

DYNAMIC SPECTRUM SHARING IN HETEROGENEOUS WIRELESS NETWORKS USING DEEP REINFORCEMENT LEARNING

BY

SULAIMON OYENIYI ADEBAYO

A Thesis Presented to the
COLLEGE OF GRADUATE AND INTERDISCIPLINARY STUDIES
KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN, SAUDI ARABIA

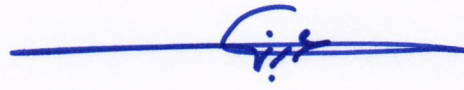
In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE
In
COMPUTER ENGINEERING

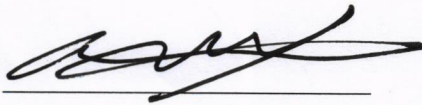
DECEMBER 2024

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN- 31261, SAUDI ARABIA
DEANSHIP OF GRADUATE STUDIES

This thesis, written by **SULAIMON OYENIYI ADEBAYO** under the direction of his thesis advisor and approved by his thesis committee, has been presented and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER ENGINEERING.**



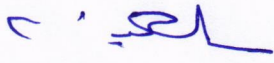
DR. ABDULAZIZ Y. BARNAWI
(Advisor)



DR. AIMAN EL-MALEH
Department Chairman



DR. TAREK RAHEL SHEL TAMI
(Member)



DR. SULIMAN S. AL-HOMIDAN
Dean of Graduate Studies



DR. MOHAMMED FELEMBAN
(Member)

Date



© Sulaimon Oyeniya ADEBAYO

2024

This thesis is dedicated to my family.

ACKNOWLEDGMENT

All praise and gratitude are due to the Almighty Allah for His infinite guidance, protection, and mercy throughout my life. His blessings have been my source of strength and perseverance. I extend my heartfelt appreciation to my parents and family members, whose unwavering support, prayers, and encouragement have been instrumental in my success. I am deeply grateful to my advisor, Dr. Abdulaziz Barnawi, for his invaluable guidance, constructive feedback, and continuous support during my research. My gratitude also goes to my committee members, Prof. Tarek Sheltami and Dr. Muhammed Felemban, for their insightful suggestions and contributions that enriched this work.

I wish to acknowledge the support and leadership of the Chairman of the Department of Computer Engineering, Dr. Aiman El-Maleh as well as the faculty and staff of the COE Department, for fostering an excellent academic and research environment. Special thanks to the KFUPM Administration for providing me with the opportunity and scholarship to pursue my Master's degree.

To my beloved wife, thank you for your understanding, patience, and immense support throughout this journey. I am also deeply indebted to my past lecturers and faculty members from my previous university, whose mentorship and encouragement laid the foundation for my academic pursuits. In particular, I would like to thank Dr. Munish Saini from Guru Nanak University, India, for his invaluable guidance and support throughout my Master's program.

To my friends and colleagues, both at KFUPM and beyond, thank you for your friendship, collaboration, and support during this challenging but rewarding journey. Your

companionship made the experience more enjoyable and motivating. I am equally thankful to the Nigerian community at KFUPM, whose camaraderie and encouragement made my stay more fulfilling, and to my fellow graduate students in the COE Department for their collaboration and shared experiences.

Lastly, I express my gratitude to the Interdisciplinary Research Center for Intelligent Secure Systems (IRC-ISS) for its resources and support that facilitated my research. To all who have contributed to this journey in any capacity, your efforts are sincerely appreciated.

TABLE OF CONTENTS

ACKNOWLEDGMENT	v
TABLE OF CONTENTS	vii
LIST OF TABLES	x
LIST OF FIGURES	xi
LIST OF ABBREVIATIONS	xiii
ABSTRACT.....	xvi
ملخص الرسالة.....	xviii
CHAPTER ONE	1
INTRODUCTION.....	1
1.1 OVERVIEW	1
1.2 BACKGROUND	4
1.2.1 Spectrum Sharing.....	7
1.2.2 Deep Reinforcement Learning.....	11
1.3 PROBLEM STATEMENT	14
1.4 RESEARCH OBJECTIVES.....	15
1.5 THESIS CONTRIBUTION	16
1.6 THESIS OUTLINE.....	16
CHAPTER TWO	18

LITERATURE REVIEW	18
2.1 REVIEW OF RELATED STATE OF THE ART LITERATURE	18
2.2 REVIEW OF BACKGROUND TECHNOLOGIES	29
2.2.1 LTE Coexistence With NB-IoT	29
CHAPTER THREE	37
METHODOLOGY	37
3.1 SYSTEM MODELING	38
3.1.1 Overview of the Selected Base Station for LTE and NB-IoT	40
3.1.2 Experimental Setup	42
3.1.3 LTE-Only Environment Scenario Modelling	44
3.1.4 NB-IoT-Only Environment Scenario Modelling	45
3.1.5 Heterogeneous Environment Scenario Modelling	47
3.1.6 LTE and NB-IoT Systems	48
3.2 REINFORCEMENT LEARNING MODEL	49
3.2.1 Reward Function	51
3.2.2 Q-Learning	51
3.2.3 RNN Construction	52
CHAPTER FOUR	55
RESULTS AND DISCUSSION	55
4.1 LTE AND DRL AGENT COEXISTENCE	57

4.1.1	Probabilities	59
4.1.2	Packets Expiry Duration	63
4.1.3	Learning Rates	65
4.1.4	Discount Factors (Gamma)	67
4.1.5	Fairness Index (F)	69
4.1.6	Recurrent Neural Network	71
4.2	NB-IoT AND DRL AGENT COEXISTENCE	74
4.3	LTE AND NB-IoT SYSTEMS COEXISTENCE	79
4.4	PERFORMANCE COMPARISON	86
CHAPTER FIVE		87
CONCLUSION AND FUTURE WORK		87
REFERENCES.....		90
VITAE.....		95

LIST OF TABLES

Table 1: Summary of Related Work.	24
Table 2: List of notations and their representation	43
Table 3: LTE-DRL parameters	57
Table 4: LTE and DRL with varied packet expiry duration (D) values from 1 to 9.....	64
Table 5: LTE and DRL with Varied Discount Factors.	68
Table 6: LTE and DRL with varied RNN configurations.....	73
Table 7: NB-IoT-DRL parameters.....	75
Table 8: LTE-NB-IoT Parameters	80

LIST OF FIGURES

Figure 1: 5G Network Slicing.....	5
Figure 2: DSS and Spectrum re-farming as NB-IoT demand increase.....	9
Figure 3: Illustration of Resource Allocation Pattern between LTE and NB-IoT	10
Figure 4: Illustration of MDP	12
Figure 5: NB-IoT Downlink Frame Structure	31
Figure 6: NB-IoT Frame Structure [36].....	32
Figure 7: LTE Structure	33
Figure 8: Illustration of DSS Deployment.....	35
Figure 9: Research Workflow	37
Figure 10: LTE and DRL spectrum sharing illustration	39
Figure 11: System Modeling.....	40
Figure 12: RNN Architecture	54
Figure 13: LTE and DRL Coexistence Throughput.	58
Figure 14: LTE and DRL with 100% Probabilities.	60
Figure 15: LTE and DRL with 50% Probabilities.	61
Figure 16: LTE and DRL with varied p_t values.	63
Figure 17: LTE and DRL with varied Learning Rate values.....	67
Figure 18: LTE and DRL with varied Fairness Index.	71
Figure 19: Architecture of Deep Recurrent Neural Network Model Used.....	72
Figure 20: NB-IoT and DRL Coexistence Throughput.	75
Figure 21: NB-IoT and DRL with 100% Probabilities.....	76
Figure 22: NB-IoT and DRL with p_t of 0.2.....	77

Figure 23: Impact of pt on DRL agent while coexisting with NB-IoT.....	78
Figure 24: NB-IoT and DRL each with 50% Probabilities.	79
Figure 25: LTE and NB-IoT Systems Coexistence Throughput.	80
Figure 26: LTE and NB-IoT Systems with 100% Probabilities	81
Figure 27: LTE and NB-IoT Systems with 0.9 and 0.2 Probabilities.....	82
Figure 28: LTE and NB-IoT Systems with 0.8 and 0.2 Probabilities.....	83
Figure 29: LTE and NB-IoT Systems with 0.8 and 0.1 Probabilities.....	84
Figure 30: LTE and NB-IoT Systems with 50% Probabilities.	85
Figure 31: Performance Comparison	86

LIST OF ABBREVIATIONS

3GPP:	Third-generation Partnership Project
ACEDA:	Advanced Cloud-based Double Auction
ACK:	Acknowledgment
AP:	Access Point
BS:	Base Station
CSMA/CA:	Carrier Sense Multiple Access with Collision Avoidance
DLMA:	Deep Reinforcement Learning Multiple Access
DQN:	Deep Q-Network
DRL:	Deep Reinforcement Learning
DRLA:	DRL Agent
DRLN:	DRL Node
DSA:	Dynamic Spectrum Access
emBB:	Enhanced or Extended Mobile Broadband
GRU:	Gated Recurrent Unit
IIoT:	Industrial Internet of Things
IoT:	Internet of Things
IoV:	Internet of Vehicles

ISM:	Industrial, Scientific, and Medical
ITS:	Intelligent Transportation System
LSTM:	Long Short-Term Memory
LTE:	Long Term Evolution
LTE-M:	Long-Term Evolution for Machines
MAC:	Media Access Control
MBIN:	Mobile Broadband Internet of Things
MBSFN:	Multimedia Broadcast Single Frequency Network
MCTS:	Monte Carlo Tree Search
MDP:	Markov Decision Process
MDQN:	Modified Deep Q-Network
mMTC:	Massive Machine Type Communication
MRL:	Multi-agent Reinforcement Learning
MSE:	Mean Squared Error
NACK:	No Acknowledgment
NB-IoT:	Narrowband Internet of Things
NSSS:	Narrowband Secondary Synchronization Signal
OFDMA:	Orthogonal Frequency Division Multiplexing

PSSAC: Power Spectrum Sensing and Allocation Control

PU: Primary User

RB: Resource Block

RE: Resource Element

ReLU: Rectified Linear Unit

RL: Reinforcement Learning

RNNs: Recurrent Neural Networks

SAC: Spectrum Allocation Control

SOC: Self-Organizing Coalition

SU: Secondary User

TD: Temporal Difference

TDD: Time Division Duplexing

TSRA: Tiny State-Space R-learning Random Access

UE: User Equipment

uRLLC: Ultra-Reliable Low-Latency Communication

ABSTRACT

Full Name : Sulaimon Oyeniya ADEBAYO

Thesis Title : Dynamic Spectrum Sharing in Heterogeneous Wireless Networks
Using Deep Reinforcement Learning

Major Field : Computer Engineering

Date of Degree : December, 2024

The rapid expansion of wireless networks demands efficient spectrum allocation. Dynamic Spectrum Sharing (DSS) is an effective technique for optimizing spectrum use, especially in heterogeneous environments. This thesis proposes an enhanced DSS technique based on Deep Reinforcement Learning (DRL). The proposed method enables an effective sharing of the available spectrum between two access technologies, namely LTE and NB-IoT. The study optimizes throughput through Q-learning and neural network-based methods, conducting experiments in three phases: LTE-DRL coexistence, NB-IoT-DRL coexistence, and LTE-NB-IoT coexistence. Results show that deep learning enhances the LTE-DRL system's convergence rate and throughput, achieving over 85% throughputs with convergence times as low as 24 microseconds (μ s). The study highlights the trade-offs between parameters such as probabilities (arrival, successful transmission, and retransmission), packet expiry duration, learning rate, discount factor, fairness index, and the neural network architecture and the parameters' impact on the overall system throughput. NB-IoT coexistence with DRL shows similar results with a slight decrement in throughput and negligibly longer convergence rate, while the coexistence of LTE and

NB-IoT results in throughput of around 70% for each of the LTE and NB-IoT systems due to increased spectrum competition and increased complexity of the operating environment. This work offers insights into optimizing spectrum sharing using DRL and underscores the balance between various parameters for efficient spectrum management.

ملخص الرسالة

الاسم الكامل : سليمان أوييني أديبايو
عنوان الأطروحة : المشاركة الديناميكية للطيف في الشبكات اللاسلكية المتنوعة باستخدام التعلم العميق المعزز
التخصص : هندسة الحاسوب
تاريخ الحصول على الدرجة : ديسمبر 2024

تتطلب التوسعات السريعة في الشبكات اللاسلكية تخصيص طيف ترددي فعال. تُعتبر تقنية المشاركة الديناميكية للطيف الترددي (DSS) تقنية فعالة لتحسين استخدام الطيف، خاصة في البيئات المتنوعة. تقترح هذه الرسالة تقنية محسّنة لـ DSS تعتمد على التعلم العميق المعزز (DRL). تمكّن الطريقة المقترحة من مشاركة فعالة للطيف المتاح بين تقنيتين للوصول، وهما LTE و NB-IoT. تعمل الدراسة على تحسين عرض النطاق الترددي من خلال طرق مبنية على Q-learning والشبكات العصبية. ويتم تقييم المقترح من خلال تجارب من ثلاث مراحل: التعايش بين LTE-DRL، التعايش بين NB-IoT-DRL، والتعايش بين LTE-NB-IoT. تظهر النتائج أن التعلم العميق يعزز معدل التقارب وعرض النطاق الترددي لنظام LTE-DRL، محققاً أكثر من 85% من عرض النطاق مع أوقات تقارب تصل إلى 24 ميكروثانية. تبرز الدراسة التوازنات بين المعلومات مثل الاحتمالات (الوصول، النقل الناجح، وإعادة النقل)، التأخير، معدل التعلم، عامل الخصم، مؤشر العدالة، ومعمارية الشبكة العصبية وتأثير المعلومات على عرض النطاق الكلي للنظام. يُظهر التعايش مع NB-IoT و DRL نتائج مماثلة مع انخفاض طفيف في عرض النطاق وزيادة غير ملحوظة في معدل التقارب، بينما يؤدي التعايش بين LTE و NB-IoT إلى عرض نطاق يقارب 70% لكل من نظامي LTE و NB-IoT بسبب زيادة المنافسة على الطيف وزيادة تعقيد بيئة التشغيل. تقدم هذه الدراسة رؤى حول تحسين مشاركة الطيف باستخدام DRL وتؤكد على التوازن بين المعلومات المختلفة من أجل إدارة فعالة للطيف.

CHAPTER ONE

INTRODUCTION

1.1 OVERVIEW

The need for portable, real-time, and more flexible modes of communication has led to the massive growth of wireless communication networks over time. This rapid growth has brought about a new age of services, ranging from 3G to 5G [1] and beyond. Spectrum allocation is a crucial aspect of wireless networks, playing a fundamental role in their efficiency and functionality. Unfortunately, the spectrum resources are limited and cannot afford further exponential growth [2], necessitating efficient and optimal spectrum sharing in wireless networks. Another problem dimension is a very wide margin between the licensed and unlicensed bands [3]. While the licensed bands are underutilized, the unlicensed bands—typically the Industrial, scientific, and Medical (ISM) bands—are overcrowded. Strategies to dynamically allocate the ISM bands and ensure higher throughput with little or no collision are a necessity of today's systems. Also, collaboration between licensed and unlicensed bands without much interference from the incumbent users will come in handy.

Dynamic Spectrum Sharing [4] and Dynamic Spectrum Access [5] have been established as major candidates to solve this problem. Dynamic Spectrum Access (DSA) and Dynamic Spectrum Sharing (DSS) are both approaches used to efficiently utilize the radio frequency spectrum. However, they differ in their fundamental concepts and goals. DSA primarily involves unauthorized access to temporarily vacant or underutilized spectrum bands. It makes spectrum bands accessible to secondary users when the primary, licensed users are not actively using them. It aims to maximize

spectrum utilization by allowing flexible access. DSS, on the other hand, focuses on sharing spectrum in a coordinated and dynamic manner among multiple users or services, both primary and secondary. It involves technologies and mechanisms for the efficient coexistence of multiple services within the same spectrum band. This work focuses mainly on DSS.

Traditional reinforcement learning (RL) faces significant challenges in wireless networks due to their dynamic and complex nature while deep reinforcement learning (DRL) enables faster learning of optimal strategies and greater robustness, even with suboptimal parameter settings [6]. DRL integrates deep learning with RL and offers significant advantages such as faster convergence rates and increased throughput. DRL algorithms converge to optimal solutions faster than traditional RL, especially when dealing with large state history lengths.

A fast convergence rate is crucial in wireless networks where channel conditions and traffic can change rapidly. For instance, a large state history length in wireless network communication helps make informed decisions. However, traditional RL struggles with this due to its tabular update rule, resulting in slow convergence. DRL, with its ability to handle high-dimensional state spaces, overcomes this limitation and converges much faster [6].

A 2019 study by Yu et al. [6] demonstrated this advantage. They compared an RL-based random-access scheme with a DRL-based scheme in a heterogeneous network and found that the DRL scheme converged to the optimal throughput significantly faster. DRL is more robust than traditional RL when dealing with non-optimal parameter settings [20]. This robustness can lead to an increase in overall system throughput which is essential in practical deployments where the optimal parameters

can vary depending on the coexisting networks and their characteristics. Additionally, we make a comparison between two models for our work, one with deep learning against the second one with traditional reinforcement learning. We established that deep reinforcement learning gives better performance than ordinary reinforcement learning.

Our system is designed such that each device communicates with the Base Station only and does not know the details of the coexisting networks. In a heterogeneous wireless environment, it is challenging to determine the optimal DRL parameters without prior knowledge of the coexisting networks. DRL's ability to perform well even with non-optimal settings makes it a more practical solution for such scenarios. By leveraging deep learning, DRL offers significant improvements over traditional RL in wireless networking applications. Its faster convergence and robustness make it a promising solution for addressing the challenges of dynamic and complex wireless environments.

We consider sharing multiple low and mid-frequency bands in a heterogeneous wireless environment where devices of different forms and capabilities, such as IoT nodes and mobile devices, among others, are used. We aim to build an intelligent reinforcement learning agent that will optimize the spectrum usage by maximizing throughput and fairness in the channel while utilizing the spectrum fully and equally sharing the resources among contending nodes. The main Agent, the Deep Reinforcement Learning Agent (DRLA), will coordinate the network activities using the Access Point (base station). The node operating the agent, called Deep Reinforcement Learning Node (DRLN), will be deployed alongside other nodes in the network.

In evaluating the performance of dynamic spectrum sharing with the proposed DRL and one of its algorithms (Deep Q-Network), several key metrics are essential.

Throughput, a measure of the data transmission rate, reflects the efficiency of the spectrum utilization, directly impacting the speed of communication and data transfer. Fairness-index, a parameter governing resource allocation equity, ensures that spectrum resources are shared equally among contending nodes, contributing to a balanced and fair distribution of resources. Fast convergence is also an important metric and crucial in wireless networks, particularly as environmental conditions change rapidly with the arrival of new nodes [6], emphasizing the need for the system to adapt quickly for optimal performance. Robustness assesses the agent's ability to navigate various environments and remains effective against non-optimal parameter settings. This ensures reliable and stable performance across diverse scenarios. These performance metrics collectively address the multifaceted challenges of dynamic spectrum sharing, encompassing efficiency, fairness, adaptability, and stability within the dynamic wireless landscape.

1.2 BACKGROUND

This research considers the network-slicing features of 5G, which divides a single physical network into virtualized sizes to guarantee Service Level Agreement regarding Throughput, Latency, and Reliability. Service type in network slicing can be divided into at least 3 primary cases;

- Enhanced or extended Mobile Broadband (eMBB) – live broadcast, VR/AR
- Ultra-reliable low latency communication (uRLLC) – automotive, health & wellness
- Massive Machine type communication (mMTC) – logistics, utilities

The use case considered in our research is the mMTC which focuses on connecting a massive number of IoT devices and sensors with low data rates and high scalability and efficiency. The typical bandwidth is low to moderate (e.g. a few Kbps to Mbps) depending on the IoT device and application requirements. Delay varied based on targets, specifically depending on the IoT use case ranging from milliseconds to seconds, emphasizing optimizing energy consumption and network efficiency.

Spectrum sharing aims to optimize spectrum utilization. Two key innovations achieving this are network slicing and dynamic spectrum sharing (DSS). Network slicing is a form of network virtualization that leverages software-defined networks and network function virtualization to create multiple virtual networks (slices) on a single physical infrastructure. This allows operators to tailor each slice to specific service requirements.

By offering customized slices, operators can efficiently use the spectrum to support diverse services with varying needs, ultimately maximizing the value and efficiency of their network resources. The pictorial representation of the available slicing is shown in Figure 1¹.

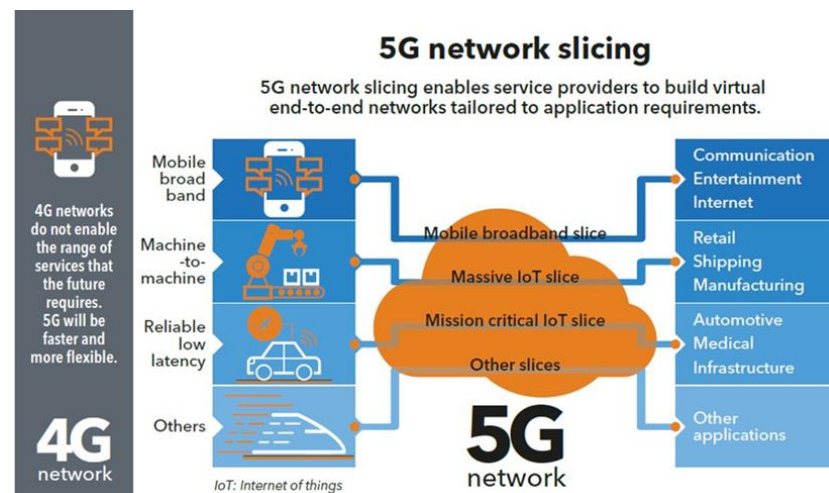


Figure 1: 5G Network Slicing

¹ [Use of shared spectrum at the national level | Digital Regulation Platform](#)

However, the latest applications introduced new trends and behaviors that require more than throughput to support mMTC. In light of this, the third-generation partnership project (3GPP) has introduced two innovative cellular IoT technologies enabling mMTC: NB-IoT and LTE-M [7]. This research focused extensively on these two new technologies, though we consider LTE, which is a broader category of LTE-M.

