American University in Cairo

AUC Knowledge Fountain

Theses and Dissertations

Student Research

Fall 12-27-2021

Machine Learning Based Critical Resource Allocation in Mixed-Traffic Cellular Networks

Mohamed Nomeir
mohamednomeir@aucegypt.edu

Follow this and additional works at: https://fount.aucegypt.edu/etds

Part of the Systems and Communications Commons

Recommended Citation

APA Citation

Nomeir, M. (2021).*Machine Learning Based Critical Resource Allocation in Mixed-Traffic Cellular Networks* [Master's Thesis, the American University in Cairo]. AUC Knowledge Fountain.
https://fount.aucegypt.edu/etds/1730

MLA Citation

Nomeir, Mohamed. *Machine Learning Based Critical Resource Allocation in Mixed-Traffic Cellular Networks*. 2021. American University in Cairo, Master's Thesis. *AUC Knowledge Fountain.*
https://fount.aucegypt.edu/etds/1730

This Master's Thesis is brought to you for free and open access by the Student Research at AUC Knowledge Fountain. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of AUC Knowledge Fountain. For more information, please contact mark.muehlhaeusler@aucegypt.edu.

Graduate Studies

Machine Learning Based Critical Resource Allocation in Mixed-Traffic Cellular Networks

A THESIS SUBMITTED BY

MOHAMED NOMEIR

TO THE

Electronics and Communications Engineering Graduate Program

September 7, 2021

in partial fulfillment of the requirements for the degree of

Master of Science in Electronics and Communications Engineering

Declaration of Authorship

I, Mohamed Nomeir, declare that this thesis titled, "Machine Learning Based Critical Resource Allocation in Mixed-Traffic Cellular Networks" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Abstract

The proliferation of cellular networks over the past two decades has encouraged the expansion of their use in many modern applications. These applications involve the use of data traffic of different quality of service (QoS) requirements. Some of these requirements are quite stringent such as in the case of critical Internet of Things (IoT) health care, military and homeland security applications. This situation resulted in imposing a variety of resource allocation requirements on the cellular network operation in a simultaneous manner.

In this thesis, we consider the challenging problem of mixed-traffic resource allocation, or scheduling, in cellular networks. We focus our attention on 5G network as the most recent version currently being deployed worldwide. In this regard, there are generally two, separate, scheduling problems in communication systems, namely, the down-link (DL) scheduling and the up-link (UL) scheduling. Each of these problems has separate requirements, even if they both share some similarities. The DL focuses on scheduling the already received packets to the intended receivers and informing the receivers with enough information to receive the data correctly. This kind of scheduling is completely implemented by and controlled at the base station of the system. On the other hand, the UL problem focuses on providing enough resources to user devices to send their data, when they have any. In this thesis, we consider the problem of uplink scheduling in 5G networks for mixed traffic that includes Ultra-Reliable Low and Latency Communications (URLLC) devices and enhanced Mobile Broad-Band (eMBB) users. Each of these types has different requirements and therefore a different mathematical model based on the scheduling technique. There are three main scheduling techniques to be considered in this case, namely, the grant-based (GB), semi-persistent, and grant-free (GF) techniques. Each of these scheduling techniques is suitable for a certain type of traffic and has its own mathematical model that describes the associated traffic behavior. Furthermore, there are three different techniques used in grant-free scheduling, namely, the reactive scheme, the k-repetitions scheme and the proactive scheme. It has been concluded, in this study, that the grant-based scheduling is the best scheme for the eMBB traffic while the grant-free scheduling is best suitable for the URLLC traffic. For this purpose, we devise a mathematical model for

the GF services using the k-repetitions Hybrid Automatic Repeat reQuest (HARQ) as the first model to define such traffic in a single cell. In addition, the GB scheduling model for eMBB traffic is adapted to fit our problem. We formulate the scheduling problem as a mixed-integer non-linear programming optimization problem. This type of problem is, in general, a complex problem due to its combinatorial nature. We introduce a complete system model that includes GF and GB subsystems. We introduce a novel mixed scheduler that combines the advantages of two well-known schedulers in the literature. We then introduce novel machine-learning based scheduling algorithms and evaluate them in comparison to some well-known algorithms in the literature in addition to the optimal bound that we also derive in this study. The results show that the proposed algorithms produce near-optimal results in real-time.

Acknowledgements

First, all my gratitude is due to Almighty God for giving me the strength, patience and determination that let me finish this thesis. I would like to thank my family for always being supportive during my M.Sc. journey.

I would like to express my deepest gratitude to my thesis advisors, Dr. Yasser Gadallah and Dr. Karim Seddik. They provided guidance and encouragement to me during different stages of this thesis and my whole masters journey. Their advice and experience in the field have had the best effects, which led to this thesis. They gave me technical as well as moral support.

Contents

List of Figures

List of Abbreviations

3GPP	**3**rd **G**eneration **P**artnership **P**roject
Best-CQI	**B**est **C**hannel **Q**uality **I**ndicator
CSI	**C**hannel **S**tate **I**nformation
eMBB	**e**nhanced **M**obile **B**road **B**and
GA	**G**enetic **A**lgorithm
GB	**G**rant **B**ased
GF	**G**rant **F**ree
gNB	next **g**eneration **N**ode- **B**
mMTC	**m**assive **M**achine **T**ype **C**ommunications
ML	**M**achine **L**earning
NN	**N**eural **N**etwork
OFDM	**O**rthogonal **F**requency **D**evision **M**ultiplexing
PDBV	**P**robability of **D**elay **B**ound **V**iolation
PRB	**P**hysical **R**esource **B**lock
RB	**R**esource **B**lock
RL	**R**einforcement **L**earning
SNIR	**S**ignal to **N**oise and **I**nterference **R**atio
SNR	**S**ignal to **N**oise **R**atio
sTTI	**s**hort **T**ransmission **T**ime **I**nterval
TTI	**T**ransmission **T**ime **I**nterval
URLLC	**U**ltra **R**eliable **L**ow **L**atency **C**ommunications
BW	**B**and**W**idth

List of Symbols

T^{RTT}	URLLC packet round trip time
M	Maximum allowable re-transmissions
τ	URLLC latency constraint
P_F	probability of delay bound violation
A_m	URLLC active probability at the m^{th} re-transmission
p_m	URLLC access success probability at the m^{th} re-transmission
R	number of resource blocks for URLLC traffic
$R_{assigned}$	number of assigned resource blocks for URLLC traffic by Algorithm 2
p_a	URLLC packet arrival probability
γ_{th}	SINR threshold
R_e	Accumulated rate of eMBB user e
N_a	NUmber of URLLC devices
ϵ	URLLC reliability constraint
N_f	number of frequency slots each TTI
N_t	number of sTTIs in each TTI
$\pi(.)$	RL training policy
$V(.)$	Discount accumulative reward function
E	Number of eMBB users in the cell
S_{ij}^{e}	Binary scheduling parameter
$\hat{S}_{ij}$	Integer scheduling parameter
$\mathcal{L}(.)$	Laplace Transform function
E[.]	Expected value of a random variable
log(.)	Logarithmic function of base 2

$h_{ij,e}$	Channel gain for eMBB user e for the (i, j) resource block
B	channel Bandwidth
N_0	Noise power spectral density
P_e	Transmission power for eMBB user e
$r(.)$	RL reward function
$a(.)$	RL action function
$s(.)$	RL state function

Chapter 1

Introduction

Currently, the existence of different traffic types that need to be carried by cellular networks presents a major challenge. This is due to the difference in the nature and needs of these traffic types which renders the resource allocation, or traffic scheduling, task quite challenging. The scheduling problem can accordingly be described as follows. The cellular base station, known as gNB in 5G systems, needs to receive the data from the senders and get it sent to the intended receivers. It needs to schedule the senders on a 2-D time-frequency grid to inform them when to send their data, which what is termed "uplink transmission". The receivers will then get informed as to when on the 2-D grid the data will be received by them, in what is called "downlink transmission". Since the resources (i.e. the bandwidth) are scarce, the scheduling process needs to be optimized to ensure these resources are not wasted, in addition to fulfilling the users' requirements. The problem of scheduling the receiving of the data from the transmitters is therefore called the uplink scheduling problem, while the transmission of data from gNB to the receivers is called the downlink scheduling problem. Each of these problems is solved separately and each has its own models. There are three types of users in 5G systems, namely, the enhanced Mobile BroadBand (eMBB), the massive Machine Type Communications (mMTC) devices, and the Ultra-Reliable and Low Latency Communications (URLLC) users as shown in Figure 1.1. The number of URLLC devices is generally greater than the eMBB nodes due to the wide number of applications supported by such traffic, as will be discussed in the next sections. However, the mMTC devices number surpasses the number of URLLC nodes as they are used in various applications in our daily life.

In the downlink direction, the eMBB users are scheduled first then the URLLC traffic is punctured in such a way that the loss of the accumulated eMBB rate is minimal. In this case, the gNB knows exactly the packets required to be transmitted and can be scheduled perfectly based on each traffic requirement and the scheduling policy. In adddition, if there is any available resources, it is granted for the mMTC devices. However, the problem is different in the uplink direction where the gNB does not know exactly when the URLLC nodes will send their data due to their sporadic nature. On the other hand, the eMBB nodes send scheduling requests to the gNB before transmission. As stated, each problem, the uplink, and downlink is solved separately even if there are some similarities. In the following sections, the traffic requirements of the main two traffic types, the eMBB and URLLC traffic, are studied, the nature and applications of each type are discussed. The traffic scheduling problem is discussed and its importance for any communications system is stated. Then, the problem addressed in this thesis is stated to fully describe the main problem. Machine learning approaches and potential for solving this kind of problems are discussed and different solution approaches are suggested. Finally, the contributions of our work and the structure of the thesis are discussed.

1.1 eMBB Versus URLLC Traffic Requirements

There is a significant difference between the eMBB traffic requirements and those of the URLLC traffic. The eMBB users require mainly high data rates while the URLLC users require, generally low data rates, with very low latency, in the order of 1 ms, and very high reliability, with a probability of error that is less than 10^{-5} [1]. In addition, in the uplink direction, each of these traffic types has its own way of communicating to the gNB. The eMBB requests a resource grant before sending its data, Then the gNB sends the assigned time slot and bandwidth for the node to send the data on. This handshaking needs about 10 ms to be established. This type of scheduling is called Grant-Based (GB) scheduling and it is suitable with the eMBB traffic requirements. The URLLC users, on the other hand, cannot use the handshaking techniques because their latency constraint will be violated. Therefore, the URLLC sends the data in an arrive-and-go manner. This is called the Grant-Free (GF) technique. To solve this problem, the

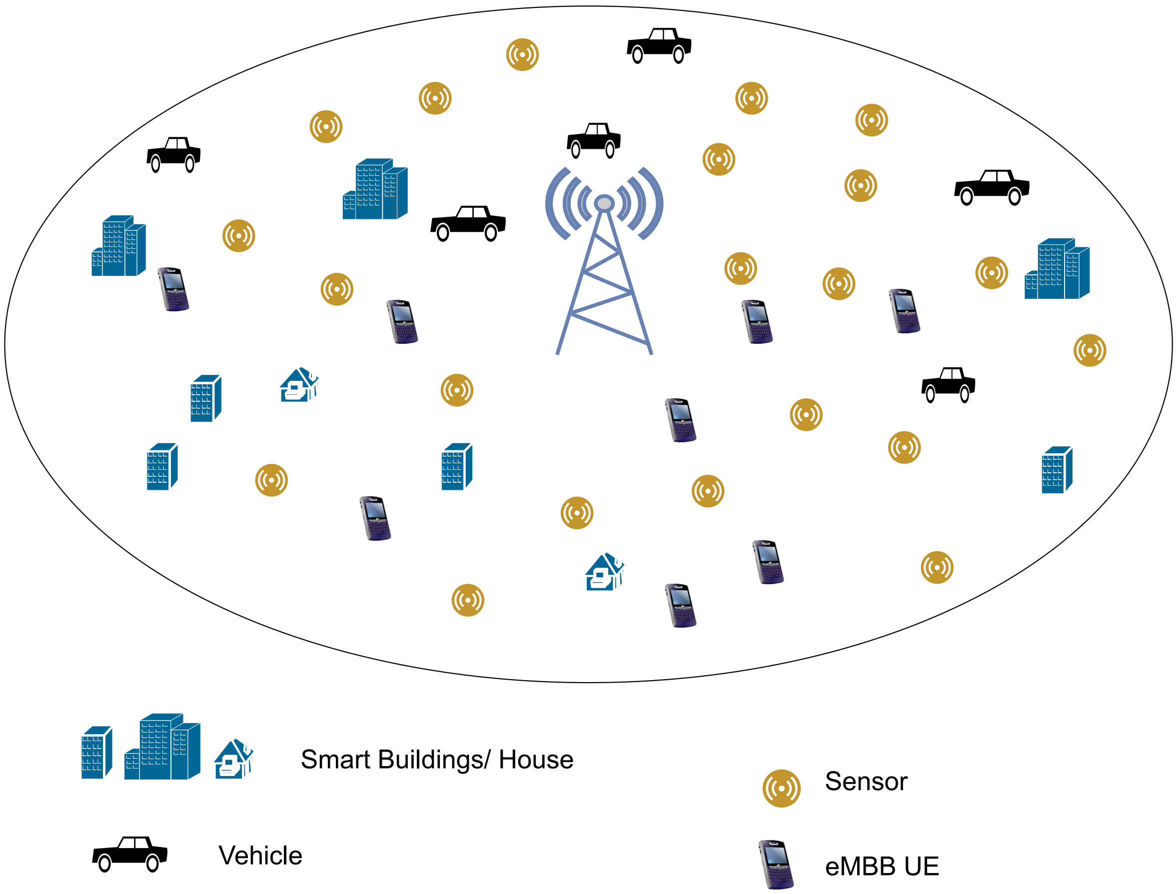


FIGURE 1.1: A General Overview on 5G system Traffic Types

gNB must have a resource in which the URLLC user can send all the time. Assigning a separate resource to each user the whole time will lead to wasting the network resources. The proposed solution is to make all the URLLC users share a common pool of resources. The main problem in GF scheduling is collisions due to competition between the shared resources.

