; therefore, the PPO is used on the BS side. In contrast, the IoT users are resource and computation constrained and can handle discrete actions; hence, such algorithms cannot be applied to IoT users.

For the decentralized part (GF users act as agents), we define a Q-function as the expected cumulative discounted reward to find the optimal policy $\pi^*$, which can be given as follows:

$$Q_i^{\pi}(s_i, a_i) = \mathbb{E}^{\pi}[\hat{\mathcal{R}}^{(t)} \big| {s_i}^{(t)} = s, {a_i}^{(t)} = a], \tag{4.15}$$

where $\hat{\mathcal{R}}$ is the discounted reward $\hat{\mathcal{R}}$=$\sum_{n=0}^{N_e} \beta^n r^{(t+n+1)}$.

To train the Q-network, stochastic gradient descent (SGD) is used to update the weights and minimize the error rate between the target Q-network and the primary Q-network.

$$\mathcal{L}(\theta) = (y_i^{(t)} - Q_i^{(t)}(s_i^{(t)}, a_i^{(t)}))^2, \tag{4.16}$$

where $y_i^{(t)} = r_i^{(t)} + \max_{a_i} Q(s_i^{(t+1)}, {a_i}^{(t+1)}; \theta)$.

For the centralized part (agent BS), we apply the PPO algorithm to find the optimal transmit power for GB users. The PPO is a policy gradient method that utilizes the actor-critic method and can be used in environments with continuous action space. In the stochastic policy, the actor maps an observation to an action, and the critic calculates the reward for the given observation. A stochastic gradient ascent optimizer is used to update the policy, and an SGD technique is used to fit the value function. As shown in equation 4.17, the loss function of the proposed model can be calculated as follows:

$$\mathcal{L}(\theta) = \hat{\mathbb{E}}^{(t)}[min\big(r^{(t)}(\theta)\hat{A}^{(t)}, clip\big(r^{(t)}(\theta)1-\epsilon, 1+\epsilon\big)\hat{A}^{(t)}\big)], \tag{4.17}$$

where $\hat{\mathbb{E}}^{(t)}$ represents the empirical expectation over time steps and $\theta$ represents the policy parameter. At time step $(t)$, $\hat{A}^{(t)}$ shows as estimated advantage, the reward $r^{(t)}$ denotes the ratio of the probability under both new and old policies. This equation has two parts; first it minimizes the loss of conservative policy iteration $(min(r^{(t)}(\theta)\hat{A}^{(t)})$, and in the second part, we have $(\text{clip}(r^{(t)}(\theta)1-\epsilon, 1+\epsilon)\hat{A}^{(t)})$, where we clip the policy ratio between $1+\epsilon$ and $1-\epsilon$.

### 4.3.2 Proposed Semi-Centralized Algorithm

To maximize the EE of GF users, agents, i.e., BS or GF users learn the optimal policy. **Algorithm 2** shows the details of semi-centralized model. In the distributed part, we initialize the network and training parameters before the start of agents training. All agents (GF users) jointly explore the environment using a $\epsilon$-greedy policy. The agents

Table 4.1: Resources saving comparison between centralized and semi-centralized algorithm with increasing number of IoT users

| **GB users** | **GF users** | $\hat{\mathcal{N}}$ **users** | **Required Operations (Proposed)** | **Required Operations (Centralized)** | **Resource Saving** |
|---|---|---|---|---|---|
| 2 | 2 | 4 | 10 | 64 | 84.38% |
| 2 | 3 | 5 | 11 | 125 | 91.20% |
| 3 | 2 | 5 | 29 | 125 | 76.80% |
| 3 | 3 | 6 | 30 | 216 | 86.11% |
| 4 | 4 | 8 | 68 | 512 | 86.72% |
| 10 | 10 | 20 | 1,010 | 8,000 | 87.38% |
| 20 | 20 | 40 | 8,020 | 64,000 | 87.47% |
| 50 | 50 | 100 | 125,000 | 1,000,000 | 87.50% |
| 100 | 100 | 200 | 1,000,100 | 8,000,000 | 87.50% |

receive states from the environment, and they take a joint action. Based on the action taken, agents obtain a reward and next state from the environment. All agents save experiences to their replay memories (line 6). To train the primary network, all the agents randomly sample mini-batches from the memory and compute the loss (line 9). For the centralized part (line 11), first, we initialize the policy parameters. We run the policy $\pi_{\theta_{old}}$ for $T_e$ time steps to calculate the advantage estimates. Finally, we calculate the loss with respect to $\theta$ using a mini-batch of size $M$ and update $\theta_{old}$ with $\theta$ (line 17).

### 4.3.3 Complexity of the Semi-Centralized

The complexity results from a number of GB $N_W$ and GF $N_F$ users connecting to BS via sub-channels $J$. The benchmark which is known as centralized framework can calculate the CC as $\mathcal{O}\big[N_e \times T_e(\hat{\mathcal{N}}^J)\big]$. Where the idea in the proposed model is to separated the agent(s) into two different groups namely distributed and centralized. This will help the agent if it BS to decide the channel for all GB users and each GF user (agent) is responsible to chose his own channel. The CC is calculated as $\hat{\mathcal{N}}$ denote the number of GB users and GF users. At each time step $t$, the CC of the proposed algorithm is given by, $\mathcal{O}\big[N_e \times T_e\big((N_W^J) + N_F\big)\big]$. For example, if we have five GB users and five GF users in a centralized framework, the complexity is increased exponentially. On the other hand, in our proposed algorithm, the complexity is distributed among the BS and GF users. Therefore, the different between the number of sub-channel in the centralized is control

Table 4.2: Training and simulation parameters for networks and ML algorithms

| **Parameter** | **Value** |
|---|---|
| GB users | (3 – 15) |
| GF users | (3 – 15) |
| Power levels | $[0.1, ..., 0.9]$ W |
| Sub-channels | 3 |
| Sub-channel bandwidth | 10 KHz |
| $\alpha$ | 2.8 |
| Min rate (GB users) | 10 bps/Hz |
| Episodes | 300 |
| Min rate (GF users) | 4 bps/Hz |
| Learning rate | 0.001 |
| DNN activation | ReLU |
| Optimizer | Adam |

be only BS.

Table 4.1 shows the complexity increasing in the centralized model and the proposed one. With increase number of IoT users, the proposed model can save more resource compare to the centralized model.

## 4.4 Simulation Results

In chapter 3, we optimize the resource and improve the sum data rate for downlink users. BS controls the allocation of resources to users. On the other hand, in chapter 4, we optimize resource allocation and EE for two different types of users, namely GB and GF. All techniques used are ML-based to enhance resource allocation in NOMA networks.

In this section, we evaluate the performance of both the GB and GF users. The parameters given in Table 4.2 are used to obtain the simulation results.

### 4.4.1 Proposed Algorithm Convergence Analysis

Fig. 4.3 shows the convergence of the PPO algorithm at the BS side to allocate the power to GB users and the MA-DQN algorithm for the GF users to find the optimal power level. The centralized agent, i.e., the BS, finds the optimal power level for each

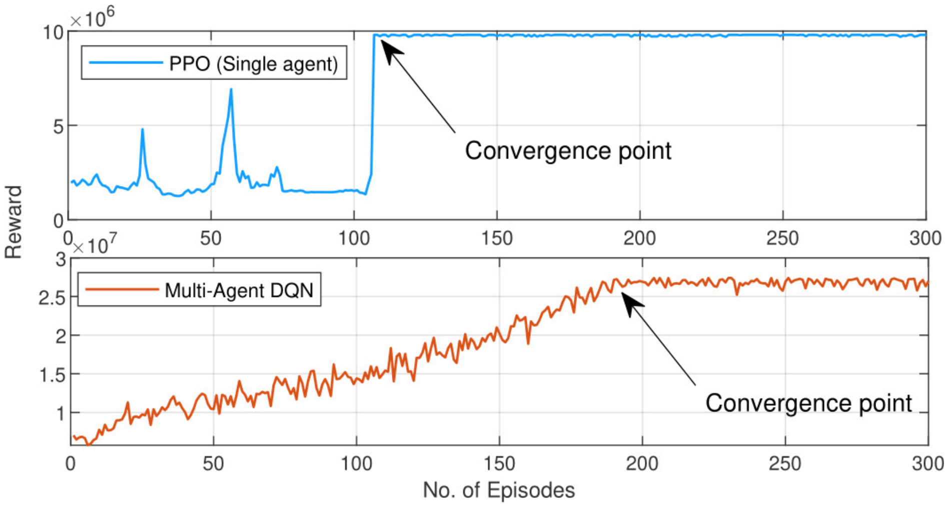


Figure 4.3: Shows the convergence of the PPO and MA-DQN

GB user after 100 episodes, as seen in the top sub-figure of Fig. 4.3. Compared with the decentralized MA-DQN, the PPO converges quickly. However, for a large number of GB users, the PPO may require more training time because of the continuous action space. The bottom sub-figure shows the convergence of MA-DQN. There is a fluctuation in the reward because the actions of one agent affect other agents in the environment. Therefore, MA-DQN requires more episodes for convergence.

### 4.4.2 Performance Comparison of the Proposed Algorithm with Benchmark Schemes

Fig. 4.4 provides a comparison of the EE of both types of users (GB and GF) with other methods. As shown in the upper sub-figure, the EE of the propose algorithm shows better performance as compared to the benchmark scheme (random power allocation and fixed power allocation methods). Because the BS identifies the accurate power levels according to the user channel gain, maintaining the QoS requirements of those GB users with a minimum power consumption. In contrast, in the other two methods, users transmit power without considering the channel gain, which increases intra-cluster interference, hence recording a low EE. The EE of GF users is depicted in the bottom sub-figure of Fig. 4.4. For comparison, we use the conventional GF method as a benchmark, where

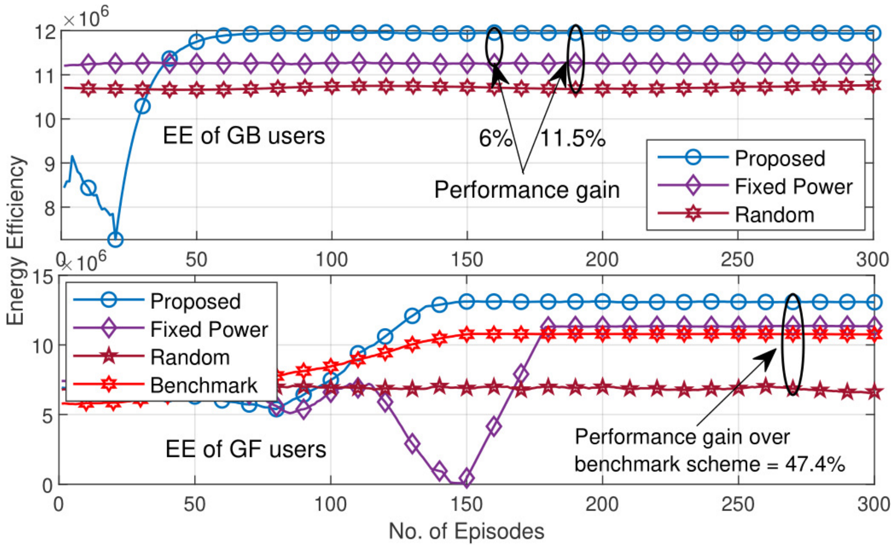


Figure 4.4: Performance comparison of the proposed scheme with fixed power, random power, and the benchmark scheme

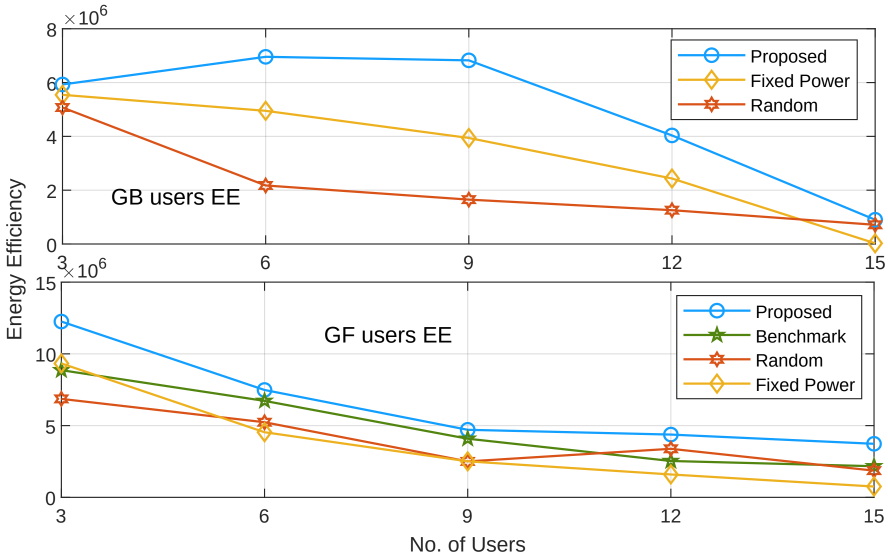


Figure 4.5: Energy efficiency with different IoT users

the users directly transmit their data to the central BS. It is observed that our proposed scheme performs well in comparison to random power and fixed power selection methods. Unlike the benchmark scheme, the GF users in our proposed scheme transmit their data to the nearest cluster head, which requires a minimum transmit power and enhances the EE.

### 4.4.3 Energy Efficiency with Different IoT Users

Fig. 4.5 displays a further analysis in both centralized and decentralized algorithms EE with different number of users. In top of Fig. 4.5, three different convergences represents the proposed, fixed power and random power in centralized algorithm where the BS is the agent. The blue curve (the proposed algorithm) shows better EE compared to other techniques (fixed and random powers). Moreover, with the increased number of GB users, the achievable EE with the agent (BS) becomes lower and vice versa. In the bottom of Fig. 4.5, four different convergences (decentralized) are proposed i.e., proposed algorithm, fixed power, random power, and benchmark. The blue curve (proposed algorithm) shows the best EE among all convergence. With the decrease number of users the EE become better and vice versa. Even with high number of GF users, the proposed model perform the best. The benchmark here is the traditional method of directly sending the signal from GF users to the BS. Where both random and fixed power are considered as alternative power allocation methods to test the performance of the proposed model. Finally, with technique of group-head, the proposed model success to achieve high EE.