It is worth noting that the requirements of these technologies are similar to those of mMTCs. For example, NB-IoT has bandwidth specifically tailored for low-power, wide-area IoT applications with intermittent data transmission and long battery life requirements. The spectrum allocation (e.g., 180kHz) is optimized for low data rates (e.g., 10s of Kbps) and is suitable for IoT sensor readings, status updates, and telemetry data. Finally, the delay targets also depend on the IoT application, typically ranging from seconds to minutes, emphasizing energy-efficient communication and extended coverage. It is safe to say that Narrowband (IoT) is a competing Massive Type Communication (5G) technology [8].

This research aims to have a system that can perform well in complex wireless environments such as the Internet of Things (IoT) and the Industrial Internet of Things (IIoT). In the context of IoT or IIoT environments, dynamic spectrum sharing (DSS) faces unique challenges. The proliferation of connected devices in these environments intensifies the demand for efficient spectrum utilization. The diversity of communication protocols and the coexistence of various devices operating on different technologies amplify the complexity of spectrum management. The sheer volume of IoT/IIoT devices introduces a dynamic and heterogeneous ecosystem, making it challenging to coordinate spectrum allocation effectively. Moreover, the intermittent and bursty nature of IoT/IIoT communication patterns demands adaptable and

intelligent spectrum-sharing mechanisms to optimize resource utilization. Security and privacy concerns also emerge in these environments, necessitating robust and resilient DSS solutions to safeguard sensitive data transmissions. Additionally, the scalability of DSS solutions becomes crucial in accommodating the ever-expanding IoT/IIoT networks. It is imperative to address these issues to guarantee the seamless integration and sustainable performance of dynamic spectrum sharing within IoT and IIoT ecosystems.

1.2.1 Spectrum Sharing

Through the safe sharing of the same frequency bands among various user categories, spectrum sharing maximizes the utilization of airwaves or wireless communications channels². This is especially pertinent in the context of wireless networks, including 4G, 5G, and future generations of wireless technologies. Multiple techniques can be used to share the spectrum. These techniques include the open sharing model, hierarchical access model, interweave, overlay, underlay, cooperative, and dynamic spectrum sharing [9]. These techniques have their use case and multiple factors they consider or ignore. It is safe to say that the choice of techniques depends to do with the application and goals one has in mind. Different factors to be considered when selecting spectrum-sharing techniques include possible improvements in spectrum efficiency, interference prevention, network congestion, and mobility support [9]. Dynamic Spectrum Sharing takes care of most of these factors, which makes it one of the most optimal spectrum-sharing techniques [10].

² [Spectrum Sharing | NIST](#)

Dynamic spectrum sharing (DSS) allows multiple technologies like LTE and NB-IoT to coexist and dynamically share the same frequency spectrum. This is particularly important for efficiently managing limited spectrum resources and supporting diverse communication needs.

Figure 2 illustrates the DSS compared to the traditional spectrum re-farming approach. Figure 2 (a) shows spectrum re-farming, where a fixed portion of the 20 MHz bandwidth is permanently allocated to LTE at the initial stage. The spectrum is statically split into two bands—10 MHz for LTE and 10 MHz for NB-IoT. This is done to accommodate NB-IoT traffic but leaves fixed allocations for each technology. As NB-IoT demand grows (shown by the blue arrow), the fixed allocation becomes insufficient, leading to limited NB-IoT resources and potentially slower speeds or congestion. LTE, meanwhile, might have underutilized spectrum.

Figure 2 (b) shows Dynamic Spectrum Sharing. Here, both LTE and NB-IoT can access the entire 20 MHz bandwidth. The entire spectrum (20 MHz) is shared dynamically between LTE and NB-IoT using DSS technology. Resources are allocated dynamically based on real-time traffic demand. If NB-IoT traffic is low, more spectrum is available for LTE. As NB-IoT demand increases, spectrum is shifted towards NB-IoT. The system dynamically allocates resources based on real-time demand. When NB-IoT traffic is low, LTE can use most of the spectrum. As NB-IoT demand increases, the system automatically assigns more resources to NB-IoT, ensuring efficient utilization.

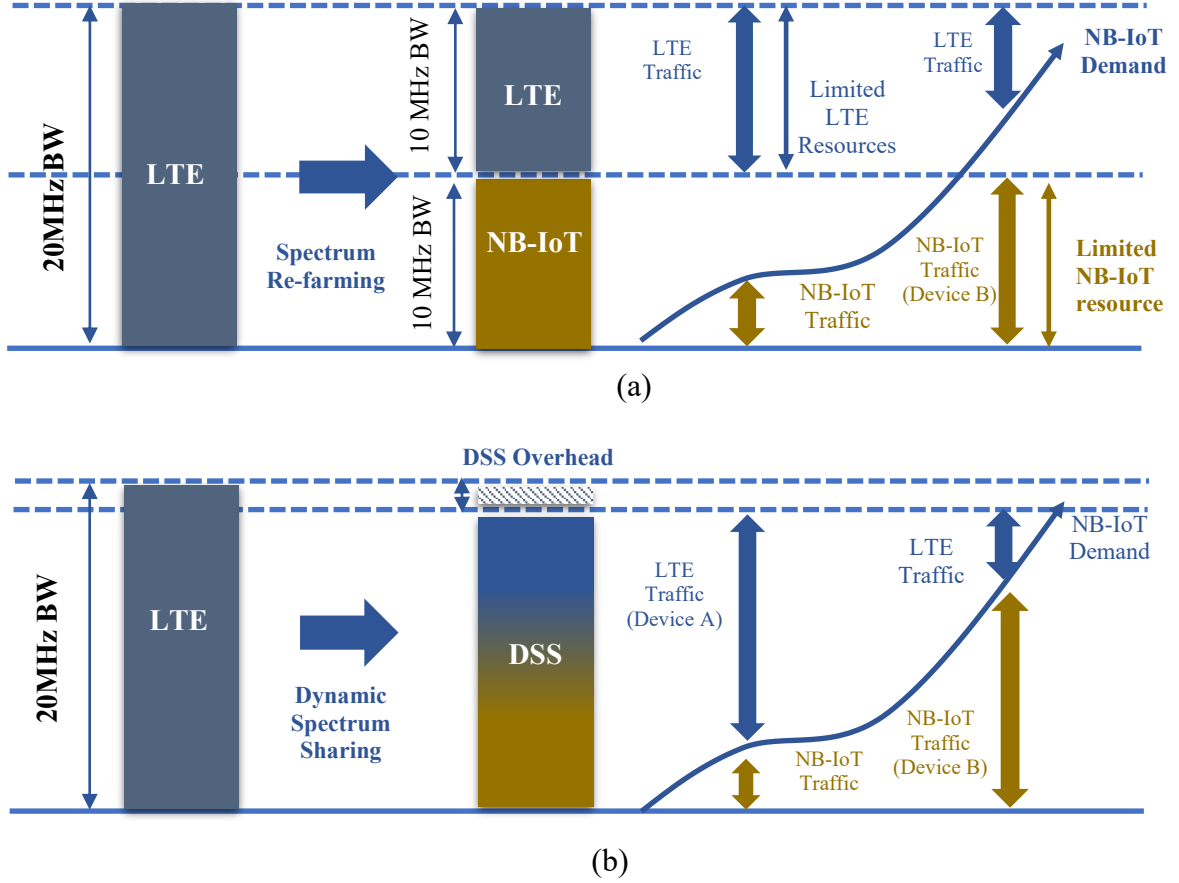


Figure 2: DSS and Spectrum re-farming as NB-IoT demand increase

We can imagine an interactive scenario between two devices: Device A: An LTE-enabled smartphone and Device B: An NB-IoT (Narrowband Internet of Things) sensor device.

- **Initial State:** Both devices are in an area with a dynamic spectrum sharing (DSS)-enabled network. Device A (LTE) is using the majority of the bandwidth, as the NB-IoT device (Device B) has minimal traffic needs, primarily sending small packets of data intermittently.
- **Increased Demand:** Device B starts transmitting a burst of data, such as sensor readings for an urgent application (e.g., monitoring critical environmental changes), increasing the demand for NB-IoT resources.

- **Dynamic Adjustment:** The DSS-enabled network detects the increased demand from Device B and dynamically reallocates spectrum. It assigns more resources to NB-IoT to support the sudden traffic increase, while adjusting LTE's allocation accordingly.
- **Shared Resources:** Device A's LTE bandwidth is slightly reduced but remains functional for standard smartphone activities. Both devices share the spectrum efficiently, ensuring seamless operations for both LTE and NB-IoT traffic demands.

Figure 3 illustrates the resource allocation pattern between LTE and NB-IoT. It shows how available radio frequency resources are divided over time proportionally and dynamically, ensuring coexistence and optimized utilization for both technologies. NB-IoT receives 40% and 70%, while LTE receives 60% and 30% in the first and second scenarios, respectively. The blue represents LTE, the gold represents NB-IoT, while the green indicates slot Type B. The system dynamically adjusts the resource allocation between LTE and NB-IoT within each 10ms frame. The technology with more traffic gets more sub-frame allocation. This dynamic adjustment ensures efficient spectrum utilization and caters to the varying demands of both technologies. The sharing ratios (e.g., 60% LTE / 40% NB-IoT and 30% LTE / 70% NB-IoT) can be adjusted dynamically based on traffic demand, ensuring efficient use of resources.

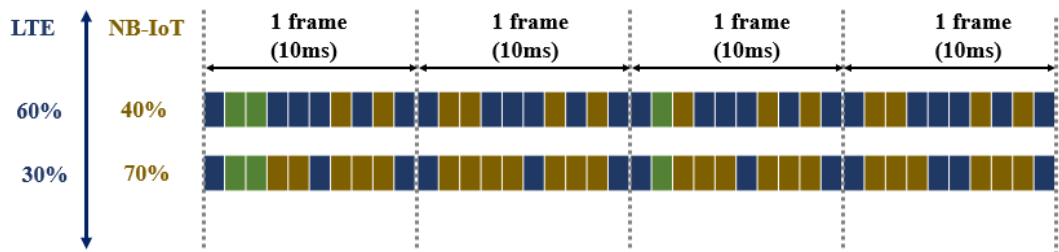


Figure 3: Illustration of Resource Allocation Pattern between LTE and NB-IoT

Overall, dynamic spectrum sharing is a valuable technique for enabling efficient and flexible use of spectrum resources, supporting the coexistence of LTE and NB-IoT to meet the growing demands of diverse wireless communication applications. Though, DSS introduces a small overhead for managing the dynamic allocation, it enables flexible spectrum use, aligning resource availability with actual demand for LTE and NB-IoT services.

1.2.2 Deep Reinforcement Learning

Reinforcement learning (RL) is a branch of machine learning that is concerned with teaching agents how to make decisions in an environment to accomplish a goal. The idea of using reinforcement learning has been around for over 2 decades [11], but it started finding its way into state-of-the-art applications recently. RL has 3 main components; Agent, Environment, and Reward Function.

The agent is the learner or decision-maker interacting with the environment. It observes the present state of the environment and takes actions to maximize a cumulative reward. The external system the agent interacts with is called the environment. The environment is the context in which the agent operates and where the actions have consequences. The environment's state can change in response to the agent's actions. The agent receives feedback from the reward function regarding the quality of its actions. It assigns a numerical reward to each action taken by the agent. The agent's objective is to maximize the total rewards that accrue over time.

The agent interacts with the environment discretely [12]. An RL agent will interact with the environment in a discrete time such as $\{0,1,2 \dots\}$. At each time step t , the agent performs an action a_t from an action space $\mathcal{A}$, following policy $\pi(a_t|s_t)$ which is an action-reward mapping from states s_t to action a_t . With this, it receives a reward r_t and

transits to the next state s_{t+1} , based on the model and the environment dynamics for state transition probability $\mathcal{P}(s_{t+1}|s_t, a_t)$ and reward function $\mathcal{R}(s, a)$ respectively. This procedure is carried out in an episodic problem until the agent achieves a terminal state, which is restarted [13]. The return $\mathcal{R}_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$ is the discounted accumulated reward with the discount factor gamma belonging to the interval $\{0,1\}$.

One of the easiest ways to implement an RL is to model it as a Markov Decision Process -MDP [11], which is simply a combination of states, actions, and rewards. The visual representation of MDP as adapted from Wikipedia³ is given in Figure 4 below. The agent can move from states $\{S_0, S_1, S_2\}$ with probabilities ranging from 0 to 1 by performing actions $\{a_0, a_1\}$ and getting a reward of either -1 or +5.

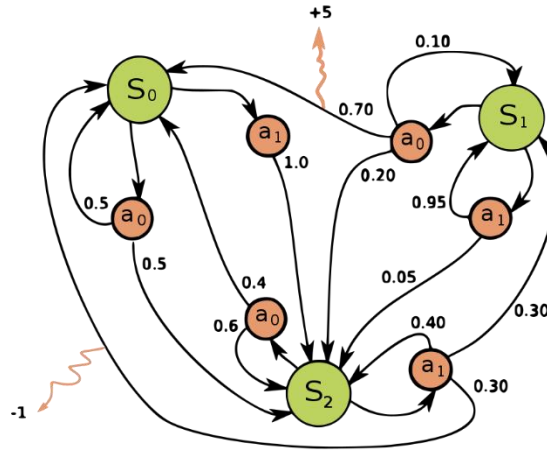


Figure 4: Illustration of MDP

Additionally, the RL algorithms can be categorized into two main types: Value-Based Methods and Policy-Based Methods. The Value-based methods aim to learn the value of being in a particular state or taking a particular action. They include algorithms like Q-learning and Deep -Networks (DQN). Policy-based, on the other hand, focuses on

³ [File:Markov Decision Process.svg - Wikipedia](#)

directly learning the policy, which is the mapping of states to actions. An example of this approach includes various policy gradient algorithms (e.g. REINFORCE).

This study uses DQN, a DRL algorithm that integrates deep neural networks with Q-learning. Q-learning maintains a data structure table known as a Q-table that stores the estimated value (Q-value) for each state-action pair. The Q-value denotes the expected cumulative reward an agent can attain by taking a specific action in a particular state. It is based on the Bellman equation, as given in equation 1, which relates the Q-values of state-action pairs to the Q-values of future states and actions.

$$Q(s, a) = (1 - \alpha) * Q(s, a) + \alpha * [R + \gamma * \max(Q(s', a'))] \dots \text{equation 1}$$

Where:

$Q(s, a)$ = Q-value for state and action pair

α = learning rate, which controls the amount each iteration updates the Q-value.

R = instant reward obtained after taking action a in state s

γ = discount factor, which represents the importance of future rewards.

s' = next state, and a' is the action chosen in the state.

Utilizing an exploration-exploitation approach, Q-learning strikes a balance between exploring new actions and using the best-known actions. Common strategies include epsilon-greedy, in which the agent uses a predetermined probability to explore randomly and take advantage of the best action otherwise.

The limitation of Q-learning comes with complex environments where the table values are extremely large. DQN, by incorporating deep learning, can take care of this. It uses

deep learning to approximate the value function, which enables it to handle high-dimensional or continuous spaces. Deep reinforcement learning, at its simplest definition, is the incorporation of deep learning into reinforcement learning. This is particularly useful in real-world applications where the state space is too vast for a tabular representation. Also, it was established that using DRL with the aid of DQN achieves fast convergence to almost optimal solutions and robustness against suboptimal parameter configurations [6]. This implies that fine-tuning parameters and optimization are unnecessary with DRL. The pseudocode of the original DQN was given by [14]. Our agent is based on DQN but with some modifications tailored around our environment and goals.

1.3 PROBLEM STATEMENT

The spectrum is a fixed and limited resource, with only a specific range of frequencies available. Once a portion of the spectrum is allocated, its availability for others becomes restricted. To address this limitation, a dynamic and intelligent allocation system is essential for managing these scarce resources effectively.

The rapid growth of wireless network applications has made static spectrum allocation insufficient, necessitating a dynamic and efficient approach in sharing the spectrum among multiple devices in a network. This challenge is further amplified with the advent of new technologies in 5G mobile networks. In particular, the low and mid-band frequencies, which are heavily utilized, are becoming oversaturated—especially in developed regions. This leaves high-band frequencies as the only alternative, but these have a much shorter range, posing additional limitations.

The previous works focused primarily on homogenous environments or traditional communication technologies like TDMA and ALOHA. The primary goal of this work

is to develop a system that uses reinforcement learning (RL) to dynamically and efficiently share frequencies between legacy and emerging technologies. Because of the drawback of traditional reinforcement learning as discussed in earlier sections and as demonstrated in the *Performance Comparison Section*, we bring deep learning to the picture and use deep reinforcement learning for our work. Also, traditional wireless Media Access Control (MAC) protocols are often designed for homogeneous environments [12], making them inefficient in heterogeneous setups. This study focuses on enabling effective spectrum utilization in such heterogeneous environments, where multiple wireless communication technologies coexist. Specifically, this work leverages DRL to facilitate spectrum sharing between NB-IoT and LTE technologies, ensuring efficient allocation and usage of resources among diverse devices and applications.

1.4 RESEARCH OBJECTIVES

This research work aims to propose an algorithm that dynamically allocates spectrum resources among multiple devices operating on one or more wireless communication technologies using deep reinforcement learning. LTE and NB-IoT are considered as use cases. The specific objectives are to:

- Perform a comprehensive review of the state-of-the-art mechanism for spectrum sharing in wireless networks.
- Develop a DRL model (agent) that uses DRL to navigate different wireless communication technologies in a heterogeneous environment. The DRLA will maximize throughput, robustness, and convergence rate in the environment.
- Model each considered technology separately and merge them into a single environment.

- Perform simulation analysis and comparisons with existing spectrum-sharing approaches.

1.5 THESIS CONTRIBUTION

The main contributions of this thesis include:

- Proposing a unique approach for dynamic spectrum sharing (DSS) in heterogeneous wireless networks by employing deep reinforcement learning (DRL).
- Modeling an effective sharing of the available spectrum between two access technologies, Long Term Evolution (LTE) and Narrowband IoT (NB-IoT), to optimize throughput.
- Demonstrates that deep learning enhances the LTE-DRL system's convergence rate and throughput.
- Analyses the trade-offs among parameters such as probabilities (arrival, successful transmission, and retransmission), packet expiry duration, learning rate, discount factor, fairness index, and the neural network architecture and their impact on overall system throughput.
- Provide insights for optimizing spectrum sharing in the rapidly evolving landscape of wireless communication technologies.

1.6 THESIS OUTLINE

The rest of the thesis is arranged as follows: Chapter two includes a review of related literature and the background technologies related to spectrum sharing using the modern machine learning approach. Chapter 3 covers the methodology used, including

the experimental stages and their setups. The results obtained are extensively discussed in chapter four, also known as the result and discussion chapter. The thesis is rounded up in chapter five with the conclusion and some recommendations for future work.

CHAPTER TWO

LITERATURE REVIEW

This section reviews prior research endeavors that have employed Deep Reinforcement Learning (DRL) for spectrum allocation in various communication and network environments. The chapter is divided into two parts: the first focuses on literature related to the research, and the second reviews related technologies used in the study.

2.1 REVIEW OF RELATED STATE OF THE ART LITERATURE

There have been several works related to spectrum sharing in wireless environments, ranging from the traditional ways of sharing the spectrum to the modern way that employs machine learning. We focus only on notable literature that uses the modern approach (machine learning), specifically deep reinforcement learning, for sharing the spectrum. The detailed insights into each work, as outlined in Table 1, encompass the contributions of different authors, the methodologies they adopt, the specific technologies or domains under focus, the characteristics of the environments considered, and an evaluation of the strengths and limitations, with potential avenues for expansion.

In the study by Liu et al. [15] in 2023, a novel approach utilizing a Deep Reinforcement Learning (DRL)-based training method is proposed for secondary users within communication systems. The primary objective is to control transmission power and facilitate spectrum sharing effectively. The methodology involves the implementation of Deep Q-Network (DQN) and Deep Recurrent Q-Network structures, focusing on the realm of Dynamic Spectrum Access (DSA) and DRL. The research investigates a

cognitive radio system, considering multiple Primary and Secondary Users (PUs & SUs). Notable strengths of this approach include an increase in system efficiency, a decrease in PU interference, and a superior performance compared to OMC-ALOHA in reward outcomes. However, it is crucial to acknowledge the study's limitations, including a somewhat limited baseline comparison (OMC-ALOHA) and simplifying assumptions that may not accurately represent real-world cognitive radio systems.

Cha & Kim [5] in 2019, contributed to the field with a reinforcement learning (RL)-based spectrum sensor-aided Dynamic Spectrum Access (DSA) system, aiming to optimize spectrum consumption in Internet of Things (IoT) networks. The proposed methodology incorporates RL algorithms and sensor-aided DSA systems, presenting a focused analysis of a network with transmitters, receivers, and an existing user network. The study demonstrates the strength of the RL-based DSA system in outperforming the distributed CSMA/CA protocol in channel access coordination. However, the authors highlight the need for careful consideration of system architecture in real-world implementations to ensure the practicality and effectiveness of the proposed approach.

Shi et al. [2] present a noteworthy contribution in 2020 by proposing a solution for sharing spectrum resources in an Internet of Things (IoT) network. The emphasis is on enabling limited spectrum sharing among various sensors, and the methodology involves formulating a new reward function and compressing the action space to enhance the learning effectiveness of the DQN algorithm. The study explores the intersection of Dynamic Spectrum Access (DSA), Reinforcement Learning (RL), and IoT in an uplink wireless network. The significant strength lies in achieving low complexity with commendable performance. However, the authors acknowledge the

necessity of addressing power allocation concerns in future works to further refine the proposed solution.

Continuing the exploration of spectrum management and optimization, [16] proposes a spectrum resource management system within an Industrial Internet of Things (IIoT) environment. The study introduces the Modified Deep Q-Network (MDQN), emphasizing the importance of condensing the action space and suggesting a priority experience replay method based on reducing temporal difference (TD) error. The research examines an uplink wireless IIoT network with shared orthogonal sub-channels and sensors, showcasing good performance and fast convergence speed. Despite these strengths, the authors highlight computational complexity as a potential drawback, along with the need to consider power allocation in future works.

Wei et al. in 2023 [17] delve into spectrum resource allocation issues within Intelligent Transportation Systems (ITS) by employing Deep Reinforcement Learning (DRL). The study introduces a Spectrum Allocation Control (SAC)-based scheme, incorporating Dynamic Spectrum Access (DSA), Self-Organizing Coalition (SOC), Power Spectrum Sensing and Allocation Control (PSSAC), Multi-agent Reinforcement Learning (MRL), and the Internet of Vehicles (IoV). The research is recognized for achieving lower complexity. However, the observed environment might not be entirely realistic, and the study did not consider the complexities introduced by real ITS environments, which often involve weather, passersby, and variable vehicle density.