There are many well-established scheduling models and techniques for the eMBB traffic. In contrast, the URLLC nodes are fairly new and their requirements are different. Precise modeling will aid us to solve the network optimization problem that involve this kind of traffic.

1.2 URLLC Traffic Nature and Applications

The main reason for adopting the URLLC in communication systems is the Internet of Things (IoT) critical applications requirements for connecting several communicating nodes with high reliability and low latency. It is predicted that in the near future the number of URLLC devices will surpass the normal data traffic [2]. The main reason is the high applicability of such traffic in various scenarios such as in health care, automation, control, intelligent transportation systems and natural disasters predictions [3]. GF scheduling is considered the main scheduling technique that has the potential to provide such requirements. There are three main types of GF scheduling and each has a different model; reactive, k-repetitions, and proactive [4]. Each of these type is suitable for and has its merits in different scenarios. As mentioned in the previous section, the latency requirements of the URLLC traffic is the main reason to adopt the GF scheduling. The main reason for choosing the suitable model of the mentioned three models is the reliability requirements since allocating the same resources to more than one node introduces collision and affects the system performance. There is a wide range of the latency of the URLLC traffic, as shown in Table 1.1. Adoption for the URLLC traffic in the domain of 5G systems is mainly due to the variety of applications from fire alarms, extreme phenomena alarms, e-health to vehicle-to-vehicle communications and remote surgeries. All the mentioned services require very low latency and high reliability quality of service. URLLC is considered the IoT enabler for next-generation communications systems [5]. The main problem in

TABLE 1.1: Applications and Requirements of the URLLC Traffic

Application	Latency	Reliability
Vehicle-to-Vehicle	1 ms	99.9999%
Alarms and Geo-sensors	$1 \sim 5$ ms	99.9999%
Factories of Future	10 ms	99.9999%
Automation and Control	$10 \sim 20$ ms	99.99%

the URLLC traffic is the absence of a mathematical model that fully grasps its nature and requirements. Therefore, the first goal in our thesis is to build a profound mathematical model that mimics its behavior and different interactions in order to build a complete system model.

1.3 eMBB Traffic Nature

The eMBB traffic is considered the main traffic type of cellular communications. Requirements of the eMBB users have evolved since the deployment of the first communication system. The requirement has evolved, from human-to-human communications to data streaming and multimedia traffic. As shown in Figure 1.2, the rate requirements for wireless traffic have increased within the past few years [6]. In addition, it is anticipated that the data traffic size will represent 71% of the total traffic by 2022 [7]. The mobile data traffic has increased seven folds from 2016 to 2021 and the video traffic has increased three folds during the same period. In fact, it is anticipated that wireless traffic will surpass the wired traffic by 2030. The eMBB traffic is characterized by rate greedy requirements. The main usage of this type of traffic is human-to-human communications, web surfing, streaming, and various other rate greedy applications. In contrast with the URLLC traffic, eMBB users can tolerate delays while the low rate is intolerable. The main quality of service requirements of such traffic are the minimum rate guarantee, availability and coverage. These requirements require the usage of a certain type of schedulers. The Grant-Based (GB) scheduling approach is usually used with this type of traffic since it fulfills the rate requirements while the accompanied hand-shaking procedure does not affect the delay requirements.

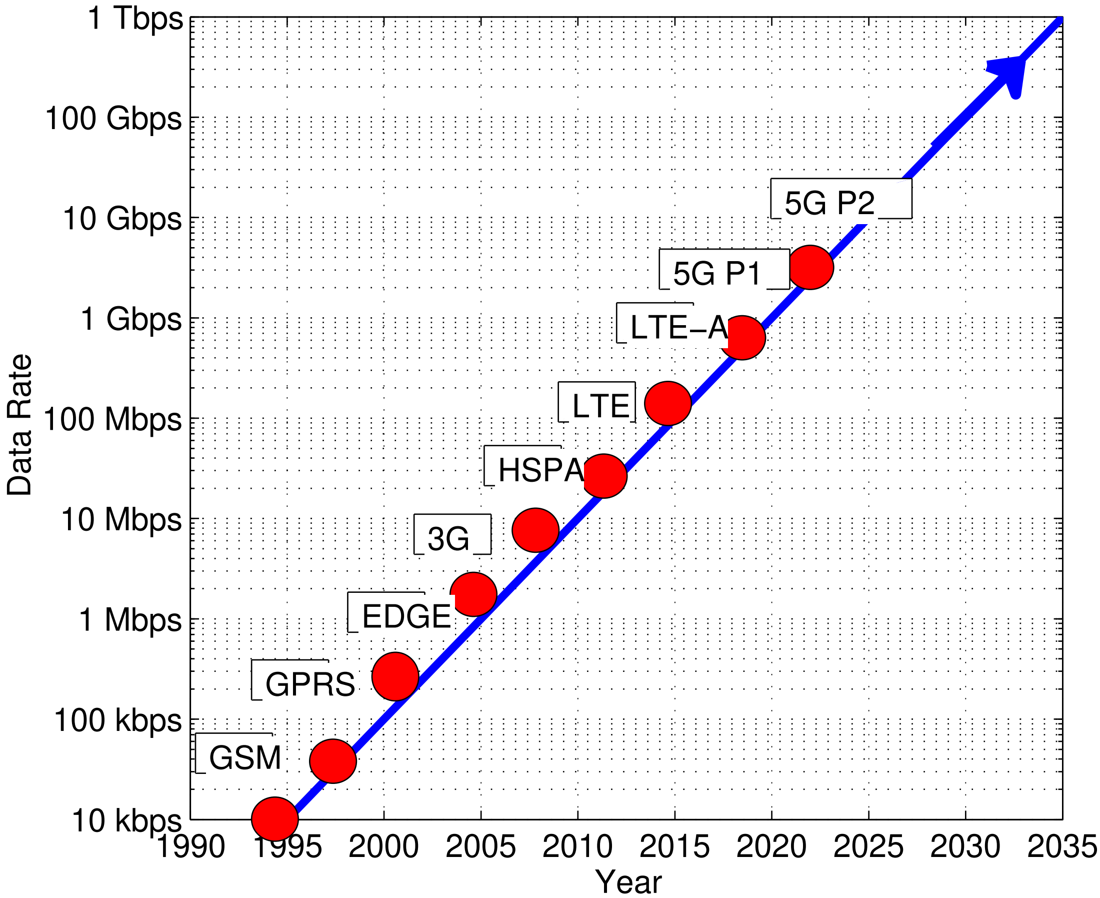


FIGURE 1.2: The Trend of eMBB Traffic Requirements [6]

1.4 The Need for Traffic Scheduling

As evident from the previous sections, traffic scheduling is a vital task in any communications network. Any communications system cannot function properly without a good scheduler. A scheduler provides, both the transmitters and receivers, with the required resources to establish and sustain a connection. Scheduling is therefore the task of allocating the resources of the system among the different users [8]. The main reason that scheduling is a hot research topic is due to the fact that both the design and functionality of the scheduler are left for the network operator, i.e. it is not part of the standard. A good scheduler must utilize all the system's resources optimally in order to provide the users with their required quality of service which results in the highest revenue for the network operators. In addition, providing a model for URLLC traffic will enable the scheduler to optimize its resources and the IoT devices to be accommodated perfectly within the 5G systems, and beyond, to fully utilize their potential.

1.5 Machine Learning Approach

Applications of Machine Learning (ML) in communication systems are wide and diverse. It is used in traffic prediction and classification, routing, scheduling, radio control, and many other topics [9]. One of the most promising fields that ML can aid communications engineers in, is the optimization of traffic scheduling. Classical optimization techniques are too complicated to use, if at all feasible, due to the variety and diverse types of traffic nowadays, as each type has different requirements. These diverse requirements when modeled correctly leads to an NP-hard optimization problem. Previously, engineers used to approximate the model and relax the parameters in order to generate an easier problem that can be solved by the classical tools. Most of the time, these simplifications lead to a model that does not fully cover the system requirements.

Reinforcement Learning (RL) and deep Neural Networks (NN) are the most promising solutions to solve network scheduling optimization problems. They can learn by themselves the optimum method, as closely as practical, of resource allocation to fulfill the requirements of both types of traffic.

1.6 Problem Statement

The uplink scheduling problem in the 5G communications system includes many parameters and requirements that need to be considered. The combination between the eMBB and URLLC presents a much complicated problem in the uplink direction than in the downlink direction since the gNB doesnot know when each URLLC user will need a Resource Block (RB) to send the packet. Allocating an RB to each URLLC user in a GB scheduling will violate their strict latency requirement. In addition, Allocating an RB to each URLLC user/node in a GF environment wastes the network resources due to their sporadic, probabilistic, nature and massive numbers. Another approach is used where the URLLC users share a common pool of resources and send their packets to gNB whenever they have data to transmit. This approach causes collisions among users, which need to be analyzed to avoid violating the reliability requirements. Three common HARQ techniques are used, namely, reactive, k-repetitions, and proactive schemes. The first two schemes are the ones adopted by 3GPP as a solution for the URLLC collision problem. In addition, the reactive scheme can be considered a special case of the k-repetitions scheme.

Our goal is to devise a formulation that guarantees reliability for the URLLC traffic in a single cell given a certain pool of resources and a repetition factor. This is an optimization problem that considers the overall system requirements.

An exact solution to this optimization problem provides an insight as to what is the optimal bound of the uplink scheduling problem. This can help determine the quality of non-optimal solutions which will have to be devised in larger environments since the optimal solution in such environments may not be feasibly possible to obtain. In addition, a comparison among different machine-learning (ML) techniques, with approximate solutions, will show the ability and the limitations of ML in solving the scheduling problem.

1.7 Thesis Contributions

The objective of this thesis is to build a robust uplink system model that covers the mixture of all the eMBB and URLLC traffic requirements of the system's users and uses the RL and NN techniques to solve the optimization problem and compare their results to find the optimal approach.

The main contributions of this thesis can therefore be summarized as follows

- We present a novel model of the URLLC probabilistic traffic in the uplink direction for a single cell in 5G systems. To the best of our knowledge, this is the first work done to model the URLLC traffic in a single cell in order to include the URLLC requirements in the scheduling problem.
- We formulate the uplink scheduling problem of both the eMBB users and the URLLC devices in the form of an optimization problem. We aim to maximize the overall eMBB rate while ensuring a minimum rate for each user. In addition, we aim to guarantee the latency and reliability requirements of the URLLC devices.
- We analyze different system parameters and their effect on URLLC traffic. This will help network designers to understand and develop better parameters to satisfy the traffic requirements.
- We propose a novel scheduling algorithm that provides near-optimal solutions in real-time.
- A reinforcement learning approach to solve the aforementioned scheduling problem in real-time with different policies is proposed and discussed. The policies are compared and discussed. The high dimensionality problem in large systems is also discussed.
- A neural network model is designed to fit our problem and utilized to solve the scheduling problem in different types of environments.

- We Compare different scheduling techniques for different scenarios in order to understand their advantages and disadvantages. The results are simulated for different environments and dimensions and their results are discussed.

1.8 Thesis Structure

The rest of this thesis is organized as follows

- Chapter 2 presents a literature review on recent advances in the traffic scheduling problem in 5G, both in downlink and uplink directions, in addition to the recent techniques to solve the scheduling problem using Machine Learning techniques.
- Chapter 3 discusses the URLLC traffic modeling in the uplink direction. An analysis is conducted to model the probabilistic nature of the URLLC traffic in the uplink direction. Finally, the complete mixed traffic scheduling problem is formulated. Insights on how to simplify the problem and different approaches are discussed. In addition, an algorithm is proposed to solve the same problem in a more efficient manner with low complexity requirements. Different simulation results are plotted to understand the interactions between different system parameters with the URLLC traffic requirements. In addition, various operating environments are simulated to show the applicability of our scheduling technique compared to the techniques in the literature.
- In Chapter 4, different Machine learning (ML) techniques are discussed. First, the reinforcement learning approach is discussed and analyzed. The scheduling problem is formulated in the reinforcement learning domain, where the reward function, state-space, and action space are defined. The dimensionality problem is discussed and analyzed. Different environments are simulated for different learning policies to show their applicability in our scheduling problem. Second, the Neural Network (NN) approach is presented and the problem transformation is discussed to fit the model. The model is tuned to produce the best results and simulations are done to validate our selection. In addition, different environments are simulated to show the applicability of the technique. All the results are compared to the well-known algorithms in literature.

- In chapter 5, we provide the conclusions of this thesis and propose some directions for future research.

Chapter 2

Background and Related Studies

2.1 Introduction

In this chapter, we present and discuss the background as well as the previous work in the area of resource allocation. Classification of traffic types is presented and their requirements are discussed in detail. The traffic scheduling process is also discussed to fully understand the scheduling problem especially in the uplink direction. We discuss the different approaches in the literature to solve the scheduling problem in both the downlink and uplink. The downlink problem is discussed as it provides many insights on the uplink problem.