### 4.4.4 Energy Efficiency vs Increasing GB Users

Fig. 4.6 shows the trade-off between the number of GB users and EE of both types of users. It can be observed that the EE of GF users increases with the number of GB users. Because the GF users have an increased choice in cluster head selection, they transmit to their nearest cluster head. On the other hand, as the number of GB users increases, it decreases the EE of GB users because this increases the number of users in each cluster, which increases the intra-cluster interference. To achieve the required data rate threshold with increased interference, GB users are required to transmit with high transmit powers.

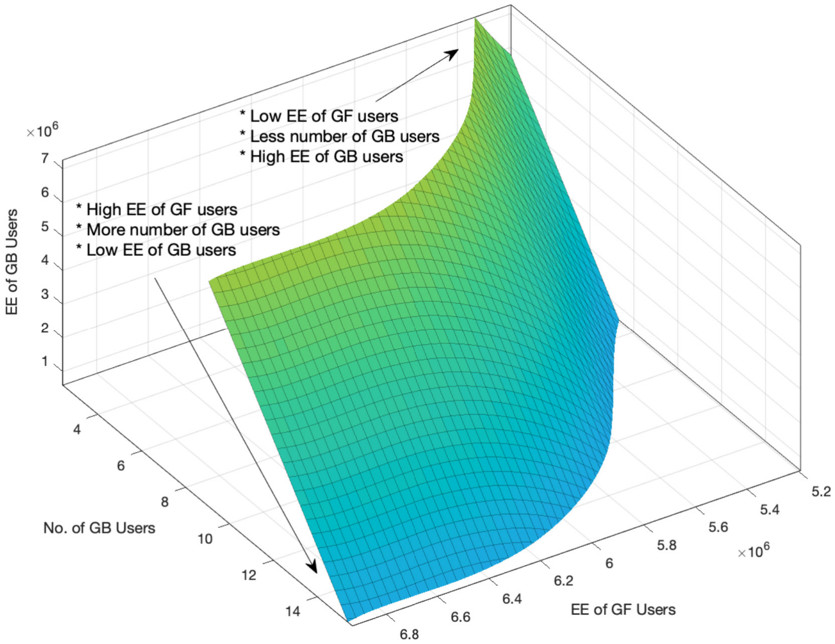


Figure 4.6: The EE of GF and GB users with respect to increasing GB users

### 4.4.5 System Energy Efficiency Comparison

Fig. 4.7 compared the proposed method with the centralized and distributed framework in terms of EE. Fig. 4.7 represents the total network EE with respect to different number of episodes. It is concluded that the fully centralized method provides the highest EE. However, as the number of episodes increases the EE of the proposed method approaches the EE of centralized method. Because distributed methods need a long learning time to fully explore the environment. Therefore, the advantage of the proposed method is that the agent can achieve similar performance (in terms of EE) to the centralized method with lower complexity. At all time, the network EE for distributed method is always the lowest.

### 4.4.6 Complexity Comparison

Fig. 4.8 compared the proposed method with both centralized and distributed framework in terms of CC with different number of IoT users. When the number of clients (IoT users) increases, the proposed model receives less CC than the centralized method. From Fig. 4.8, it can be seen that the complexity of the centralized model is increasing

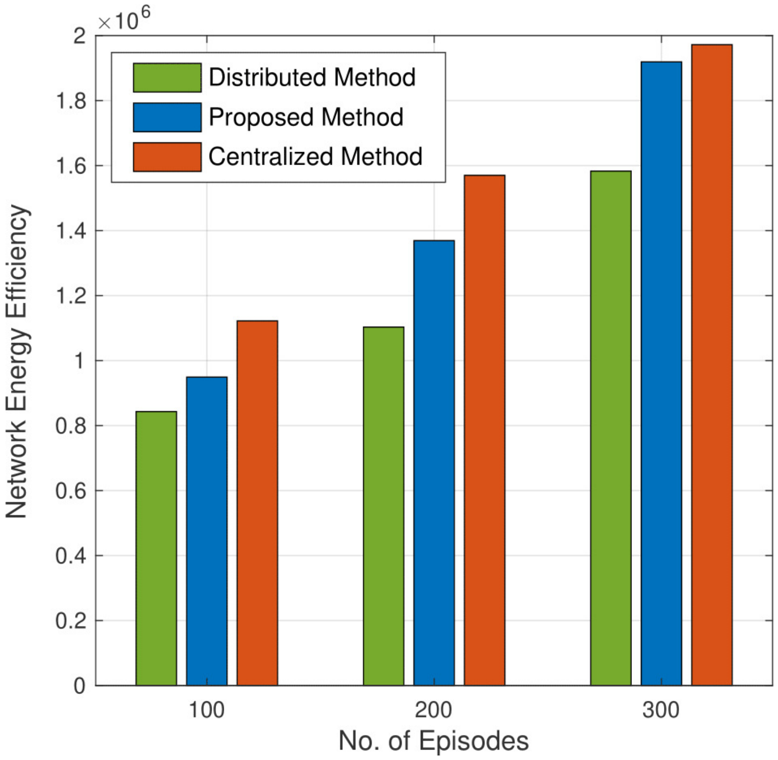


Figure 4.7: Network energy efficiency with different number of episodes

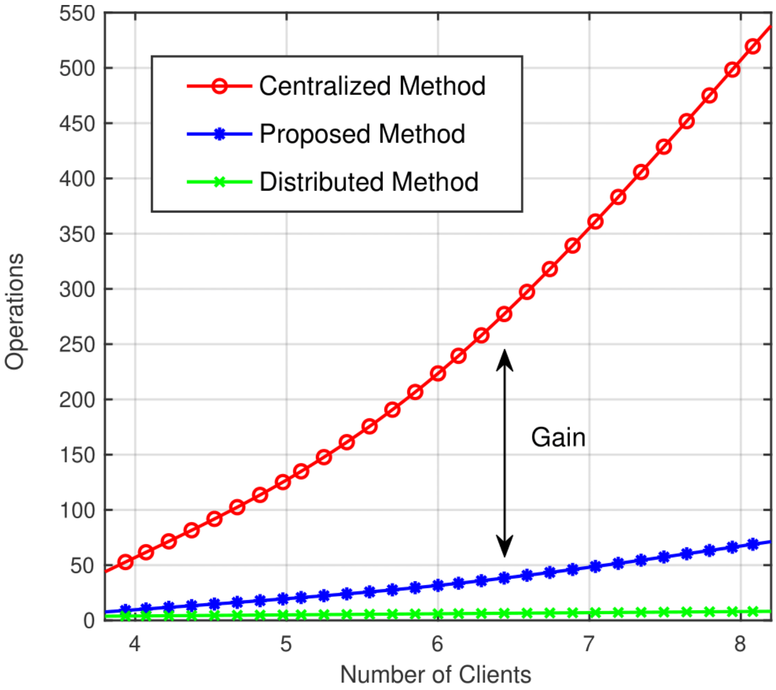


Figure 4.8: An illustration of operations with increase number of IoT users

exponentially as the number of users increases. In our proposed framework, the complexity is distributed between the BS and GF users. In a fully distributed model, all the users are independently searching for resources without any centralized entity (BS), which reduces the CC. However, a fully distributed framework requires a long learning time to reach the Nash Equilibrium. On the other hand, the centralized method can easily find the optimal resources for users but at the cost of a high CC.

## 4.5 Chapter Summary

In this chapter, we have proposed a low-complexity semi-centralized framework for NOMA networks to avoid the disadvantages of fully centralized and fully distributed systems. The proposed scheme improves the EE of GB and GF users and outperforms the fixed and random power allocation methods. The EE of GF users surpasses the EE of the conventional GF scheme where no group head exists.

# Chapter 5

# Soft Actor Critic Based Resource Allocation in Simultaneously Downlink and Uplink NOMA Networks

## 5.1 Introduction

In contrast to downlink NOMA (discussed in chapter 3) and uplink NOMA (discussed in chapter 4), simultaneous uplink and downlink NOMA use full-duplex communication, where transmissions occur simultaneously over the same frequency. However, it presents notable challenges, including self-interference, in which the base station's own transmission interferes with its reception, and increased signal processing complexity for decoding overlapping signals. Maintaining network quality and performance requires efficient management of transmit power and dynamic resource allocation. To address these challenges, the utilization of machine learning for resource and power management, accurate beamforming, and efficient network coordination are potential solutions [39, 46, 50, 90].

These strategies are crucial for maximizing the advantages of simultaneous uplink and downlink NOMA, making it a promising yet intricate advancement in wireless communication technology. Therefore, we proposed a SAC model for efficient resource allocation in simultaneous uplink and downlink NOMA transmission. SAC, with its robustness in handling complex and dynamic environments, offers a powerful approach to solving the challenges of simultaneous uplink and downlink NOMA transmission. Its ability to learn optimal policies for interference management, signal decoding, power control, and resource allocation makes it well-suited for improving the efficiency and performance of these advanced wireless communication systems.

### 5.1.1 Contributions

In contrast to the distinct difficulties faced by conventional uplink and downlink broadcasts, simultaneous uplink and downlink transmission raises the crucial and intricate problem of self-interference. To fully utilize simultaneous transmission systems, such as those anticipated in the upcoming 5G and 6G networks, it is crucial to efficiently manage interference. The main contributions of this chapter are listed below.

- Novel multi-downlink IoT users and multi-uplink backscatter devices are considered. The aim is to maximize the sum rate of backscatter users by jointly optimizing the transmit power for downlink users and the reflection coefficient for backscatter devices subject to the QoS requirements of downlink IoT users.

- The optimization problem of maximizing the sum rate is formulated as MDP problem, which is extremely difficult and complex to be solved by conventional optimization approaches. Therefore, the formulated MDP is solved using the RL-based model-free SAC algorithm.

- The proposed SAC algorithm uses the online optimization strategy with an entropy regularization process to effectively explore and exploit the dynamic BAC-NOMA environment to solve the formulated problem optimally.

- Numerical results indicate that the suggested algorithm outperforms the conventional optimization (benchmark) method in terms of the achievable sum rate of uplink backscatter devices. With a large number of iterations, the network with multiple downlink users obtains a higher reward. Moreover, with different numbers of backscatter devices, the proposed algorithm outperforms the benchmark scheme and BAC with OMA. Furthermore, our proposed algorithm improves sum rate efficiency under different self-interference coefficients and noise levels. As a final step, we evaluate and demonstrate the sum rate efficiency of the proposed algorithm with different QoS requirements and cell radii.

## 5.2 System Model And Problem Formulation

The practical scenario for backscatter communication can be an agricultural farm or an industrial floor [75], where the backscatter sensors are deployed to carry out the application-specific tasks. For example, the sensor's node can estimate the water stress of a plant by finding the difference in temperature between the leaf and the atmosphere. As shown in Fig. 5.1, the left side represents the environment of backscatter and NOMA network, where the FDBS is connected to downlink and uplink devices simultaneously. The blue devices represent the downlink devices where these devices receive signals from the FDBS. The red devices are the backscatter devices that send signals to FDBS in an uplink manner. The red arrows represent the channel gain from FDBS to all devices. The blue arrows represent the channel gain between the uplink backscatter devices and the downlink devices. The right side of the figure (at the top) shows the handshake between FDBS and the downlink user, while at the bottom is the handshake between the uplink backscatter devices and the FDBS. Next sub-sections illustrate the proposed system model and problem formulation.

### 5.2.1 Single Downlink User and Multiple Uplink Backscatter Devices

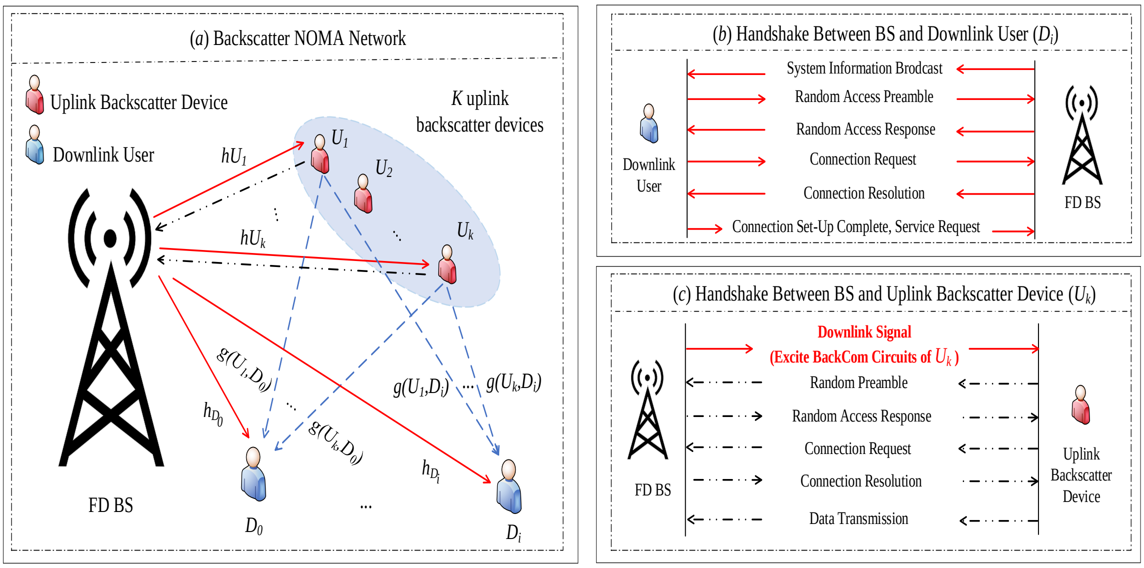

(a) Backscatter NOMA Network
Uplink Backscatter Device
Downlink User
K uplink backscatter devices
hU1
U1
U2
Uk
hUk
g(U1,D0)
hD0
g(Uk,D0)
g(U1,Di)
g(Uk,Di)
hDi
FD BS
D0
Di
(b) Handshake Between BS and Downlink User (Di)
Downlink User
System Information Brodcast
Random Access Preamble
Random Access Response
Connection Request
Connection Resolution
Connection Set-Up Complete, Service Request
FD BS
(c) Handshake Between BS and Uplink Backscatter Device (Uk)
FD BS
Downlink Signal
(Excite BackCom Circuits of Uk )
Random Preamble
Random Access Response
Connection Request
Connection Resolution
Data Transmission
Uplink Backscatter Device

Figure 5.1: An illustration of the environment and handshakes process of the BAC-NOMA network

In this sub-section, we considered a BAC-NOMA network where we have a FDBS, downlink users known as $D_0$, and the uplink backscatter devices known as $U_k$, where the integer $U_k \in \{1, \cdots, U_K\}$. We assume in each time slot that both $D_0$ and $U_K$ users are simultaneously served. The BS transmits the downlink signal to the downlink user $D_0$, which excites the circuits of uplink backscatter devices. Based on the signal received signal from the BS, the uplink backscatter devices then modulate and reflect the incident signal via a reflection coefficient $\eta_k$ (adjustable parameter and $\eta_k \in [0, 1])$ [98].