A proactive dynamic spectrum-sharing approach between 4G and 5G systems was proposed by Challita & Sandberg in 2021 [18]. The controller used by the authors determines how much resource to allocate between NR and LTE in each subframe while taking future network conditions like multimedia broadcast single frequency network

(MBSFN) subframes and high interference subframes into consideration. They solve this problem by proposing a DRL algorithm based on Monte Carlo Tree Search (MCTS). By simulating hypothetical bandwidth splits over time, starting from the current network state, the controller of the suggested deep reinforcement learning architecture is trained offline and can forecast a series of future states of the wireless access network. Their simulation results affirm that the proposed scheme can operate while taking future states into account.

Xu et al. in 2024 [19] propose an innovative Mobile Broadband Internet of Things (MBIN) architecture that is designed to facilitate efficient progress in the IoT revolution and especially focuses on intelligent dynamic spectrum sharing. The authors also introduced an intelligent Advanced Cloud-based Double Auction (ACEDA) mechanism that aids in fair spectral resource sharing within the MBIN framework. Their mechanism outperforms conventional structures in terms of spectrum and energy efficiency. Future work includes further exploration and optimization of the proposed ACEDA mechanism.

Examining a heterogeneous wireless network in 2019, Yu et al. [6] contributed to the literature by constructing a MAC protocol based on Deep Reinforcement Learning (DRL). This research incorporates various elements such as DRL, Deep Q-Network (DQN), 4G technologies, 5G technologies, and Medium Access Control (MAC). The proposed MAC protocol proves applicable in both homogeneous and heterogeneous environments, showcasing versatility. However, the study assumes nodes are saturated and always have something to transmit. The resource sharing is simplified to one-dimensional time slots, representing a potential limitation in certain real-world scenarios.

Motivated by the two novel IoT technologies released by the third-generation partnership project (3GPP) to support mMTC, also known as NB-IoT and LTE-M, Abou El Hassan et al. [7] in 2021 explore the performance analysis of NB-IoT and LTE-M. The authors determine the system configuration and deployment necessary for LTE-M and NB-IoT technologies to satisfy the 5G mMTC requirements regarding connection density, latency, coverage, throughput, and battery life. After a complete evaluation of NB-IoT and LTE-M against 5G mMTC requirements, the authors concluded that these requirements can be met only under certain circumstances regarding system configuration and deployment. The researchers established that The IoT technology with the best throughput, latency, and battery life is NB-IoT. In contrast, the most effective technology for IoT applications that require correspondingly faster reaction times is LTE-M.

In 2021, a delay-constrained, heterogeneous wireless network's random-access problem was investigated by Wu et al [20]. The authors consider a network where two users use an unreliable collision wireless channel to send delay-constrained traffic to an access point (AP). To quantify the performance gap of any suggested schemes, they first present a Markov Decision Process (MDP) formulation to derive a model-based upper bound. Afterward, they use reinforcement learning to create Tiny State-space R-learning random Access (TSRA), an R-learning-based random access technique. According to their simulation results, TSRA outperforms the current baseline Deep Reinforcement Learning Multiple Access (DLMA) and achieves close-to-upper-bound performance.

In conclusion, the reviewed literature demonstrates the growing interest in applying DRL to spectrum allocation problems in diverse communication and network

environments. While the approaches show strengths in enhancing system performance and reducing interference, they often make simplifying assumptions about the environments, which may limit their real-world applicability. Also, very few works consider heterogenous settings while the others assume the nodes are operating on the same MAC, this might not be feasible in certain networks such as the IoT and IIoT. This research plans to address these drawbacks. The summary of the reviewed work is given in Table 1 below.

Table 1: Summary of Related Work.

Authors	Work Summary	Approaches	Technology/ Domain	Environment	Strength	Drawback/Future Works
Liu et al., (2023) [15]	Propose a DRL-based training method to create a plan for secondary users in the communication system to control their transmission power and share the spectrum.	Neural networks are constructed using the Deep Q-Network and Deep Recurrent Q-Network architectures.	DSA, DQN, DRL	Examines M PUs and N SUs, each having a transmitter and receiver, in a cognitive radio system. If a channel is not being used by a PU, SUs can only access it for data transmission. PUs have a higher priority in the system and own the channels.	Increased system efficiency and decreased PU interference. Exceeds OMC-ALOHA in the reward by roughly 10% and 30%, respectively, in the case of a single SU and the situation with several SUs.	Limited comparison baseline: Only compares with OMC-ALOHA. Simplifying Assumptions: Some of the mentioned assumptions about the CR systems might not hold in real-world scenarios.
Cha & Kim, (2019) [5]	Propose a reinforcement learning RL-based spectrum sensor-aided DSA system to optimize IoT network spectrum consumption	Introduce sensor-aided DSA systems that improve spatial spectrum reusability by using RL algorithms to support the DSA of IoT devices.	DSA, RL, IoT	Consider a DSA IoT network in which transmitters and receivers connect via an existing user network. Define the IoT network as the secondary user network and the	For channel access coordination, the suggested RL-based DSA system performs better than the distributed CSMA/CA protocol. Additionally, the incumbent user has more reliable performance when	System architecture must be taken into account when implementing the RL-based DSA system in actual wireless communication networks in the absence of a dedicated control channel.

Authors	Work Summary	Approaches	Technology/ Domain	Environment	Strength	Drawback/Future Works
	over the current network.			incumbent user network as the primary user network.	using the IoT device's Spectrum band access with acquired spectrum access probability.	
Shi et al., (2020a) [2]	To enable limited spectrum sharing amongst various types of sensors, propose a solution for sharing spectrum resources in the Internet of Things network.	Initially, a new reward function is formulated to facilitate the learning process, considering the distinct communication needs of diverse sensors. After that, the action space was compressed to raise DQN's learning effectiveness.	DSA, RL, DQN, IoT	An uplink wireless Internet of Things network was examined. A BS, a set $N = \{1, 2, \dots, N\}$ of shared orthogonal sub-channels, and a set $K = \{1, 2, \dots, K\}$ of sensors are present.	Low complexity with good performance.	Power allocation is to be considered in future works.
Shi et al., (2020b) [16]	Propose a spectrum resource management system in an IIoT environment to enable limited spectrum sharing between various UE types.	To make environment state observation easier, a basic MAC frame structure and a realistic IIoT model were first constructed. Next, to increase learning efficiency, they condense the	MDQN, DRL, IIoT	An uplink wireless Industrial Internet of Things network was examined. A BS, a set $N = \{1, 2, \dots, N\}$ of shared orthogonal sub-channels,	Good performance and fast convergence speed	Computational Complexity. Power allocation is to be considered in future works.

Authors	Work Summary	Approaches	Technology/ Domain	Environment	Strength	Drawback/Future Works
		action space and suggest a priority experience replay method based on reducing temporal difference (TD) error.		and a set $K = \{1, 2, \dots, K\}$ of sensors are present.		
Wei et al., (2023) [17]	Employing spectrum sensing technologies in the ITS, investigate the DRL's spectrum resource allocation issues.	The SAC-based scheme was proposed to address the complex vehicular environment's spectrum allocation issues.	DSA, SOC, PSSAC, DRL, MRL, IoV	Consists of X V2I links, Y V2V links, several automobiles, and a single base station.	Lower complexity	Did not consider the Real ITS environment which often has weather, passerby, and vehicle density
Challita & Sandberg (2021) [18].	Using Monte Carlo Tree Search (MCTS) and a deep reinforcement learning algorithm, propose a proactive dynamic spectrum sharing approach for 4G and 5G networks.	The research introduces a deep RL algorithm based on MCTS to manage spectrum sharing between LTE and NR.	LTE, NR, DRL, MCTS	The RL environment models a wireless system where LTE and NR share bandwidth in the 3.5 GHz band. It optimizes resource block allocation based on channel conditions and traffic models.	The proposed framework intelligently manages spectrum sharing by considering future network states, improving system-level performance.	Future research can focus on addressing uncertainties in observations to enhance the algorithm's robustness.

Authors	Work Summary	Approaches	Technology/ Domain	Environment	Strength	Drawback/Future Works
Xu et al., 2024 [19].	Introduces the MBIN architecture, designed to meet the spectrum management needs of emerging 5G IoT networks. The research has the ACEDA algorithm at its core which optimizes the spectrum sharing.	The ACEDA algorithm is implemented within the MBIN framework to ensure efficient allocation.	MBIN, ACEDA, 5G, IoT	Uses a multi-agent system where each agent (IoT service) bids for spectral resources based on its state, which includes current resource availability and bid status.	The ACEDA algorithm improves spectral and energy efficiency, outperforming traditional methods. The architecture is also scalable and suitable for large IoT networks.	The ACEDA algorithm can be further optimized for better efficiency. Other algorithms can be used and compared as well.
Yu et al., (2019) [6].	Examines a heterogeneous wireless network MAC protocol based on deep reinforcement learning (DRL).	Constructs a framework to take advantage of DRL for creating a MAC protocol for a heterogeneous network.	DRL, DQN, ALOHA, TDMA, MAC	A heterogeneous environment with an access point that serves as the base station and at least one node operating on DLMA, TDMA, ALOHA	Can work in both homogenous and heterogenous environments	They assume the nodes are saturated and always have something to transmit. The resource sharing is also one-dimensional of time slots only.
Abou El Hassan et al., (2021) [7].	Evaluate NB-IoT and LTE-M for 5G mMTC requirements,	Accesses system configurations for NB-IoT	NB-IoT, LTE-M, mMTC, 5G	The study is set in a 5G network context, analyzing NB-IoT and	The study offers a detailed performance evaluation of NB-IoT and LTE-M,	Performance varies with deployment specifics, which may limit general applicability.

Authors	Work Summary	Approaches	Technology/ Domain	Environment	Strength	Drawback/Future Works
	establishing that both technologies can meet these under specific conditions.	and LTE-M to meet 5G mMTC targets.		LTE-M performance based on deployment conditions like base station density and channel transmission.	highlighting their respective advantages.	Future optimization strategies for both technologies could be explored.
Wu et al., 2021 [20].	Addresses the random access problem in delay-constrained heterogeneous wireless networks with two users and a common collision-prone channel	The study formulates the problem using a Markov Decision Process (MDP) to set a model-based upper bound for performance.	MDP, TSRA, RL, DLMA	The RL environment models a two-user wireless system with delay-constrained traffic, where one user uses slotted ALOHA and the other's transmission policy is optimized via RL.	The paper introduces a novel approach to random access in delay-constrained networks, with TSRA achieving near-optimal performance.	Random access for multi-user networks could be explored for future research. Also, further investigation is needed to refine the approach for broader applications and more complex network scenarios.

2.2 REVIEW OF BACKGROUND TECHNOLOGIES

The Internet of Things (IoT) has taken over major activities in the connected living environment and industrial processes [21]. The fifth generation (5G) is a major contributor to this positive evolution and has led the development alongside other state-of-the-art technologies. This work incorporates Long-Term Evolution (LTE), also called 4G, and Narrowband IoT (NB-IoT). The 3rd Generation Partnership Project (3GPP) has also made significant efforts in deploying these technologies and ensuring they coexist.

3GP release 13, for example, introduced two categories of IoT technologies called LTE-M and NB-IoT [22]. The 3GPP created the low-power wide-area network radio communication standard known as LTE-M for machine-to-machine and Internet of Things applications. LTE-M is an extension of the 3GPP 4G LTE standard that takes advantage of many of the benefits of LTE, including mobility, priority handling, security, and an internationally recognized base station [23]. For easy categorization and modeling, this research uses the traditional LTE as it has been established that LTE-M uses an identical frame structure and numerology as LTE [7].

2.2.1 LTE Coexistence With NB-IoT

LTE and NB-IoT can coexist within the same spectrum and network infrastructure. TE and NB-IoT, both standardized by 3GPP, share similar frame structures, numerology, and modulation, enabling seamless coexistence within mobile networks, especially during the transition to 5G. Their compatibility in shared spectrum and design features supports interoperability and integration [24]. Below is an overview of these technical compatibilities:

- **Shared Spectrum:** NB-IoT can be deployed within existing LTE carriers, utilizing a small portion of the LTE spectrum. It can be carried out as an In-band or Guard-band deployment. For in-band, NB-IoT operates within the LTE carrier, sharing resources without compromising the performance of LTE. In the case of the guard band, NB-IoT is deployed in the guard band of LTE carriers, which allows it to function alongside LTE without interference [25], [26]. For our use case, the in-band deployment mode is most suitable. It enables both technologies to share the same LTE carrier frequency, optimizing spectrum utilization without requiring extra spectrum. This mode minimizes infrastructure changes, reduces costs, and aligns well with our DRL approach to efficiently allocate resources while mitigating interference. It supports seamless coexistence in heterogeneous wireless environments, making it ideal for our research goals.
- **Numerology and Modulation:** NB-IoT employs a 15kHz subcarrier spacing similar to LTE, which enhances the coexistence of the two technologies. This alignment in numerology allows efficient resource allocation and minimizes interference between the technologies [27].
- **Design Features:** NB-IoT's design is optimized for low power consumption and extended coverage, making it suitable for IoT applications [25]. It is also capable of supporting a massive number of devices, which is crucial in IoT ecosystems.
- **Frame Structure:** The frame structures of NB-IoT and LTE are designed to meet different communication needs, with NB-IoT optimized for low-power, wide-area (LWA) applications and LTE for high-speed, high-capacity data transmission. Not all the frames are usable for data transmission. For example, frames 0, 5, and 9 are used for other purposes in NB-IoT. Subframe number 0 carries the Narrowband Physical Broadcast Channel (NPBCH), 1 to 4 and 6 to 8 the Narrowband Physical

Downlink Control Channel (NPDCCH)/Narrowband Physical Downlink Shared Channel (NPDSCH), and 5 and 9 the Narrowband Primary Synchronization Signal (NPSS)/Narrowband Secondary Synchronization Signal (NSSS), as shown in Figure 5. When carrying control channels in a subframe (A) and data (B) [28].

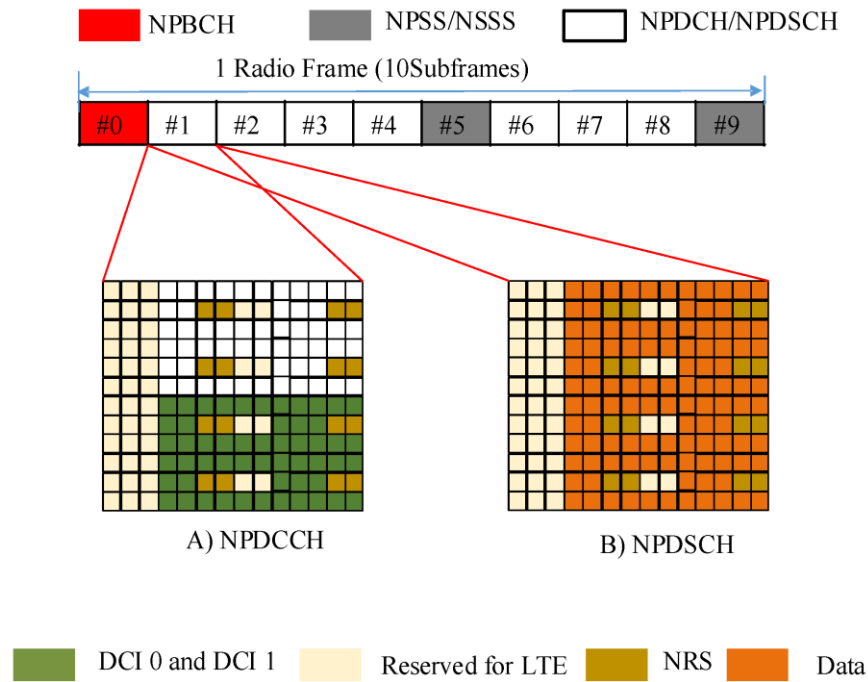


Figure 5: NB-IoT Downlink Frame Structure

Both NB-IoT and LTE share several similarities in their frame structures, which are essential for their operation and coexistence in wireless communication networks. These similarities include Subframe Length, Time Domain Structure, and Resource Elements. A brief description of the Frame structure features is given below.

- **Subframe Length:** Both technologies utilize similar subframe lengths. In NB-IoT for example, each frame consists of 10 subframes, each with a duration of 1 ms, mirroring the frame structure used in LTE systems. The

NB-IoT and LTE frame structures are represented in Figures 6 and 7⁴ respectively.

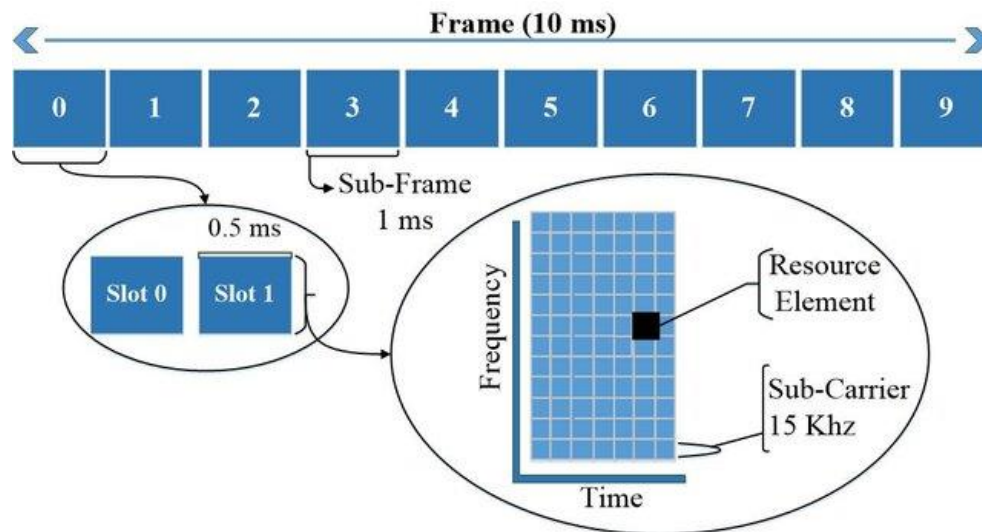


Figure 6: NB-IoT Frame Structure [36].

⁴ [Explain LTE Frame Structure both for FDD and TDD - Techplayon](#)

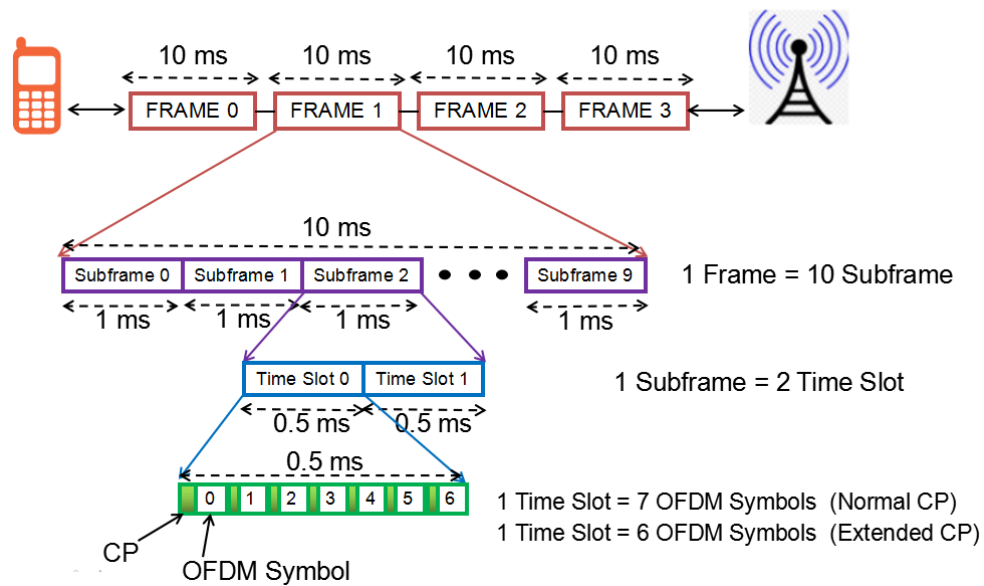


Figure 7: LTE Structure

- **Time Domain Structure:** The time domain structure is largely retained in NB-IoT from LTE, allowing for efficient scheduling and resource allocation. This implies that NB-IoT can operate within the existing LTE framework, making it easier to deploy alongside LTE networks⁵.
- **Resource Elements (RE):** Both use RE as the smallest unit for physical channels. In NB-IoT, a Physical Resource Block (RB) spans 12 subcarriers over 7 OFDMA symbols, similar to LTE's approach.

Their coexistence enables operators to offer a comprehensive range of services, including high-speed broadband, IoT connectivity, and machine-to-machine communications, while efficiently utilizing spectrum resources and infrastructure. This can give birth to multiple use cases some of which include:

➤ ***Industrial IoT (IIoT) Monitoring and Control***

- **OVERVIEW:** Designing a system for monitoring and controlling industrial equipment and processes remotely.

⁵ [4G | ShareTechnote](#)

- **NB-IoT for Low-Power Sensors:** NB-IoT is used to connect low-power sensors and measure parameters like temperature, sensor, and vibration.
- **LTE/LTE-M for High-Speed Data:** Use LTE or LTE-M for high-speed data transmission in industry environments for real-time control signals and video surveillance.
- **Coexistence Management:** Evaluate coexistence strategies to prevent interference between NB-IoT sensors and LTE/LTE-M devices.

➤ ***Smart City Infrastructure Monitoring***

- **OVERVIEW:** Develop a system for monitoring various aspects of smart city infrastructure, such as traffic lights, waste management systems, and environmental sensors.
- **NB-IoT DEVICES:** Use NB-IoT devices for low-power, long-range communication to collect data from sensors deployed across the city.
- **LTE INTEGRATION:** Explore how LTE and LTE-M can complement NB-IoT by providing higher data rates for critical applications or real-time monitoring.
- **COEXISTENCE CHALLENGES:** Investigate interference mitigation techniques and resource allocation strategies to ensure seamless coexistence between NB-IoT and LTE devices in the shared spectrum bands.