2.2 Classification of Traffic Types

There are three main types of traffic in the 5G systems, namely, the eMBB traffic, the URLLC traffic and the mMTC traffic. The eMBB traffic is characterized by its high data rate requirements and delay tolerance. It is considered as the human-centric data traffic for various multimedia services. In contrast, URLLC traffic is extremely delay intolerant. However, its rate requirements are minimal due to its small payload. On the other hand, the mMTC are a variant of the URLLC traffic with moderate delay requirements [10].

There are three types of scheduling techniques for the uplink traffic adopted in literature, namely, the Grant-Based (GB) scheduling [11], the Grant-Free (GF) scheduling [12] and the Semi-Persistent Scheduling (SPS) [13]. The GB scheduling uses a hand-shaking procedure between the transmitter and gNB. The transmitter receives a packet from the upper layer, then sends to the gNB requesting for resources. The gNB replies with the timing and location to send its packets on. The handshaking procedure is suitable for applications with delay tolerance since the hand-shaking requires about 10 ms [4]. On the other hand, SPS is suitable for the periodic traffic where the gNB reserves some of the network resources every time period for a specific device or user. Finally, the GF scheduling does not require a hand-shaking procedure where the node sends directly its packets in an arrive-and-go manner without waiting for the gNB to provide resources. These resources can be assigned to more than one user or device. This type of scheduling is suitable for the sporadic traffic since the gNB does not know exactly when it will receive a packet from the device, which is the main reason to schedule more than one user on the same resources. The main drawback of the GF traffic is collision. When the same nodes compete for the same resources, collisions occurs where the Signal-to-Interference plus Noise (SINR) ratio decreases below a certain threshold rendering the decoder unable to decode the packet thus leading to a transmission failure.

For the GF scheduling there are three Hybrid Automatic Repeat reQuest (HARQ) schemes used to avoid and mitigate the collision problem, namely, the reactive scheme, the k-repetitions scheme and the proactive scheme [4]. In the reactive scheme, the node sends a packet whenever it arrives and waits for an ACK/NACK packet from the gNB. If an ACK is received, the packet is dropped, else the packet is resent if its latency requirement is not violated. In the k-repetitions scheme, the node sends k replicas of the packet to the gNB and waits for the feedback to respond accordingly as shown in Figure 2.1. It is considered that the reactive scheme is a special case of the k-repetitions scheme. Finally, in the proactive scheme, shown in Figure 2.2, the device sends the packet along with the replicas and at the same time senses if there is a response from the gNB. The first two techniques are accepted by the 3GPP [14]. However, the proactive scheme is not accepted due to the high processing requirements imposed by the algorithm to transmit and receive at the same time which might not be available to the URLLC nodes.

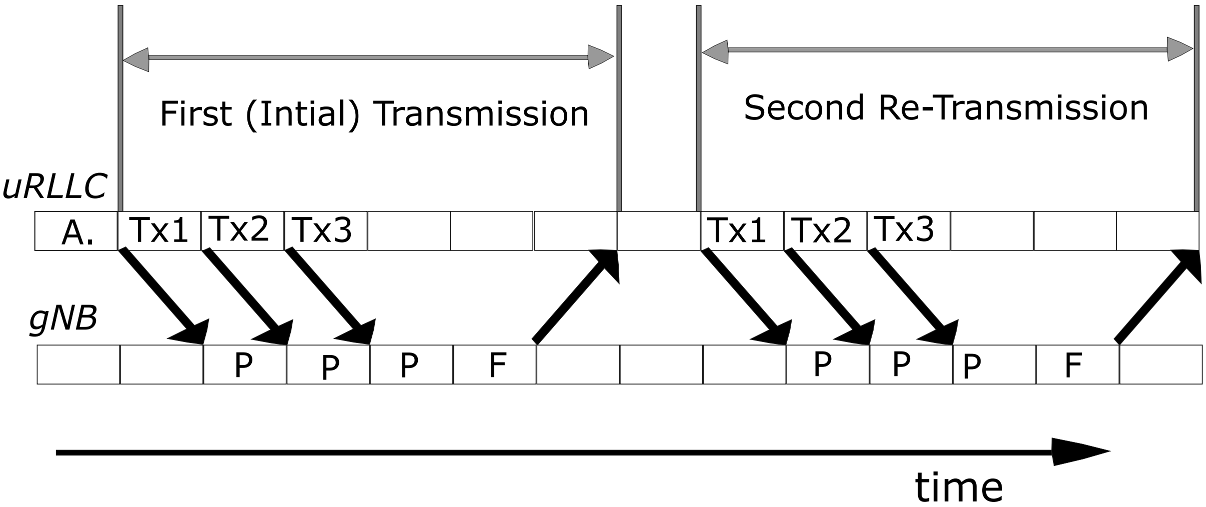


FIGURE 2.1: k-repetitions scheme with 3 Repetitions

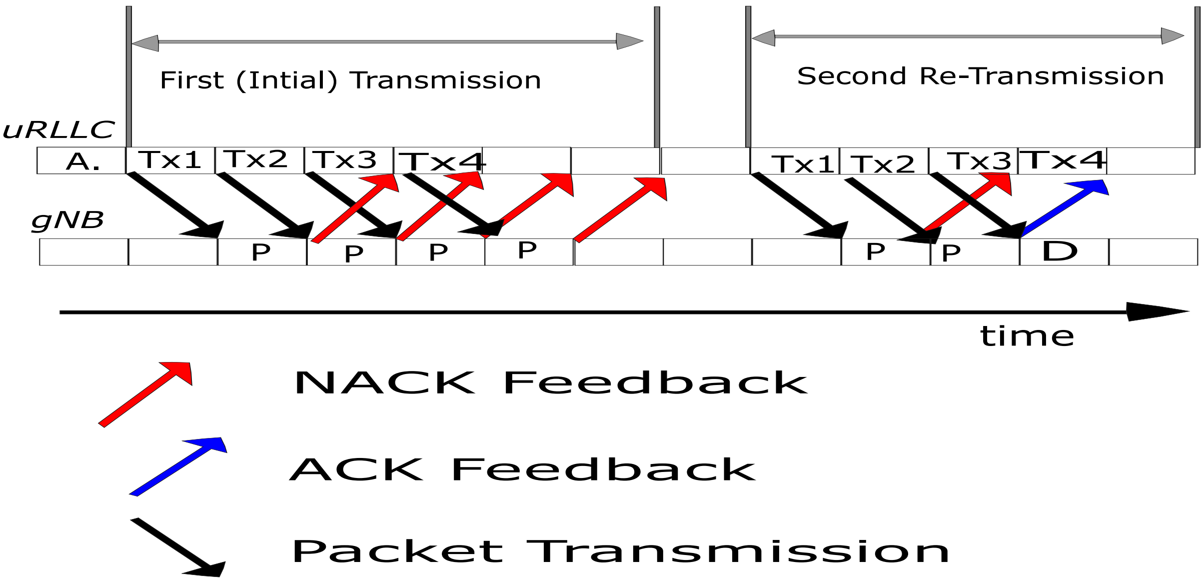


FIGURE 2.2: Proactive Scheme with 4 Repetitions

2.3 5G Background

Over the years, newly introduced communication systems provide a better upgrade in performance and applications over the previous or existing systems. Although 5G is considered an upgrade to the 4G LTE and LTE-A, it is considered as a revolution over all the previous communications systems. It is highly expected that 5G will present the main building block for the future of communication systems [15]. It provides a high boost in data rates as compared to 4G and past communications systems. It enables the use of other applications that were considered impossible in past communications system, or at least inefficient, as shown in Figure 2.3. In addition, it is considered the enabler of the IOT technology by merging the URLLC and mMTC in the system [16]. The objective of 5G is to digitize every aspect of human lives to provide comfort to all users in their daily life. It is predicted that 5G will support up to 7 trillion devices or nodes with high availability and reliability provisions [17]. To that extent, the 5G system introduced the idea of the short Transmission Time Interval (sTTI) in the system by chunking each Transmission Time Interval (TTI) into smaller blocks. The introduction of sTTIs allows the introduction of the URLLC traffic in the system without resorting to underutilizing system resources. The URLLC node can send the packet in a single sTTI, due to its small payload, 32 Bytes, without needing to reserve the whole TTI. This allows other nodes to use the same resource on the same TTI without compromising the other payload or interference. It is considered a merit as well in the downlink communications since it allows the concept of puncturing of the eMBB traffic. The gNB punctures the eMBB traffic, through the use of an sTTI, with the URLLC traffic to be sent on the same resource block within an acceptable calculated reduction of the eMBB rate.

2.4 Traffic Scheduling Process

As discussed in the previous chapter, traffic scheduling is a vital step in communication networks. It is enabled by the network operator in order to satisfy a certain policy and revenue targets. To do so, the traffic scheduling should be designed to satisfy the QoS requirements

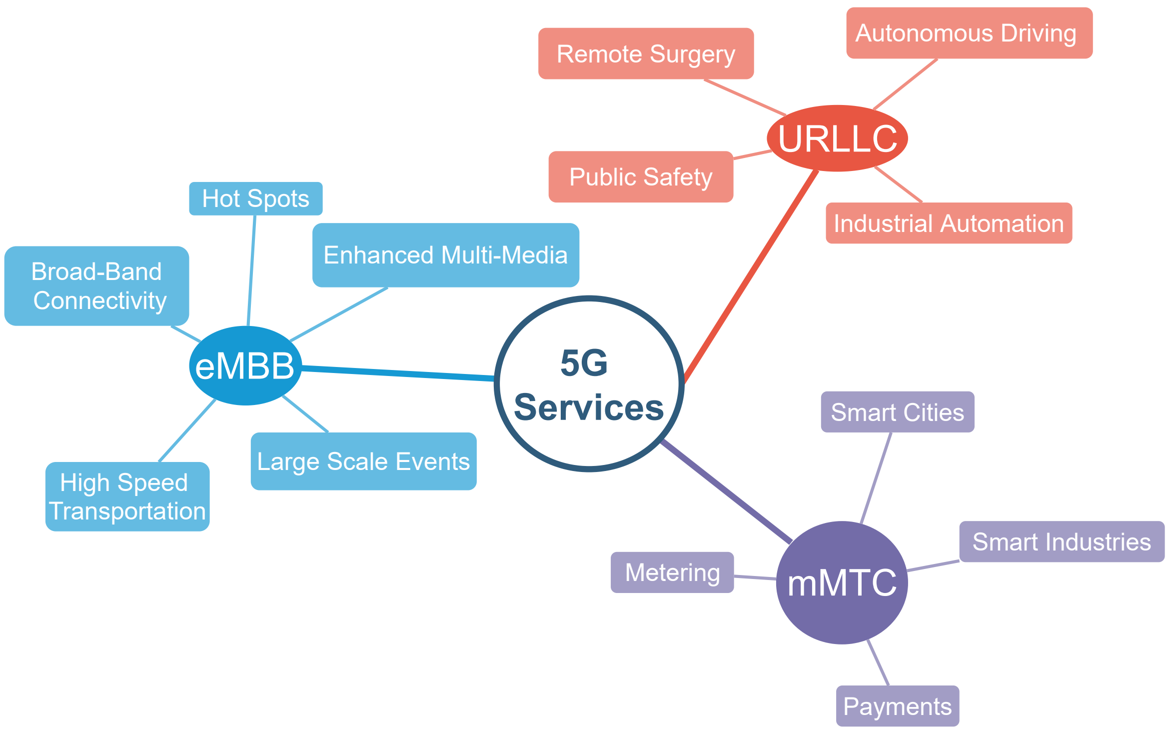


FIGURE 2.3: 5G Supported Traffic and Services

in addition to utilizing the system resources as optimally as practical. The traffic scheduling starts at every TTI for both the downlink and uplink. For the downlink, the gNB schedules the eMBB traffic on the time-frequency grid, resource grid, and notifies the receivers with their locations. Then it punctures the eMBB traffic with the URLLC packets in a way that minimizes the QoS loss for the eMBB traffic. On the other hand, in the uplink direction the gNB does not know when the URLLC traffic is received. However, it must broadcast the available resources for all URLLC nodes in case a packet is received. If more than one node sends at a time, the SINR decreases. If the SINR falls below a certain threshold, this signals the occurrence of collision. The decoder cannot decode the packet and it is considered as a failure. In addition, the gNB receives a scheduling request from the eMBB users before each TTI and tries to map it on the resource grid to satisfy the QoS requirements in addition to optimizing the system resources in order to avoid under-utilization.

As shown in Figure 2.4, at the start of each TTI, the gNB recieves all the traffic requests from all nodes. It receives requests from the eMBB users and the mMTC devices, in order to schedule the traffic optimally. On the other hand, it receives the data packets directly from the URLLC traffic on the allocated RBs in the previous TTI. The gNB must decide the best possible allocation for the eMBB users to provide their QoS requirements and optimize the resources. It allocates the available resources to the URLLC traffic and broadcasts them to all the devices based on their number without the knowledge of their channel. The number of URLLC nodes is usually large and the massive overhead will impact the system performance since the scheduling task must be established each 1 msec. The gNB must provide enough resources to satisfy their reliability and latency requirements based only on the number and path loss knowledge. If there is any remaining resources, they are left to the mMTC traffic. The allocation at each TTI must be of the same shape as seen in Figure 2.5.

In our view, a good scheduler must maximize the accumulated rate of the eMBB users as well as providing a minimum rate guarantee to each eMBB user to avoid starvation. In addition, it should provide enough resources to the URLLC traffic to satisfy their latency and reliability requirements.