The signal received at the uplink backscatter device $U_k$ from the BS is denoted by $\sqrt{P_{D_0}^{(t)}} h_k^{(t)} x_{D_0}^{(t)}$, where $P_{D_0}$, $h_k$, and $x_{D_0}$ are the downlink transmit power for downlink user $D_0$, the channel gain between the BS and $U_k$, and the signal for downlink user $D_0$, respectively. The signal reflected by uplink backscatter device $U_k$ is expressed as $\sqrt{P_{D_0}^{(t)} \eta_{U_k}^{(t)}} h_{U_k}^{(t)} x_{D_0}^{(t)} x_{U_k}^{(t)}$, where $x_{U_k}$ is the backscatter signal from device $U_k$. The channel gain is characterized by large-scale path loss and small-scale multi-path fading, as considered in [62].

Based on the aforementioned expressions, the combined signal received at the BS from the $U_k$ uplink backscatter devices can be expressed as:

$$y_{BS}{}^{(t)} = \sum_{U_k=1}^{U_K} h_{U_k}^{2}{}^{(t)} \sqrt{P_{D_0}^{(t)} \eta_{U_k}^{(t)}} x_{D_0}^{(t)} x_{U_k}^{(t)} + x_{SI}^{(t)} + n_{BS}, \tag{5.1}$$

where $x_{SI}^{(t)}$ is based on the complex Gaussian distribution and is defined as $x_{SI} \sim \mathcal{CN}(0, \varphi P_{D_0} |h_{SI}|^2)$ [99]. The $h_{SI}^{(t)}$ shows the self-interference channel that is based on the complex Gaussian distribution, that is $h_{SI} \sim \mathcal{CN}(0,1)$. The $n_{BS}$ represents the noise at the BS. The amount of FD residual self-interference ($\varphi$) is defined as ($0 \leq \varphi \ll 1$) [62].

At the same time, the downlink user $D_0$ receives the signal from the BS with added interference from the uplink backscatter devices, as the downlink user utilizes the same time slot with the uplink backscatter devices. Consequently, downlink user $D_0$ receive the signal $y_{D_0}$ as follows:

$$\begin{aligned} y_{D_0}^{(t)} =& \underbrace{h_{D_0}{}^{(t)} \sqrt{P_{D_0}^{(t)}} x_{D_0}^{(t)}}_{\text{Desired Signal}} + \\ & \underbrace{\sum_{U_k=1}^{U_K} g_{U_k}{}^{(t)} h_{U_k}{}^{(t)} \sqrt{P_{D_0}^{(t)} \eta_{U_k}^{(t)}} x_{D_0}^{(t)} x_{U_k}^{(t)}}_{\text{Intra-Cell } (U_k) \text{ Interference}} + \underbrace{n_{D_0}}_{\text{Noise}}. \end{aligned} \tag{5.2}$$

The first part of (5.2) is the intended signal for user $D_0$ from the BS, and the second part represents the interference from uplink backscatter devices. The channel gain between the downlink user and BS is denoted as $h_{D_0}^{(t)}$. Moreover, the channel gain between the uplink backscatter device and downlink user is denoted as $g_{U_k}{}^{(t)}$. Finally, the noise is denoted as $n_{D_0}$.

The sum rate for uplink backscatter devices that is achievable by BAC-NOMA transmission can be given as:

$$R_{\text{sum}}^{(t)} = \log\Big(1 + \frac{\sum_{U_k=1}^{U_K} |h_{U_k}|^{4(t)} \eta_{U_k}^{(t)} P_{D_0}^{(t)} |x_{D_0}|^{2(t)}}{\varphi^{(t)} P_{D_0}^{(t)} |h_{SI}|^{2(t)} + \sigma^2}\Big), \tag{5.3}$$

where in this system model we assume that noise for both BS and downlink user $D_0$ have the same power; it is denoted as $\sigma^2$. Finally, the data rate for the downlink user is calculated as:

$$R_{D_0}^{(t)} = \log\Big(1 + \frac{{P_{D_0}}^{(t)}|h_{D_0}|^{2^{(t)}}}{\sum_{U_k=1}^{U_K} |h_{U_k}|^{2^{(t)}}|g_{U_k}|^{2^{(t)}}\eta_{U_k}^{(t)}P_{D_0}^{(t)} + \sigma^2}\Big). \tag{5.4}$$

### 5.2.2 Multiple Downlink Users and Uplink Backscatter Devices

In this sub-section, we consider more general scenario where a single FDBS simultaneously serves multiple downlink users and multiple uplink backscatter devices, as shown in Fig. 5.1. Without losing the generality, perfect CSI is available at the BS. Downlink users are defined as $D_i$, where the integer $D_i \in \{0, \cdots, D_I\}$, and the first downlink user $D_0$ is considered to be in close proximity to the BS and has the strongest channel gain condition. In effect, the downlink user $D_1$ is far away from the BS and has a poor channel gain compared to $D_0$. Therefore, based on this description, the received signal given in (5.2) for multiple downlink users can be rewritten as:

$$\begin{aligned} y_D = & \underbrace{h_{D_0}^{(t)}\sqrt{P_{D_0}^{(t)}}{x_{D_0}}^{(t)}}_{\text{Desired Signal}} + \underbrace{\sum_{D_i \neq D_0} h_{D_i}^{(t)}\sqrt{P_{D_i}^{(t)}}x_{D_i}^{(t)}}_{\text{Intra-Cell } (D_i) \text{ Interference}} + \\ & \underbrace{\sum_{U_k=1}^{U_K}\sum_{D_i=0}^{D_I} {g_{U_k}}^{(t)}h_{U_k}^{(t)}\sqrt{P_{D_i}^{(t)}\eta_{U_k}^{(t)}}x_{D_i}^{(t)}x_{U_k}^{(t)}}_{\text{Intra-Cell } (U_k) \text{ Interference}} + \underbrace{n_D}_{\text{Noise}}, \end{aligned} \tag{5.5}$$

where $n_D$ is the noise, and $D_i$ is the $i$-th downlink user in the intra-cell interference part. Based on NOMA decoding order principles, the downlink user $D_0$ employs SIC to decode its own signal, and then downlink user $D_1$ is considered next as it has the second strongest channel gain.

The SINR is calculated as:

$$SINR_{D_0}^{(t)} = \frac{P_{D_0}{}^{(t)}|h_{D_0}|^2(t)}{I_d^{(t)} + I_u^{(t)} + \sigma^2}, \tag{5.6}$$

where $I_d$ is the interference from other downlink users and $I_d = \sum_{D_i \neq D_0} h_{D_i}^{(t)} \sqrt{P_{D_i}^{(t)}}$. The signal reflected by uplink backscatter devices is denoted as $I_u$, where $I_u$ is denoted as $I_u = \sum_{U_k=1}^{U_K} |h_{U_k}|^{2(t)} |g_{U_k}|^{2(t)} \eta_{U_k}^{(t)}$. The SINR for the last user $D_1$ is calculated as:

$$SINR_{D_1}^{(t)} = \frac{P_{D_1}{}^{(t)}|h_{D_1}|^{2(t)}}{I_u^{(t)} + \sigma^2}. \tag{5.7}$$

Below equation is used to calculate $D_i$ downlink user data rate:

$$R_{D_i}^{(t)} = \log\Big(1 + SINR_{D_i}^{(t)}\Big). \tag{5.8}$$

For the uplink backscatter devices, the signal received at the BS is calculated as:

$$y_{BS}{}^{(t)} = \sum_{U_k=1}^{U_K} \sum_{D_i=0}^{D_I} h_{U_k}^2{}^{(t)} \sqrt{P_{D_i}^{(t)} \eta_{U_k}^{(t)}} x_{D_i}^{(t)} x_{U_k}^{(t)} + x_{SI}^{(t)} + n_{BS}. \tag{5.9}$$

The decoding order is based on the strength of the signal received [46]. Therefore, the uplink backscatter device with higher received power will be decoded first. The sum data rate for all uplink backscatter devices is calculated as:

$$R_{\text{sum}}^{(t)} = \log\Bigg(1 + \frac{\sum_{U_k=1}^{U_K} \sum_{D_{i=0}}^{D_I} |h_{U_k}|^{4^{(t)}} \eta_{U_k}^{(t)} P_{D_0}^{(t)} |x_{D_0}|^{2^{(t)}}}{\varphi \sum_{D_i \neq D_0}^{(t)} P_{D_i}^{(t)} |h_{SI}|^{2^{(t)}} + \sigma^2}\Bigg). \tag{5.10}$$

### 5.2.3 Problem Formulation

We maximize the sum rate of uplink backscatter devices by optimizing the $P$ and $\eta_k$. Therefore, considering the QoS requirements of downlink users, the optimization problem for long-term communications over the time period $T$ can be formulated as follows:

$$\max_{P,\eta_{U_k}} \sum_{t=1}^{T} R_{\text{sum}}(t)/T, \tag{5.11a}$$

$$\text{s.t} : R_{D_i}^{(t)} \geq \hat{R}_{D_i}, \tag{5.11b}$$

$$0 \leq \eta_{U_k} \leq 1, \quad U_k \in U_K, \tag{5.11c}$$

$$0 \leq \eta_{U_k} P_{D_I}^{(t)} \leq P_{D_I}, \tag{5.11d}$$

$$0 \leq P_{D_I} \leq P_{max}, \tag{5.11e}$$

where constraint (5.11b) ensures the minimum QoS requirements for the downlink users, (5.11c) ensures the BAC reflection coefficient should be between 0 and 1, (5.11d) is the amount of power to be allocated to uplink device $U_k$ from the power allocated to downlink users, and (5.11e) represents the maximum transmit power limit for the downlink users. The optimization of the problem defined in (5.11a) is considered as an NP-hard problem. The detailed proof is provided in [100].

## 5.3 Intelligent BAC-NOMA Resource Allocation Systems

### 5.3.1 Markov Decision Process Model for BAC-NOMA

This sub-section shows the problem formulation to optimize resource allocation for BAC-NOMA users as a MDP. Choosing the SAC algorithm for this problem is considered a suitable solution since SAC is known for its stability and sample efficiency. It employs a stochastic policy, which can be more robust in complex and noisy environments where channel conditions and interference may vary significantly. Unlike other algorithms such as DDPG, which are more sensitive to perturbations due to their reliance on deterministic policies, selecting SAC resolves this issue. Furthermore, SAC introduces an entropy term into its objective function, encouraging exploration of the action space. In scenarios where exploration is crucial, SAC can be advantageous as it effectively balances exploration and exploitation. Moreover, SAC naturally handles continuous action spaces, which are prevalent in resource allocation problems within wireless networks. In the context of NOMA and backscatter networks, continuous resource allocation decisions are common, making SAC a suitable choice. Finally, SAC often requires fewer samples to converge compared to DDPG. This can help on dealing with limited resources or when rapid policy updates are necessary.

The significant elements of MDP are agent/s, states, actions, rewards, and transition probability. To begin the decision making process, the agent starts interacting with the specified environment (BAC-NOMA network in our case). To learn the policy $\pi$, the agent performs an action $a^{(t)}$ for a current state $s^{(t)}$ to move to the next state $s^{(t+1)}$. Based on the action, the agent receives the action evaluation (feedback) in the form of reward or punishment before moving to the next state $s^{(t+1)}$. These rewards and punishments are used to train the agent to optimize the action-selection process to find the optimal policy $\pi*$. When the training process is finished, all the actions and states are stored in the brain of an agent. That brain is in the form of a Q-table, denoted by $Q^T_\pi(s^{(t)}, a^{(t)})$. Traditional Q-learning is considered one of the solutions to the MDP problem by learning the best path for the state value optimization function.

The downside of this method is the requirement for a huge amount of memory to accommodate a Q-table for complex state space. Furthermore, DRL solves this problem by introducing a neural network to solve memory requirements. More research in this area will help improve the performance of DRL by introducing more neural networks, because neural networks solve the problem of a high-dimensional state or continuous state and action space.

The DDPG added deterministic policies to improve the learning process. It uses a replay buffer whereby it can draw samples from past experiences during the learning process, which sometimes is referred to as sample-efficient learning. However, obtaining good results with the DDPG algorithm is usually a challenge in some environments [89, 101]. SAC, an off-policy algorithm, introduces an entropy term to combat this instability. SAC aims to have this entropy high at each training step update to encourage exploration and therefore assign equal probabilities to all actions rather than repetitively assigning a high probability to a particular action.

Therefore, this work implements SAC to optimize the resource allocation for all uplink backscatter devices and to ensure the QoS for the downlink users. In summary, the proposed model solves the MDP with the help of SAC for long-term resource optimization.

## 5.3.2 BAC-NOMA-SAC Algorithm

### 5.3.2.1 A Design Overview

SAC is the extended version of DDPG that is from the family of RL algorithms. Traditional RL algorithms are based on simple Q-table and epsilon-based simple exploration/exploitation methods (greedy approaches), and therefore are prone to poor policy learning. To overcome these problems, SAC employs actor/critic networks and maximizes the entropy (unpredictability) of the best action that the agent can possibly take and thus maximizes the agent's long-term rewards. Additionally, SAC uses an off-policy formulation that is based on the previously stored data to enhance efficiency. The critic network assists the actor network to further improve the quality of the learning.