Dynamic Spectrum Sharing (DSS) is essential for ensuring a smooth transition to 5G while supporting the existing Internet of Things (IoT) ecosystem. Transitioning between cellular generations can be complex and costly, often requiring significant infrastructure upgrades and device modifications. DSS alleviates these challenges by enabling the coexistence of legacy technologies like LTE and NB-IoT with the newer

5G NR. This allows mobile network operators (MNOs) to avoid the cumbersome process of spectrum refarming, as illustrated in Figure 8⁶.

Figure 8 demonstrates how DSS dynamically allocates spectrum resources across different technologies, optimizing utilization and ensuring seamless connectivity. With the minimal network overhead required for LTE-M and NB-IoT, a significant portion of the spectrum can be dedicated to 5G NR services as needed. This adaptability not only extends the lifecycle of existing IoT devices but also facilitates the efficient integration of new technologies, promoting a more sustainable and future-proof network.

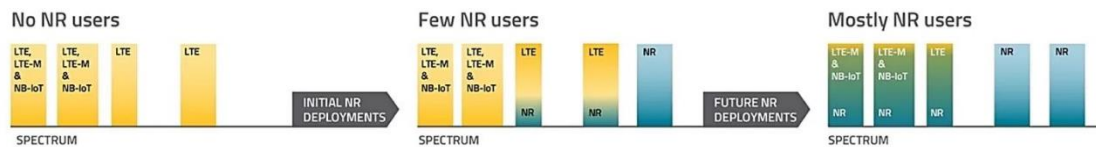


Figure 8: Illustration of DSS Deployment

While LTE and NB-IoT inherently support Dynamic Spectrum Sharing (DSS), researchers continue to make significant efforts to optimize DSS performance, particularly by using Deep Reinforcement learning, enabling advanced features, and addressing challenges for these technologies [29]. The key features to consider include:

- **Optimization:** DSS algorithms in LTE and NB-IoT may rely on predefined rules or heuristics to determine how spectrum resources are allocated between different technologies or services. However, DRL offers the potential to optimize spectrum allocation dynamically based on real-time network conditions and user demands, leading to more efficient spectrum utilization and improved network performance.

⁶ [Dynamic Spectrum Sharing \(DSS\) in 3GPP Release 17: Bridging 4G LTE and 5G for IoT Continuity - 5G HUB TECHNOLOGIES, INC](#)

- **Adaptability:** DRL algorithms can adapt to changing environments and network dynamics by learning from experience and adjusting their strategies accordingly. This adaptability is particularly beneficial in dynamic and heterogeneous networks where the availability and demand for spectrum resources may vary over time.
- **Complex Scenarios:** DRL can more effectively handle complex scenarios and interactions between multiple entities than traditional optimization techniques. In the context of DSS for LTE and NB-IoT coexistence, DRL can consider various factors such as interference levels, traffic patterns, user priorities, and quality of service requirements to make informed decisions about spectrum allocation.
- **Interference Mitigation:** DRL can more efficiently mitigate interference between LTE and NB-IoT deployments by learning adaptive strategies for power control, beamforming, and frequency allocation. By optimizing these parameters dynamically, DRL algorithms can minimize interference and improve the overall performance of both technologies.
- **Enhanced Performance:** Research in DSS with DRL can potentially enhance the performance of LTE and NB-IoT networks by enabling more intelligent and adaptive spectrum management strategies. This can improve key performance metrics such as throughput, latency, coverage, and spectral efficiency.

Overall, while LTE and NB-IoT already support DSS capabilities, research in DSS for LTE and NB-IoT with the use of DRL can further enhance spectrum management and coexistence between the two technologies by enabling more adaptive, efficient, and intelligent allocation of spectrum resources. This research especially focuses on exploring the key parameters of DRL and how they affect the spectrum sharing between LTE and NB-IoT.

CHAPTER THREE

METHODOLOGY

This section details the approach and framework used in this research work. As depicted in Figure 9, there are three major phases incorporating the review and experimental phases. The review phase encompasses a comprehensive literature review and a review of related technologies. In contrast, the experimental phases include the three stages of experiments alongside the exploration and optimization of parameters.

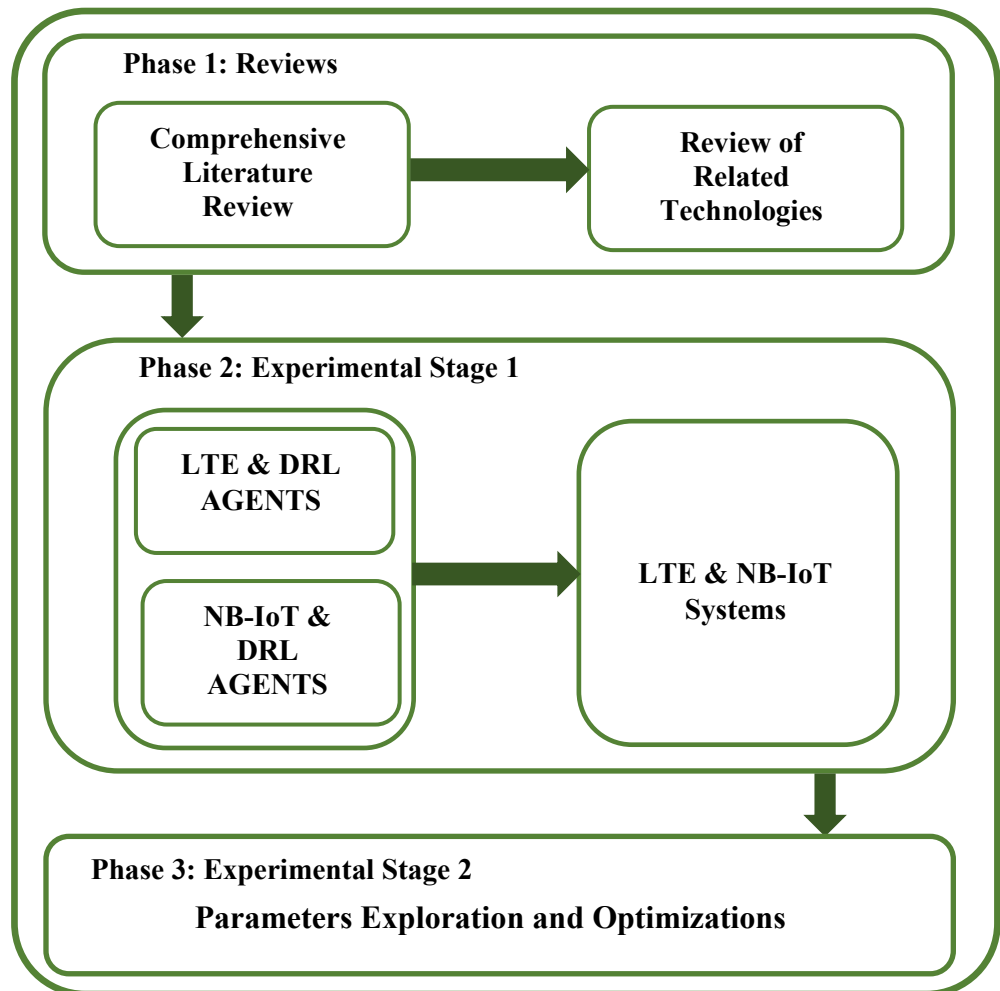


Figure 9: Research Workflow

The details of the first phase are extensively discussed in chapter two, which includes a review of related literature and technologies. The second phase comprises 3 stages of experiments.

- The incorporation of the LTE system and DRL agent in the same environment,
- The incorporation of the NB-IoT system and DRL agent in the same environment and
- The incorporation of the LTE and NB-IoT systems in the same environment.

The experimental stage 2 (phase 3) includes the areas we focused on for exploration to give insights into how different parameters affect the overall throughput of the environment. The pictorial representation of the workflow is given in Figure 9 above.

3.1 SYSTEM MODELING

Our system considers heterogeneous networks in which N user equipment (UE) transmit packets to a Base Station (AP) via a shared wireless channel. We have type one user equipment running on LTE and type II running on NB-IoT. The Base Station functions as a single agent, centrally controlling the spectrum resources without requiring communication or cooperation from the user equipment as represented in Figure 10 below.

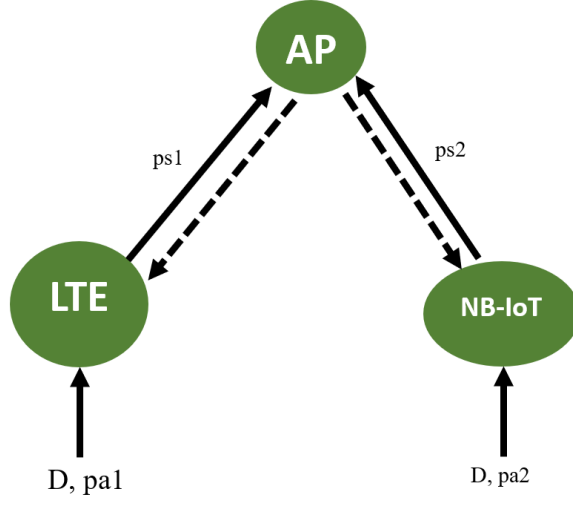


Figure 10: LTE and DRL spectrum sharing illustration

Consider a wireless environment with devices operating on LTE and NB-IoT serving a set of X and Y users, respectively, in a slotted time slot $T = \{0, 1, \dots, 9\}$. Each Radio Access Technology (RAT), R_a serves a set of $K_{Ra} \subseteq \{X, Y\}$ User Equipment UE with $K = \{1, 2, \dots, N\}$. Total Bandwidth is divided into a set B of resource blocks (RB). Each Radio Access Technology R_a is allocated a set $B_{Ra} \subseteq B$ of RBs. Each User Equipment $UE \in K_{Ra}$ is allocated $B_{UE,Ra} \subseteq B_{Ra}$ of $B_{UE,Ra}$ by it serving RAT R_a . We have type I and type II using LTE and NB-IoT, respectively. As depicted in Figure 11, all UEs are randomly distributed around the BS.

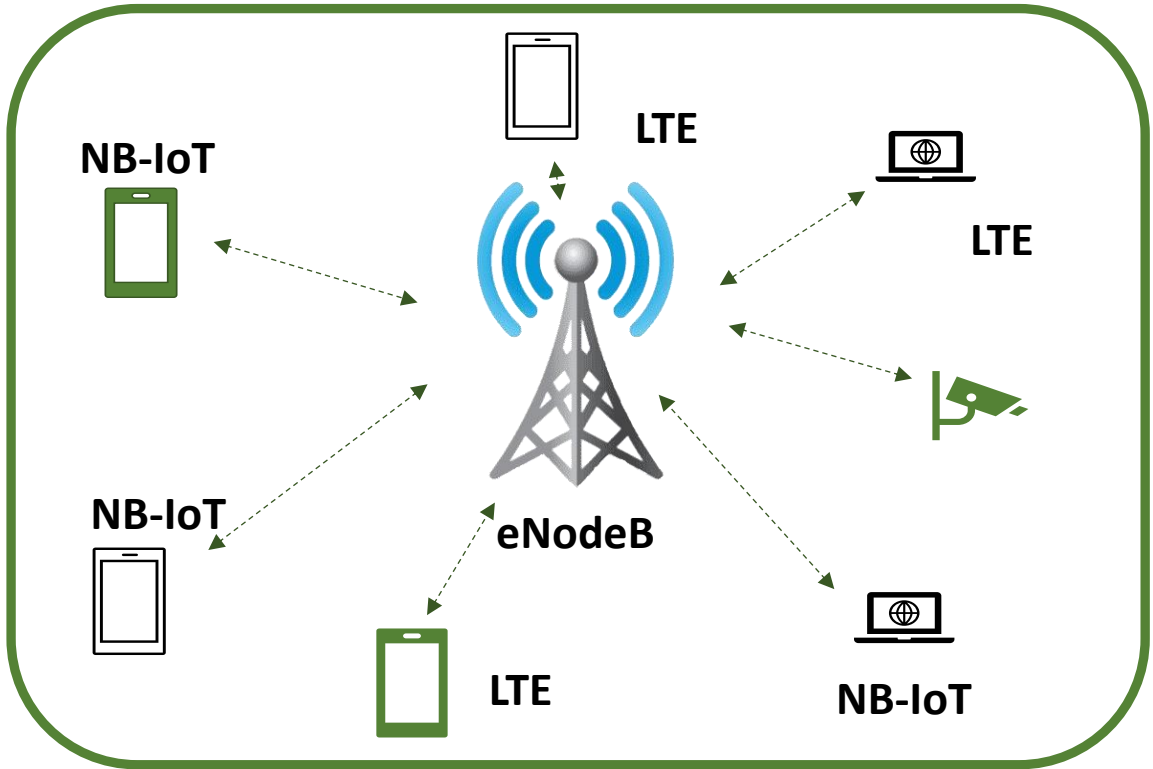


Figure 11: System Modeling

LTE and NB-IoT are assumed to operate in the 3.5GHz band, and Time Division Duplexing (TDD) is applied as the duplexing method. Also, 15kHz sub-carrier spacing and 10 ms Frame length are considered to align LTE and NB-IoT subframes in time and frequency division. Each UE is allowed to access a single subchannel in a time slot. Only the same type of UEs can transmit together in the same time slot.

3.1.1 Overview of the Selected Base Station for LTE and NB-IoT

The Access Point is assumed to be one of the commonly used Evolved node B (eNodeB) such as the Nokia AirScale Base Station which supports LTE and NB-IoT technologies. The Nokia AirScale Base Station is a scalable, modular, and energy-efficient solution designed to support multiple wireless technologies, including LTE,

NB-IoT, and 5G. It enables smooth evolution from LTE to 5G while supporting dynamic spectrum sharing (DSS) between technologies.

Key Features of the Nokia AirScale Base Station include:

- **Multi-Radio Capability:** Supports multiple standards, such as LTE, NB-IoT, and 5G NR, from a single hardware unit. Offers DSS to allow LTE and NB-IoT devices to coexist efficiently.
- **Scalability:** Modular design allows deployment in small cells, macro cells, or massive MIMO configurations. Flexible to serve a few users in rural areas or dense urban centers with high user density.
- **Performance:** Supports high throughput for LTE users while maintaining low-power, narrowband connectivity for NB-IoT devices. Up to 20 Gbps capacity per cell (for LTE and 5G combined) and 128 users per antenna port.
- **Efficient Spectrum Utilization:** Implements advanced scheduling algorithms to allocate spectrum dynamically between LTE and NB-IoT, ensuring optimal performance. Narrowband resources (180 kHz) for NB-IoT coexist with wider bands for LTE and 5G.
- **Energy Efficiency:** Designed with energy-efficient hardware to reduce operational costs. Includes Nokia's proprietary "Liquid Cooling" technology to reduce power consumption by up to 30%.
- **Backhaul and Integration:** Supports seamless integration with existing LTE EPC and future 5G core networks.

3.1.2 Experimental Setup

The environment consists of two primary categories of equipment: Type I (using LTE) and Type II (using NB-IoT). While the implementation focuses on a two-user equipment (UE) scenario—one Type I device and one Type II device—it can be extended to include multiple devices within each user type. For instance, within the LTE category, devices can be further classified as LTE1, LTE2, LTE3, and so on. Similarly, the NB-IoT category can encompass multiple devices, such as NB-IoT1, NB-IoT2, among others. Whenever a spectrum resource is allocated to LTE, it can be dynamically assigned to any LTE-based device depending on predefined criteria, such as priority levels or the availability of data packets to transmit. The same principle applies to NB-IoT devices, where spectrum allocation depends on similar factors.

This approach ensures that both LTE and NB-IoT devices can efficiently share the available spectrum while prioritizing network needs, improving the overall utilization of finite spectral resources.

For both types of UEs, we assume a delay-constrained Bernoulli traffic pattern as given in [20]: Type I has packet arrival with $pa1 \in (0,1)$, and Type II has $pa2 \in (0,1)$ in any slot. All packets have a hard *packet expiry duration* of D slots, which implies that a value of D equals 1 is equivalent to a 1ms *packet expiry duration* per slot. If $D = 1$, it indicates that a packet will expire at the end of slot t if it arrives at the beginning and is not transmitted or if the transmission fails.

A packet not successfully delivered to the *BS* upon reaching D slots will be discarded. If both Types transmit to the *BS* at the same time slot, then a collision will occur, and both packets cannot be delivered. The wireless channel is unreliable and can fail even

if a single user is transmitting due to fading and shadowing. We can model this unreliable scenario by the *success probability* as presented in [20].

We brought deep learning to the picture using the Deep Q-Network (DQN) algorithm. We slightly modified the original DQN algorithm given by [14]. Our agent is based on DQN but with some modifications tailored around our environment and goals. Our environment is dynamic, and we model the wireless environment as closely as possible. The environments adopted for the LTE and NB-IoT are similar, with minor changes in the parameter names. However, there is a significant difference in the LTE-NB-IoT environment compared to when each operates alone with the DRL Agent. The DRL agent is modeled to represent any random wireless technology. The list of notations used is given in Table 2 below.

Table 2: List of notations and their representation

Symbol	Meaning
$\mathcal{P}$	State Transition Probability
$\mathcal{A}$	Action Space
θ	The parameter vector; made up of the weights of the edges in the neural
$\mathcal{M}$	State History Length, denoting how many states in the past the agent can
s_t	State of the agent at a particular time t
a_t	Action taken by the agent at a specific moment in time.
π	Policy: action-reward mapping from states s_t to action a_t
$\triangleq$	Equal by definition
$\mathcal{R}_t$	Reward accrues by the agent at a specific moment in time.
γ	Discount factor
co_t	Channel Observation at a particular time t
cs_t	Channel State at a particular time t
R_{t+1}	Reward at time $t+1$
(s_t, a_t)	State Action pair
UE	User Equipment
X	Type I user equipment, representing LTE devices
Y	Type II user equipment, representing NB-IoT devices
T	Time duration for each slot, in milliseconds

K	<i>Number of user equipment from 1-N</i>
R_a	<i>Radio Access Technology e.g LTE and NB-IoT</i>
K_{Ra}	<i>User Equipment for a particular Radio Access Technology, R_a</i>
B	<i>Total Bandwidth</i>
B_{Ra}	<i>Bandwidth for a particular R_a</i>
$B_{U,R}$	<i>The bandwidth of a particular user equipment U operating on a particular</i>
pa	<i>Probability of Arrival</i>
ps	<i>Probability of Success</i>
pt	<i>Retransmission Probability</i>

3.1.3 LTE-Only Environment Scenario Modelling

As depicted in Algorithm 1, the LTE environment modeling simulates the Dynamic Spectrum Sharing between the LTE and devices running a DRL agent. The environment models how these two entities interact based on their actions (whether to transmit or wait) and the quality of their respective communication channels. During each simulation step, the environment assesses the actions taken by the LTE and the DRL agents. If both decide to wait, the environment remains idle, returning an "idle" observation. If one of the entities chooses to transmit, the success of that transmission is determined probabilistically based on the channel quality, with a reward given for successful transmission and different observations indicating success, failure, or collisions. Collisions occur if both attempt to transmit simultaneously, resulting in failed communication for both parties. This environment provides a framework to study the effects of dynamic spectrum sharing between LTE and other communication technologies, particularly in scenarios where channel interference and collision risks are critical factors. The pseudocode of the environment is given in algorithm 1.

Algorithm 1: Pseudocode of LTE Environment

INPUT: lte_channel, agent_channel, lte, agent

OUTPUT: lte_reward, agent_reward, observation

INITIALIZE:

 lte_reward = 0

 agent_reward = 0

 observation = 'I'

STEP:

if lte.action == 0: # LTE waits

if agent.action == 0: # agent waits

 observation = 'I'

else if agent_channel > random_uniform(): # agent transmits

 agent_reward = 1

 observation = 'S'

else: # agent channel error

 observation = 'F'

else: # LTE transmits

if agent.action == 0: # agent waits

if lte_channel > random_uniform(): # LTE transmits

 lte_reward = 1

 observation = 'B'

else:

 observation = 'F' # LTE channel error

else:

 observation = 'F' # collision

RETURN:

 lte_reward, agent_reward, observation

3.1.4 NB-IoT-Only Environment Scenario Modelling

The NB-IoT environment is modeled in a similar way to LTE. The NB-IoT system behaves differently than the LTE system, given that the NB-IoT takes special characteristics (such as power-saving modes and smaller packet sizes, among others) into consideration. The environment is modeled to incorporate an NB-IoT System and a DRL agent coexisting and sharing the spectrum. The environment models the decision-making process of both entities, where each decides whether to transmit or wait based on their respective channel conditions. During each step, the environment evaluates the actions of the NB-IoT system and the agent. If both entities decide to wait, the system remains idle, returning an "I" observation. If one of them chooses to

transmit, the success of the transmission is determined probabilistically, based on the channel quality, with rewards granted for successful transmissions. A successful NB-IoT transmission results in an "S" observation, while a successful agent transmission results in a "B" observation if the NB-IoT is observing the environment. However, if both entities attempt to transmit simultaneously, a collision occurs, leading to a failed transmission for both, indicated by an "F" observation. This environment framework is particularly useful for analyzing the dynamics of spectrum sharing, especially in scenarios where interference and collisions are critical factors in communication reliability. The pseudocode of the NB-IoT environment is given in Algorithm 2.

Algorithm 2: Pseudocode of NB-IoT Environment

INPUT: nbiot_channel, agent_channel, nbiot, agent

OUTPUT: nbiot_reward, agent_reward, observation

INITIALIZE:

 nbiot_reward = 0

 agent_reward = 0

 observation = 'I'

STEP:

if nbiot.action == 0: # NB-IoT waits

if agent.action == 0: # agent waits

 observation = 'I'

else if agent_channel > random_uniform(): #agent transmits

 agent_reward = 1

 observation = 'S'

else: # agent channel error

 observation = 'F'

else: # NB-IoT transmits

if agent.action == 0: # agent waits

if nbiot_channel > random_uniform(): #NB-IoT transmits

 nbiot_reward = 1

 observation = 'B'

else:

 observation = 'F' # NB-IoT channel error

else:

 observation = 'F' # collision

RETURN:

 nbiot_reward, agent_reward, observation

3.1.5 Heterogeneous Environment Scenario Modelling

The heterogeneous environment models the interaction of the LTE, NB-IoT, and DRL agents within a shared spectrum. Each agent operates based on its channel quality and probabilistic transmission strategy, with their actions being either to transmit or wait. The environment then assesses the outcomes of these actions during each step, determining whether transmissions are successful or result in collisions. The *step* function models this process, checking the state of each agent's action. If all agents choose to wait, the environment returns an idle observation. If only one agent transmits, the environment checks the corresponding channel's quality to determine if the transmission is successful or fails, awarding rewards accordingly. However, if multiple agents attempt to transmit simultaneously, a collision occurs, resulting in failed transmissions for all involved. This setup allows for the evaluation of dynamic spectrum-sharing strategies, highlighting the trade-offs between successful communication and potential collisions in a multi-agent wireless network. The pseudocode of the NB-IoT environment is given in Algorithm 3.