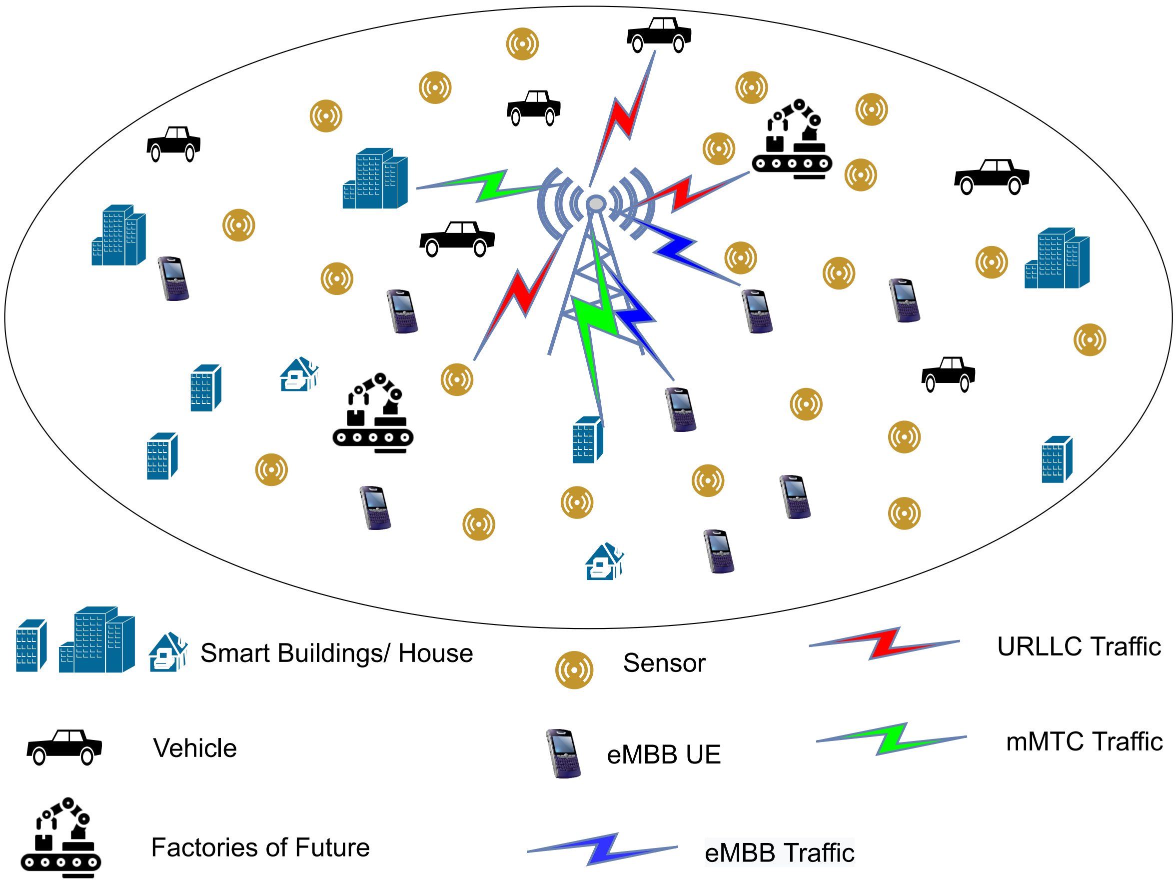


FIGURE 2.4: A Sample of Traffic Requests to the gNB in the Uplink Direction

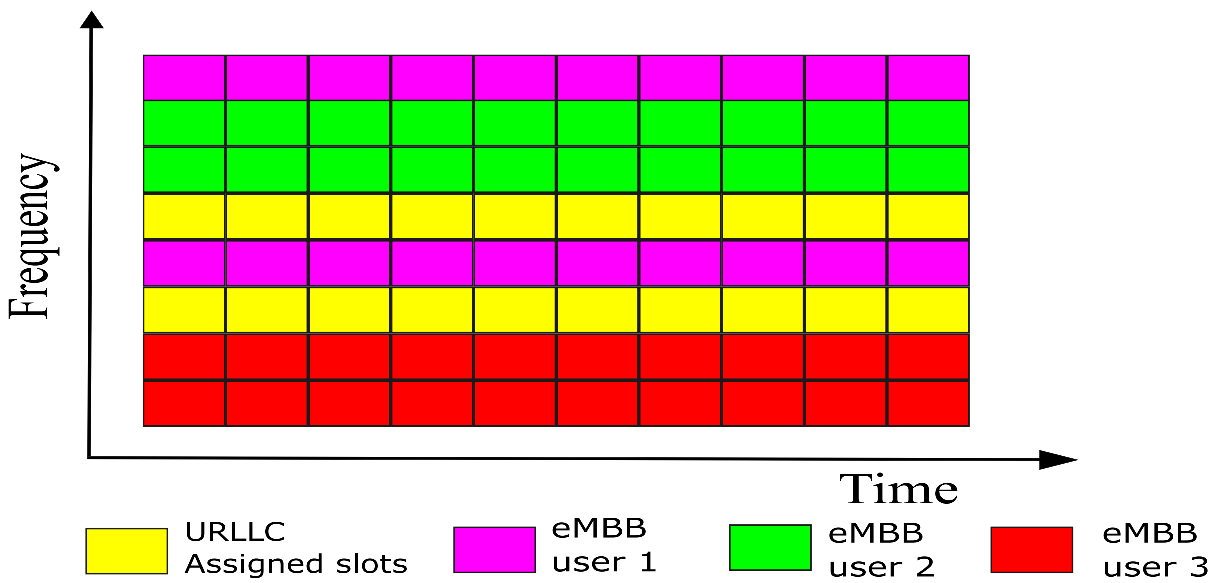


FIGURE 2.5: A Sample of the Scheduling Process

2.5 Related Studies

The scheduling process is divided into two parts, namely, uplink and downlink scheduling. The uplink scheduling is responsible for giving access to different types of devices to send their data depending on the network's resources. These devices can be eMBB, URLLC, or massive Machine Type Communications (mMTC) devices. The eMBB uses Grant Based (GB) scheduling, while Grant Free (GF) is most suitable for URLLC traffic. The mMTC is given the remaining network resources using the GB scheduling as the latency requirements of such devices can withstand the handshaking procedure, if any exists. In the downlink direction, the gNB re-transmits the packets to the intended receivers. This is done by scheduling the eMBB traffic on the resource grid. The gNB then punctures it with the URLLC traffic. In [18], the expected requirements for 5G systems are discussed in detail to demonstrate how the demands of Machine-to-Machine (M2M) communications are accommodated.

In addition, machine learning approaches are used in several optimization problems to produce near-optimal results in real-time. In the following sections, different techniques in literature used for solving scheduling problems are discussed. The adopted approaches for the network optimization problems are the reinforcement learning approach and the neural network approach. In addition, similar scheduling problems with the combinatorial nature are discussed to provide insights for solving our problem.

2.5.1 Downlink Scheduling

In the downlink, [19] differentiates between 2 types of M2M communication. The first is mMTC which is the low-cost and energy device, considered the enabler of the Internet of Things (IOT). This type of communication does not have delay constraints. The latter is the URLLC, where the communication is required to be of low latency and the required reliability is high. The authors split the requirements of URLLC into 3 main dimensions according to the previously discussed issues, namely, reliability to guarantee packets transmissions within the latency demands, latency which is the end-to-end time required to send the message from the sender to the intended receiver correctly and availability. The paper discusses 3 ways to achieve the aforementioned demands, first to provide latency requirements, the Transmission Time Interval (TTI) should be minimized compared to that of OFDM systems. Redesigning the physical channel for fast channel estimation and processing. High order antennas' diversity to combat fading and channel uncertainties is used. Different ways of scheduling using optimization techniques and near-optimal algorithms. Actually, this paper tackles various important problems concerning the scheduling in 5G system. It aims to maximize the rates of eMBB as much as possible for assuring the quality of service for users while maintaining the URLLC packets delay and reliability requirements. At the edge of each slot, the gNB divides the resources in the downlink phase among eMBB users. After that, the incoming URLLC traffic is punctured/superimposed over the eMBB traffic. Each URLLC traffic takes only one mini-slot to prevent any latency. A joint eMBB/URLLC optimized scheduling is required to consider the puncturing effects on the eMBB users rate. In this paper, the authors proposed 3 eMBB rate loss models for URLLC puncturing, namely linear, convex, and threshold rate loss models. Each of the mentioned models has a placement, puncturing, policy for the URLLC packets in the eMBB traffic. The joint scheduler for the eMBB and URLLC traffic is compared to the separated traffic scheduler for each policy. The results shows that for the linear model the joint scheduling can be decomposed, for faster processing, to separate schedulers for each traffic type. In contrast, for both the convex and threshold models, the scheduling cannot be separated.

In [20], a downlink scheduling technique is proposed to protect the eMBB users with low data rate while ensuring the URLLC latency requirements. They proposed adding an extra factor called the Conditional Value at Risk (CVaR). Where the α-VaR is the α percentile of the distribution of the random variable, i.e. the smallest value such that the probability that a random variable is smaller or equals to this value is greater than or equal to α. By subtracting the CVaR from the objective function the optimization problem became a mixed integer non linear programming problem. This problem of exponential complexity order which requires searching the state space. So the problem is decomposed into two sub-problems. First, the eMBB users scheduling and, second, the URLLC placement. First, an optimal allocation for eMBB users is derived without considering the URLLC placement and in the second step the placement is considered with regard to the CVaR. The two sub-problems are of a convex optimization problems type so they can be solved using any convex optimization tool. The study in [21] discusses the modeling of the URLLC downlink traffic as a queueing system. The arrival process of the URLLC packets is modeled as a Poisson arrival process with an arrival rate that they wish to maximize. However, the co-existence of eMBB traffic is not discussed. The goal in [21] is to determine how much traffic the system can support and how to increase the arrival rate as much as possible, given some system parameters and arrival rates of the URLLC services. Two different approaches are discussed in the study. The first is the one-shot approach where the gNB sends the URLLC packets once and drop them from the system. The other approach is to keep the packet and wait for feedback from the receiver to check if the packet is received and decoded correctly for a specific number of trials. In both cases, the system was divided to a set of classes with the same SINR. For the first case, the methods of allocation are discussed in the time-frequency domain, namely, the tall allocation where the packet takes a small fraction of time while taking a large bandwidth (BW) and the wide allocation where it takes a large amount of time with low BW. It was shown that the wide allocation is better since it satisfies the delay constraints. This technique decreases the probability of blocking of the gNB to URLLC packets. The result reached after formulating the optimization problem is that the maximum optimal arrival rate of each class is a concave function with the SINR of each class. Also, for a large BW, the optimal arrival rate increases linearly with it.

In [22], an energy consumption model is built for the DownLink (DL) transmission and allocation process. The goal is to minimize the expected energy consumption in DL transmissions for the 5G system packets for all kinds of traffic, eMBB, mMTC and cMTC. In addition, the Chase Combining Hybrid Automaitc Repeat reQuest (CC-HARQ) is applied, which is a variation of HARQ, with a threshold on retransmissions to compare it with frequency diversity transmissions energy usage since CC-HARQ is considered of the time diversity type. The outage probability measure is used, in order to measure the violation of URLLC delay requirements. The Nakagami m-channel model is adopted as their channel model. The optimization problem is formulated with objective of minimizing the expected energy for transmissions, subjected to outage probability threshold, maximum power threshold and minimum rate to not violate the latency requirement. Since the optimization problem cannot be solved analytically, an alternative problem is proposed to find the optimum number of retransmissions to satisfy the latency and power constraints. A protocol is designed to fit the URLLC requirements based on the alternative problem. The results show that this protocol provides high energy savings compared with a benchmark obtained numerically by optimizing the average signal-to-noise ratio guaranteeing an outage probability. Also, it was shown that the CC-HARQ outperforms the frequency diversity to provide better diversity with energy saving.

A joint scheduler is proposed in [23] for assigning the network resources for eMBB and URLLC DL traffic in 5G systems. The model is based on maximizing the overall spectral efficiency of the system while maintaining the URLLC requirements and providing a minimum throughput for each eMBB user. The network, in their model, is based on separate slices for eMBB and URLLC users and the scheduling is done every TTI. The optimization problem was a mixed-integer non-linear programming optimization problem. Therefore, a relaxed version is proposed that is based on [24] which is a convex optimization problem. The problem was solved using the Powell–Hestenes–Rockafellar (PHR) augmented Lagrangian algorithm to find the optimal subcarrier and power allocation which is then fed to the Joint carrier Power and Subcarrier Allocation (JPSA) algorithm to obtain the optimal resource allocation and spectral efficiency. The results are compared to Equal Power Allocation (EPA), Equal Subcarrier Allocation (ESA) and Adaptive Particle Swarm Allocation (APSA) algorithms. The results show the superiority of their allocation algorithm in spectral efficiency for the whole traffic, reliability for URLLC traffic and maintaining spectral efficiency for different eMBB rate demands.