As shown in Fig. 5.2, the environment consists of two downlink users, referred to as $D_0$ and $D_1$, along with $K$ uplink backscatter devices and FDBS. This configuration is depicted in (a). Furthermore, the colored boxes represent all three SAC neural networks (b, c, and d). The first neural network receives the state information directly from the environment through the actor network (online), which is represented by the red box (b). Similarly, after processing the action, the output of the actor network becomes the input of the critic network, as depicted in the yellow box (c). To assess the quality of each action performed by the actor network online at each time step, the critic network evaluates the output of the actor network. Therefore, to ensure the quality of each action, another input of the critic network is also based on the state $s^{(t)}$. The quality of each action and state pair is determined by the $Q$ values. For this reason, the output

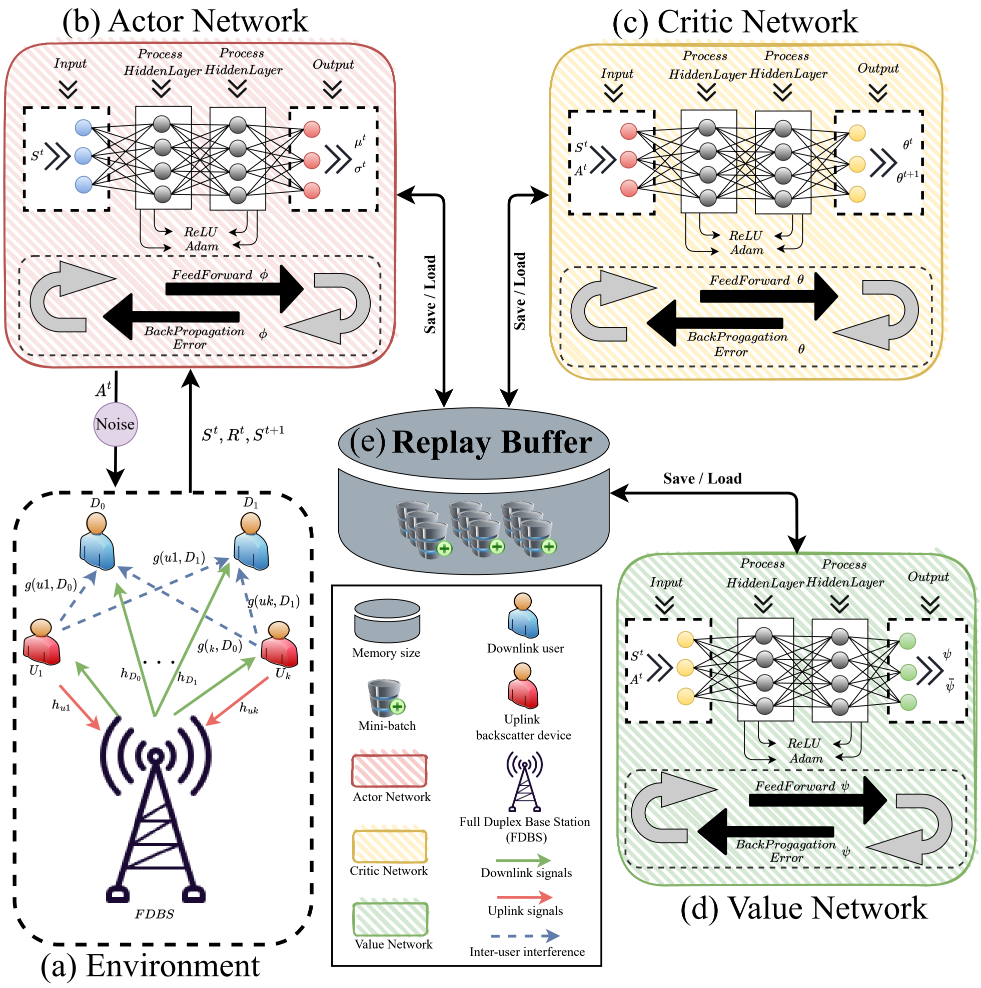


Figure 5.2: An illustration of the BAC-NOMA-SAC network model

for the critic network is the current $Q^{(t)}$ value and the predicted next $Q^{(t+1)}$ value for the future state and action pair. The green box represents the value network (d). The input to the value network is the state, which is used to predict the current and future value function. All the information is stored in the replay buffer $\mathcal{D}$, which is represented as a memory bank (colored in gray, as shown in (e)).

Next, we introduce the proposed SAC approach to optimize BAC-NOMA systems. First, the basic BAC-NOMA-SAC design and significant elements of the proposed learning algorithm are introduced. Second, we introduce the optimization process performed by the proposed algorithm.

#### 5.3.2.2 Key Design Elements

In this sub-section, we introduce an intelligent BAC-NOMA-SAC system for the long-term BAC-NOMA network sum rate maximizing optimization, where the agent learns a policy to jointly optimize the transmit power for downlink users and the BAC reflection coefficient under QoS requirements of downlink users. In the formulated MDP, which is a tuple of $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, the BAC-NOMA-SAC agent selects a state $s^{(t)} \in \mathcal{S}$ and takes a step by performing an action $\mathcal{A}$ to obtain the feedback from the BAC-NOMA environment in the form of reward $\mathcal{R}$. The $\mathcal{P}$ represents the probability of transition from the current state to the next state in a time step $t$. We use $\rho_\pi(s^{(t)})$ and $\rho_\pi(s^{(t)}|a^{(t)})$ to represent the state and state-action marginals of the trajectory distribution induced by a policy $\pi(s^{(t)}|a^{(t)})$.

A detailed explanation of the elements of the formulated MDP is given below.

- **Environment:** A BAC-NOMA network is the environment for the proposed SAC agent where there are one FDBS, a number of $K$ uplink backscatter devices, and multiple downlink users, as shown in Fig. 5.2.

- **Agent:** In the formulated MDP, the BS works as an agent to jointly optimize the power of downlink users and the BAC reflection coefficient of the uplink backscatter devices.

- **State space:** The state is the information relevant to the environment the agent accesses during the interaction. The proposed state space is a matrix characterized by the BAC reflection coefficient $\eta_k$ of uplink backscatter devices and the transmit power for downlink users $P_{D_i}$. At each time step $t$, the state can be given as:

$$s^{(t)} = \big((P_{D_i}), (\textstyle\sum \eta_{U_k} \times P_{D_i})\big). \tag{5.12}$$

  The state space is a finite set with ${U_K}^{(\eta_{U_k} \times P_{D_i})}$ number of states through which the agent (BS) can navigate. Furthermore, based on the received reward, if the agent selects 1 then the agent moves to the next power allocation coefficients subset from 2 dimensions set of states that are bounded by ${U_K}^{(\eta_{U_k} \times P_{D_i})}$ total number of states. The whole state space can be defined as $\mathcal{S} = \{s^{(t)}, s^{(t+1)}, s^{(t+2)}, \dots, s^N\}$. The values of state space parameters are listed in Table 5.1.

- **Action space:** The action is the swap operation between the states. Three different levels of action help the agent to explore and exploit the environment and to optimize the resource allocation for all users. The action can be given as:

$$a^{(t)} = \{-1, 0, 1\}, \tag{5.13}$$

where action $-1$ implies that the agent shifts back to the previous state, 0 implies that the agent does not change its state but remains in the current state, and 1 implies that the agent shifts to the next state. To optimize the resource allocation, the agent navigates the environment by switching to different power allocation levels for each downlink user and BAC reflection coefficient for uplink backscatter devices. In this way, the agent explores the dynamic environment to optimize long-term resource allocations for BAC-NOMA systems.

- **Rewards:** The agent receives feedback from the BAC-NOMA environment in the form of reward $r^{(t)}$. The agent receives positive feedback in the form of 10 from the BAC-NOMA environment if the current sum rate of the uplink backscatter devices is greater or equal to the previous sum rate and the constraints are not violated ((5.11b)-(5.11e)). Otherwise, the agent receives a reward of 0 as a penalty for the wrong action. Finally, the reward function is calculated as:

$$r^{(t)}(s^{(t)}, a^{(t)}) = \begin{cases} 10, & \text{if } {R_{sum}}^{(t)} \geq {R_{sum}}^{(t-1)} \text{and satisfy constraints} \\ & \text{given in ((5.11b)-(5.11e)).} \\ 0, & \text{otherwise.} \end{cases} \tag{5.14}$$

The following function $Z_{(\pi)}$ maximizes the expected reward by adding an entropy term $\mathcal{H}$ as indicated below [89],

$$Z_{(\pi)} = \sum_{t=0}^{T} \mathbb{E}_{(s^{(t)},a^{(t)})\sim\rho_\pi} \left[ r(s^{(t)}, a^{(t)}) + \bar{\alpha}\, \underbrace{\mathcal{H}\big(\pi(\cdot|s^{(t)})\big)}_{\text{Entropy}} \right], \tag{5.15}$$

where $\mathcal{H}$ is weighted by a temperature parameter $\bar{\alpha}$ to regulate the randomness of the optimal policy. For the SAC agent, the concept of exploration and exploitation of the wireless network environment is important to learn a stable action selection policy. $\bar{\alpha}$ temperature parameter is between 0 and 1. This $\bar{\alpha}$ determines the $\mathcal{H}\big(\pi(\cdot|s^{(t)})\big)$ to set the learning path for the agent.

The modified Bellman equation for the policy $\pi$ is utilized in any $Q$ function that is calculated iteratively for operator $\chi^\pi$ as follows:

$$\chi^\pi Q(s^{(t)}, a^{(t)}) \triangleq r(s^{(t)}, a^{(t)}) + \gamma \mathbb{E}_{s^{(t+1)}\sim\mathcal{P}} \left[ V(s^{(t+1)}) \right], \tag{5.16}$$

where $V(s^{(t)})$ is the soft state value function for policy $\pi$, which is shown in the following equation:

$$V(s^{(t)}) = \mathbb{E}_{a^{(t)}\sim\pi} \left[ Q(s^{(t)}, a^{(t)}) - \log \pi(a^{(t)}|s^{(t)}) \right]. \tag{5.17}$$

SAC trains functions to approximate, a state value function $V_\psi(s^{(t)})$, a soft $Q$ function $Q_\theta(s^{(t)}, a^{(t)})$, and a policy function $\pi_\phi(a^{(t)}|s^{(t)})$. The actor, critic, and value networks' parameters are respectively denoted by $\phi, \theta$, $\psi$, and $V_\psi$ minimizes the squared residual error as follows:

$$Z_V(\psi) = \mathbb{E}_{s^{(t)}\sim\mathcal{D}} \Big[ \tfrac{1}{2} (V_\psi(s^{(t)}) - \mathbb{E}_{a^{(t)}\sim\pi_\phi} [Q_\theta(s^{(t)}, a^{(t)}) - \log \pi_\phi(a^{(t)}|s^{(t)})])^2 \Big], \tag{5.18}$$

where $\mathcal{D}$ denotes a previously experienced state and action distribution, which is used as experience memory. The gradient update estimation of equation (5.18) is performed with the help of the following function. Generally, at each time step, the squared difference between predictions and the expectation of the soft $Q$-function is minimized to obtain the policy $\pi$. The parameters of the above objective function are updated as follows:

$$\hat{\nabla}_{\psi} Z_V(\psi) = \nabla_{\psi} V_{\psi}(s^{(t)}) \Big( V_{\psi}(s^{(t)}) - Q_{\theta}(s^{(t)}, a^{(t)}) + \log \pi_{\phi}(a^{(t)}|s^{(t)}) \Big), \tag{5.19}$$

where $\hat{\nabla}_{\psi}$ shows the update function of the $Z_V(\psi)$ based on the gradient step. The soft $Q$-function is optimized using the equation below:

$$Z_Q(\theta) = \mathbb{E}_{(s^{(t)},a^{(t)})\sim\mathcal{D}} \Big[ \tfrac{1}{2} \Big( Q_{\theta}(s^{(t)}, a^{(t)}) - \hat{Q}(s^{(t)}, a^{(t)}) \Big)^2 \Big], \tag{5.20}$$

where the definition of $\hat{Q}(s^{(t)}, a^{(t)})$ is as follows:

$$\hat{Q}(s^{(t)}, a^{(t)}) = r(s^{(t)}, a^{(t)}) + \gamma \mathbb{E}_{s^{(t+1)}\sim\mathcal{P}}[V_{\bar{\psi}}(s^{(t+1)})]. \tag{5.21}$$

The objective here is to minimize the squared difference between what the soft $Q$-function predicts and the reward plus the discounted expected value of the next state. The soft $Q$-functions parameters are updated as below:

$$\hat{\nabla}_{\theta} Z_Q(\theta) = \nabla_{\theta} Q_{\theta}(a^{(t)}, s^{(t)}) \Big( Q_{\theta}(s^{(t)}, a^{(t)}) - r(s^{(t)}, a^{(t)}) - \gamma V_{\bar{\psi}}(s^{(t+1)}) \Big). \tag{5.22}$$

#### 5.3.2.3 BAC-NOMA-SAC Algorithm Details

Based on the above discussion, we describe the significant features of the proposed BAC-NOMA-SAC **Algorithm 3** that are used to enhance the achievable sum rate of uplink backscatter devices while preserving the QoS requirements of the downlink users. The details for these features of the proposed algorithm are introduced in the following points.

- **Initialization:**

  To begin the optimization processes which is line 1 in **algorithm 3**, we initialize network environment parameters and training hyper-parameters. The brain of the

**Algorithm 3** The Intelligent BAC-NOMA-SAC Framework

```
Initialize parameter vectors $\mathcal{S}$, $\mathcal{A}$, $\mathcal{R}$, BAC-NOMA network environment, episodes, iterations, replay memory $\mathcal{D}$, batch-size, actor network ($\phi$), critic network ($\theta$), value network ($\psi$), and target value network ($\bar{\psi}$).
for each episode $N_e$ do
    for each iteration $T_e$ do
        $a^{(t)} \sim \pi_{\phi}(a^{(t)}|s^{(t)})$
        if action $< 0$ then
            $a^{(t)} = -1$
        else if action $= 0$ then
            $a^{(t)} = 0$
        else
            $a^{(t)} = 1$
        end if
        Calculate reward $r^{(t)}$ using equation (5.14)
        $\mathcal{D} \leftarrow \mathcal{D} \cup \left\{ \left( s^{(t)}, a^{(t)}, r(s^{(t)}, a^{(t)}), s^{(t+1)} \right) \right\}$
    end for
    Update; actor network ($\phi$), critic network ($\theta$), value network ($\psi$), and the next target value network ($\bar{\psi}$).
    for each gradient step do
        $\psi \leftarrow \psi - \lambda_V \hat{\nabla}_{\psi} Z_V(\psi)$
        $\theta \leftarrow \theta - \lambda_Q \hat{\nabla}_{\theta} Z_Q(\theta)$
        $\phi \leftarrow \phi - \lambda_{\pi} \hat{\nabla}_{\phi} Z_{\pi}(\phi)$
        $\bar{\psi} \leftarrow \tau\psi + (1-\tau)(\bar{\psi})$
    end for
end for
```

SAC agent is initialized as three different neural networks (actor, critic, and value) to learn the optimal policy. The hyper-parameters used for this algorithm are listed in Table 5.1.