Algorithm 3: Pseudocode of Combined Environment

INPUT: lte_channel, nbio_channel, agent_channel, lte, nbio, agent, channel_quality, threshod_quality

OUTPUT: lte_reward, nbio_reward, agent_reward, observation

INITIALIZE:

lte_reward = 0
nbio_reward = 0
agent_reward = 0
observation = 'I'

STEP:

```
if lte.action==0 & nbio.action==0 & agent.action==0: #ALL waits
    observation = 'I'
else if lte.action==1 & nbio.action==0 & agent.action==0: # LTE transmits, others wait
    if lte_channel > random_uniform():
        lte_reward = 1
        observation = 'S'
    else:
        observation = 'F'
else if lte.action==0 & nbio.action==1 & agent.action==0: #NB-IoT transmits, others wait
    if nbio_channel > random_uniform():
        nbio_reward = 1
        observation = 'S'
    else:
        observation = 'F'
else if lte.action==0 & nbio.action==0 & agent.action==1: # Agent transmits, others wait
    if agent_channel > random_uniform():
        agent_reward = 1
        observation = 'B'
    else:
        observation = 'F'
else: # Any collision or simultaneous transmissions
    observation = 'F'
```

RETURN:

lte_reward, nbio_reward, agent_reward, observation

3.1.6 LTE and NB-IoT Systems

The LTE and NB-IoT Systems are modeled to reflect the true nature of the technologies in real applications. The LTE_UE is designed to model an LTE-communication agent, managing resources and decisions based on various network parameters. It is initialized with parameters such as arrival rate, transmission probability, number of subcarriers, number of users, channel quality, and a threshold for acceptable channel quality. The agent maintains a queue to track incoming and outgoing packets and allocates resources

to users based on channel conditions. The transmission probability and the queue status determine the action decision, whether to transmit or wait. The agent continually updates its queue and channel quality to reflect the dynamic network environment, ensuring efficient spectrum usage and communication reliability.

The NB-IoT_UE, on the other hand, is tailored for NB-IoT communication, focusing on energy efficiency and small packet transmissions. Similar to the LTE_UE, it initializes with key network parameters but includes additional features specific to NB-IoT, such as power-saving mode and a smaller packet size. The power-saving mode significantly influences the action selection process, making the agent inactive during certain cycles to conserve energy. The queue management reflects the transmission of smaller packets, enhancing efficiency for IoT applications. This agent's design highlights NB-IoT's emphasis on low power consumption and optimal resource utilization, making it suitable for devices requiring extended battery life and consistent connectivity [25].

3.2 REINFORCEMENT LEARNING MODEL

The above-discussed scenarios of the three environments can be formulated as an MDP problem, distinguished by its reward function, state, action, and transition probability [30]. The state of the system can be defined as given in equation 2 below:

$$s_t = (l_{t,1}, l_{t,2}, o_t) \dots \text{equation 2}$$

Where,

$l_{t,i}$ is the lead time vector for the user i at time t , representing whether user i has packets in the system that are due to expire within a specific range

A lead time vector is a representation used to track and manage time-sensitive items in systems with packet expiry duration. It specifically indicates how much time remains before each item (like a packet or resource) expires or becomes unusable. This vector is particularly useful in systems where actions or resources have delayed effects, such as inventory management, communication networks, or scheduling systems with resource constraints.

$$l_{t,i,k} = \begin{cases} 1 & \text{if user } i \text{ has a packet expiring at slot } t \\ 0 & \text{otherwise} \end{cases}$$

Where,

- k represents the expiration period, meaning the packet has a time-to-live (TTL) of k slots remaining before expiration.
- D is the maximum expiration threshold for packets in the system, setting the upper limit for TTL values.

Observation o_t at state-time s_t can be defined as $o_t \in \{BUSY, SUCCESS, IDLE, FAIL\}$.

Where,

$o_t = BUSY$ implies the other type transmits successfully in time slot $t-1$

$o_t = SUCCESS$ means the type observing the channel transmits successfully in slot $t-1$

$o_t = IDLE$ means no one is transmitting and

$o_t = FAIL$ implies collision or channel failure.

It is worth noting here that we use delay gratitude for rewarding the agent, which implies the reward R at time t is based on the action performed at time $t-1$. The Set of all possible states is given as S . Action space is defined as; $a_t = \{TRANSMIT, WAIT\}$.

The action observation pair (a_t, o_t) can be at least any of the six possibilities:

$\{TRANSMIT, BUSY\}$, $\{TRANSMIT, SUCCESS\}$, $\{TRANSMIT, FAIL\}$, $\{WAIT, IDLE\}$, $\{WAIT, BUSY\}$, and $\{WAIT, FAIL\}$.

3.2.1 Reward Function

The reward R is the feedback gotten in a time-slot t for an action a_t performed in a state s_t is defined as $R(s_t, a_t)$:

$$R(s_t, a_t) \triangleq 1 \{o_t \in \{BUSY, SUCCESS\}\}, \forall s_t \in S, \quad a_t \in A \dots \text{equation (3)}$$

We can say $R(s_t, a_t) = 1$ if the system receives an *ACK* either for its packet ($o_t = SUCCESS$) or for another type ($o_t = BUSY$). Here, we are using delayed gratification, which means the action performed in $t-1$ will be rewarded in t . The three experiment scenarios have different reward functions. The reward functions for each of the three environments are given in Algorithms 1, 2, and 3.

3.2.2 Q-Learning

Equation 3 is the adopted version of the Q-learning for our implementation. It is worth noting that this equation is an updated version of the Bellman equation given in Equation 1 of Section 1. It iteratively updates Q-function $Q(s, a)$ as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha R(s_t, a_t) + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t) \dots \text{equation (3)}$$

Where,

$\alpha \in (0,1)$ is the learning rate

$\gamma \in (0,1)$ is the discount factor and

$Q(s_t, a_t)$ is the state-action value at time t .

3.2.3 *RNN Construction*

We incorporated RNN in our modified DQN. Recurrent Neural Networks (RNNs) are a type of neural network architecture intended to process sequential data. Unlike feedforward neural networks, RNNs have internal memory that allows them to process inputs of variable lengths and maintain a state that depends on the previous computations. Because of this, they are excellent at jobs like time series prediction, machine translation, and language modeling [31], [32]. Reusing the output from the previous time step as extra input for the next time step is the fundamental principle underlying RNNs [31]. This allows the network to build an internal memory that stores information about the sequence it has already processed. However, because of the vanishing or expanding gradient problem, early RNNs had trouble learning long-term dependencies. More sophisticated RNN architectures, including the Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM), were created to solve this problem.

Cho et al. [33] created Gated Recurrent Units (GRUs); a sort of recurrent neural network (RNN) architecture intended to capture relationships in sequential input effectively. GRUs utilize two primary gating mechanisms: the update gate, which determines how much of the previous hidden state to retain, and the reset gate, which controls how much past information to forget. This design simplifies the architecture compared to Long Short-Term Memory (LSTM) networks by combining the functions of the input and forget gates into a single update gate, resulting in fewer parameters and faster training times. GRUs are known to be more computationally efficient than their variants like LSTM, given their less complex structure, which in turn makes it more

suitable for applications requiring more intelligent management of computational resources [34].

We construct the Recurrent Neural Network (RNN) model using the Keras framework, specifically leveraging a Gated Recurrent Unit (GRU) layer followed by fully connected (Dense) layers. This architecture is particularly suitable for tasks involving sequential data, where the model can capture temporal dependencies. The model begins by defining an input layer that accepts data sequences with a specified shape and determines the length of the sequences and the number of features per time step. The first hidden layer is a GRU layer with 64 units and a Rectified Linear Unit (ReLU) activation function, which allows the model to learn complex patterns in the input sequences.

The kernel weights for this layer are initialized using the He normal distribution, which is beneficial for layers with ReLU activation as it helps mitigate issues related to vanishing gradients. Following the GRU layer, two Dense layers with 128 units each are added, also utilizing ReLU activation and He normal initialization. These layers further process the features extracted by the GRU, enabling the model to learn higher-level representations of the data. The final output layer is another Dense layer, configured to produce many outputs equal to the product of the number of actions and nodes. This design closely corresponds to the model's goal of predicting multiple actions across various nodes in a multi-dimensional output space. The model is compiled with a mean squared error (MSE) loss function suitable for regression tasks and the RMSProp optimizer. It effectively trains deep learning models due to its adaptive learning rate capabilities. This architecture is particularly effective for applications such as our case of Dynamic Spectrum Sharing. We would like to note

here that these are the configurations that give us the best output. The details of other explored configurations are discussed in the results and discussion section. The pictorial representation of the RNN Architecture is given in Figure 12.

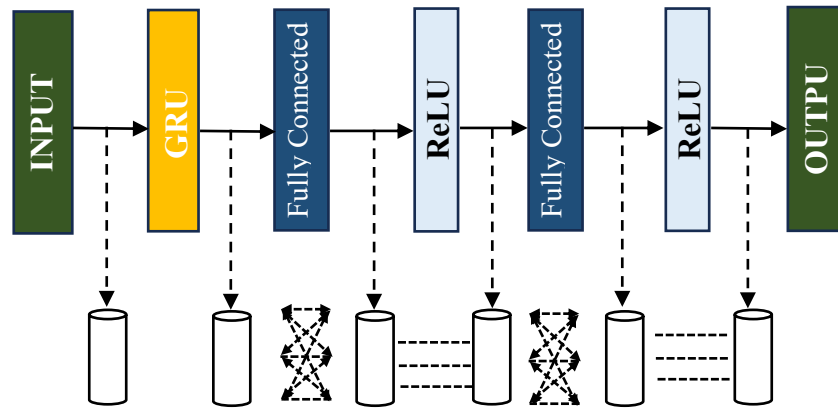


Figure 12: RNN Architecture

CHAPTER FOUR

RESULTS AND DISCUSSION

To evaluate the performance of the proposed DLR model, multiple experiments were conducted for the three environment scenarios discussed earlier. We have explored different parameters to optimize the throughput for each of the scenarios. These parameters are essential for Q-learning, Deep Neural Network, and Channel Probabilities for exploration. The Q-learning parameters include Gamma and learning rates. The deep neural network optimization parameters include the overall RNN architecture, which encompasses the input, middle, and output layers. In contrast, channel probabilities encompass packet arrival probabilities (pa), packet successful delivery probabilities (ps), packet transmission & retransmission probabilities (pt), and Packet Expiry Duration (D).

- Throughput refers to the rate at which data is successfully transmitted over the network. It is a critical measure of the efficiency of spectrum utilization and the network's capacity to handle traffic. Throughput in computer networks is usually calculated using the formula given in Equation 4 below:

$$Throughput = \frac{Total\ Data\ Transferred}{Time\ Taken} \dots equation\ 4$$

Where;

Total Data Transferred is the amount of data (in bits or bytes) successfully delivered over a specified period.

Time Taken is the time interval during which the data was transmitted. It is worth noting that throughput can be calculated for both LTE and NB-IoT systems individually or for the entire system.

Higher throughput indicates better spectrum utilization and network performance.

- Convergence time refers to the amount of time it takes for the system to stabilize and reach an optimal or near-optimal solution for spectrum allocation. This is especially important in dynamic systems where the network conditions are constantly changing. The convergence rate can be calculated as the time taken to reach a stable throughput or resource allocation. The time is measured from the start of the simulation or the system's operation until throughput or resource allocation stabilizes within a pre-defined threshold.

The first stage of the experimental setup includes the LTE and Deep Reinforcement Learning (DRL) agents in the same environment communicating through the Base Station (Access Point) with LTE depicted as Type I and NB-IoT as Type II. If Type I transmits to the BS, the packet can be successfully delivered with probability $ps1 \in (0,1)$ and $ps2 \in (0,1)$ for Type II. Otherwise, a channel retransmission probability pt for Type I and Type II UEs will be necessary.

Both Type I and Type 2 users need not communicate with each other before transmitting. It will interact with the environment through the BS to learn information about another User type. The BS will send ACK to both User types if it successfully decodes a packet, a NACK if it receives a packet but does not decode it successfully, and sends nothing if it does not receive anything in the time slot. By observing the BS, each user can infer the behavior of the other user.

4.1 LTE AND DRL AGENT COEXISTENCE

This experiment evaluates dynamic spectrum sharing between two agents—LTE and reinforcement learning (RL) agents—in a heterogeneous wireless environment. The experiments were conducted using the baseline RL algorithm and the configuration shown in Table 3. After a series of experiments to check the optimal parameters while using the baseline RL agent, the LTE system was configured with arrival and success probabilities of 90% each, and the RL agent transmission probability of 20%.

The values given in Table 3 are also the parameters that produce the optimal performance. It is worth noting that this combination is arrived at after trying several values. Extensive discussion about each of the parameters and their impacts are given later in this section from 4.1.1 – 4.1.6. As shown in Figure 13(a), the RL model achieved approximately 45% throughput with a convergence time of 519 microseconds.

Table 3: LTE-DRL parameters

Parameters	Values
State history (M)	20
Learning rate	0.01
Fairness Index (F)	1
Discount factor (Γ)	0.9
LTE packet expiry duration ($D1$)	3
DRL packet expiry duration ($D2$)	3
LTE Arrival Prob. ($Pa1$)	0.9
DRL Arrival Prob. ($Pa2$)	0.2
LTE Success Prob. ($Ps1$)	0.9
DRL Success Prob. ($Ps2$)	0.2
Trans/retransmission Prob. (Pt)	1

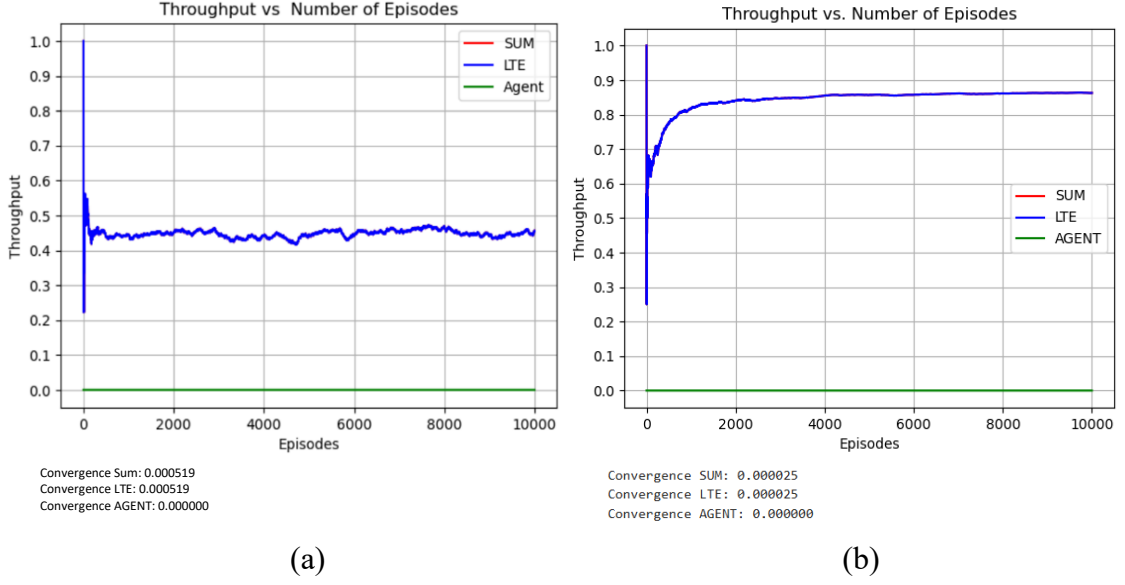


Figure 13: LTE and DRL Coexistence Throughput.
 $pa1, ps1 = 0.9, pa2, ps2 = 0.2, pt = 1$

By bringing deep learning into the picture, we have more state history, up to 50 states with negligible convergence rates. The highest convergence rate recorded with all the experiments is around 5 milliseconds. A typical RL algorithm will have approximately 10 million possible values, which implies that there will be more than 5 hours for each state-action pair if a 1-millisecond time slot and 10 states are used [6]. Need to say that the value of states indicates how many episodes or states the agent will remember in the past. This usually improves the robustness and decision-making accuracy of the agent but mainly with a convergence rate trade-off.

As shown in Figure 13(a), the RL model achieved approximately 45% throughput with a convergence time of 519 microseconds. On the other hand, the DRL implementation significantly outperformed RL, achieving over 85% throughput and a much faster convergence time of 25 microseconds as represented in Figure 13(b). This result shows that integrating deep learning into the learning process enables the use of extended state history, up to 50 states, with negligible additional convergence time. A standard RL algorithm, in comparison, would involve approximately 10 million possible values,

contributing to its low convergence rate. However, it is important to note that the DRL implementation runs slightly slower overall compared to RL due to several factors. First, the complexity of the deep neural networks used in DRL introduces considerable computational overhead, as these networks require intensive training. Additionally, DRL involves more extensive exploration of the environment to learn effective policies, demanding greater computational resources and interaction time. In scenarios where the neural network size is small and achieving higher throughput or faster convergence is not critical, RL can still be a suitable and efficient choice.

4.1.1 Probabilities

As discussed earlier, we explored the impact of various parameters on the throughput. The arrival, success, and retransmission probabilities (given in %) indicate the preferences being given to one system over the other. The probabilities are also expressed in a ratio from $[0, 1]$ with 0 indicating 0% and 1 representing 100%. Expressing probabilities as ratios or percentages will be used interchangeably throughout the thesis. It is worth noting that the probabilities are not normalized (e.g. $Pa1$ and $Pa2$ do not necessarily have to sum to 100%). These probabilities help us to model the unpredictable nature of wireless network channels. A typical wireless channel would not have 100% success, arrival, and retransmission probabilities at all times. Nevertheless, to check the correctness of our agents we evaluated our model assuming these probabilities were equal to 100%. We first have the LTE packet arrival and success probabilities to be 100% while that of the DRL agent is 0% and later reverse the configurations to give higher priorities to the DRL agent. We have interesting outputs as shown in Figures 14(a) and 14(b) for LTE and DRL, respectively.

Initially, the LTE packet arrival and success probabilities are set to 100%, while those of the DRL agent are set to 0%. Later, the configurations are reversed to prioritize the DRL agent. The results are shown in Figure 14.

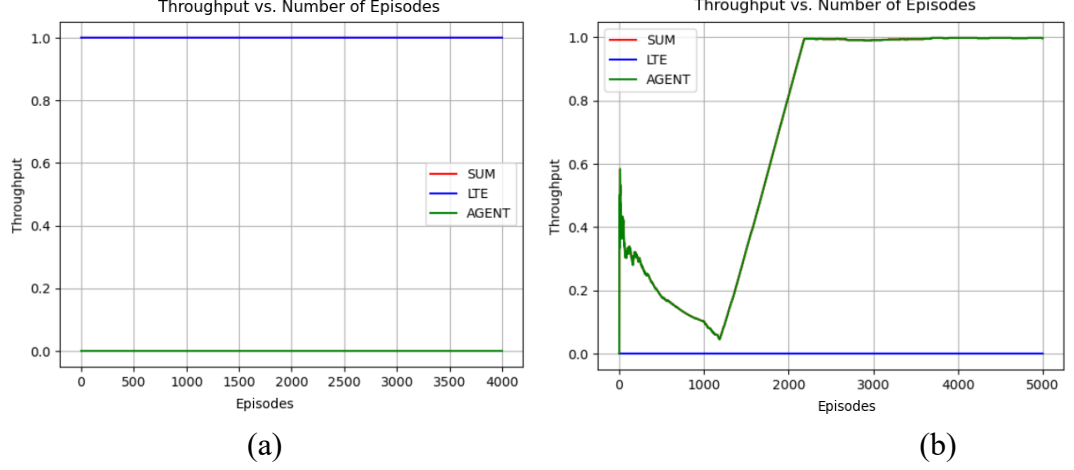


Figure 14: LTE and DRL with 100% Probabilities.

(a) $pa1, ps2, pt = 1, pa2, ps2 = 0$ (b) $pa1, ps1 = 0, pa2, ps2, pt = 1$

When LTE is given higher priority, we have a convergence rate of 0, and the throughput has hit 100% from the first episode and never comes down again, as seen in 14(a) above. On the other hand, the system takes some time to achieve 100% accuracy even with maximum probabilities given to the DRL agent. The transmission/retransmission probability for the two scenarios is also kept at 100%. The DRL agent is also responsible for coordinating the whole environment and needs some of its resources for general coordination rather than only achieving maximum throughput.

Exploring further, the overall system throughput lies above 35% if both have the same packet arrival and successful delivery probabilities. Even with the same probabilities, the LTE systems are allocated most of the spectrum for environmental transmission. DRL agent is observed using some spectrum before finally stabilizing and giving all the spectrum to the LTE system. This indicates how fast the DRL agent can understand

the environment and adapt to it even with the same probabilities as the base technology (LTE in this case). A typical example is shown in Figure 15 where both are transmitting with 50% probabilities.

Further exploration, as shown in Figure 15, reveals that the overall system throughput exceeds 35% when both LTE and the DRL agent have equal packet arrival and successful delivery probabilities, 50% each. Despite the equal probabilities, the LTE system initially occupies most of the spectrum resources. Meanwhile, the DRL agent utilizes some of the spectrum before eventually stabilizing and reallocating the entire spectrum to the LTE system. This demonstrates the DRL agent's ability to rapidly understand and adapt to its environment, even when operating under the same probabilities as the base technology (in this case, LTE).