The DL resource allocation problem in [25] is solved by dividing it into two sub-problems. The goal of the optimization problem is to maximize the minimum expected eMBB actual data rate after puncturing URLLC traffic over different scheduling techniques. A model is provided for the actual rate of the eMBB users after puncturing and is set as an optimization objective function with the URLLC requirements as constraints. Due to the fact that the main problem is an NP-hard and requires a lot of computational time, they sub-divide the problem into 2 parts. The first part is the scheduling of the eMBB users before puncturing and the objective is to maximize the minimum rate before puncturing, this also requires a high computational power that can not satisfy the time required for scheduling. A heuristic algorithm is used, where the algorithm allocates the resources based on the eMBB users requirements in this slot and the allocations that occurred in the previous slots, so in a way, it can know the actual rates given to the eMBB users after puncturing in the previous time slots. The second step is puncturing for URLLC in the scheduled eMBB slots. The authors assume that the gNB knows the CSI for each eMBB and URLLC device and used a matching game in order to allocate the URLLC packets to their preferred channels after arranging their preferences with taking in consideration not to violate both the URLLC and eMBB requirements. Their techniques show better results for the Minimum Expected Achieved Rate (MEAR) and the fairness among users is achieved better than legacy algorithms.

In [26], the main optimization problem is to maximize the MEAR of eMBB while having the requirements of URLLC as constraints. The authors divide the problem into two sub-problems with the same objective. The first objective is assigning the resources to eMBB users and the second objective is to puncture them using the URLLC packets. For the first sub-problem, the Penalty Successive Upper bound Minimization (PSUM) is used. For the second problem, the Transportation Model (TM) is used and defined as a cost function for puncturing the eMBB users and the objective was to minimize that cost. Due to the high computational requirement of the first algorithm used in the first sub-problem they use a heuristic algorithm in which, in the first iteration, it assigns all the eMBB devices the same resources while in the subsequent iterations it assigns the RB according to the punctured eMBB in the previous slots giving the punctured eMBB more resources. The optimization function in [27] is based on maximizing the eMBB rate after puncturing while satisfying the latency and reliability constraints of the URLLC traffic, in addition to granting a minimum rate for eMBB UEs. Before puncturing, the eMBB traffic is optimized to achieve the maximum rate while fulfilling a minimum rate for each user. In the puncturing process, two techniques are discussed, immediate scheduling for URLLC packets upon arrival and waiting in a queue while served in another mini-slot, without breaking the latency constraint, in order to achieve better eMBB performance. To solve the problem in real-time, a matching system is proposed where each eMBB and URLLC users arrange their preferred channels to send in descending order and the choice is based on fulfilling the constraints.

In [28], Reinforcement Learning (RL) techniques are used to solve the joint resource allocation scheduling problem for eMBB and URLLC demands of high speed vehicular devices in the DL. A Deep Neural Network (DNN) model is designed that can predict the eMBB user channel state information (CSI) using the CSI of the URLLC. The paper considers the environment of the moving vehicle that has URLLC and eMBB demands. The vehicle motion changes the channel conditions so as to prevent additional CSI information overhead. The procedure is as follows. The CSI acquired by the gNB when transmitting a URLLC packet to the vehicle will be used to infer the CSI of the same vehicle when requiring an eMBB packet. It proposes a heuristic resource allocation policy as the gNB satisfies the URLLC demands first and the remaining resources are given to satisfy the eMBB requirements. The non linear relationship between CSI for geographically separated areas is investigated. A DNN that can infer such a relationship and enable the gNB to design a beam-former that provide spatial diversity, is proposed. The DNN is trained using the beam-forming loss function offline, then this inferred CSI is passed for the DL allocation algorithm. The results of the proposed DNN compared to the ideal case where the actual CSI is known are given for two scenarios, namely, the sparse scenario, where the gNB resources are more than the network requirement and the dense scenario, where the network resources is less than the network requirement. Another approach that uses RL is presented in [29], it proposes two ways of dynamic allocation for the DL phase in a URLLC packets system. It proposes to solve the allocation problem using the Markov Decision Process (MDP). The performance metrics they propose is to maximize the proposed reward function which is based on the number of frequency slots given to each user in each time slot, while minimizing the risk function, which is related to entering an absorbing state. An absorbing state is the state in which the process might reach a policy that will violate the threshold requirements for URLLC traffic even for one user. The first approach assumes that all the channel statistics are known to the controller (gNB). A finite horizon value iteration algorithm is used. In the second approach, the CSI and channel parameters are not known, which is more realistic because the delay requirements of URLLC packets will not allow the controller to send pilot signal for channel estimation. They applied a reinforcement learning algorithm, namely the Q-learning algorithm. The learning function is also related to the reward and risk functions described in the first approach. However, in the learning process if a risk state is entered, which might happen in the case of learning procedure, the learning stops and the system is restarted with a non-risk state.

In [30], optimization problem is devised for DL resource allocation for both URLLC and eMBB packets. It aims to maximize the average throughput of the eMBB users while decreasing their variability. Decreasing the variability is advantageous for users that have bad channel conditions, where if the variability in the optimization problem is disregarded the algorithm will puncture the eMBB users with bad channel conditions with the URLLC packets and keep the users with good channel conditions. This way the users with bad channel conditions will have low rate compared to other eMBB users. The optimization goal is to find three main vectors, the optimal RB allocation for eMBB users, the optimal power allocation for eMBB users and the optimal number of punctured mini-slots in each time slot. The optimization problem constraints are as follows. Ensuring a minimum reliability for URLLC packets, which is formulated as an outage probability and decreasing the variance in rate among the eMBB users. This optimization problem is a Mixed Integer Non-Linear Programming (MINLP) and an NP-hard problem. A relaxation is done to the optimization problem in which all the integers are turned into a real valued continuous variables and the complexity caused by the variance parameter in the objective function is smoothed using an exponential function. The main optimization problem is further sub-divided into three optimization problems, allocating the eMBB users, optimal power allocation for eMBB users and URLLC packets allocation. The first two problems are convex optimization problems and can be solved with any convex optimization tool. However, the third one is a combinatorial optimization problem which is NP-hard, and hence, further relaxation is required for the third optimization sub-problem to a convex optimization problem. They devised a Decomposition Relaxation based Resource Allocation (DRRA) algorithm to solve the three optimization problems. Due to the relaxations done in the third subproblem, the URLLC constraint might be violated in some cases. So the authors devised a Deep Reinforcement Learning (DRL) algorithm to find the numbers of punctured mini-slots in each RB, aided with the DRAA results for the first 2 optimization sub-problems. The Policy Gradient actor-critic Learning (PGACL) algorithm is used for solving the URLLC scheduling problem instead of Q-learning algorithm due to the slow learning of the latter where it takes a lot of time till convergence. It is shown that using the DRRA in the DRL algorithm fastens the convergence of the learning procedure compared to random initial learning. The Jain's fairness index is used for comparison purposes and shown that their algorithm outperforms the sum rate algorithm in the fairness domain however the sum rate gives better average rate. The algorithm proposed outperforms the sum-rate and sum-log algorithms in terms of reliability for eMBB users for different URLLC loads and has a less rate variability for eMBB users although it has the least average rate.

In [31], deep RL for the downlink scheduling is studied. The study did not define a model for URLLC traffic and left the algorithm to train by itself. The optimization problem goal is to minimize power and to take the reliability and latency constraints of URLLC traffic into account, as well as the eMBB rate. The approach was to train the agent on a virtual environment first before deployment in order to gain experience before dealing with real traffic. The problem of the training data scarcity was solved using GANs where using samples of real data. It mimics their behavior and generate as much data as required for training, in addition to generating extreme conditions which happens rarely in the network and needs the scheduler to act upon.

2.5.2 Uplink Scheduling

As explained in previuos sections, the uplink problem is more complex, as the URLLC traffic model is probabilistic in nature since the gNB does not exactly know when these devices send their packets. In addition, the URLLC traffic requirements differ from the eMBB and hence a probabilistic traffic model should be established to fully understand and formulate the problem. [32] and [33] discuss the theoretical framework of the existence of different traffics in the uplink channel and the GF URLLC scheduling. In [32], the uplink multiple access techniques are discussed. Their model is composed of 3 different users; eMBB, URLLC and mMTC. The co-existence of these services in the same Radio Access Network (RAN) is therefore discussed [32]. The objective of the mMTC is not latency or throughput but the goal is to maximize the ability to support a high arrival rate for a given radio resource. There are two main uplink multiple access techniques, namely, orthogonal multiple access (OMA) and non-orthogonal multiple access (NOMA). To simplify the study of the requirements of the heterogeneous service, the paper discusses the co-existence of eMBB and URLLC together on the one hand and the mMTC with eMBB on the other hand, as separate cases. In the first case, a Successive Interference Cancellation (SIC) decoder is used at the gNB where the URLLC is decoded first because the reliability requirement of the URLLC is higher than that of the eMBB. Hybrid NOMA (H-NOMA) appears to be more efficient than Hybrid OMA (H-OMA) in regimes of high rates for eMBB and vice versa. In the second case, the SIC decoder is used as well, the eMBB packets are decoded after one or multiple of mMTC packets received on the same Resource Block (RB), according to the rate of eMBB users. The H-NOMA outperforms the H-OMA in the same regime, moderate rates.

The study in [33] discusses the uplink scheduling for both the URLLC and eMBB users in both the Orthogonal Multiple Access (OMA) and the Non-Orthogonal Multiple Access (NOMA) schemes from an information-theoretic point of view. The model was based on a Cloud RAN (C-RAN) with a BaseBand unit (BBU) connected to the Edge Nodes (ENs). The model for OMA was based on dividing the time slots for URLLC and eMBB users, which is not efficient due to the sporadic nature of the URLLC traffic as well as the low utilization of network resources. On the other hand, in the NOMA technique, the GB scheduling is used for eMBB traffic while the GF scheduling is used for URLLC traffic. Therefore, puncturing in the eMBB traffic occurs. This hybrid technique is efficient if it can be mitigated in EN due to the delay constraint of the URLLC traffic. In NOMA, it is proposed that the C-RAN receives the eMBB data through the front haul connection which will be responsible for interference mitigation for eMBB traffic, while the URLLC traffic decoding will be the ENs responsibility. In NOMA, there are two approaches to deal with URLLC traffic while decoding in ENs. The first approach discards the punctured slots and treats them as erasures so the quantized versions of the data sent in the front-haul are of high resolution. The other approach deals with them as noise and forward them to the C-RAN. The first approach can be applied using a SIC decoder which shows significant results compared to the second one.

2.5.2.1 GF Scheduling

In [4], the authors discuss the common types of the GF HARQ access schemes in the 5G New Radio (NR) to mitigate the latency and reliability requirements of the URLLC packets. The schemes are reactive, proactive and the K-repetitions schemes. Each of them is modeled mathematically in order to show which of them is more suited in each scenario. The main difference between the GF and GB schemes is the collision between the packets due to the free grant given to all URLLC users. It is shown that the k-repetitions scheme is better in avoiding collision between URLLC devices especially if the duplicates are sent with different pilots or on different RB. However, it takes more time compared to the previous scheme, and hence, some measures need to be taken in consideration while applying this scheme, e.g. the maximum number of allowed repetitions. The proactive scheme provides the benefits of both the previous schemes. This technique is more complex than the aforementioned techniques. The main problem in GF schemes that should be considered is the collision probability. The interference in the GF is composed of intra-cell and inter-cell interference. The results show that the proactive scheme provides the lowest latent access failure probability than the reactive scheme in the high SINR threshold scenario. For lower SINR threshold and high density URLLC devices K-repetitions provides lower access failure probability.

In [34], two ways of GF access in 5G are discussed; separate band for URLLC transmission and overlaying the eMBB and URLLC traffic. The outage probability for the two modes of operation is calculated analytically and the achievable rates are plotted for both cases. It is shown that the SIC decoder provides a good performance in the overlaying mode in low or high SNR scenarios. In [35], the uplink Random Access (RA) procedures in 5G communications are discussed. It discusses the URLLC traffic requirements for the industrial domain, Factories of the Future (FoF). The main requirements for URLLC traffic in FoF are the 10 ms latency and 99.99% reliability. The standard RACH procedure is reviewed and compared with the 3GPP Early Data Transmission (EDT) RACH procedure discussed in 3GPP release 15 [14]. In the standard RACH procedure the message is considered a failure if the gNB detects a collision when two URLLC devices use the same preamble. The failed devices tries to reconnect after the Back-Off (BO) time. The 3GPP added 3 enhancements to the standard RACH procedure. Two of these enhancements are to decrease the access delay, EDT and flexible numerology. The last enhancement is to decrease the collision probability and the reserved preamble. Three types of enhancement to the EDT are discussed to decrease the probability of collision and access delay. For decreasing collision, the authors suggested using DC where the URLLC device sends two different preambles to two different gNBs and it connects with whichever responds first. By sending two different preambles the probability of collision will decrease. The enhanced back-off technique is used when the URLLC devices detect its failure where the authors suggest that it shouldnot wait for 20 ms as provided by 3GPP release 15 and sends directly while the non-URLLC devices send after a back-off equal to 10 ms. The third and final enhancement is the dynamically reserved preambles where the gNB dynamically changes its reserved preambles according to the number of failed connections and the numbers of established connections for both URLLC and non- URLLC users. The results show that their techniques combined together provide lower access delay, lower collision probability, and better preamble utilization for both mixed traffic and URLLC only traffic environments compared to the reserved preamble, EDT or both combined. They also state that the requirements can be met if the 5G numerology with 2 symbols/ slot with 15 KHz spacing.