- **Brain Architecture:**

In the proposed model, there are FCNNs architecture for the brain of the proposed agent because FCNNs are considered efficient architecture of artificial neural networks to process the dynamic environment. The feed-forward propagation mainly performs the functions of neuron activation, neuron transfer, and forward propagation. First, the neuron activation computes the weighted sum for the input and the bias. The neuron transfer invokes ReLU activation function to activate the neurons. Finally, forward propagation is the process of providing input to the next layer. This process happens for all the remaining layers. Last, to get robust stable learning and optimize the dynamic BAC-NOMA network, we use the optimization for a dynamic BAC-NOMA network with the three following neural networks.

– **Actor Network ($\phi$):**

This model is based on the throughput maximization policy $\pi_\phi(s^t, a^t)$ that also considers downlink user's QoS requirements, which is tuned by the actor network ($\phi$). The architecture of this network consists of one input layer, two hidden layers with ReLU activation functions, feed-forward propagation, back propagation, loss function, Adam optimizer, and output mechanisms to perform efficient action in the dynamic network environment. Starting with the inputs, the actor network receives states as input from the environment (BAC-NOMA).

The first hidden layer receives the network environment information that is output propagated from the first layer that is activated by the ReLU activation function. The output of this hidden layer is in the form of weights and bias. The same process continues with the second hidden layer until the final output. We utilize the Adam optimizer to compute the gradients used in updating the weights of the neural networks, thus minimizing the overall loss when predicting the output that is an action $a^{(t)}$. Generally this back-propagation process helps the neural network to minimize the weight prediction errors by adjusting neural network weights during the learning process.

Last, when the agent is experienced enough by obtaining multiple allocation policies. The updated parameters of the actor network are:

$$\phi \leftarrow \phi - \lambda_\pi \hat{\nabla}_\phi Z_\pi(\phi). \tag{5.23}$$

– **Critic Network ($\theta$):**

Similar to the first neural network architecture (Actor), the critic network follows the same architectural design. The input of this network is different from that of the actor network, which is based on state and action at each time slot $t$. This is the function of the critic network is to learn the current in future key value by calculating the Bellman equation (5.16). For this reason, the input of the critic network is different from the actor network. As the name suggests, the bellman equation is updated with soft $Q$ updates. The soft $Q$-function is denoted as $Q_\theta(s^{(t)}, a^{(t)})$. Finally, the $Q$-function update is as follows:

$$\theta \leftarrow \theta - \lambda_Q \hat{\nabla}_\theta Z_Q(\theta). \tag{5.24}$$

– **Value Network and Target Value Network ($\psi, \bar{\psi}$):**

Value network denoted by $V^{(t)}(\psi)$, and the target value network is denoted by $V^{(t+1)}$ $(\bar{\psi})$. The architecture of the value network follows the same design as the actor and critic networks. The input is the state which predict the current and target values. To learn the efficient resource allocation via policy $\pi$, the value network output $V^{(t)}$ seeks to minimize the error between the two value networks to assist the agent efficiently. The value network is updated with the help of the following equation:

$$\psi \leftarrow \psi - \lambda_V \hat{\nabla}_\psi Z_V(\psi). \tag{5.25}$$

Similarly, the target value network $V^{t+1}$ is updated with the following equation,

$$\bar{\psi} \leftarrow \tau\psi + (1 - \tau)(\bar{\psi}), \tag{5.26}$$

where $\tau$ represents the smoothing coefficient of target value. The function of

$\tau$ is utilized to stabilize the training process of the SAC agent.

We use the same architecture for all the neural networks. This shows the strength of the proposed design, which can learn the dynamic environment of the actor, critic, and value networks.

### 5.3.3 Complexity of the BAC-NOMA-SAC

In this sub-section, we discuss the complexity of the proposed model. According to the given network environment, the complexity of our model depends on the network size (i.e., active uplink backscatter devices and downlink users) and three neural networks (actor, critic, and value networks). Each network consists of a different number of inputs and output features. The actor network takes input from the environment in the form of a state. After processing the state, the DNN produces output action in the form of mean and standard deviation. Before producing the output, the feed-forward and back-propagation mechanisms are adopted to fine tune the DNN online. Similarly, activation of all the neurons is performed using the ReLU activation function. The $(e)$ denotes the input layer size that depends on the number of active devices. Every network contains two hidden layers $(L)$, and each layer contains $(\hat{x}_l)$ neurons.

These parameters follow $\hat{E} \overset{\Delta}{=} e\hat{x}_1 + \sum_{l=1}^{L-1}(\hat{x}_l)\hat{x}_{l+1}$. The real-time computational complexity of the feed forward and back propagation for the downlink users and uplink backscatter devices in this BAC-NOMA-SAC model is $\mathcal{O}(\hat{E})$. According to the total number of episodes $N_e$ and iterations $T_e$ that the agent takes, the calculation for the computational complexity is $\mathcal{O}(N_e T_e \hat{E})$.

## 5.4 Simulation Results

Simulation results using the experimental setup given below are listed in this section.

### 5.4.1 BAC-NOMA-SAC Experimental Setup

This section presents the system parameters and the setup of the simulation to demonstrate the BAC-NOMA-SAC algorithm performance. Our setup includes multiple downlink users and multiple uplink backscatter devices connected via the same sub-channel to a single FD BS within different radius sizes of 5 meters, 25 meters, and 50 meters. The location of the BS, downlink users, and uplink backscatter devices are set at $(0,0)$ meters, $((3,0),(4,0))$ meters, and randomly distributed in the area, respectively. We treat the noise $(\sigma^2)$ as a hyper-parameter and test different values. The system model (BAC-NOMA-SAC) uses fully connected hidden layers, and there are (256) neurons per layer. The actor, critic, and value networks are used to enhance the learning process. Different parameters, such as the temperature parameter represented by $\bar{\alpha}$, the discount factor represented by $\gamma$, and $\tau$ are used to modulate the parameters of our target value network. Moreover, all hidden layers are processed by the ReLU function. To balance between exploration and exploitation, SAC uses entropy from equation (5.15). We use Rayleigh fading in the proposed model. When a channel experiences more fading, the received signal strength drops, and this can lead to decreased performance in terms of data rates, throughput, increased error rates, and poorer overall communication quality. Therefore, achieving the same performance is challenging. However, by adjusting the transmission power based on the channel quality, a system can ensure significant performance. Tuning the parameters can lead to a faster learning process and convergence. Additional system parameters and their values used for the simulation (for both the proposed and benchmark schemes) are given in Table 5.1. A MacBook Pro macOS system with a 3.1 GHz Intel Core i5 processor, 8 GB of memory (random access memory), and 2133 MHz LPDDR3 is used for the simulation. Python 3.6 is used to implement the proposed system model.

### 5.4.2 The BAC-NOMA-SAC Convergence

Fig. 5.3 shows the convergence of the BAC-NOMA-SAC algorithm with respect to the different number of iterations in each episode. It can be seen that the agent obtained

Table 5.1: Network and training ML parameters

| **Parameter** | **Value** |
|---|---|
| FD BS | 1 |
| Downlink users | $\{1-2\}$ |
| Uplink backscatter devices | $\{2-8\}$ |
| $P_{max}$ | 20 dBm |
| Channel type | Fading |
| Noise | $\{-94, -84, -74\}$ dBm |
| Radius | $\{5, 25, 50\}$ meters |
| Target data rate for $D_i$ | $\{0.5, 1, 2, 3\}$ BPCU |
| BAC reflection coefficient | $\{0.1, 0.2, \dots 0.8, 0.9\}$ |
| Self-interference coefficient | $\{0.001, \dots, 0.1\}$ dBm |
| Episodes | 500 |
| Trials | $\{400, 500\}$ |
| Learning rate | 0.1 |
| Discount factor | 0.99 |
| Target value smoothing coefficient | 0.001 |
| Batch size | 100 |
| DNN activations | ReLU |
| Optimizer | Adam |
| Hidden layers | 2 |
| Neurons for each layer | 256 |

a higher average reward with 500 iterations in each episode. The agent with a lower number of iterations (400 iterations) in each episode cannot explore the environment completely and converges to a non-optimal solution with a low reward. In order to reach the optimal solution for the given problem, RL algorithms require considerable learning steps; therefore, we kept the number of iterations at 500 so that the agent can fully explore the environment and find good states and actions.

### 5.4.3 Performance with Respect to Different QoS Requirements

Fig. 5.4 illustrates the sum rate of backscatter users with regard to different QoS requirements and the different number of downlink users. The sum rate of backscatter devices increases with multiple downlink users when we set the QoS requirements to 0.5 bit per channel use (BPCU). Because of the small QoS requirements, the downlink users can achieve the target date rate with a small amount of transmit power, and the rest of the power is allocated to backscatter devices, which increases their sum rate. In the

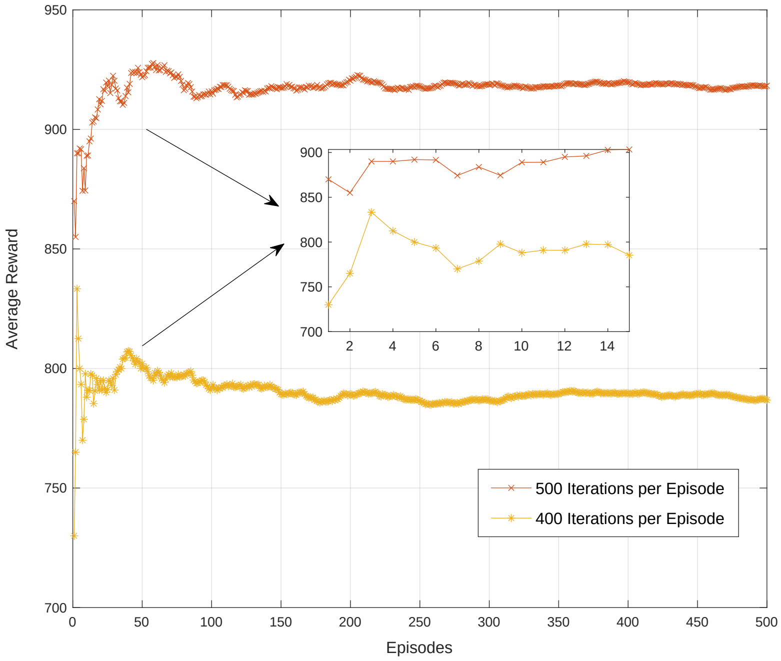


Figure 5.3: Shows the convergence and the reward obtained in the different number of iterations at each episode

same way, with a single downlink user and multiple uplink backscatter devices, the 0.5 BPCU requirements enhance the sum rate of backscatter devices compared to the large (3 BPCU) requirements. In a nutshell, the BS (agent) is able to allocate the transmit power and reflection coefficient effectively while considering the QoS requirements of downlink users.

### 5.4.4 Performance Comparison with a Varying Number of Backscatter Devices

In this section, we compare the performance of our proposed scheme with conventional optimization (benchmark) and random power allocation and compare the performance of BAC with OMA in terms of the achievable sum rate against varying numbers of $U_K$ uplink backscatter devices. The performance of all schemes is checked for two different target data rate requirements, that is 0.5 BPCU and 3 BPCU. As seen in Fig. 5.5, our proposed scheme (red curves) outperforms the rest of the schemes with respect to both

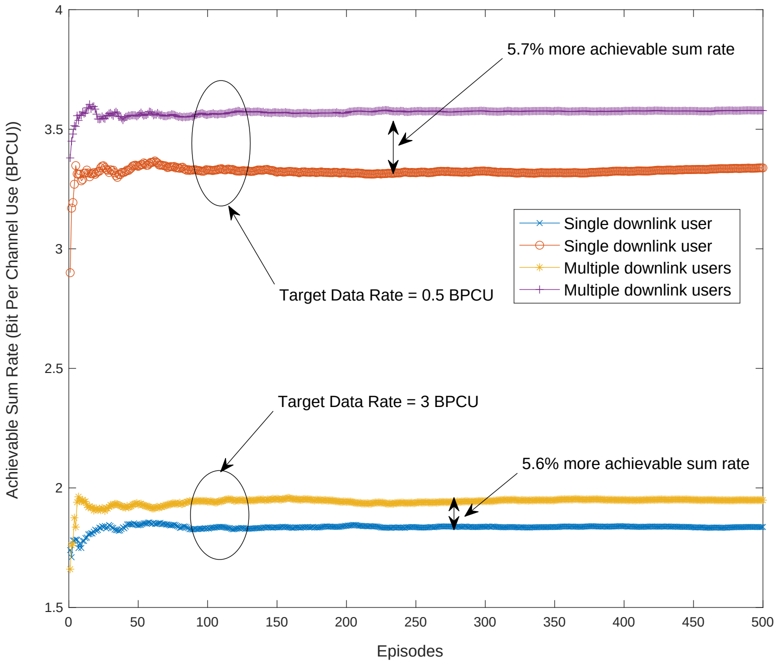


Figure 5.4: Shows sum rate with different target data rate and downlink users

QoS requirements. With an increased number ($U_K = 10$) and QoS of 0.5 BPCU, the sum rate almost reaches 8 BPCU. Increasing QoS for downlink users from 0.5 BPCU to 3 BPCU leads to a decrease in the achievable sum rate; that is, it drops from 8 BPCU to 6.5 BPCU. The benchmark scheme (black curves) outperforms the random power allocation method (blue curves) and backscatter communication with OMA (green curves). Therefore, by applying the proposed model, the system can be more flexible in allocating resources (power and subcarriers) to users. Also, the proposed model can manage and consider the complexities of SIC order and power allocation to enhance the performance of the system. Furthermore, the proposed model can handle non-linear constraints.