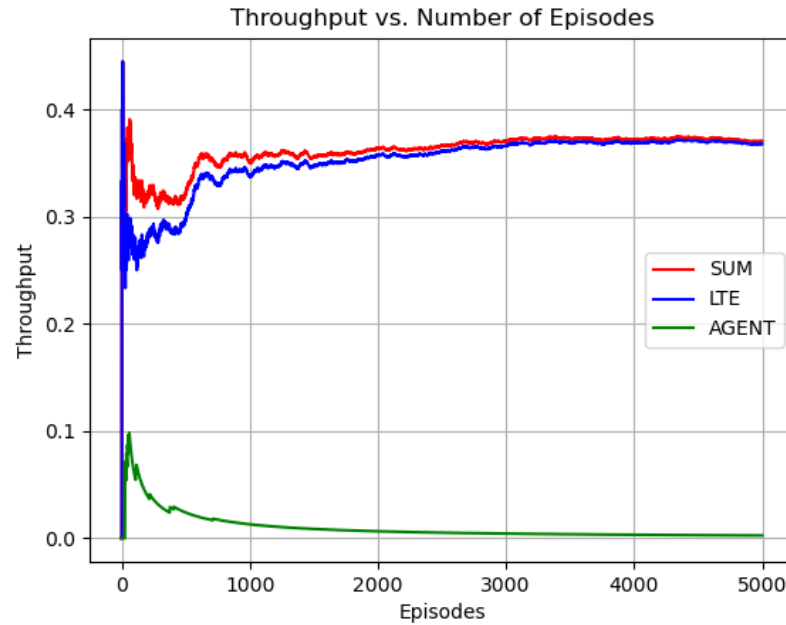


Figure 15: LTE and DRL with 50% Probabilities.
 $pa1, ps1, pa2, ps2 = 0.5, pt = 1$

The transmission/retransmission probability (pt) represents the possibility of agents retransmitting the packet in case of collision and/or channel error. The pt has a great

impact on the whole system's throughput. As illustrated in Figure 16, the overall system throughput with priority given to the base systems is depreciated to below 50% (compared to the initial 85% achieved with pt of 1) when the retransmission probability is set to 0.5. When pt was decreased to 0.2, the overall system throughput further dropped to below 20%. This shows the significant impact of retransmission on overall system throughput.

We achieve higher throughput if we allow the system to retransmit some of the failed packets when it is idle as against discarding them after a single try. On the other hand, the system throughput is slightly above 40% if priority is being given to the DRL agent with a 100% retransmission rate. This value, however, sits at 80% around 2500 episodes if the retransmission rate is decreased to 20% as shown in Figure 16(b). This implies that a high value of pt affects the system positively if higher priority is being given to LTE and the opposite if higher priority is being given to the DRL agent.

We can also observe in Figure 16(a) that the system can achieve higher throughput by allowing the retransmission of some failed packets during idle periods, instead of discarding them after a single attempt. When priority is given to the DRL agent with a 100% retransmission rate, the system throughput is slightly above 40%. However, as shown in Figure 16(b), this throughput increases to 80% after approximately 2500 episodes when the retransmission rate is reduced to 20% while the arrival and success probabilities are 0.2 and 0.9 for LTE and Agent respectively. This indicates that a high pt (transmission/retransmission probability) positively impacts the system when LTE is given higher priority. On the other hand, a lower pt is more beneficial when the DRL agent is prioritized.

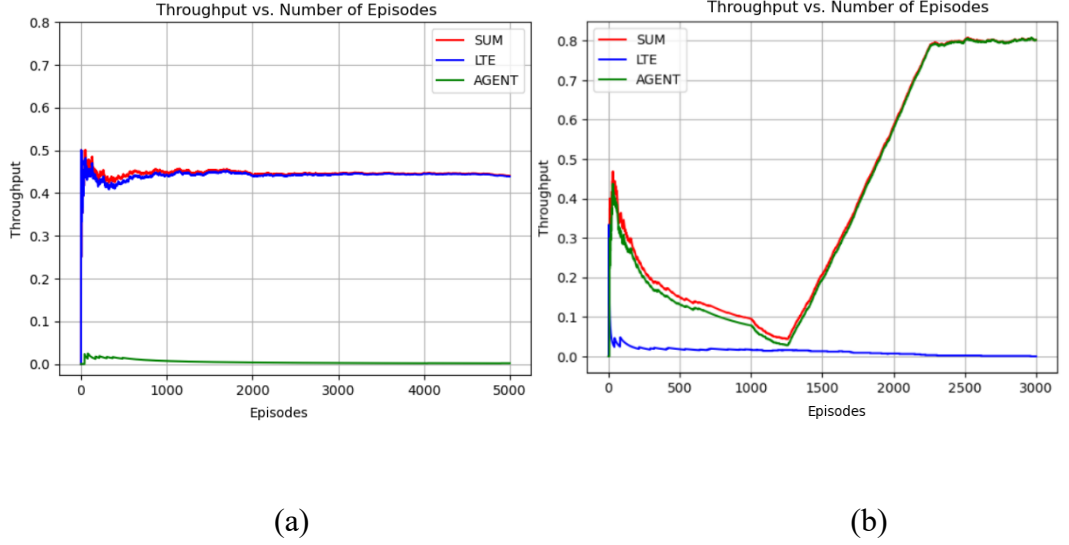


Figure 16: LTE and DRL with varied pt values.
 (a) $pa1, ps1 = 0.9, pa2, ps2 = 0.2, pt = 0.5$
 (b) $pa1, ps1 = 0.2, pa2, ps2 = 0.9, pt = 0.2$

4.1.2 Packets Expiry Duration

Packet expiry duration denotes how many time-slots a packet will wait before being discarded if it is not successfully sent. It is very important to consider different values of packet expiry duration based on the applications. For example, in real-time systems, some data might not be useful after certain time slots. This is also a common scenario in IIoT, where some sensor data are time-sensitive. It is important to have the flexibility of defining how long a packet can wait before being tagged unusable.

We investigated the influence of packet expiry duration (D) on the throughput. Table 4 below presents the observed throughput at 2000 episodes for different values of D , the packet expiry duration. As the table illustrates, shorter packet expiry durations, in the range of [1-3] slots, would generally result in lower throughput compared to longer expiry durations, 4 slots and above.

Table 4: LTE and DRL with varied packet expiry duration (D) values from 1 to 9.

Packets Expiry Duration (D)	Throughputs (%)
1	78
2	83
3	84
4	84
5	87
6	88
7	82
8	86
9	85

These results can be attributed to several factors including:

- **Packet Loss:** Shorter packet expiry durations increase the likelihood of packet loss. When multiple packets arrive simultaneously with packet short expiry durations, the system, limited to transmitting one packet at a time, may fail to transmit all packets before their deadlines, leading to losses. For instance, if three packets with a D value of 2 arrive concurrently, at least one will be lost.
- **Increased Idle Time:** Conversely, longer packet expiry durations allow more packets to accumulate in the queue, reducing the system's idle time and increasing the probability of a packet being available for transmission. This contributes to higher throughput.
- **Intelligent Scheduling:** The DRL agent effectively schedules packets to minimize collisions, even with a larger number of packets waiting in the queue. This mitigates the potential negative impact of longer packet expiry durations on collision rates.

Longer packet expiry durations generally improve throughput, but excessively high values may have diminishing returns. The results indicate that expiry durations of

around 5–6 time slots achieve the highest throughput, but cause longer delays. Increasing this value beyond this point leads to a decrease in throughput. On the other hand, short packet expiry durations result in higher packet loss. These results suggest that packet a moderate packet expiry duration ensures a balanced performance. It allows for enough time for successful retransmissions without being prematurely discarded. This ensures optimal throughput while minimizing delays.

This experiment highlights the crucial role of packet expiry duration in optimizing DRL performance while meeting application requirements. For example, real-time applications that prioritize low latency and minimal delay may favor shorter expiry durations, even at the cost of more packet loss. While applications with flexible time constraints may prefer longer expiry durations to maximize throughput. Hence, by carefully tuning the D value packet expiry duration, it is possible to balance the trade-off between packet loss and throughput, leading to efficient network operation for diverse applications.

4.1.3 *Learning Rates*

The learning rate controls how much the agent’s weights are updated during each learning episode. It directly affects how quickly the model adapts to new data. It is also a crucial factor in balancing between exploration (trying out new actions) and exploitation (leveraging known information) in the learning process. A high learning rate means the agent quickly adapts to new experiences, while a low learning rate makes it more conservative and resistant to change. A high learning rate may lead to a faster convergence rate but it can cause the learning process to be unstable, as the agent might overshoot the optimal solution or get stuck in sub-optimal policy because it is updating too drastically. However, a low learning rate can result in lower convergence rates. This

conservative approach generally makes the learning process more stable, as updates are smaller. Figure 17 shows the achievable throughput when the learning rate is 0.001, 0.01, 0.1, and 1. We can conclude that lower learning rates cause lower convergence rates compared to higher learning rates. In turn, the higher learning rates achieve higher throughput earlier than those with lower learning rates. Learning rates of 0.001 and 0.01 gives throughput below 80% at 800 episodes while higher learning rates of 0.1 and 1 give above 80% around 800 episodes. Though with negligible differences, it is safe to say high learning rates learn faster and give higher throughput compared to lower values. The learning rate of 0.01, Figure 17(a), is optimal for throughput improvement. It provided a good balance; ensuring stable learning and relatively faster convergence. Higher learning rates (e.g., 0.1 and 1 in Figures 17(c), and 17(d)) resulted in faster convergence but often overshoot optimal solutions, leading to sub-optimal performance.

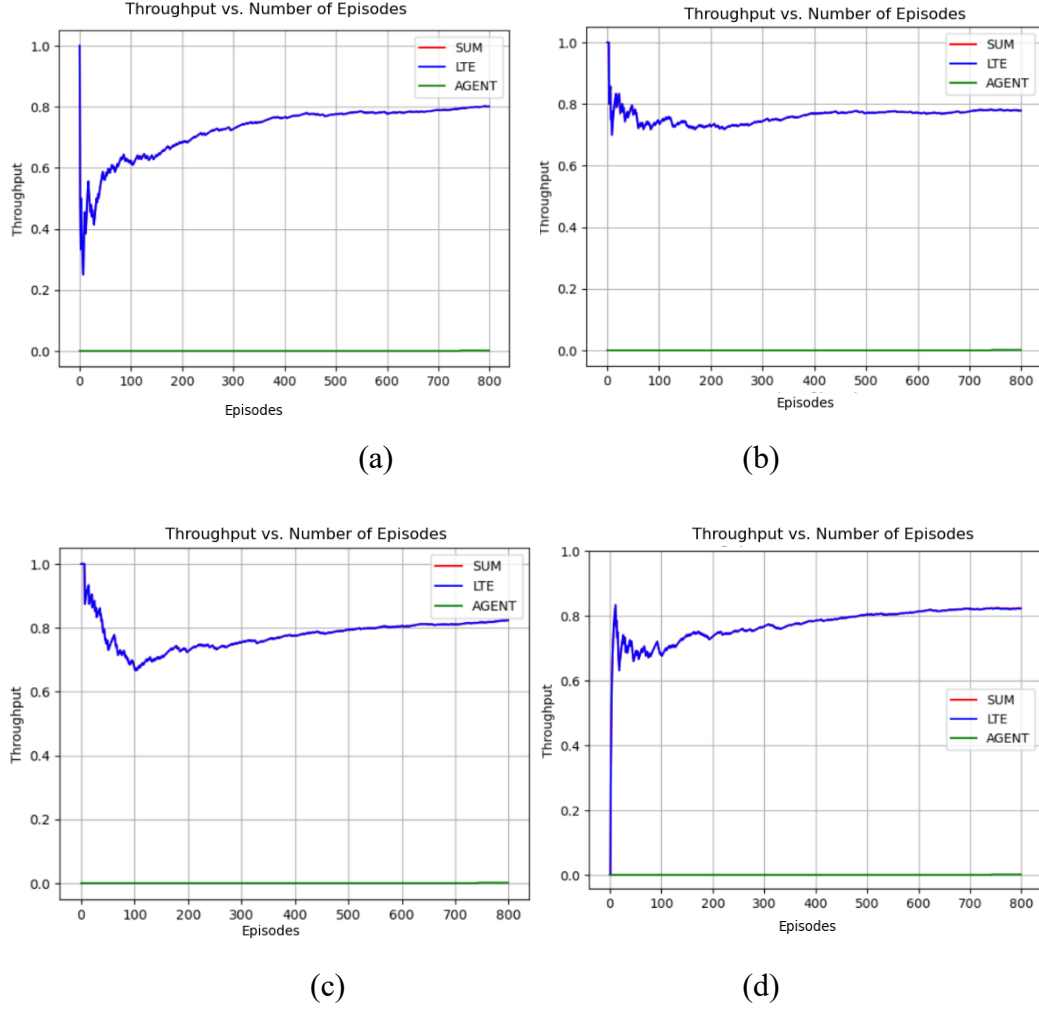


Figure 17: LTE and DRL with varied Learning Rate values.
(a) 0.001 (b) 0.01 (c) 0.1 (d) 1

4.1.4 Discount Factors (Gamma)

The discount factor denoted as gamma (γ) determines the weight of future rewards relative to immediate rewards. It influences the agent's ability to prioritize long-term goals over short-term ones. It is a crucial parameter used to balance immediate rewards versus future rewards in the agent's decision-making process. The discount factor takes values in the range between 0 to 1. When it is 0, the agent only considers immediate rewards, i.e. the agent prioritizes short-term gains and does not consider potential long-term benefits or consequences of its actions. As it approaches 1, the agent puts more emphasis on maximizing cumulative rewards over time, even if it means sacrificing

immediate rewards. The discount factors also interact with the exploration-exploitation trade-off. A high discount factor may lead the agent to explore more since future rewards are valuable, while a low value could lead to more exploitation of immediate rewards. Also, a high discount factor can lead to higher variance in the estimated returns, especially in stochastic environments, which can complicate learning while lower γ reduces this variance but at the cost of potentially missing out on better long-term strategies.

We have examined how the discount factor (γ) impacts the throughput of dynamic spectrum sharing in a wireless network controlled by a deep reinforcement learning (DRL) agent. Table 5 below presents the throughput achieved at 2000 episodes for different discount factors.

Table 5: LTE and DRL with Varied Discount Factors.

Discount Factor (γ)	Throughputs (%)
0	78
0.2	81
0.4	83
0.6	86
0.8	84
1	83

As can be concluded from the table, lower values of γ , in the range [0-0.4], generally lead to lower throughput compared to higher values, in the range [0.6-1]. This trend can be attributed to the following:

- **Short-term vs. Long-term Optimization:** Lower γ values encourage the DRL agent to prioritize immediate rewards, potentially sacrificing long-term performance. This may result in suboptimal policies that do not fully exploit the network's capacity.

- **Convergence Speed:** Lower γ values lead to faster convergence during training.

However, this rapid convergence may come at the cost of settling on a suboptimal policy that focuses on short-term gains. Higher γ values, while converging slightly slower, allow the agent to explore a wider range of strategies and potentially discover a more optimal long-term policy.

The results suggest that a discount factor of 0.6 achieves the highest throughput in this scenario. It strikes a balance between short-term and long-term optimization, allowing the agent to learn a policy that maximizes overall performance. The choice of the discount factor depends on the specific application and its objectives. Applications that require rapid adaptation and prioritize immediate performance may benefit from lower γ values. However, for applications where long-term stability and optimal resource utilization are critical, higher γ values are more suitable.

4.1.5 Fairness Index (F)

The fairness index controls how equally resources are distributed among all users or devices in the network. In this scenario, it controls how equitably spectrum resources are distributed among all users (i.e. LTE and NB-IoT devices). A fairness index closer to 1 indicates that fair allocation of spectrum is highly desired, while values closer to 0 indicate fairness is not desired.

Impact of Fairness Index on Agent Performance: The effect of F on the dynamic resource allocation by a deep reinforcement learning (DRL) agent was analyzed with respect to throughput, particularly under varying arrival and transmission probabilities.

- High Throughput Scenario ($F=0$)

When $F=0$, fairness is not enforced, allowing the DRL agent to maximize throughput by allocating all resources to optimize performance. For instance, with high arrival and transmission probabilities for the DRL agent, the system achieves a maximum throughput of approximately 45%, as shown in Figure 18(a).

- High Fairness Scenario ($F=1$)

At $F=1$, the agent prioritizes serving all entities equitably, resulting in a reduced throughput as depicted in Figure 18(b). The agent's allocation strategy shifts towards ensuring fairness, which sacrifices some performance efficiency to meet the fairness objective.

➤ **Impact of Fairness on LTE System Performance:** When higher probabilities are assigned to the LTE system, the impact of F on throughput is less pronounced. For both $F=0$ and $F=1$, throughput reaches approximately 85%, as illustrated in Figures 18(c) and 18(d). This suggests that fairness has a minimal effect on overall system performance when the LTE system is given higher priorities.

A low fairness index ($F \approx 0$) leads to higher throughput, as the DRL agent focuses solely on maximizing rewards without considering fairness constraints. A high fairness index ($F \approx 1$) reduces throughput due to the added complexity of balancing fairness and optimal resource allocation strategies. The impact of fairness is more noticeable when the DRL agent is given higher arrival and transmission probabilities, emphasizing its role in shaping resource distribution policies.

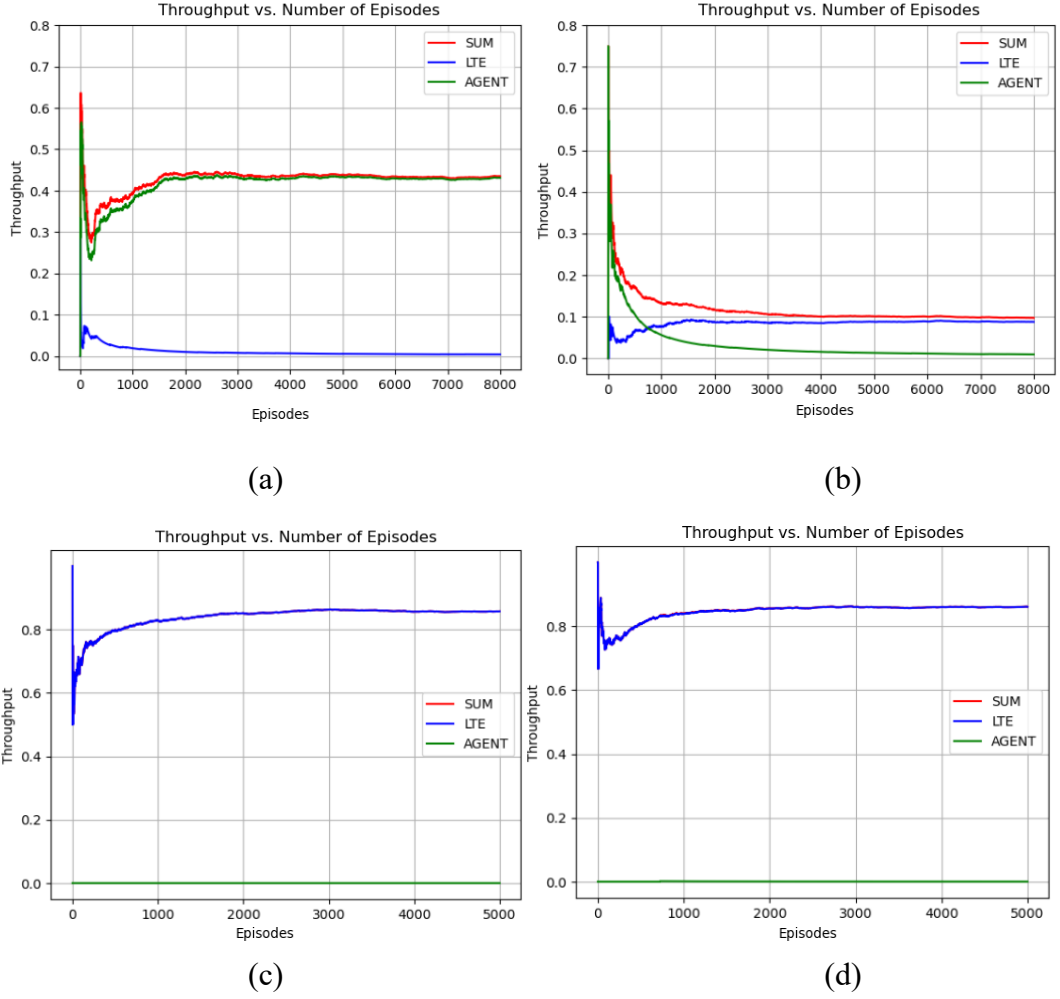


Figure 18: LTE and DRL with varied Fairness Index.
(a, b) $pa1, ps1=0.2, pa2, ps2=0.9$ (c, d) $pa1, ps1=0.9, pa2, ps2=0.2$
(a, c) $F = 0$ (b, d) $F = 1$

4.1.6 Recurrent Neural Network

The RNN architecture construction has been discussed in section 3.1.6. We used a Recurrent Neural Network (RNN) with a Gated Recurrent Unit (GRU) layer followed by fully connected (Dense) layers with ReLU activation functions. GRUs are a type of RNN that is designed to capture temporal dependencies in sequential data while addressing the vanishing gradient problem that can occur with standard RNNs. The GRU layer processes the input sequences and captures temporal dependencies. Our data is shaped by state length and feature numbers indicating that the model is processing

sequences where the state length represents sequence length (number of time steps), and features represent the number of features at each time step.

After the GRU layer, the model has one or more Dense layers with ReLU activation functions. These layers are used to transform the high-level temporal features extracted by the GRU into the final output prediction. The final Dense layer has many units, which are the product of the number of actions and the number of nodes, which suggest that the model is producing a vector of predictions. The model is trained using Mean Squared Error (MSE) loss and the RMSprop optimizer. With State Length, features, Number of actions, and number of nodes as 20, 8, 2, and 3, respectively, we have our input values as 20, 64, 64, 64 and output as 64, 64, 64, 6, as given in Figure 19. Figure 12 from section 3 also gives the pictorial representation of the architecture.

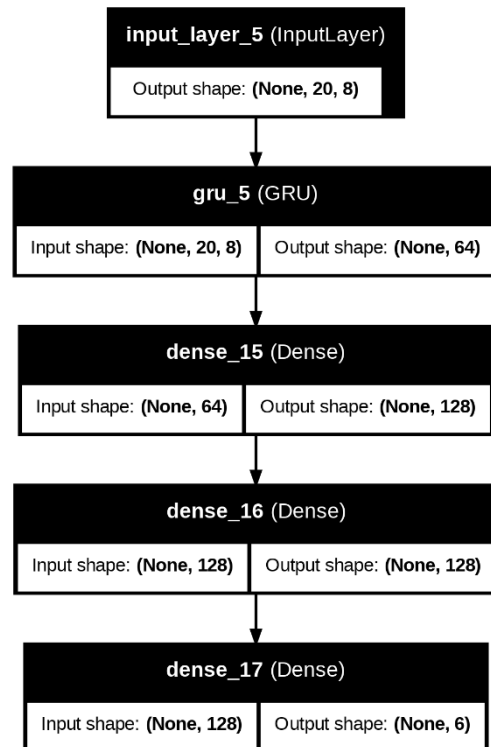


Figure 19: Architecture of Deep Recurrent Neural Network Model Used

To investigate how the architecture of the Recurrent Neural Network (RNN) affects the DRL model in this work, we varied the number of nodes in the dense layers and the number of dense layers themselves. The number of nodes, dense layers, and the throughputs achieved are given in Table 6 below.