In [36], the GF procedure is discussed in detail along with its requirements when sharing the resources to enable multi-users decoders. The authors also show that the performance can be enhanced by using frequency hopping for HARQ and advanced receivers. In [37], the power-boosting technique in GF uplink scheduling for URLLC is discussed to fulfill its requirements. The used GF transmission scheme is the reactive scheme with power-boosting with each transmission. The two techniques compared in this paper are full path loss compensation and partial path loss compensation. The results show that the power-boosting with full path loss compensation achieves better performance than the partial path loss. Also, using the boosting technique doubles the system capacity and provides 20% outage gain. In [38], different enhanced GF NOMA uplink techniques are discussed, besides the 3 main candidates reactive, proactive and k-repetition techniques, to satisfy the URLLC requirements. The 2 main components of delay, as suggested in the paper, are scheduling and transmission delays. GF schemes can overcome the first problem by sending the packets directly. However, collisions might occur especially in a high-density system. The first enhancement scheme, repetition with hybrid resource allocation, the scheme begins by giving each user a dedicated channel to send the first packet then they send the same packet multiple times over shared channels. The second approach is to use advanced receivers. Coded Random Access (CRA) schemes are used to resolve the collision problem. In this matter, they propose 3 receiving techniques that can be used in gNB; selection combining, where the packets are repeated and the receiver chooses the best slots for decoding. Chase combining; where the packets are repeated and the receiver performs MRC before decoding. Low rate channel coding performs packet encoding over all the transmission slots and the receiver considers all slots for decoding.

A GF algorithm to mitigate the collision problem between URLLC packets is discussed in [39]. The gNB offers each URLLC device a dedicated RB and/or a shared pool with other URLLC devices. The algorithm assigns the RB based on the number of repetitions the system can offer, the channel conditions, and the traffic loads. The two repetition scenarios considered are the k-repetitions and proactive, each one the probability of success is calculated analytically. Given the mode of operation and the available resources, the algorithm's goal is to try different allocation algorithms to achieve the reliability constraint with the least possible resources. In [40], an analysis is done on GF K-repetitions scheduling where the URLLC traffic takes a separate band without the interference with eMBB traffic. The analysis is done for different modulation schemes; QPSK 1/8, 1/4, 1/2, and 2/3 where the difference is the number of RBs the URLLC device will take at each time of transmission. Frequency hopping is adopted and collision can occur due to the number of URLLC devices in each RB can be more than one. Collisions probability is analyzed as a function of the coding scheme adopted. By increasing the number of sub-bands, the probability of collisions decreases but the transmission power of URLLC device needs to be increased or the number of repetitions will increase.

In [41], a RL environment is built to optimize the network Energy Efficiency (EE). The RL problem is divided into states, actions and rewards. States are the channel state, Signal to Interference and Noise Ratio (SINR) and traffic load. Actions are channel allocation and transmission power. And the reward is defined as the network energy efficiency with a cost subtracted when transmission of URLLC fails or SINR exceeds the threshold. Using the Q-learning algorithm, a Q-table is calculated and the optimum policy is found. It is worth noting that the original optimization problem in [41] was NP-hard and cannot be solved by normal optimization methods. This approach outperform other heuristic and ML approaches. It is clear that the RL approach performs better that other techniques which make it one of the most promising techniques to solve network optimization problems.

2.5.2.2 GB Scheduling

In [42], an optimization problem for uplink GB scheduling is discussed. In this paper, the power requirements of the URLLC devices are taken into consideration. An objective function with an aim to decrease the power consumption of URLLC and maximizing the system utility function is formulated. The authors develop a qeueuing model for each service for both eMBB and URLLC. The latency requirements for URLLC packets are translated to a maximum queue length for each service, where some services might require latency of 1 ms and other of 10 ms. The optimization problem created was a stochastic optimization problem which is difficult to solve given the time constraints in scheduling. To mitigate the difficulty of the problem, the authors use Lyapunov optimization decision algorithms. Two algorithms are developed, namely, the short and long time scales algorithms. The long time scale is used for dynamic resource allocation subjected to URLLC time constraints while the short time scale is used to optimize the power consumption of URLLC devices and the Qos of eMBB users and services. The results show that their algorithm nearly reaches the optimal solution for the proposed optimization function. In addition to balancing between latency and power constraints compared to other scheduling algorithms. Another GB uplink scheduling technique using the matching process is discussed in [43]. The adopted model depends on the assumption that the gNB is of massive numbers of antennas greater than the number of users in the cell (HTC and cMTC), so more than one user can use the same RB. An uplink optimization problem is formulated for maximizing the rate of eMBB users while satisfying the constraints of URLLC. The authors use the concepts of effective BW and effective capacity to model the Qos requirements of URLLC devices. The optimization problem is BNLP which is NP-hard. In order to solve the problem with lower complexity, the concepts of matching theory are applied to reach near-optimal results. The matching was based on each user to select their preferred RB taking into consideration the other links that will share the channel with them; this means that each user, both the URLLC and eMBB must know the CSI of all other devices, as well as a preference list for each RB of the users so the gNB should know the CSI of all the devices.

2.5.2.3 Mixed Traffic Scheduling

The overlaying between eMBB traffic and URLLC traffic is discussed in [44]. The authors discuss how the power control for both types of traffic will affect the throughput of eMBB and the reliability of URLLC. The high power level of URLLC in transmitting will increase the SINR for eMBB traffic and affects the throughput as well, and vice-versa. According to their results, the overlaying is feasible for low URLLC loads however it will affect the capacity of eMBB.

In [45], a deep learning technique is adopted and tuned to solve the mixed traffic problem for the eMBB and URLLC with puncturing for eMBB traffic. The designed model generated near-optimal solutions to the uplink scheduling optimization problem defined in their work with the aim to increase the accumulated rate for eMBB traffic after puncturing.

It is evident from the above discussions that the work in literature for mixed traffic is quite limited. An optimization problem for the uplink traffic is scarce in literature due to the absence of a mathematical model for the URLLC traffic.

2.5.3 Other Scheduling Problems using ML Techniques

In [46], a scheduling technique is discussed to solve the scheduling problem of big data using deep reinforcement learning and long short-term memory. The technique used to automate the scheduling task proved efficient in CPU usage compared to other big data scheduling techniques. In [47], deep learning is used to bypass the channel estimation for wireless links and use only the geographical location of the device for the scheduling step. The results show that the adopted technique reaches near-optimal solutions for different environments.

Similar studies adopted the NN approach to solve similar scheduling problems. In [48], The NN approach is adopted in the task scheduling problem with the aim to minimize energy consumption. It is shown that the adopted NN approach decreases the energy consumption and required overhead compared to the other adopted algorithms. In [49], the NN approach is used in smart grids to provide a scheduler for energy management scheme and a corresponding pricing scheme based on the model derived.

In [50], the NN approach is used to predict the video streaming rate in the uplink prediction. It is used to predict the uplink traffic rate for the upcoming 100 ms based on historical values, to provide insights for the network operators of the quality of service provided to the user. In [51], the deep learning approach is used to schedule edge services at real-time. It used to schedule the tasks received from edge devices that does not have the processing capabilities to perform the provided tasks. The proposed scheme provides higher accuracy and no deadline misses compared to legacy algorithms.

In [52], a deep learning approach is used to increase the energy efficiency in fog and cloud computing using a two step approach. It is used to schedule what resources are given to which task to minimize the energy consumption. In [53], a NN approach is used to schedule the resources for edge device to use the cloud resources efficiently. The results show that the NN approach reduced energy consumption by a third compared to algorithms in literature.

As discussed, the literature is rich with different scheduling problems and many algorithms that aim to solve the problem in real-time. The main obstacle with the scheduling problem is its combinatorial nature which prevents the real-time algorithms to solve it optimally.

In addition, different ML techniques are used as a tool to generate sub-optimal results in real-time. RL and NN approaches are the most known approaches to solve the scheduling problems as discussed.

2.6 Chapter Summary

In this chapter, different types of traffic that are enabled by the 5G systems are discussed. The requirements of each traffic are discussed. In addition, the suitability of each traffic type with the appropriate scheduling technique is discussed. Furthermore, different HARQ techniques are discussed along with their merits and applicability. Moreover, the 5G system and traffic scheduling process are discussed in a way that paves the way for the introduction of our problem. Finally, a thorough literature review on both the scheduling of the downlink and uplink scheduling is conducted, followed by a literature review on recent advancements on the ML applications in the domain of scheduling. The discussion in this chapter revealed the strong need to address the problem of the uplink scheduling of mixed traffic in 5G cellular networks, which is what we discuss in the upcoming chapters.

Chapter 3

System Model, Problem Formulation and Solution Approaches

3.1 Introduction

In this chapter, our system model is introduced, the GB and GF traffic for eMBB users and URLLC nodes respectively will be analyzed. A model for GF traffic is derived to satisfy the latency and reliability requirements for URLLC nodes. In addition, the eMBB traffic model is defined based on our model. Our resource allocation scheduling problem is defined as an optimization problem and sub-divided into two sub-problems without affecting the optimization problem optimality. The effect of various system parameters on the URLLC requirements is analyzed and simulated. Different algorithmic approaches in the literature are discussed and analyzed. Moreover, a novel scheduling technique is proposed and its results are compared to the aforementioned algorithms.

3.2 System Model

Our system is composed of N_a URLLC nodes and E eMBB users with a single gNB. At each TTI, the gNB receives the requests from the eMBB users, updates its information about the number of URLLC devices in the system and their latency and reliability requirements, decides on the suitable number of resources to satisfy their requirements, R. Finally, it allocates, on the grid with N_f frequency slots and N_t sTTIs, both the eMBB and URLLC traffics in a way that maximizes the eMBB rate. Then, it broadcasts the new locations to the URLLC nodes and transmits individually the granted resources for eMBB. The HARQ technique adopted is the k-repetitions technique as it is considered the general case of the reactive scheme with no repetitions. In addition, the proactive scheme is not yet accepted by the 3GPP.

3.2.1 URLLC Traffic

In this section, the Probability of Delay Bound Violation (PDBV) for the GF model is derived for our system with k-repetitions as our HARQ protocol as discussed in the previous sections. In this scheme, the node receives a packet from higher layers with probability p_a, the packet arrival probability, modeled as a Bernoulli arrival process, the Bernoulli arrival process is considered the best fit than other arrival models since the traffic of the URLLC devices is both discrete and sporadic and the transmission of the packets is in an arrive and go manner. In addition, the probability of an arrival of more than one packet is low, negligible, since the applications associated with the URLLC traffic has low packet arrival probabilities [5]. It is sent to the gNB in an arrive and go manner with the $k-1$ replicas. If an ACK is received, the packet is dropped, else the packet is re-transmitted with the $k-1$ replicas again if the latency threshold, τ, is not violated. Since all the URLLC nodes share a common pool of resources, there will be interference between the transmitted packets. If the SINR is below a certain threshold, γ_{th}, a collision occurs and the transmission is considered as a failure, however, if only one packet of the k replicas is decoded correctly, the transmission is considered as a success and an ACK is transmitted to the node. As shown in Figure 3.1, the transmission time to and from the gNB is one sTTI and the processing time is one sTTI. As discussed, the gNB does not know the locations of the URLLC nodes, so the path loss is taken for the worst-case scenario. The full path loss inversion power control, ρ, is used to compensate for the worst-case scenario. A power level control mechanism, g_m is defined to ensure that the URLLC nodes can change the power level at each re-transmission. Based on the discussed power scheme, the targeted received signal power at the gNB is $g_m\rho$. The channels are modeled as flat fading Rayleigh channels with channel gains h since the Orthogonal Frequency Division Multiplexing (OFDM) is the modulation scheme adopted in 4G and 5G systems. In addition, it is assumed to be constant for

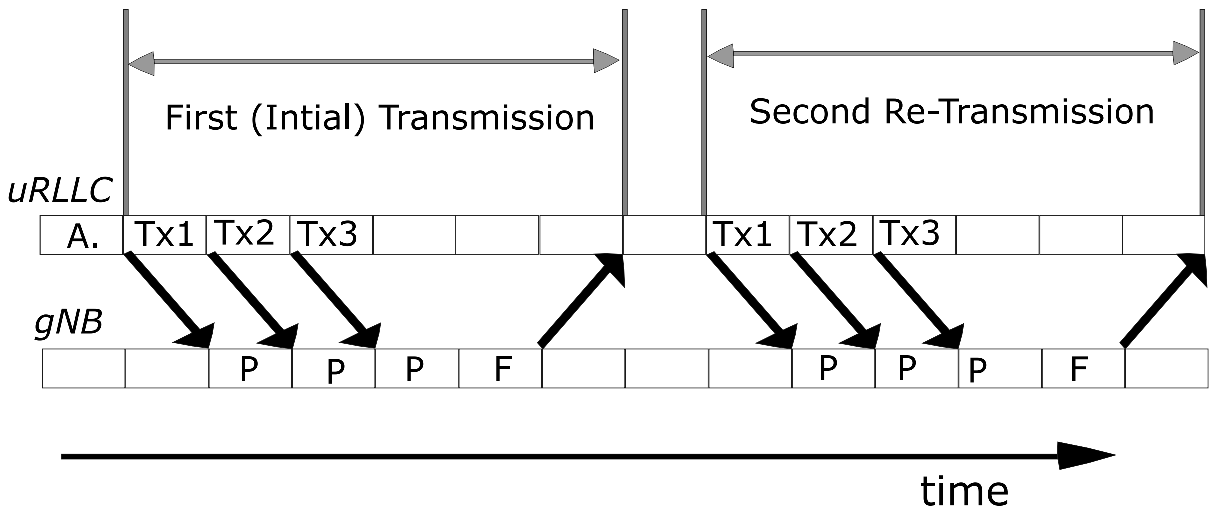


FIGURE 3.1: k-repetitions Diagram with $k = 3$

each TTI since each TTI is 1 ms. The receiver noise is modeled as white Gaussian noise with variance $\sigma^2 = N_0 B$.