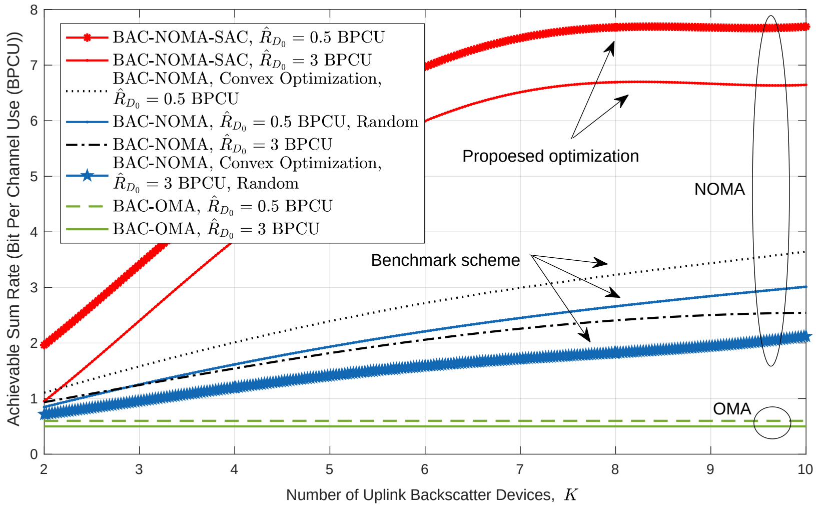


Figure 5.5: The achievable sum rate against the different target data rate and different number of $K$ devices

### 5.4.5 Varying Self-Interference Coefficient and Different Uplink Backscatter Devices

Fig. 5.6 shows the performance comparison of the proposed BAC-NOMA-SAC scheme with the conventional optimization (benchmark) schemes with regard to different values of $(\varphi)$ and $U_K$ in terms of sum rate. The proposed scheme with $U_K = 8$ provides the highest achievable sum rate. However, as the value of $(\varphi)$ increases towards 0.1, the achievable sum rate decreases to almost 3 BPCU. With the same number of backscatter users $(U_K = 2)$, our proposed algorithm obtained higher sum rate as compared to the benchmark scheme and BAC-OMA method. We attribute the performance gains made by our proposed model to the fact that the BS allocates the power and BAC reflection coefficient dynamically to downlink and uplink backscatter users.

### 5.4.6 Impact of the Noise

Fig. 5.7 shows the impact of noise $\sigma$, uplink backscatter devices, and different QoS requirements for downlink user on the performance of the proposed BAC-NOMA-SAC algorithm. We also compare the performance with that of BAC-OMA.

For all the cases, the achievable sum rate decreases as the noise level increases from $(-94$ dBm$)$ to $(-74$ dBm$)$. Moreover, the proposed scheme achieves a better sum rate

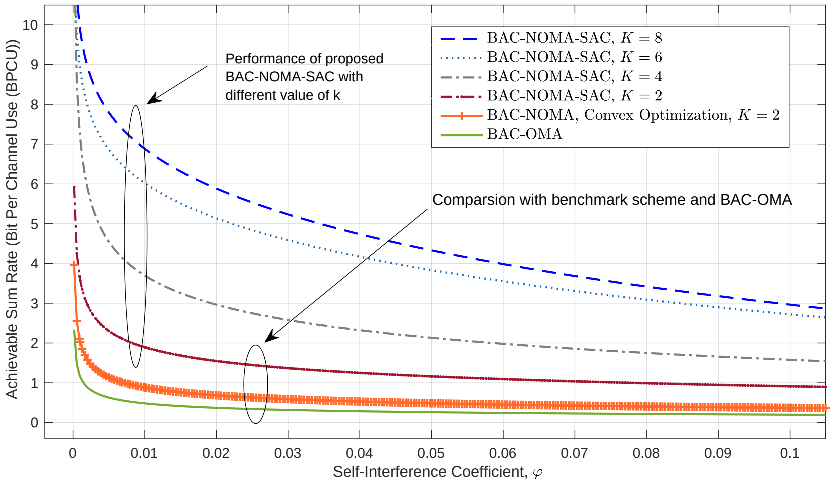


Figure 5.6: Achievable sum rate comparison with respect to increasing self-interference coefficient ($\varphi$)

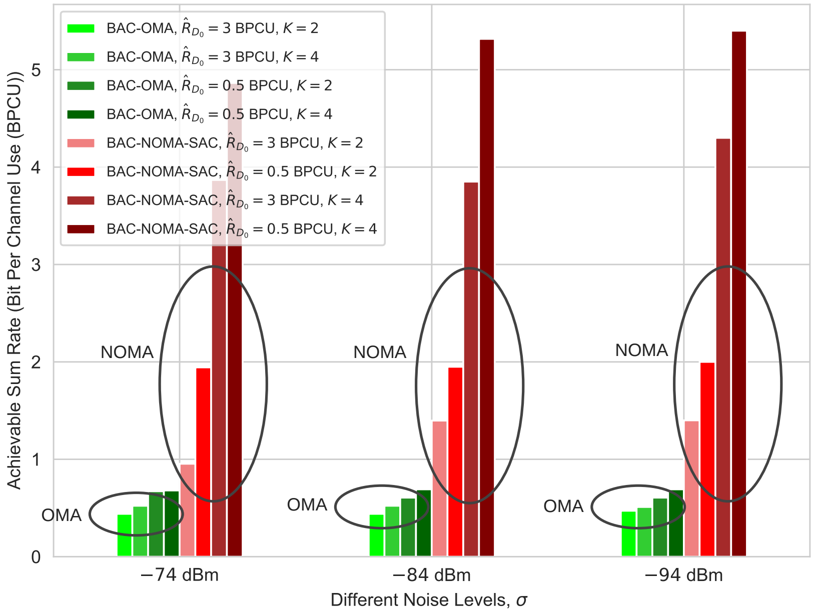


Figure 5.7: Achievable sum rate vs different noise ($\sigma$) levels

compared to BAC-OMA with an increased number of backscatter devices and when the QoS requirements are set to 0.5 BPCU (low QoS requirements). Additionally, the conventional BAC-OMA provides the lowest sum rate against all parameters.

### 5.4.7 Impact of the Cell Radius Size

Fig. 5.8 illustrates the comparison of the proposed BAC-NOMA-SAC and BAC-OMA in terms of the achievable sum rate. The figure depicts the achievable sum rate with

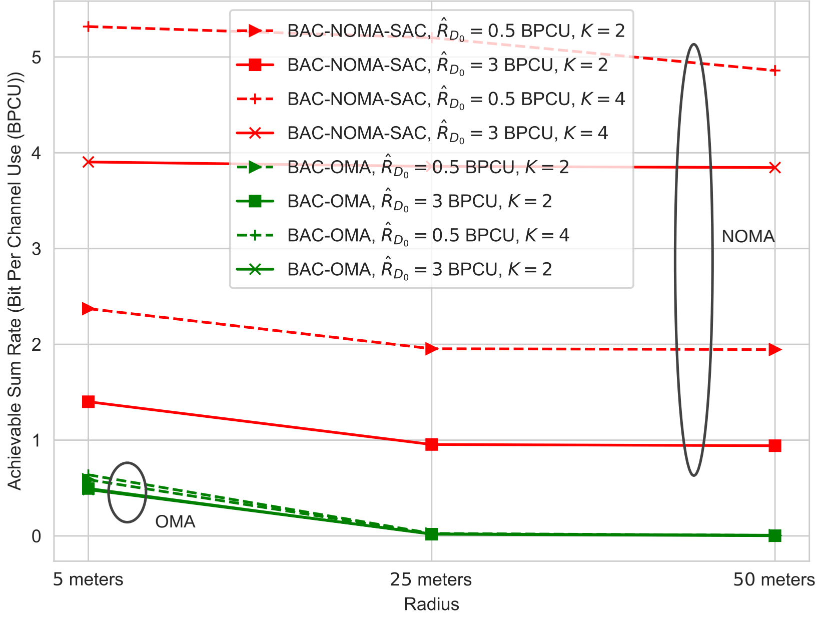


Figure 5.8: Achievable sum rate comparison with different radius and different target data rates $\hat{R}_{D_0}$

different radius sizes, different values of $U_K$, and different QoS requirements for the downlink users. The achievable sum rate with the high number of $U_K$ uplink backscatter devices is a higher sum rate compared to BAC-OMA for a low number of $U_K$ uplink backscatter devices for different radii. As the radius size increases, the sum rate of the proposed algorithm (red curves) gradually decreases because of the large-scale distance-dependent path loss.

Moreover, based on different radius settings, BAC-NOMA-SAC and BAC-OMA with $\hat{R}_{D_0} = 3$ BPCU perform worse than BAC-NOMA-SAC and BAC-OMA with $\hat{R}_{D_0} = 0.5$ BPCU. The BAC-OMA performance (green curves) also decreases with the increase in the radius size and has a low sum rate for all scenarios compared to the proposed BAC-NOMA scheme.

### 5.4.8 Performance Comparison with BAC-OMA with regard to Different QoS Requirements

Fig. 5.9 provides a performance comparison of the proposed BAC-NOMA with BAC-OMA against different QoS $\hat{R}_{D_0}$ and number of $U_K$ uplink backscatter devices. The

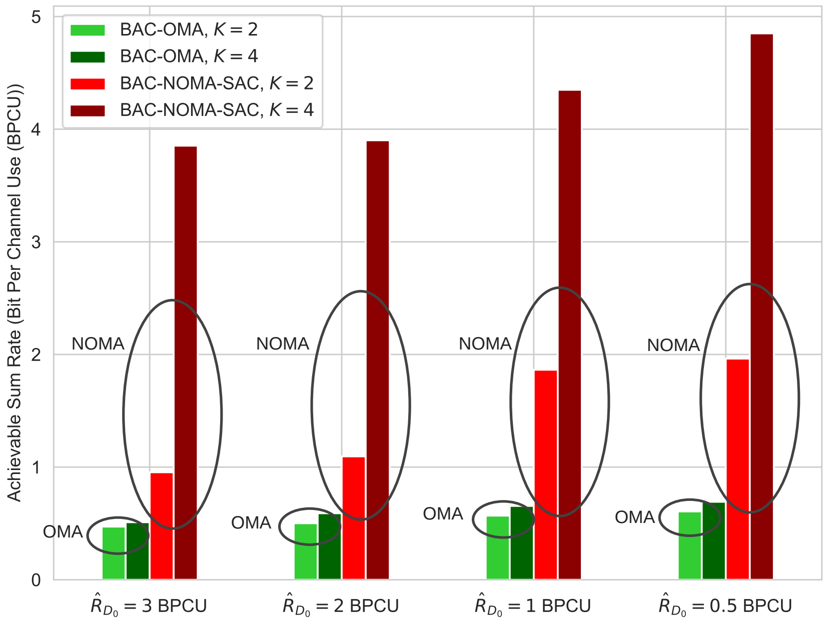


Figure 5.9: Achievable sum rate of vs different numbers of K devices and different target data rates

light green bar represents two uplink backscatter devices with the BAC-OMA network, and the dark green bar represents four uplink backscatter devices with the BAC-OMA network. In contrast, the light red bar represents the two uplink backscatter devices with BAC-NOMA-SAC, and the dark red represents the four uplink backscatter devices with BAC-NOMA-SAC.

We can see that with decreased QoS requirements, the sum rate of our proposed algorithm optimizing the reflection coefficient of four backscatter devices produces a higher sum rate. Generally, with different target data rates and uplink backscatter devices, BAC-NOMA-SAC consistently achieves a better sum rate compared to the BAC-OMA system.

## 5.5 Chapter Summary

In this chapter, we have suggested SAC-based BAC-NOMA algorithm to maximize the sum rate of uplink backscatter devices. The proposed SAC framework ensures the QoS requirements of downlink users are not compromised and learns long-term resource optimization in a dynamic BAC-NOMA network. We have shown that the proposed algorithm converges to an optimal solution with 500 iterations. Moreover, the simulation results demonstrate that the suggested algorithm obtained better sum rate with multiple downlink users and small QoS requirements. Additionally, the proposed algorithm outperforms the benchmark scheme, random power allocation, and the BAC-OMA method in terms of the achievable sum rate given the varying number of uplink backscatter devices. Similarly, the proposed algorithm shows superiority in terms of the sum rate against different values of self-interference and different noise levels. Finally, we have shown that the BAC-NOMA algorithm surpass the BAC-OMA method with different radii and target data rates in terms of the achievable sum rate.

# Chapter 6

# Conclusions and Future Work

This chapter summarises the content of the thesis, lists the contributions and findings, and discusses some possible future directions. Sub-section 6.1 summarises the contributions.

## 6.1 Contributions Summary

As promising technology in future wireless communication 5G, this thesis concentrates on how to improve NOMA network in different structures. With three different aspects represented in this thesis, we illustrated the knowledge and principles of IoT NOMA network as follows: 1) The knowledge and principles of NOMA, which include key technologies, such as SC and SIC, single-cell downlink NOMA network, multi-cell downlink NOMA network, and investigation about NOMA network in general with downlink framework setting. 2) Uplink IoT NOMA network, where there are different uplink IoT NOMA users. For example, IoT users with only GB or GF or both scenarios are as known as SGF. Moreover, we used different techniques, such as relay node and semi-centralized algorithm, to improve the EE of GF users and reduce the system complexity. 3) With FDBS, downlink and uplink NOMA scenarios were applied. Moreover, the BAC technique was applied to improve the EE for uplink backscatter devices without affecting downlink users. In this work, we took the advantages of the downlink signal from the BS to excite the circuits of uplink backscatter devices. More details about the main contributions that applied in this thesis are described as follows.