Table 6: LTE and DRL with varied RNN configurations

Number of nodes	Dense Layers	Throughputs (%)
64	2	78
128	2	86
128	3	70
256	1	61
256	2	76
256	2	73

- **Optimal Number of Nodes:** The results show that 128 nodes in the dense layers yield the highest throughput exceeding 85%. Increasing the number of nodes to 256 leads to a decrease in throughput (76% and 73%), suggesting that having too many nodes can negatively impact performance. This could be due to overfitting, where the model starts to memorize the training data instead of generalizing well to new situations.
- **Optimal Number of Dense Layers:** Two dense layers appear to be the optimal configuration for our scenario. Using only one dense layer results in significantly lower throughput (61%). Increasing the number of dense layers to three also reduces throughput (70%), indicating that adding more layers does not necessarily improve performance and might even hinder it.
- **Runtime Considerations:** Increasing the number of layers and nodes generally leads to higher runtime. This is an important consideration when designing RNN architectures, as a more complex model requires more computational resources and time to train and execute.

This experiment highlights the importance of carefully choosing the architecture of an RNN (or other deep learning) for optimal performance in wireless network management. In our case, two dense layers with 128 nodes achieved the highest throughput. However, it is crucial to consider the potential for overfitting and the trade-off between model complexity and runtime when designing RNNs for similar tasks.

4.2 NB-IoT AND DRL AGENT COEXISTENCE

The second phase of the experiment includes the coexistence of NB-IoT and DRL in the environment. The NB-IoT System considers features such as the power-saving mode and smaller packet size. After multiple experiments, we achieved above 80% throughput in the wireless network environment. This is significant when considering the challenging nature of the wireless channel. A major difference between this output and the LTE system is the fact that it takes more episodes to achieve this performance. While we comfortably recorded throughput of above 80% for LTE in approximately 1000 episodes (see Figure 20), we approached almost 2000 episodes before achieving the same for the NB-IoT System. This is evident because the two environments are different. The NB-IoT System uses power saving mode which implies that the system is not sending any packets in certain time slots, among other factors. The results in Figure 20 were achieved based on the system parameters given in Table 7 and the RNN model in Figure 12.

Table 7: NB-IoT-DRL parameters

Parameters	Values
State history (M)	20
Learning rate	0.01
Fairness Index (F)	1
Discount factor (Γ)	0.9
NB-IoT packet expiry duration ($D1$)	3
DRL packet expiry duration ($D2$)	3
NB-IoT Arrival Prob. ($Pa1$)	0.9
DRL Arrival Prob. ($Pa2$)	0.2
NB-IoT Success Prob. ($Ps1$)	0.9
DRL Success Prob. ($Ps2$)	0.2
Trans/retransmission Prob. (Pt)	1

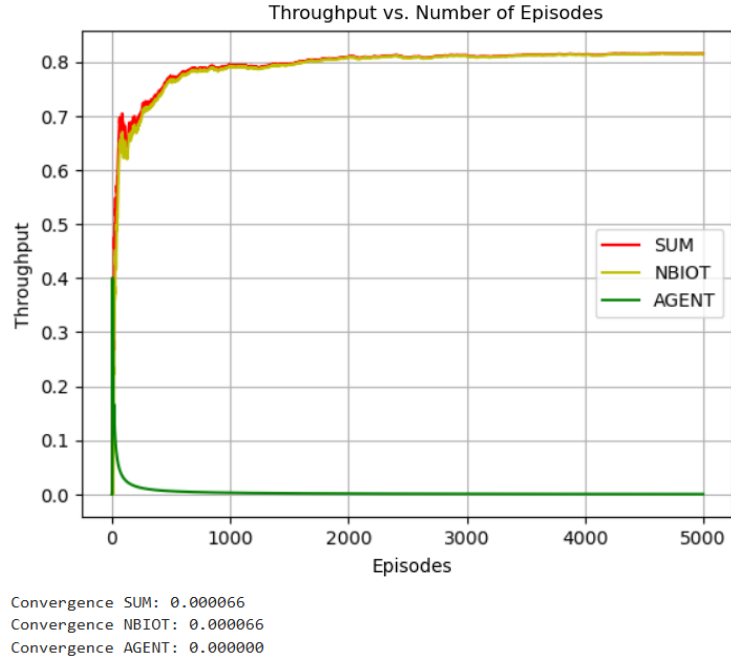


Figure 20: NB-IoT and DRL Coexistence Throughput.
 $pa1, ps1 = 0.9, pa2, ps2 = 0.2, pt = 1$

Even though it is not typical to have a wireless environment with 100% arrival, success, and retransmission probabilities, we model the system to operate under such an

assumption in favor of the NB-IoT. Specifically, we set the probabilities of NB-IoT to 100% and that of DRL agents to 100%. The results are shown in Figure 21.

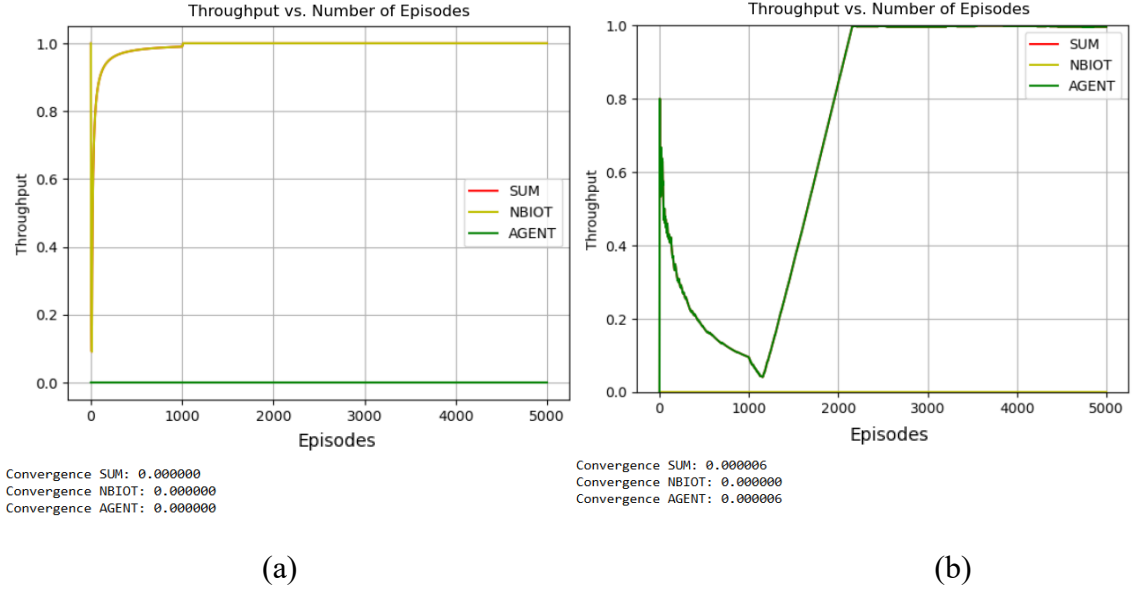


Figure 21: NB-IoT and DRL with 100% Probabilities
(a) $pa1, ps1, pt = 1, pa2, ps2 = 0$ (b) $pa1, ps1 = 0, pa2, ps2, pt = 1$

Figure 21 shows different trends from those presented in an earlier experiment when studying the LTE-only system. The LTE-only system gave 100% throughput from the first episode if given 100% probabilities for packet arrival and success transmission. NB-IoT on the other hand did not achieve 100% throughput with similar configurations till after 1000 episodes. The DRL agent behaves similarly in both experiments apart from the negligible difference in convergence rates. The retransmission probability (pt) significantly influences throughput, especially when the NB-IoT system is assigned high arrival and success probabilities. For instance, as shown in Figure 22, a pt value of 0.2 results in a throughput of approximately 20%. The system performs optimally with higher pt when the NB-IoT system is prioritized, but lower pt values are more favorable when the DRL agent has higher arrival and success priorities.

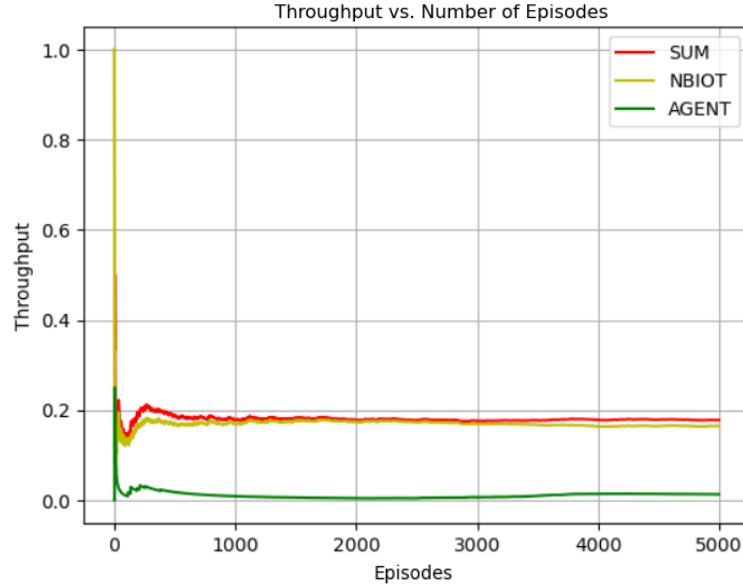


Figure 22: NB-IoT and DRL with pt of 0.2

Figure 23 shows the throughput of the system when pt is 0.2 and 1. It is worth noting that $pa1$ and $ps1$ are set to 0.2 while $pa2$ and $ps2$ are set to 0.9 implying higher priorities are given to the DRL agent. A lower retransmission rate, 0.2 for example gives a throughput of 70% as shown in Figure 23(a) which is 30% higher when the retransmission rate is set to 1 as shown in Figure 23(b). The DRL agent is the brain of the system and has an adaptive nature that allows it to learn and optimize in environments where low retransmission probabilities are present reducing the adverse effects and maintaining performance in coexistence scenarios. Higher retransmission probabilities introduce a range of challenges that can degrade the performance of DRL agent in a coexistence environment with NB-IoT. The increased latency, resource contention, learning instability, and interference all contribute to a more complex and less predictable environment, making it harder for the DRL agent to effectively learn and adapt.

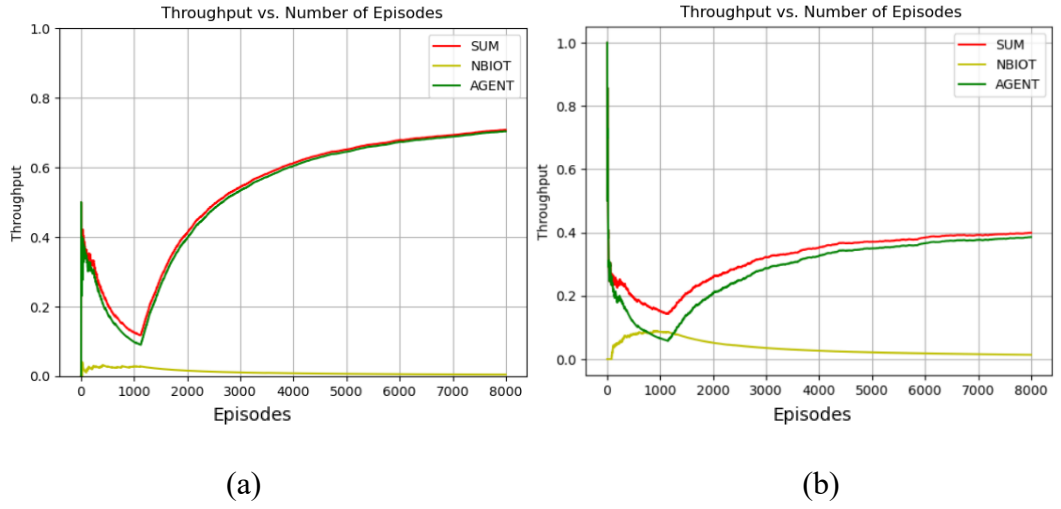


Figure 23: Impact of pt on DRL agent while coexisting with NB-IoT.
 $pa1, ps1 = 0.2, pa2, ps2 = 0.9$ (a) $pt = 0.2$ (b) $pt = 1$

Exploring further, we observe the behavior of the system if the transmission and arrival probabilities of the two agents are set to 50%. Transmission/Retransmission probabilities also have a notable impact on the system behavior. As shown in Figure 24 a, NB-IoT has recorded a throughput of around 30% from 500 episodes while the system is unstable till after 3000 episodes in Figure 24(b). An interesting observation here is that both Figure 24(a) and (b), though having the same probability configuration, give different outputs due to the values of pt being set to 1 and 0.2, respectively. Also, a pt value of 1 gives the spectrum to the NB-IoT System while a pt value of 0.2 allocates most of the spectrum to the DRL agent. It is worth noting that with probabilities of 50% for the 2 agents, the system is highly unstable and there will be lots of collisions since the agents have the same probabilities and are 50%.

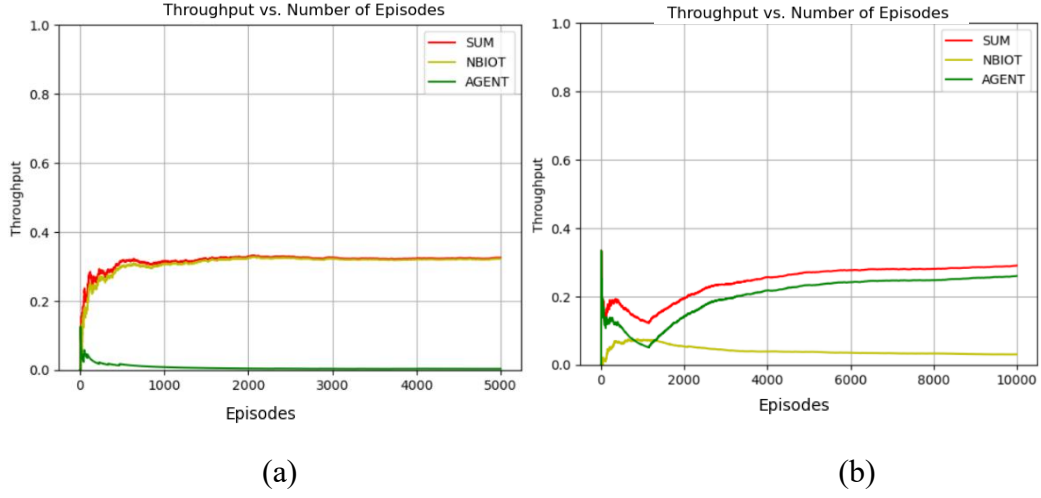


Figure 24: NB-IoT and DRL each with 50% Probabilities.
 $pa1, ps1, pa2, ps2 = 0.5$ (a) $pt = 1$ (b) $pt = 0.2$

4.3 LTE AND NB-IoT SYSTEMS COEXISTENCE

The third phase of our work involves the LTE and NB-IoT coexisting in the same environment. This environment is more complex than the 2 environments explored earlier. The DRL agent is also present to coordinate the environment. The LTE and NB-IoT Systems do not have full knowledge of each other's operation but derive inference from the feedback received from the base station. For example, after sending a packet, each agent can know the status of the other based on the feedback from the Base Station, whether ACK, NACK, or nothing. Setting the arrival and success probabilities of LTE to 0.9 and that of NB-IoT to 0.1 and other parameters as given in Table 8, we achieve a throughput of approximately 70% for LTE and NB-IoT as given in Figure 25(a) and (b), respectively.

Table 8: LTE-NB-IoT Parameters

Parameters	Values
State history (M)	50
Learning rate	0.01
Fairness Index (F)	1
Discount factor (Γ)	0.9
LTE packet expiry duration ($D1$)	3
NB-IoT packet expiry duration ($D2$)	3
Trans/retransmission Prob. (P_t)	1

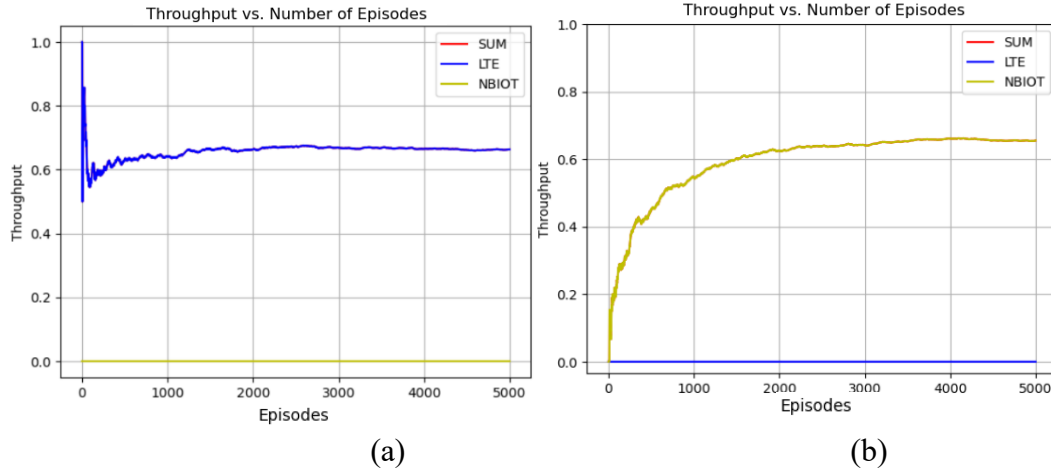
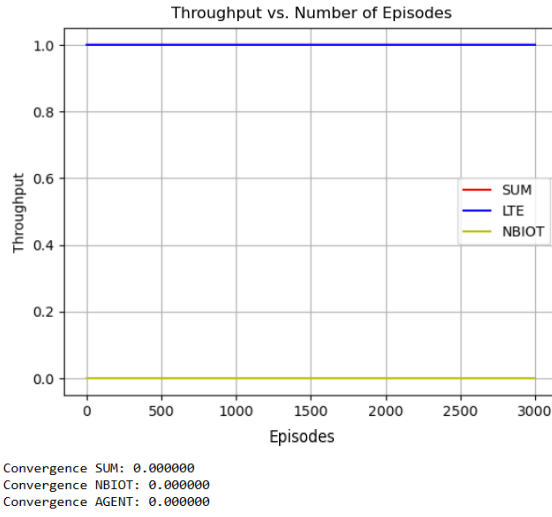


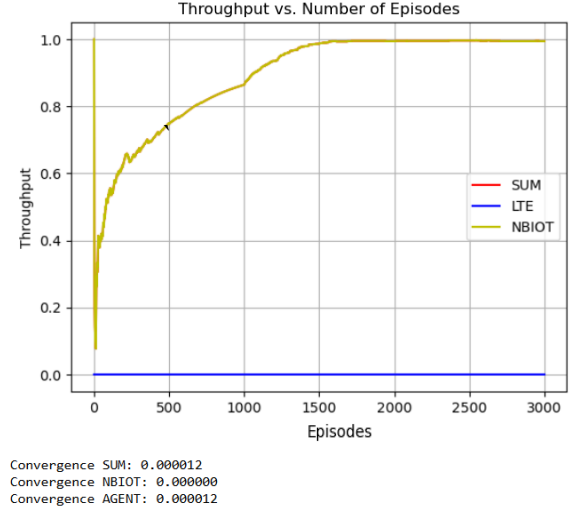
Figure 25: LTE and NB-IoT Systems Coexistence Throughput.
 (a) $pa1, ps1 = 0.9, pa2, ps2 = 0.1$ (b) $pa1, ps1 = 0.1, pa2, ps2 = 0.9$

The patterns of both LTE and NB-IoT devices are similar to those they exhibit when operating alone with the DRL agent in the same environment discussed in previous sections. The LTE system achieves maximum throughput in approximately 1000 episodes. NB-IoT, on the other hand, takes more time than LTE; it achieves both maximum throughput and stability after 4000 episodes.

Following the same approaches used in Phases 1 and 2, we also model the system to operate with 100% probability to test the workability of our system. Just like the earlier behavior, the LTE achieves stability and maximum throughput from the first episode while NB-IoT does so around 1500 episodes, as shown in Figures 26(a) and (b).



(a)



(b)

Figure 26: LTE and NB-IoT Systems with 100% Probabilities

Exploring the environment further, we experimented with multiple probabilities for the agents and observed their behavior. In Figure 27(a), we set the probabilities of LTE and NB-IoT to 0.9 and 0.2, respectively, and reverse the values in Figure 27(b). We recorded an output of approximately 50% for both.

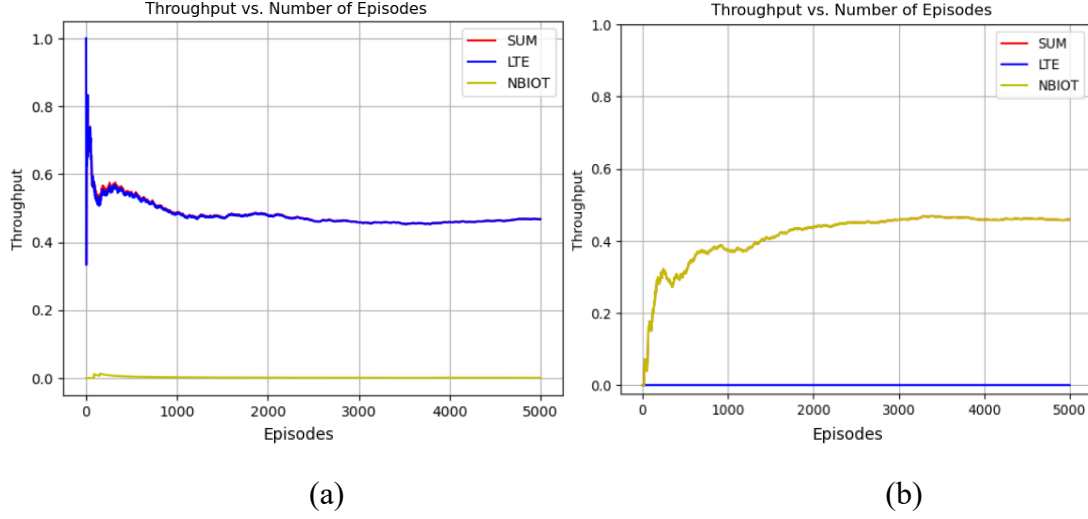


Figure 27: LTE and NB-IoT Systems with 0.9 and 0.2 Probabilities.
 (a) $pa1, ps1 = 0.9, pa2, ps2 = 0.2$ (b) $pa1, ps1 = 0.2, pa2, ps2 = 0.9$

In Figure 28, we set the probabilities to 0.8 and 0.2, giving a higher value to LTE, as shown in Figure 28(a), and to NB-IoT in Figure 26(b). The throughputs are lower compared to what we achieved in Figure 25. The LTE system yields 40% throughput, while the NB-IoT yields slightly below 40%.