The round trip transmission time, T^{RTT} can be calculated directly from the discussed procedure for any k as

$$T^{RTT} = k + 3 \text{ sTTIs}, \tag{3.1}$$

the extra 3 sTTIs included in Equation (3.1) are based on our assumptions of the packet transmission time and the processing time at the gNB. The maximum allowable re-transmissions, M, can be calculated as

$$M = \left\lfloor (\tau - 1)/T^{RTT} \right\rfloor, \tag{3.2}$$

where τ is the URLLC latency threshold, in sTTI units. The PDBV is calculated as the complement of all the possible re-transmissions that can be done by the URLLC

$$P_F = P(T \geq \tau) = \begin{cases} 1 & M = 0 \\ 1 - \sum_{m=1}^{M} A_m p_m & M \geq 1 \end{cases}, \tag{3.3}$$

where A_m is the probability that the URLLC node is still active at the m-th re-transmission and p_m is the GF access success probability, as defined in [4]. The probability A_m can be calculated as the failure of the past $m-1$ re-transmissions

$$A_m = \begin{cases} 1 & m = 1 \\ 1 - \sum_{i=1}^{m-1} A_i p_i & m \geq 2 \end{cases}, \tag{3.4}$$

and the GF access success probability of a single URLLC node, p_m, can be calculated as the probability of the number of interfering nodes multiplied by the probability of success transmission

$$p_m = \sum_{n=0}^{N_a} \binom{N_a}{n} (A_m p_a/R)^n (1 - A_m p_a/R)^{N_a - n} \\ .\Theta[n,m,k](1 - \Theta[n,m,k])^n, \tag{3.5}$$

where $\Theta[n,m,k]$ is the probability of successful transmission of the m-th re-transmission for k replicas given that the number of interfering nodes is n. And the number of allocated resources for URLLC traffic is denoted by R. The probability of success transmission can be written as

$$\Theta[n,m,k] = \sum_{l=1}^{k} (-1)^{l+1} \binom{k}{l} \frac{\exp(-l\gamma_{th}\sigma^2/g_m\rho)}{(1+\gamma_{th})^{ln}}. \tag{3.6}$$

Proof. To prove Equation (3.6), recall that the packet transmission is considered successful if at least one of the k replicas is received correctly with $SINR > \gamma_{th}$. The simplest form of Equation (3.6) can be written as

$$\Theta[n,m,k] = 1 - \prod_{l=1}^{k} (1 - Prob(SINR_l^m \geq \gamma_{th}), \tag{3.7}$$

where $SINR_l^m$ is the $SINR$ of the m^{th} re-transmission of the l^{th} replica.

Equation (3.7) can be rewritten using the Binomial theorem as [4]

$$\Theta[n,m,k] = \sum_{l=1}^{k} (-1)^{l+1} \binom{k}{l} Prob(SINR_1^m \geq \gamma_{th}, SINR_2^m \geq \gamma_{th}, ..., SINR_k^m \geq \gamma_{th} | N = n), \tag{3.8}$$

and,

$$Prob(SINR_1^m \geq \gamma_{th}, SINR_2^m \geq \gamma_{th}, ..., SINR_k^m \geq \gamma_{th}|N = n) = \exp(-\frac{l\gamma_{th}\sigma^2}{g_m\rho})\mathcal{L}_{intra}^m(\frac{\gamma_{th}}{g_m\rho}|N = n), \tag{3.9}$$

where $\mathcal{L}_{intra}^m$ is the Laplace Transform of the aggregate intra-cell interference of the m^{th} re-transmission, and it is derived as

$$\mathcal{L}_{intra}^m(s|N = n) = \mathbf{E}[\exp(-s\sum_{Q=1}^{n} g_m\rho \sum_{l=1}^{k} h_Q^l)] = (\frac{1}{1 + sg_m\rho})^{ln}. \tag{3.10}$$

Substituting Equation (3.10) in Equation (3.9), then substituting in Equation (3.8) yields Equation (3.6). □

In the next section, the eMBB traffic model is derived to fit our model. However, it is important to note that the model developed for the URLLC traffic in this section is the only model, to the best our knowledge, that is derived in the context of a single cell in order to provide the gNB with enough information to schedule the traffic efficiently.

3.2.2 eMBB Traffic

In this section, the eMBB rate equations are formulated. As discussed in the previous sections, the main requirement for eMBB users is increasing the data rate. To that extent, the rate equation for each eMBB user is defined as

$$R_e = \sum_{i,j} S_{ij}^e B \log(1 + SNR_{ij}^e), \tag{3.11}$$

where R_e is the rate of the eMBB user e, and S_{ij}^e is the gNB scheduling parameter for user e on the (i, j) RB, ie:

$$S_{ij}^e = \begin{cases} 1 & \text{eMBB user } e \text{ is allocated the } (i,j) \text{ RB} \\ 0 & \text{elsewhere.} \end{cases}, \tag{3.12}$$

and the SNR^e_{ij} is the SNR of the e-th eMBB user on the (i, j) RB, defined as

$$SNR^e_{ij} = \frac{|h_{ij,e}|^2 P_e}{N_0 B}, \tag{3.13}$$

where $h_{ij,e}$ is the channel gain of the e-th eMBB user on the (i, j) RB. P_e is the transmission power of the e-th eMBB user and $N_0 B$ is the noise variance. Equation (3.11) calculates the rate of the e-th eMBB user, where it accumulates the rates of each RB scheduled for that user, ie: when $S^e_{ij} = 1$.

In the next section, the optimization problem is formulated for our system based on the derived equations in this and the previous section. Then, a transformation for the problem is done to ease the calculations and solution approaches are discussed.

3.3 Problem Formulation and Proposed Algorithms

In this section, the optimization problem is formulated and the solution approaches are discussed. In addition, an algorithm is proposed to overcome the shortcomings of the other algorithms.

According to the discussions in the previous section, a good scheduler should provide the URLLC nodes with enough resources, based on their requirements, to satisfy their latency and reliability requirements. In addition, it should maximize the accumulated eMBB users' rate to avoid the under-utilization of the system's resources. To avoid starvation, a minimum rate for each eMBB user should be guaranteed. In our system, there is no overlapping between different types of traffic, and no puncturing process is adopted. Based on the previous discussion, the resource allocation optimization problem can be formulated as

$$\max_{S^e_{ij}, R, k} \sum_e R_e, \tag{3.14}$$

subject to

$$p(T \geq \tau) \leq \epsilon, \tag{3.14a}$$

$$S_{ij}^{e} \in \{0,1\}, \qquad \forall i, j, e \tag{3.14b}$$

$$R_e \geq R_e^{min}, \qquad \forall e \tag{3.14c}$$

$$\sum_{e} S_{ij}^{e} = 1, \qquad \forall i, j \tag{3.14d}$$

$$R \in \{0, 1, 2, ..., N_f\}, \tag{3.14e}$$

$$S_{ij}^{e} = 0, \qquad \text{for } i \in \{i_1, i_2, ..., i_R\}. \forall j, e \tag{3.14f}$$

Equation (3.14) aims to maximize the accumulated eMBB users' rate based on the optimization parameters. Equation (3.14a) ensures that the PDBV for a certain latency, τ, is below the reliability threshold, ϵ. Equation (3.14b) ensures that the scheduling decision is a binary parameter. Equation (3.14c) prevents starvation for each eMBB user by providing a minimum rate, R_e^{min}. Equation (3.14d) ensures that each RB is scheduled for one eMBB only and Equation (3.14f) ensures no overlapping between both types of traffic. Equation (3.14e) limits the number of the allocated RBs for URLLC traffic based on the system's resources.

In the problem formulated, some decision variables are integers, k, and R, while S_{ij}^{e} is binary. It can be further simplified, without relaxation, in order to simplify the operating algorithms. If we consider the mapping of S_{ij}^{e} to $\hat{S}_{ij}$ as

$$S_{ij}^{e} = 1 \longrightarrow \hat{S}_{ij} = e. \tag{3.15}$$

The new, integer, decision variable $\hat{S}_{ij}$ can be defined as

$$\hat{S}_{ij} = \begin{cases} e & \text{eMBB user } e \text{ is allocated the } (i,j) \text{ RB} \\ 0 & \text{URLLC nodes are allocated the } (i,j) \text{ RB} \end{cases}. \tag{3.16}$$

Using Equation (3.16), the resource allocation optimization problem can be simplified to

$$\max_{\hat{S}_{ij},R,k} \sum_{e} R_e, \tag{3.17}$$

subject to

$$p(T \geq \tau) \leq \epsilon, \tag{3.17a}$$

$$\hat{S}_{ij} \in \{0, 1, ..., E\}, \qquad \forall i, j \tag{3.17b}$$

$$R_e \geq R_e^{min}, \qquad \forall e \tag{3.17c}$$

$$R \in \{0, 1, 2, ..., N_f\}. \tag{3.17d}$$

Equation (3.17) is the same as Equation (3.14) except for changing the binary decision parameter into an integer parameter using Equation (3.15). Equations (3.14d) and (3.14f) are removed since the new scheduling parameter cannot take two values on the same RB by definition. The resource allocation optimization problem defined in (3.17) is an integer non-linear optimization problem. The combinatorial nature of this problem makes it, in general, hard to solve and mostly requires exhaustive search to reach the optimal solutions. Exhaustive, grid, search becomes cumbersome and unrealistic when the dimensions of the problem increase. The two main factors that affect the problem dimensionality are the number of RBs, N_f, and the number of eMBB users, E.

The problem can be further divided into two smaller sub-problems. The first problem is choosing the two URLLC parameters, the number of RBs for URLLC traffic, R, and the repetition factor, k, to satisfy Equation (3.17a). This, of course, requires a good understanding of the PDBV behavior based on the system's parameters. To that extent, Equation (3.17a) is studied when varying several system parameters to aid designers and network engineers in choosing the optimal decision variables. The second step is to use the values of the previous step to optimally allocate the eMBB users' traffic to maximize their accumulated rate while satisfying their minimum rate requirements.

3.3.1 Solution Approach

In solving the first sub-problem, Algorithm 1 is used. Algorithm 1 requires the system's parameters to be known in order to calculate the PDBV. Due to the iterative nature of Equation (3.17a), Algorithm 1 calculates each instance of A_m and p_m then calculates the PDBV. Equation (3.17a) requires high computational power especially if the number of URLLC nodes is large. This is the main reason for the problem separation, as the resource allocation problem needs to be done each TTI, it will be time-consuming and unrealistic to calculate it each time. Equation (3.17a) can be calculated once and used each TTI if all the system's parameters included remained the same. In addition, Equation (3.17a) is irreversible as the repetitions factor, k, and the number of allocated RBs for URLLC nodes, R, cannot be calculated directly. That is the main reason this equation is studied separately in the next section to give network designers insights to choose the optimum values based on the network parameters.

Algorithm 1: Calculating the PDBV in Equation (3.14a)

```
Require: τ, N_a, k, γ_th, σ, g_m and ρ
  Calculate M using Equation (3.2)
  if M = 0 then
    PDBV=1
  else
    for m = 1 to M do
      Calculate A_m using Equation (3.4)
      Calculate p_m using Equation (3.5)
    end for
  end if
  Calculate PDBV using Equation (3.3)
```

The second sub-problem is of combinatorial nature, which generally requires exhaustive search methods to find the optimal solution. The exhaustive search method becomes unrealistic when the dimensions of the problem increase. That is why many algorithms are developed in the past few years to find sub-optimal solutions in real-time. The Best Channel Quality Indicator (Best CQI) algorithm and the Proportional Fair (PF) algorithm [54] are the most well-known algorithms in the literature. The Best CQI solves the resource allocation optimization problem without taking into consideration the minimum rate constraint. It aims to maximize the the accumulated rate of the eMBB users without taking in consideration to the actual rate granted to each eMBB user. This might cause starvation for the eMBB users with bad channel conditions, however, it serves as a threshold line for the other scheduling algorithms. On the other hand, the PF algorithm provides fairness among the eMBB users, where the eMBB users are allocated channels to have approximately equal resources. The most common parameter used to define fairness is known as the Allocation Fairness (AF) parameter and it is defined as

$$\mathrm{AF} = \frac{(\sum_{e=1}^{E} N_f^e)^2}{E \times \sum_{e=1}^{E} (N_f^e)^2}, \tag{3.18}$$

where N_f^e is the number of RBs allocated to the e^{th} eMBB user and the system has E eMBB users. The aim of the PF to maintain the AF equals to unity, providing equal resources to all the eMBB users. This might decrease the accumulated rate of the system especially if at least one of the users has bad channel conditions on all the available RBs. It aims to maximize the minimum rate of the eMBB user, in addition to, decreasing the rate variances among them. Another well-known approach that is used in solving the same kind of problems is the Genetic Algorithms (GA) [55]. GA uses reproduction and mutation to reach a sub-optimal solution. In addition, in our design, GA ensures the starvation problem is resolved, unlike the Best CQI algorithm. In our design, the GA objective function is maximizing accumulated eMBB rate. The constraints are satisfying the minimum eMBB user rate, the integer property of the selection parameters and the number of slots allocated to the URLLC traffic. If the minimum rate constraint is violated, the objective function is set to zero. In addition, if the number of slots dedicated for the URLLC traffic is greater than or less than number chosen by Algorithm 1, the objective function output is set to zero. It is clear that this setup will produce, at least, a sub-optimal solution satisfying all the problem constraints. On the other hand, as the dimensions of the problem become larger, the GA becomes less accurate and consumes a lot of time that is not suitable for real-time operation.