In Chapter 3, single and multiple cell downlink NOMA networks were investigated. Moreover, different techniques, such as DRL, ACDRL-D, and ACDRL-C ML, were investigated. A framework for downlink NOMA was proposed based on a model-free RL approach for dynamic resource allocation in a multi-cell network structure. With the aid of ACDRL, we optimized the active power allocation for multi-cell NOMA systems under an online environment to maximize the long-term sum rate. To exploit the dynamic nature of NOMA, this work utilized the instantaneous data rate for designing the dynamic reward. The state space in ACDRL contains all possible resource allocation realizations depending on a three-dimensional association among users, base stations, and sub-channels. We proposed an ACDRL algorithm with this transformed state space, which is scalable to handle different network loads by utilizing multiple deep neural networks. Lastly, the simulation results validated that the proposed solution for multi-cell NOMA outperforms the conventional RL, DRL algorithms, and OMA schemes in terms of the evaluated long-term sum rate.

In Chapter 4, different types of uplink NOMA IoT users were investigated. Moreover, different techniques to improve the EE such as relay node, were considered. Moreover, we considered two different network frameworks, such as centralized and fully distributed. We proposed a semi-centralized optimization framework for NOMA IoT networks to maximize the EE of different types of users (GB and GF). We used a proximal policy optimization algorithm at the BS to maximize the EE of GB users and a multi-agent DQN to optimize the resource allocation for GF users with aid of a relay node. The proposed algorithm combines the advantages of fully centralized and fully distributed frameworks to compensate for their shortcomings (complexity and long learning time). The numerical results showed that the proposed algorithm enhances the EE of GB users by 6% and 11.5%, respectively, compared with the fixed power allocation and random power allocation strategies. Moreover, the results demonstrated a 47.4% increase in the EE of GF users over the benchmark scheme. Additionally, we showed that the increase in the number of GB users has a significant impact on the EE of GB and GF users.

In Chapter 5, with the use of power domain NOMA and BAC, future 6G ultra massive machine-type communications networks are expected to connect large-scale IoT devices. However, due to NOMA co-channel interference, the power allocation to large-scale IoT devices becomes critical. The existing convex optimization-based solutions are highly complex, and therefore, it is difficult to find the optimal solution to the resource allocation problem in a highly dynamic environment. To alleviate this problem, this work developed an efficient model-free BAC approach with a NOMA system to assist the base station with complex resource scheduling tasks in a dynamic BAC-NOMA IoT network. The objective was to increase the sum rate of uplink backscatter devices. More specifically, we jointly optimized the transmit power of downlink IoT users and the reflection coefficient of uplink backscatter devices using a reinforcement learning algorithm, namely the SAC algorithm. With the advantage of entropy regularization, the SAC agent learns to explore and exploit the dynamic BAC-NOMA network efficiently. The proposed algorithm ensures the QoS requirements of downlink users while enhancing the sum rate of uplink backscatter devices. Numerical results revealed the superiority of the proposed algorithm over the conventional optimization (benchmark) approach in terms of the average sum rate of uplink backscatter devices. We showed that the network with multiple downlink users obtained a higher reward with respect to a large number of iterations compared to episodes with a lower number of iterations. Moreover, the proposed algorithm outperformed the benchmark scheme and BAC with OMA in terms of the average sum rate with the different number of backscatter devices. Additionally, we showed that our proposed algorithm enhances sum rate efficiency with respect to different self-interference coefficients and different noise levels. Finally, we evaluated and showed the sum rate efficiency of the proposed algorithm with different QoS requirements and cell radii.

## 6.2 Further Work

Wireless networks and AI (ML) have emerged within the last few years and created a new wireless generation called 6G communication technologies. Today, there are many requirements, tests, and improvements needed to be added in this area. Below is a brief list of some future directions for solving part of these requirements and for improvements in the near future:

- **Incorporating Deep Learning:** It is refers to the use of DL models, techniques, or methods to solve problems, build systems, or develop applications. DL considered as subset of ML which involves NN with more than one layer to learn patterns from data. By adding DL to a task or system, the performance of this system or task is enhanced. DL has shown great potential in resource allocation for IoT networks with NOMA. Incorporating DL for channel estimation/ CSI acquisition can improve system performance. As acquiring CSI in NOMA-based IoT networks is a challenging task, especially combining NOMA with some other technologies, such as reconfigurable intelligent surface (RIS) and multiple input multiple output (MIMO), the CSI can be acquired via DL through extensive training on the input data of existing channel models.

- **User Mobility Prediction:** User location and movement trajectory impact resource allocation, especially in NOMA-assisted networks. Therefore, predicting users' geographical location or position can improve the resource optimization process. Unsupervised learning is a better solution to predict users' positions.

- **K-repetitions:** Reliability is one of the main issues for 5G and beyond wireless communication. The K-repetitions method can be adopted especially for GF-NOMA IoT networks, where the users can send multiple copies of the same packets to enhance reliability.

- **MIMO:** MIMO is very important subject within 6G and beyond. Design and implement beamforming in MIMO can be considered as follows, 1) complex signal representation, 2) MIMO channel model, 3) beamforming vector, 4) transmit and receive beamforming. Adding MIMO in future designs helps enhance the performance of wireless communication systems in many ways. First, it can reduce interference from specific directions, thereby enhancing the QoS and performance of the wireless network. Also, it enhances both data rate and EE since multiple data streams can be transmitted within the same frequency band. Moreover, it can increase the range and coverage of the wireless network. Therefore, integrating MIMO with the proposed algorithms is another future research direction.

- **Impact of channel fading:** In resource allocation for NOMA-IoT networks with RL-based approaches, it is crucial to examine the impact of significant channel fading on delay, complexity, and real-time implementation. The following aspects require further investigation.

    1. Delay: Fluctuations in the quality of a wireless link due to channel fading can lead to delays in communication. In a resource allocation system based on reinforcement learning , the algorithm may need to adapt by re-optimizing resource allocations in response to these changes. However, this adaptation process can result in additional delays as the RL agent gathers information about the current channel conditions and updates its policy. It is essential to analyze and reduce these delays, especially for applications with strict latency requirements.

    2. Complexity: The complexity of resource allocation decisions can be heightened by channel fading. As channels fluctuate, the RL agent must take into account additional states, actions, or observations, depending on the fading model's intricacy and the extent of resource allocation. In situations with complex fading patterns, the use of more advanced RL algorithms or architectures may be necessary to manage the increased intricacy.

    3. Real-time implementation: Implementing real-time systems can be difficult

when dealing with channel fading, as the RL agent needs to make decisions quickly to adapt to changing conditions. If the RL algorithm is computationally intensive and cannot make decisions within the required time frame, it may not be suitable for real-time applications. Therefore, it is important to analyze the real-time performance of the RL algorithm under fading conditions to ensure that it meets the application's timing constraints.

- **UAV-assisted Networks:** The UAVs is an essential part of future wireless communication to achieve high data rates. Adding UAVs as flying BS with ML algorithms (specifically, the multi-agent system) can enhance system efficiency.

## References

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

[2] F.-L. Luo and C. J. Zhang, *Signal processing for 5G: algorithms and implementations.* John Wiley & Sons, 2016.

[3] Q. C. Li, H. Niu, A. T. Papathanassiou, and G. Wu, "5G network capacity: Key elements and technologies," *IEEE Veh. Tech. Mag.*, vol. 9, no. 1, pp. 71–78, 2014.

[4] Y. Tao, L. Liu, S. Liu, and Z. Zhang, "A survey: Several technologies of non-orthogonal transmission for 5G," *China commun.*, vol. 12, no. 10, pp. 1–15, 2015.

[5] A. W. Scott and R. Frobenius, "Multiple access techniques: FDMA, TDMA, and CDMA," *Wiley-IEEE Press*, 2008.

[6] H. Li, G. Ru, Y. Kim, and H. Liu, "OFDMA capacity analysis in MIMO channels," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4438–4446, 2010.

[7] L. Dai, B. Wang, Z. Ding, Z. Wang, S. Chen, and L. Hanzo, "A survey of non-orthogonal multiple access for 5G," *IEEE Commun. Surveys Tuts*, vol. 20, no. 3, pp. 2294–2323, 2018.

[8] A. Slalmi, H. Chaibi, A. Chehri, R. Saadane, and G. Jeon, "Toward 6G: Understanding network requirements and key performance indicators," *Trans. Emerging Tele. Tech.*, vol. 32, no. 3, p. e4201, 2021.

[9] H. Yang, A. Alphones, Z. Xiong, D. Niyato, J. Zhao, and K. Wu, "Artificial-intelligence-enabled intelligent 6G networks," *IEEE Network*, vol. 34, no. 6, pp. 272–280, 2020.

[10] S. R. Islam, N. Avazov, O. A. Dobre, and K.-S. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surveys & Tutorials*, vol. 19, no. 2, pp. 721–742, 2016.

[11] J. Mestoui and M. El Ghzaoui, "A survey of NOMA for 5G: Implementation schemes and energy efficiency," in *WITS 2020: Proceedings of the 6th Int. Conf. Wireless Tech., Embedded, and Intelligent Systems.* Springer, 2022, pp. 949–959.

[12] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Non-orthogonal multiple access for 5G and beyond," *Proc. IEEE*, vol. 105, no. 12, pp. 2347–2381, 2017.

[13] Y. Shi, L. Lian, Y. Shi, Z. Wang, Y. Zhou, L. Fu, L. Bai, J. Zhang, and W. Zhang, "Machine learning for large-scale optimization in 6g wireless networks," *IEEE Commun. Surveys & Tutorials*, 2023.

[14] B. Ji, Y. Han, S. Liu, F. Tao, G. Zhang, Z. Fu, and C. Li, "Several key technologies for 6g: Challenges and opportunities," *IEEE Commun. Standards Magazine*, vol. 5, no. 2, pp. 44–51, 2021.

[15] R. Alghamdi, R. Alhadrami, D. Alhothali, H. Almorad, A. Faisal, S. Helal, R. Shalabi, R. Asfour, N. Hammad, A. Shams *et al.*, "Intelligent surfaces for 6g wireless networks: A survey of optimization and performance analysis techniques," *IEEE access*, vol. 8, pp. 202 795–202 818, 2020.

[16] Z. M. Fadlullah, B. Mao, and N. Kato, "Balancing QoS and security in the edge: Existing practices, challenges, and 6G opportunities with machine learning," *IEEE Commun. Surveys & Tutorials*, 2022.

[17] B. M. Lee, "Improved energy efficiency of massive MIMO-OFDM in battery-limited IoT networks," *IEEE Access*, vol. 6, pp. 38 147–38 160, 2018.

[18] W. U. Khan, J. Liu, F. Jameel, V. Sharma, R. Jäntti, and Z. Han, "Spectral efficiency optimization for next generation NOMA-enabled IoT networks," *IEEE Trans. Veh. Tech.*, vol. 69, no. 12, pp. 15 284–15 297, 2020.

[19] K. Wang, Y. Liu, Z. Ding, A. Nallanathan, and M. Peng, "User association and power allocation for multi-cell non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5284–5298, 2019.

[20] P. Xu, Z. Ding, X. Dai, and H. V. Poor, "A new evaluation criterion for non-orthogonal multiple access in 5G software defined networks," *IEEE Access*, vol. 3, pp. 1633–1639, 2015.

[21] X. Shao, C. Yang, D. Chen, N. Zhao, and F. R. Yu, "Dynamic IoT device clustering and energy management with hybrid noma systems," *IEEE Trans. Ind. Info.*, vol. 14, no. 10, pp. 4622–4630, 2018.

[22] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *IEEE 77th Veh. Tech. Conf. (VTC Spring)*, 2013, pp. 1–5.

[23] Z. Ding, Z. Yang, P. Fan, and H. V. Poor, "On the performance of non-orthogonal multiple access in 5g systems with randomly deployed users," *IEEE signal processing lett.*, vol. 21, no. 12, pp. 1501–1505, 2014.

[24] S. Timotheou and I. Krikidis, "Fairness for non-orthogonal multiple access in 5G systems," *IEEE signal processing lett.*, vol. 22, no. 10, pp. 1647–1651, 2015.

[25] Y. Liu, Z. Ding, M. Elkashlan, and H. V. Poor, "Cooperative non-orthogonal multiple access with simultaneous wireless information and power transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 938–953, 2016.

[26] 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 Trans. Veh. Tech.*, vol. 67, no. 4, pp. 3377–3389, 2017.

[27] S. Zhang, L. Li, J. Yin, W. Liang, X. Li, W. Chen, and Z. Han, "A dynamic power allocation scheme in power-domain NOMA using actor-critic reinforcement learning," in *IEEE Int. Conf. Commun. in China (ICCC)*, 2018, pp. 719–723.

[28] S. Wang, T. Lv, W. Ni, N. C. Beaulieu, and Y. J. Guo, "Joint resource management for MC-NOMA: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, pp. 5672–5688, Sept. 2021.

[29] J. Choi, "Non-orthogonal multiple access in downlink coordinated two-point systems," *IEEE Commun. Lett.*, vol. 18, no. 2, pp. 313–316, 2014.

[30] Z. Zhang, H. Sun, and R. Q. Hu, "Downlink and uplink non-orthogonal multiple access in a dense wireless network," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2771–2784, 2017.

[31] C. Chaieb, F. Abdelkefi, and W. Ajib, "Deep reinforcement learning for resource allocation in multi-band and hybrid OMA-NOMA wireless networks," *IEEE Trans. Commun.*, 2022.

[32] M. B. Shahab, R. Abbas, M. Shirvanimoghaddam, and S. J. Johnson, "Grant-free non-orthogonal multiple access for IoT: A survey," *IEEE Commun. Surveys Tuts*,

vol. 22, no. 3, pp. 1805–1838, 2020.

[33] X. Bai and X. Gu, "Noma assisted semi-grant-free scheme for scheduling multiple grant-free users," *IEEE Systems J.*, 2022.

[34] C. Zhang, Y. Liu, and Z. Ding, "Semi-grant-free NOMA: A stochastic geometry model," *IEEE Tran. Wireless Commun.*, vol. 21, no. 2, pp. 1197–1213, 2021.

[35] M. Al-Imari, P. Xiao, M. A. Imran, and R. Tafazolli, "Uplink non-orthogonal multiple access for 5G wireless networks," in *IEEE 11th int. symp. wireless commun. sys. (ISWCS)*, 2014, pp. 781–785.

[36] M. Al-Imari, P. Xiao, and M. A. Imran, "Receiver and resource allocation optimization for uplink NOMA in 5G wireless networks," in *IEEE Int Symp. Wireless Commun. Sys. (ISWCS)*, 2015, pp. 151–155.