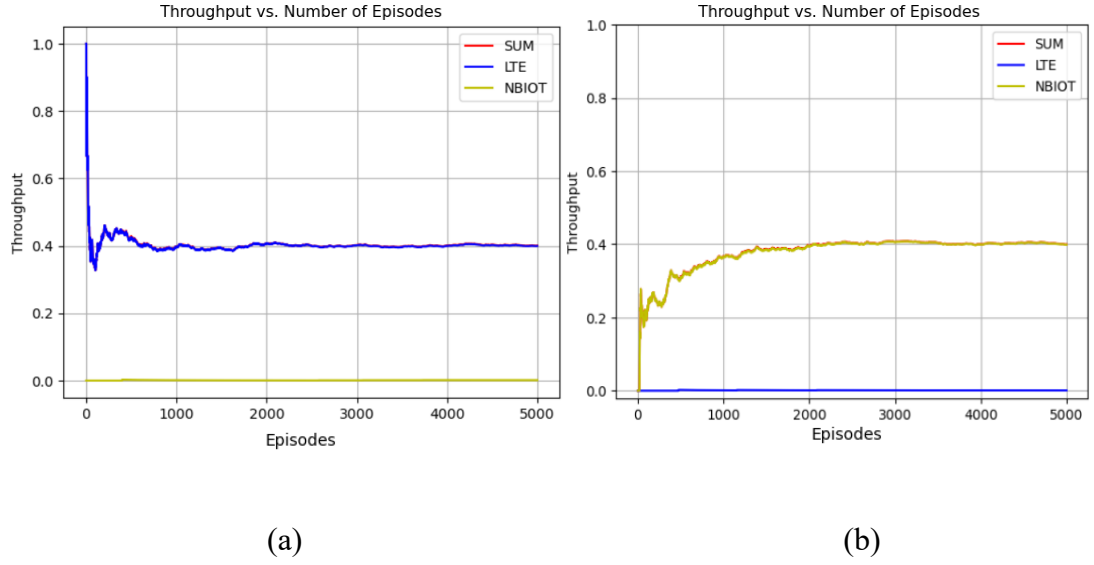


Figure 28: LTE and NB-IoT Systems with 0.8 and 0.2 Probabilities.
(a) $pa1, ps1 = 0.8, pa2, ps2 = 0.2$ (b) $pa1, ps1 = 0.2, pa2, ps2 = 0.8$

Both LTE and NB-IoT have their throughputs around 60% when the probabilities are set to 0.8 and 0.1, as shown in Figures 29(a) and (b), respectively. The primary agent is sending with the same probabilities as in Figure 28, but the other agent reduced by 10% which produced increased throughput of approximately 20%.

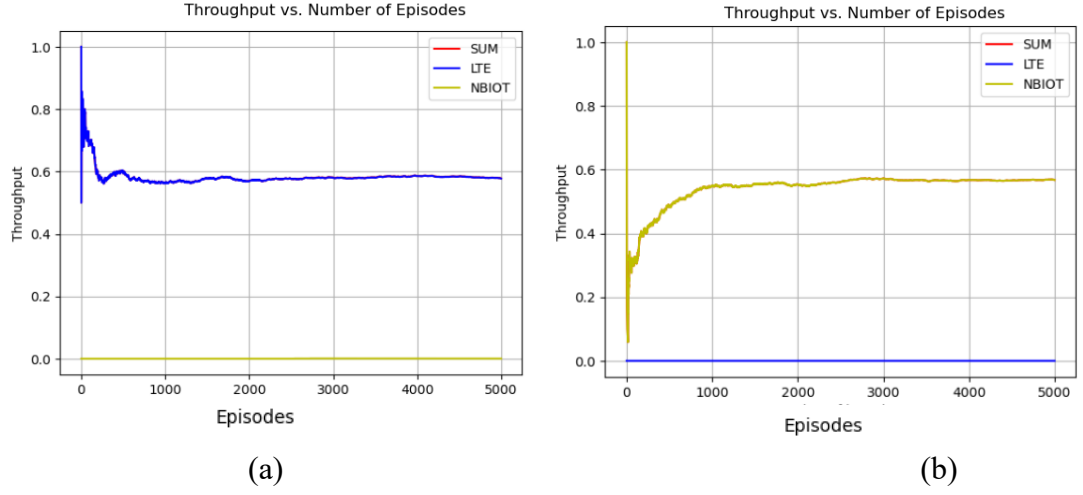


Figure 29: LTE and NB-IoT Systems with 0.8 and 0.1 Probabilities.
(a) $pa1, ps1 = 0.8, pa2, ps2 = 0.2$ (b) $pa1, ps1 = 0.2, pa2, ps2 = 0.8$

Though with a throughput of around 10%, both share very limited successful transmission when the probabilities are set to 50% for each of the LTE and NB-IoT Systems. This is represented in Figure 30 for LTE and NB-IoT, respectively. We will recall that the system has a high collision rate when the probabilities of both are the same or very close to each other. In this highly destabilized situation, our system can still fairly share the little available spectrum among the two agents of LTE and NB-IoT.

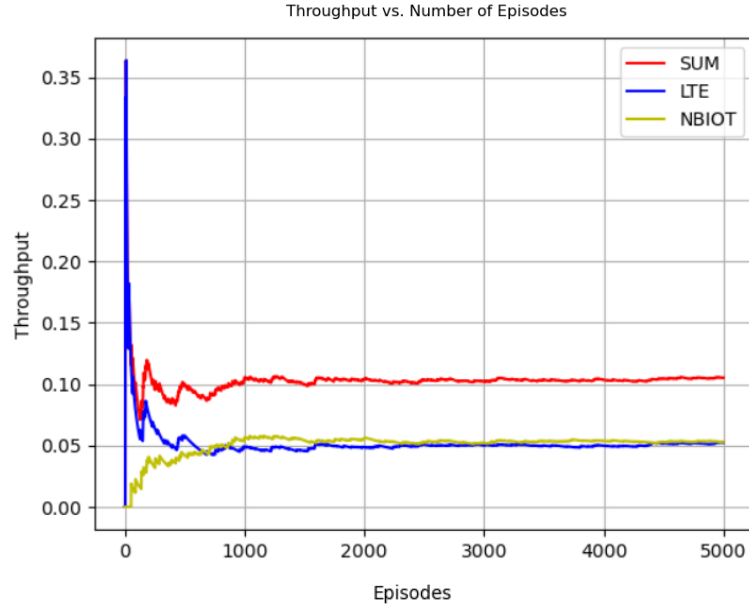


Figure 30: LTE and NB-IoT Systems with 50% Probabilities.
 $pa1, ps1, pa2, ps2 = 0.5$

With 100% probabilities for each of the agents, we achieve a throughput of 0% after around 1000 episodes. The system experiences high collision and it is very difficult to resolve as the two agents are sending with their maximum probabilities. With probabilities less than 100%, the agent has some reservations which allows the other agents to use the spectrum which is not feasible when they are both sending with 100% probabilities.

4.4 PERFORMANCE COMPARISON

To check the workability of our agent, we compare our proposed model with Deep Reinforcement Learning Multiple Access – DLMA [6] and Tiny State-space Reinforcement Learning random Access-TSRA [20]. The three research works use Deep Reinforcement Learning algorithms to navigate heterogeneous wireless networks, though considering different technologies. Comparing throughputs achieved over 10-time slots packet expiry duration, our approaches perform better in varied values of packet expiry duration time slots. Our system has approximately 10 and 40 percent increased throughput for different values of D compared to TSRA and DLMA, respectively. The pictorial representation of the comparison is given in Figure 31 below.

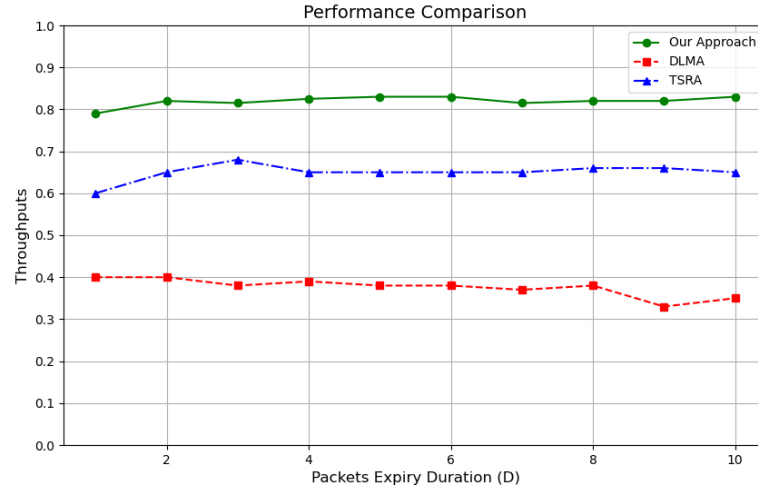


Figure 31: Performance Comparison

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

This thesis explores the dynamic sharing of spectrum resources in wireless networks, specifically focusing on heterogeneous environments where multiple devices can operate on different wireless communication technologies. The application of Deep Reinforcement Learning (DRL) techniques to dynamic spectrum sharing between Long-Term Evolution (LTE) and Narrowband Internet of Things (NB-IoT) devices was explored to address the challenge of efficient spectrum utilization in environments where diverse types of devices with varying communication requirements coexist.

The study began by investigating the performance of RL-based algorithms for spectrum allocation and management between the LTE system and the RL agent. Afterward, we added deep learning to the system to get better throughput and improved convergence rate. The study examines the performance of DRL-based algorithms for spectrum allocation and management between LTE and NB-IoT devices. Different environments were modeled including LTE, NB-IoT, and combined LTE & NB-IoT. Several experiments were conducted in three phases: LTE and DRL coexistence, the NB-IoT and DRL coexistence then finally the coexistence of LTE and NB-IoT.

Additionally, the thesis explored the impact of various parameters on the general system throughput. The results demonstrated that a DRL model in which Q-learning is combined with deep neural network-based significantly enhances the throughput and efficiency of dynamic spectrum sharing. The experiments showed that DRL agents could effectively learn and adapt their strategies to optimize throughput, achieve faster convergence rates, and improve overall network performance.

The key findings of the thesis include:

- **Improved Throughput and Convergence:** The DRL framework successfully optimized spectrum sharing, resulting in throughput improvements exceeding 80% with convergence rates as low as 66 microseconds. This indicates that DRL can efficiently balance the trade-offs between exploration and exploitation in dynamic spectrum allocation.
- **Impact of Retransmission Probabilities:** When LTE and DRL coexist, higher retransmission probabilities generally favor LTE by allowing LTE devices to maintain higher throughput levels. On the other hand, lower retransmission probabilities favor DRL agents by reducing the likelihood of congestion and improving performance.
- **Effect of packet expiry duration and Fairness Indices:** High packet expiry duration and low fairness index were found to contribute positively to the system performance. Specifically, enhancing throughput and overall efficiency. However, increasing expiry duration leads to latency in the overall packet arrival rate and a decrease in throughput.
- **Challenges with NB-IoT Coexistence:** The coexistence of NB-IoT with DRL agents showed similar throughput compared to when the DRL agent coexists with LTE but with NB-IoT having longer convergence times compared to LTE. This difference is attributed to the power-saving modes inherent in NB-IoT devices.
- **Performance in Coexistence Scenarios:** When LTE and NB-IoT systems coexist, the system experiences reduced throughput due to increased competition for spectrum resources. Despite this, the system was able to maintain good performance

given the specified parameters, including fair sharing of resources between the two technologies. This serves as a testament to the robustness of the DRL approach in handling multi-agent environments.

The key findings and insights of this work contribute to a deeper understanding of spectrum management and offer practical solutions for enhancing the performance of wireless networks. They underscore the potential of DRL in creating more efficient and adaptable spectrum-sharing strategies, paving the way for future advancements in network management and optimization.

Our future work will focus on analyzing additional coexistence scenarios involving other wireless technologies, such as 5G and Wi-Fi, to better understand the implications of DRL-based spectrum sharing across diverse network environments. We also plan to explore other reinforcement learning algorithms, integrate energy and interference in the system modeling, and examine their impacts on the overall system performance. Finally, since little attention was given to the network topology in the system's modeling, we look forward to studying the impact of different network topologies on the overall system performance.

REFERENCES

- [1] P. S. R. Henrique and R. Prasad, “The road for 6G multimedia applications,” in *2020 23rd International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2020, pp. 1–6.
- [2] Z. Shi, X. Xie, M. Kadoch, and M. Cheriet, “A spectrum resource sharing algorithm for IoT networks based on reinforcement learning,” in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, 2020, pp. 1019–1024.
- [3] W. S. H. M. W. Ahmad *et al.*, “5G technology: Towards dynamic spectrum sharing using cognitive radio networks,” *IEEE access*, vol. 8, pp. 14460–14488, 2020.
- [4] S. Bhattarai, J.-M. J. Park, B. Gao, K. Bian, and W. Lehr, “An overview of dynamic spectrum sharing: Ongoing initiatives, challenges, and a roadmap for future research,” *IEEE Trans Cogn Commun Netw*, vol. 2, no. 2, pp. 110–128, 2016.
- [5] H. Cha and S.-L. Kim, “A reinforcement learning approach to dynamic spectrum access in Internet-of-Things networks,” in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.
- [6] Y. Yu, T. Wang, and S. C. Liew, “Deep-reinforcement learning multiple access for heterogeneous wireless networks,” *IEEE journal on selected areas in communications*, vol. 37, no. 6, pp. 1277–1290, 2019.
- [7] A. Abou El Hassan, A. El Mehdi, and M. Saber, “NB-IoT and LTE-M towards massive MTC: Complete performance evaluation for 5G mMTC,” *Indonesian*

Journal of Electrical Engineering and Computer Science, vol. 23, no. 1, pp. 308–320, 2021.

- [8] S. Liu, L. Xiao, Z. Han, and Y. Tang, “Eliminating NB-IoT interference to LTE system: A sparse machine learning-based approach,” *IEEE Internet Things J*, vol. 6, no. 4, pp. 6919–6932, 2019.
- [9] S. M. Baby and M. James, “A comparative study on various spectrum sharing techniques,” *Procedia Technology*, vol. 25, pp. 613–620, 2016.
- [10] G. Barb, F. Alexa, and M. Otesteanu, “Dynamic spectrum sharing for future LTE-NR networks,” *Sensors*, vol. 21, no. 12, p. 4215, 2021.
- [11] M. M. Afsar, T. Crump, and B. Far, “Reinforcement learning based recommender systems: A survey,” *ACM Comput Surv*, vol. 55, no. 7, pp. 1–38, 2022.
- [12] Y. Yu, S. C. Liew, and T. Wang, “Non-uniform time-step deep Q-network for carrier-sense multiple access in heterogeneous wireless networks,” *IEEE Trans Mob Comput*, vol. 20, no. 9, pp. 2848–2861, 2020.
- [13] Y. Li, “Deep reinforcement learning: An overview,” *arXiv preprint arXiv:1701.07274*, 2017.
- [14] V. Mnih *et al.*, “Human-level control through deep reinforcement learning,” *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [15] S. Liu, C. Pan, C. Zhang, F. Yang, and J. Song, “Dynamic spectrum sharing based on deep reinforcement learning in mobile communication systems,” *Sensors*, vol. 23, no. 5, p. 2622, 2023.

- [16] Z. Shi, X. Xie, H. Lu, H. Yang, M. Kadoch, and M. Cheriet, “Deep-reinforcement-learning-based spectrum resource management for industrial Internet of Things,” *IEEE Internet Things J*, vol. 8, no. 5, pp. 3476–3489, 2020.
- [17] H. Wei, Y. Peng, M. Yue, J. Long, F. AL-Hazemi, and M. M. Mirza, “A Deep Reinforcement Learning Scheme for Spectrum Sensing and Resource Allocation in ITS,” *Mathematics*, vol. 11, no. 16, p. 3437, 2023.
- [18] U. Challita and D. Sandberg, “Deep reinforcement learning for dynamic spectrum sharing of LTE and NR,” in *ICC 2021-IEEE International Conference on Communications*, 2021, pp. 1–6.
- [19] J. Xu, X. Zhaojun, Y. Wenli, W. Hu, A. Cabani, and H. Xinrong, “An intelligent mechanism for dynamic spectrum sharing in 5G IoT networks,” *Expert Syst Appl*, vol. 252, p. 124122, 2024.
- [20] D. Wu, L. Deng, Z. Liu, Y. Zhang, and Y. S. Han, “Reinforcement learning random access for delay-constrained heterogeneous wireless networks: A two-user case,” in *2021 IEEE Globecom Workshops (GC Wkshps)*, 2021, pp. 1–7.
- [21] L. Sharma, “NB-IoT for 5G: How 5G Aligns With NB-IoT,” in *Principles and Applications of Narrowband Internet of Things (NB-IoT)*, IGI Global, 2021, pp. 352–372.
- [22] F. J. Dian and R. Vahidnia, “LTE IoT technology enhancements and case studies,” *IEEE Consumer Electronics Magazine*, vol. 9, no. 6, pp. 49–56, 2020.
- [23] S. R. Borkar, “Long-term evolution for machines (LTE-M),” in *LPWAN technologies for IoT and M2M applications*, Elsevier, 2020, pp. 145–166.

- [24] A. D. Zayas and P. Merino, “The 3GPP NB-IoT system architecture for the Internet of Things,” in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2017, pp. 277–282.
- [25] M. B. Hassan, E. S. Ali, R. A. Mokhtar, R. A. Saeed, and B. S. Chaudhari, “NB-IoT: Concepts, applications, and deployment challenges,” in *LPWAN Technologies for IoT and M2M Applications*, Elsevier, 2020, pp. 119–144.
- [26] A. D\’iaz Zayas, F. J. Rivas Tocado, and P. Rodr\’iguez, “Evolution and testing of NB-IoT solutions,” *Applied Sciences*, vol. 10, no. 21, p. 7903, 2020.
- [27] L. Iiyambo, G. Hancke, and A. M. Abu-Mahfouz, “A Survey on NB-IoT Random Access: Approaches for Uplink Radio Access Network Congestion Management,” *IEEE Access*, 2024.
- [28] C. B. Mwakwata, H. Malik, M. Mahtab Alam, Y. Le Moullec, S. Parand, and S. Mumtaz, “Narrowband Internet of Things (NB-IoT): From physical (PHY) and media access control (MAC) layers perspectives,” *Sensors*, vol. 19, no. 11, p. 2613, 2019.
- [29] K. F. Muteba, K. Djouani, and T. Olwal, “5G NB-IoT: Design, Considerations, Solutions and Challenges,” *Procedia Comput Sci*, vol. 198, pp. 86–93, 2022, doi: 10.1016/j.procs.2021.12.214.
- [30] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.
- [31] S. Zocholl, “Development of Recurrent Neural Network Architectures for Hydrological Time Series Forecasting,” 2021.

- [32] B. Perez Orozco, “Recurrent neural networks for time series prediction,” University of Oxford, 2019.
- [33] K. Cho, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.
- [34] U. Onyekpe, V. Palade, S. Kanarachos, and S.-R. G. Christopoulos, “A quaternion gated recurrent unit neural network for sensor fusion,” *Information*, vol. 12, no. 3, p. 117, 2021.
- [35] R. Jain, *The art of computer systems performance analysis*. john wiley & sons, 1990.
- [36] A. Nauman, M. A. Jamshed, Y. Ahmad, R. Ali, Y. Bin Zikria, and S. W. Kim, “An intelligent deterministic D2D communication in narrow-band Internet of Things,” in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, 2019, pp. 2111–2115.

VITAE

Name : Sulaimon Oyeniya ADEBAYO

Nationality : Nigerian

Date of Birth : 23/05/1996

Email : sulaimon.o.adebayo@gmail.com

Address : 13, Oke-ayo area, Ido-Osun, 232111, Osun State, Nigeria

Academic Background

M.Sc. (Computer Engineering), 2024, King Fahd University of Petroleum & Minerals (KFUPM), Dhahran, Saudi Arabia. (3.47/4.0).

B.Tech. (Computer Science and Engineering), 2022, Guru Nanak Dev University, Amritsar, India. First Class with Distinction (8.61/10.0).

National Diploma (Computer Engineering), 2019, Federal Polytechnic, Ile-Oluji, Ondo State, Nigeria. Distinction (3.94/4.0).

Selected Publications

- Saini, M., **Adebayo, S.O.** & Arora, V. IoT-Fog-based framework to prevent vehicle–road accidents caused by self-visual distracted drivers. Multimedia Tools Appl (2024). <https://doi.org/10.1007/s11042-024-19050-w>.
- Saini, M., **Adebayo, S. O.**, Singh, H., Singh, H., & Sharma, S. Sustainable development goals for gender equality: Extracting associations among the

indicators of SDG 5 using numerical association rule mining. *Journal of Intelligent & Fuzzy Systems*, 44(3), 5151-5156. 2023.

- **Adebayo, S. O., & Saini, M.**, A scientometric study for scientific research publication on gender inequality. *Quality & Quantity*, 1-29, Springer, December 2022.
- Madanjit Singh, **Sulaimon Oyeniyi Adebayo**, Munish Saini, Jaswinder Singh, “Indian Government Online Education Initiatives in Response to COVID19 Crisis: A Case Study on Indian Higher Education System”, *Education and Information Technologies*, Springer, 26, 7569-7607, 2021.
- **Sulaimon Adebayo**, Abdulaziz Barnawi, Tarek Sheltami, and Muhammad Felemban, “Dynamic Spectrum Sharing In Heterogeneous Wireless Networks Using Deep Reinforcement Learning”. *IEEE Access* [Under Review]