[37] S. Chen, K. Peng, and H. Jin, "A suboptimal scheme for uplink NOMA in 5G systems," in *IEEE Int. Wireless Commun. Mobile Computing Conf. (IWCMC)*, 2015, pp. 1429–1434.

[38] N. Zhang, J. Wang, G. Kang, and Y. Liu, "Uplink nonorthogonal multiple access in 5G systems," *IEEE Commun. Lett.*, vol. 20, no. 3, pp. 458–461, 2016.

[39] W. Ahsan, W. Yi, Z. Qin, Y. Liu, and A. Nallanathan, "Resource allocation in uplink NOMA-IoT networks: A reinforcement-learning approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5083–5098, 2021.

[40] M. Nduwayezu, Q.-V. Pham, and W.-J. Hwang, "Online computation offloading in NOMA-based multi-access edge computing: A deep reinforcement learning approach," *IEEE Access*, vol. 8, pp. 99 098–99 109, 2020.

[41] Y.-H. Xu, Y.-B. Tian, P. K. Searyoh, G. Yu, and Y.-T. Yong, "Deep reinforcement learning-based resource allocation strategy for energy harvesting-powered cognitive machine-to-machine networks," *Computer Commun.*, vol. 160, pp. 706–717, 2020.

[42] M. V. da Silva, R. D. Souza, H. Alves, and T. Abrão, "A NOMA-based Q-learning random access method for machine type communications," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1720–1724, Oct. 2020.

[43] S. Wang, X. Wang, Y. Zhang, and Y. Xu, "Resource allocation in multi-cell NOMA systems with multi-agent deep reinforcement learning," in *IEEE Wireless Com-*

*mun. Net. Conf. (WCNC)*, 2021, pp. 1–6.

[44] M. Vaezi, G. A. A. Baduge, Y. Liu, A. Arafa, F. Fang, and Z. Ding, "Interplay between NOMA and other emerging technologies: A survey," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 4, pp. 900–919, Dec. 2019.

[45] L. Li, Q. Cheng, X. Tang, T. Bai, W. Chen, Z. Ding, and Z. Han, "Resource allocation for NOMA-MEC systems in ultra-dense networks: A learning aided mean-field game approach," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1487–1500, 2020.

[46] M. Fayaz, W. Yi, Y. Liu, and A. Nallanathan, "Transmit power pool design for grant-free NOMA-IoT networks via deep reinforcement learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 11, pp. 7626–7641, 2021.

[47] J. Zhang, X. Tao, H. Wu, N. Zhang, and X. Zhang, "Deep reinforcement learning for throughput improvement of the uplink grant-free NOMA system," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6369–6379, 2020.

[48] S. Doğan, A. Tusha, and H. Arslan, "NOMA with index modulation for uplink URLLC through grant-free access," *IEEE J. Selected Topics in Signal Processing*, vol. 13, no. 6, pp. 1249–1257, 2019.

[49] N. Ye, X. Li, H. Yu, L. Zhao, W. Liu, and X. Hou, "Deep NOMA: A unified framework for NOMA using deep multi-task learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2208–2225, Apr. 2020.

[50] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep learning for an effective nonorthogonal multiple access scheme," *IEEE Trans. Veh. Tech.*, vol. 67, no. 9, pp. 8440–8450, 2018.

[51] Y. Liu, Y. Deng, H. Zhou, M. Elkashlan, and A. Nallanathan, "Deep reinforcement learning-based grant-free NOMA optimization for mURLLC," *IEEE Trans. Commun.*, vol. 71, no. 3, pp. 1475–1490, 2023.

[52] Z. Ding, R. Schober, P. Fan, and H. V. Poor, "Simple semi-grant-free transmission strategies assisted by non-orthogonal multiple access," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4464–4478, 2019.

[53] Z. Yang, P. Xu, J. A. Hussein, Y. Wu, Z. Ding, and P. Fan, "Adaptive power alloca-

tion for uplink non-orthogonal multiple access with semi-grant-free transmission," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1725–1729, 2020.

[54] N. Jayanth, P. Chakraborty, M. Gupta, and S. Prakriya, "Performance of semi-grant free uplink with non-orthogonal multiple access," in *IEEE 31st Annual Int. Symp. Personal, Indoor and Mobile Radio Commun.*, 2020, pp. 1–6.

[55] C. Zhang, Y. Liu, W. Yi, Z. Qin, and Z. Ding, "Semi-grant-free NOMA: Ergodic rates analysis with random deployed users," *IEEE Wireless Commun. Lett.*, vol. 10, no. 4, pp. 692–695, 2020.

[56] M. Fayaz, W. Yi, Y. Liu, and A. Nallanathan, "Competitive MA-DRL for transmit power pool design in semi-grant-free NOMA systems," June. 2021, arXiv preprint arXiv:2106.11190. [Online]. Available: https://arxiv.org/abs/2106.11190

[57] A. Mondal, A. M. A. Junaedi, K. Singh, and S. Biswas, "Spectrum and energy-efficiency maximization in RIS-aided IoT networks," *IEEE Access*, vol. 10, pp. 103 538–103 551, 2022.

[58] C. Yao, Y. Liu, X. Wei, G. Wang, and F. Gao, "Backscatter technologies and the future of internet of things: Challenges and opportunities," *Intelligent and Converged Networks*, vol. 1, no. 2, pp. 170–180, 2020.

[59] J. Kimionis, A. Bletsas, and J. N. Sahalos, "Increased range bistatic scatter radio," *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 1091–1104, Mar. 2014.

[60] A. N. Parks, A. Liu, S. Gollakota, and J. R. Smith, "Turbocharging ambient backscatter communication," *ACM SIGCOMM Computer Commun. Review*, vol. 44, no. 4, pp. 619–630, 2014.

[61] Z. Ding, "Harvesting devices' heterogeneous energy profiles and QoS requirements in IoT: WPT-NOMA vs BAC-NOMA," *IEEE Trans. Commun.*, vol. 69, no. 5, pp. 2837–2850, 2021.

[62] Z. Ding and H. V. Poor, "On the Application of BAC-NOMA to 6G umMTC," *IEEE Commun. Lett.*, vol. 25, no. 8, pp. 2678–2682, Aug. 2021.

[63] J. Guo, X. Zhou, S. Durrani, and H. Yanikomeroglu, "Design of non-orthogonal multiple access enhanced backscatter communication," *IEEE Trans. Wireless Commun.*, vol. 17, no. 10, pp. 6837–6852, 2018.

[64] H. Guo, Y.-C. Liang, R. Long, and Q. Zhang, "Cooperative ambient backscatter system: A symbiotic radio paradigm for passive IoT," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 1191–1194, 2019.

[65] Y. Ye, L. Shi, X. Chu, and G. Lu, "On the outage performance of ambient backscatter communications," *IEEE Internet of Things J.*, vol. 7, no. 8, pp. 7265–7278, 2020.

[66] F. Jameel, T. Ristaniemi, I. Khan, and B. M. Lee, "Simultaneous harvest-and-transmit ambient backscatter communications under rayleigh fading," *EURASIP J. Wireless Commun. Net.*, vol. 2019, no. 1, pp. 1–9, 2019.

[67] J. Qian, A. N. Parks, J. R. Smith, F. Gao, and S. Jin, "IoT communications with $M$-PSK modulated ambient backscatter: Algorithm, analysis, and implementation," *IEEE Internet of Things J.*, vol. 6, no. 1, pp. 844–855, 2018.

[68] B. Lyu, C. You, Z. Yang, and G. Gui, "The optimal control policy for RF-powered backscatter communication networks," *IEEE Trans. Veh. Tech.*, vol. 67, no. 3, pp. 2804–2808, 2017.

[69] X. Li, Y. Zheng, W. U. Khan, M. Zeng, D. Li, G. Ragesh, and L. Li, "Physical layer security of cognitive ambient backscatter communications for green Internet-of-Things," *IEEE Trans. Green Commun. Net.*, vol. 5, no. 3, pp. 1066–1076, 2021.

[70] C.-B. Le and D.-T. Do, "Outage performance of backscatter NOMA relaying systems equipping with multiple antennas," *Electronics Lett.*, vol. 55, no. 19, pp. 1066–1067, 2019.

[71] W. U. Khan, F. Jameel, N. Kumar, R. Jäntti, and M. Guizani, "Backscatter-enabled efficient V2X communication with non-orthogonal multiple access," *IEEE Trans. Veh. Tech.*, vol. 70, no. 2, pp. 1724–1735, 2021.

[72] X. Li, M. Zhao, Y. Liu, L. Li, Z. Ding, and A. Nallanathan, "Secrecy analysis of ambient backscatter NOMA systems under I/Q imbalance," *IEEE Trans. Veh. Tech.*, vol. 69, no. 10, pp. 12 286–12 290, 2020.

[73] A. Farajzadeh, O. Ercetin, and H. Yanikomeroglu, "UAV data collection over NOMA backscatter networks: UAV altitude and trajectory optimization," in *IEEE*

*Int. Conf. Commun. (ICC)*, 2019, pp. 1–7.

[74] G. Yang, X. Xu, and Y.-C. Liang, "Resource allocation in NOMA-enhanced backscatter communication networks for wireless powered IoT," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 117–120, 2019.

[75] S. Zeb, Q. Abbas, S. A. Hassan, A. Mahmood, R. Mumtaz, S. H. Zaidi, S. A. R. Zaidi, and M. Gidlund, "NOMA enhanced backscatter communication for green IoT networks," in *IEEE 16th Int. Symp. Wireless Commun. Sys. (ISWCS)*, 2019, pp. 640–644.

[76] Y. Li, M. Jiang, Q. Zhang, and J. Qin, "Secure beamforming in MISO NOMA backscatter device aided symbiotic radio networks," *arXiv preprint arXiv:1906.03410*, 2019.

[77] Y. Xu, Z. Qin, G. Gui, H. Gacanin, H. Sari, and F. Adachi, "Energy efficiency maximization in NOMA enabled backscatter communications with QoS guarantee," *IEEE Wireless Commun. Lett.*, vol. 10, no. 2, pp. 353–357, 2020.

[78] X. Li, M. Zhao, M. Zeng, S. Mumtaz, V. G. Menon, Z. Ding, and O. A. Dobre, "Hardware impaired ambient backscatter NOMA systems: Reliability and security," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2723–2736, 2021.

[79] W. U. Khan, X. Li, M. Zeng, and O. A. Dobre, "Backscatter-enabled NOMA for future 6G systems: A new optimization framework under imperfect SIC," *IEEE Commun. Lett.*, vol. 25, no. 5, pp. 1669–1672, 2021.

[80] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Machine learning for wireless networks with artificial intelligence: A tutorial on neural networks," *arXiv preprint arXiv:1710.02913*, vol. 9, 2017.

[81] F. Hussain, S. A. Hassan, R. Hussain, and E. Hossain, "Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1251–1275, 2020.

[82] J. Wang, R. Li, J. Wang, Y.-q. Ge, Q.-f. Zhang, and W.-x. Shi, "Artificial intelligence and wireless communications," *Frontiers of Inf. Tech. Elec. Eng.*, vol. 21, pp. 1413–1425, 2020.

[83] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Commun.*, vol. 14, no. 11, pp. 92–111, 2017.

[84] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms and empirical evaluation," in *Machine Learning: ECML: 16th European Conference on Machine Learning, Porto, Portugal, October 3-7, Proc. 16.* Springer, 2005, pp. 437–448.

[85] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.

[86] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[87] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[88] C.-Y. Tang, C.-H. Liu, W.-K. Chen, and S. D. You, "Implementing action mask in proximal policy optimization (PPO) algorithm," *ICT Express*, vol. 6, no. 3, pp. 200–203, 2020.

[89] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Int. Conf. Mach. Learn. (ICML)*, Jul. 2018, pp. 1861–1870.

[90] C. He, Y. Hu, Y. Chen, and B. Zeng, "Joint power allocation and channel assignment for NOMA with deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2200–2210, Oct. 2019.

[91] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[92] S. K. Mahmud, Y. Liu, Y. Chen, and K. K. Chai, "Adaptive reinforcement learning framework for NOMA-UAV networks," *IEEE Commun. Lett.*, vol. 25, no. 9, pp. 2943–2947, 2021.

[93] E. Marchesini and A. Farinelli, "Discrete deep reinforcement learning for mapless

navigation," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 10 688–10 694.

[94] C. Zhong, Z. Lu, M. C. Gursoy, and S. Velipasalar, "A deep actor-critic reinforcement learning framework for dynamic multichannel access," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 4, pp. 1125–1139, Dec. 2019.

[95] Y. Sun, Y. Wang, J. Jiao, S. Wu, and Q. Zhang, "Deep learning-based long-term power allocation scheme for NOMA downlink system in S-IoT," *IEEE Access*, vol. 7, pp. 86 288–86 296, 2019.

[96] S. Han, X. Xu, L. Zhao, and X. Tao, "Joint time and power allocation for uplink cooperative non-orthogonal multiple access based massive machine-type communication network," *Int. J. of Dist. Sensor Net.*, vol. 14, no. 5, 2018.

[97] F. Fang, Z. Ding, W. Liang, and H. Zhang, "Optimal energy efficient power allocation with user fairness for uplink MC-NOMA systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 4, pp. 1133–1136, 2019.

[98] F. D. Ardakani and W. V. WS, "Joint reflection coefficient selection and subcarrier allocation for backscatter systems with NOMA," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, May 2020, pp. 1–6.

[99] Q. Zhang, L. Zhang, Y.-C. Liang, and P.-Y. Kam, "Backscatter-NOMA: A symbiotic system of cellular and Internet-of-Things networks," *IEEE Access*, vol. 7, pp. 20 000–20 013, 2019.

[100] J. Zuo, Y. Liu, Z. Qin, and N. Al-Dhahir, "Resource allocation in intelligent reflecting surface assisted NOMA systems," *IEEE Trans. Commun.*, vol. 68, no. 11, pp. 7170–7183, Nov. 2020.

[101] S. Gu, T. Lillicrap, Z. Ghahramani, R. E. Turner, and S. Levine, "Q-Prop: Sample-efficient policy gradient with an off-policy critic," *Proc. Int. Conf. Learn. Representations (ICLR)*, 2017.