3.3.2 Proposed Resource Allocation Mixed-Scheduling Algorithm

To avoid the shortcomings of the Best CQI, PF and GA schedulers, a novel scheduling algorithm, the mixed scheduler, is proposed. The proposed scheduler combines both the benefits of the Best CQI and PF algorithms. The real-time operation is also taken into consideration since it depends solely on both the aforementioned scheduling techniques which are considered the fastest possible schedulers. The first step is to divide the number of channels equally between the eMBB users, by allocating N_{ch} to each user. It is calculated as $N_{ch} = \frac{N_F - R}{E}$, where N_F is the available number of RBs, R is the number of assigned slots to the URLLC traffic and E is the number of eMBB users. The second step is to choose the best channel for user one then the best channel for user two and so on till each eMBB user is allocated the required number of channels, N_{ch}, with the described allocation scheme. The remaining R RBs are allocated to the URLLC traffic. In some cases, the ordering of the eMBB user might matter, however, on the long run it is nearly the same and the ordering step is considered as a time-consuming extra processing step. The ordering of the users can be done randomly to decrease the processing time. Algorithm 2 explains each step in the adopted scheduling process. The proposed scheduler is evaluated and compared to the Best CQI, PF and GA schedulers and it will be shown that the proposed scheduler provides the best sub-optimal results in different scenarios without violating any of the problem's constraints.

Algorithm 2: Proposed Mixed Scheduler for the Channels Assignment problem

```
Require: R and CSI
  Calculate N_ch, the number of channels assigned to each eMBB users to ensure minimum
  rate requirements
  for j = 1 to N_ch do
    for i = 1 to E do
      Choose the best j^th channel for user i using CSI
    end for
  end for
  Reserve the worst R channels for URLLC traffic
```

The algorithm developed in this section aims to balance between two important aspects; the satisfaction of all the users requirements and the fast processing time. In the next sections,

TABLE 3.1: PDBV Analysis Parameters [4]

Parameter	Value	Parameter	Value	Parameter	Value
BW	100 MHZ	N_a	50	N_t	10
γ_{th}	0.1	ϵ	10^{-5}	τ	1 ms
P_a	10^{-4}	g_m	m	σ^2	-114 dbm

the results for various scenarios are discussed to show the applicability of our mixed scheduler in different scenarios and environments.

3.4 Simulation Results and Discussions

In this section, different scenarios are implemented and the scheduling results of the aforementioned schedulers are discussed. The Probability of delay bound violation is analyzed when varying different system parameters to understand their effect. Next, a comparison between all the aforementioned schedulers is simulated for different environments. The first environment simulated is of small dimensions in order to compare the schedulers with the optimal solution. Next, a comparison between the schedulers in a large dimensions environment is simulated and the results are discussed. In this chapter, all the schedulers implementations and simulations are done using MATLAB in addition to the genetic algorithm solver tool implemented in the aforementioned programming platform.

3.4.1 PDBV Analysis for GF Traffic

In this section, we analyze The PDBV for URLLC devices, Equation (3.14a), when varying different parameters, the repetition factor, k, the number of assigned frequency slots, R, the SINR threshold, γ_{th}, and the latency threshold, τ. Unless stated otherwise, the simulations parameters adopted from table 3.1.

Figure 3.2 shows that, at a low delay threshold, increasing the value of k will negatively affect system performance. In contrast, increasing the repetition factor k for c-MTCDS that have a higher latency threshold will decrease the PDBV and increase the reliability, as in the case of the 2 ms delay threshold. In the case of the 2 ms delay threshold, the optimal number of repetitions is 2 or 3. However, in the case of a low delay threshold, 1 ms and 1.5 ms, the optimal number of repetitions is 1 which implies that the packet is sent one time with no replicas. If the number of repetitions, k, is increased beyond optimal values, the collision probability increases due to the contention between URLLC nodes. On the other hand, if the number of repetitions is less than the optimal value, the transmission reliability decreases as the system resources are not fully utilized. In addition, Figures 3.2a and 3.2b look nearly the same, however, 2 resource blocks are granted for URLLC traffic in 3.2b instead of one resource block only. This suggests that broadcasting all the granted resource blocks to all URLLC nodes is not the optimal technique. Instead of broadcasting all the resource blocks to all URLLC nodes, the gNB should divide the nodes equally and multi-cast to each group individually the assigned RBs.

Figure 3.3 shows that increasing the latency threshold, τ, will not increase the PDBV. In fact, increasing the latency enable the URLLC nodes to re-transmit their packets if NACk is received which increases the reliability. It is important to understand that sometimes increasing the latency will not affect the PDBV, because the maximum number of re-transmissions, M, is not changed. To put it in other words, if the latency threshold is increased slightly, as in 1.2 ms to 1.4 ms in Figure 3.3, the number of available re-transmissions remains the same implying that the PDBV remains constant. Also as the activation probability, p_a, decreases along with the latency threshold, τ, a larger difference between PDBV curves increases. This is due to the fact that for higher activation probability, the system appears congested as the packets can be retransmitted several times for a longer period of time which affects the PDBV.

Finally, Figure 3.4 shows that as the SINR threshold increases, the decoding of the URLLC packets becomes difficult and the PDBV increases. As the SINR threshold increases, it becomes harder on the decoder at the gNB to decode the received packets which decrease the system reliability and increases the PDBV. In addition, the gap between one reserved frequency slot and two frequency slots, $R = 1$ and $R = 2$, curves decreases as the SINR threshold, γ_{th}, increases. It should be noted that at medium SINR thresholds, some repetition values, $k > 1$, have better performance, but this analysis is out of our scope.

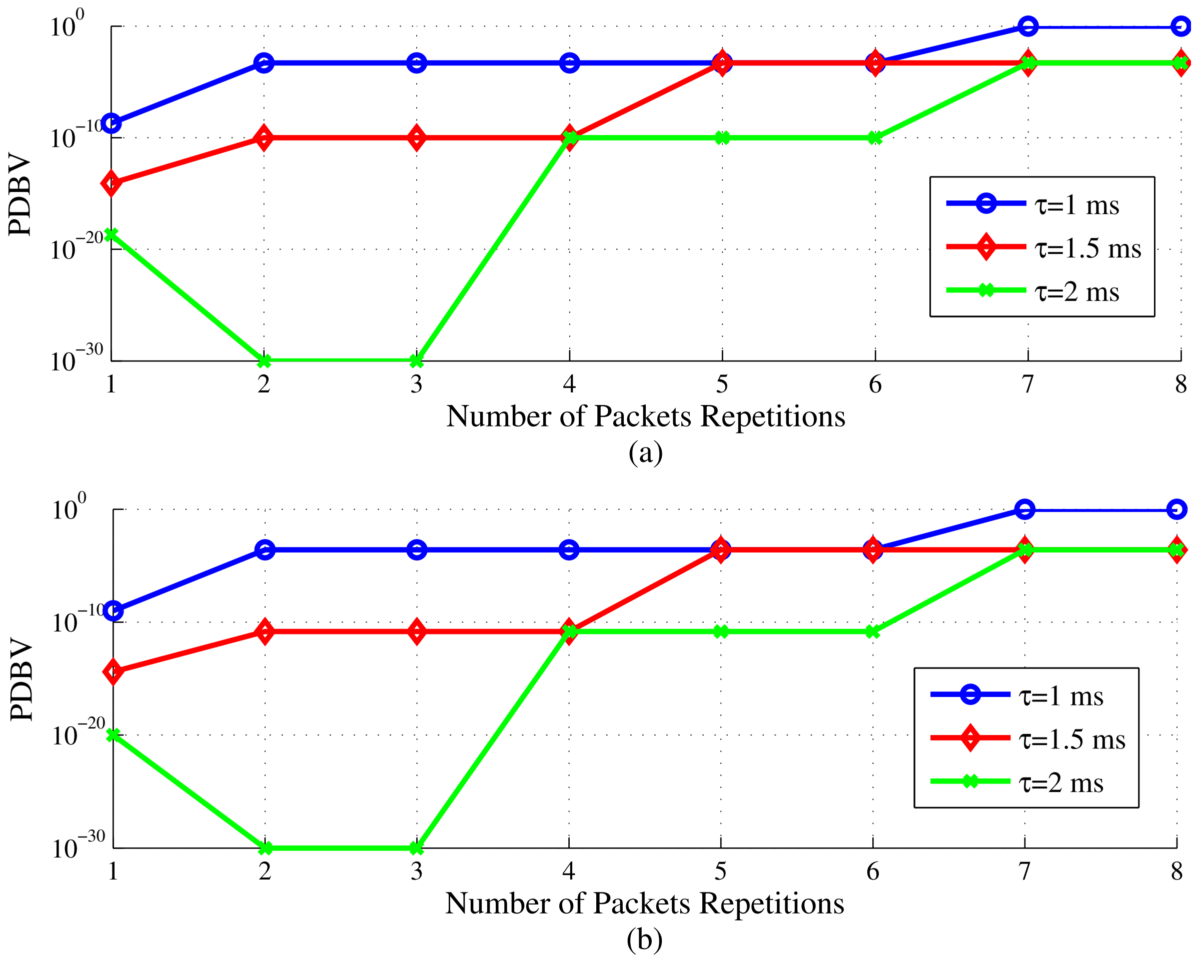


FIGURE 3.2: PDBV (P_F) vs the Number of Packet Repetitions (k). (a) One RB ($R = 1$). (b) Two RBs ($R = 2$)

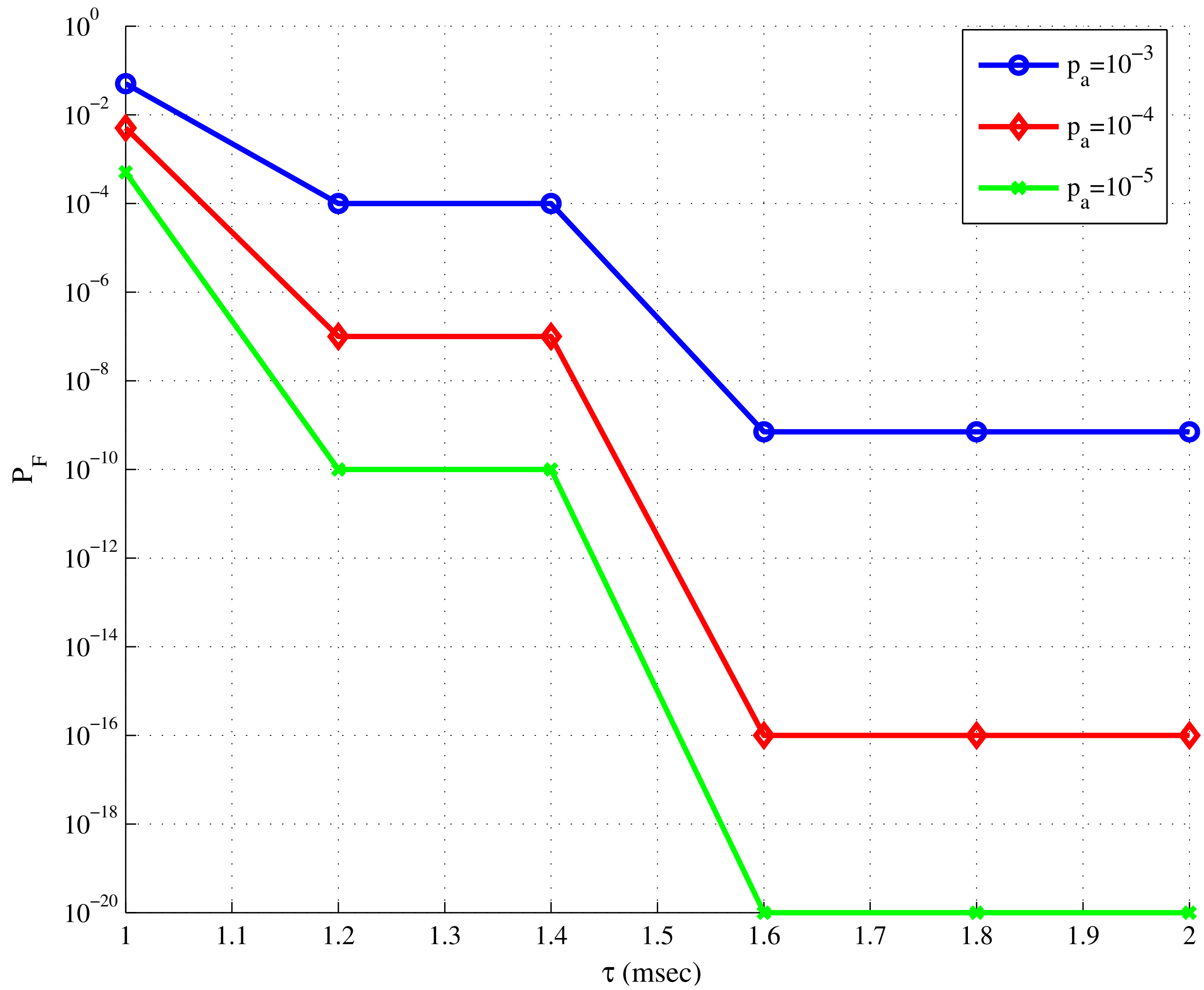


FIGURE 3.3: PDBV (P_F) vs the Latency Threshold (τ). The Number of Allocated RB (R)=1, and Number of Repetitions (k)=2

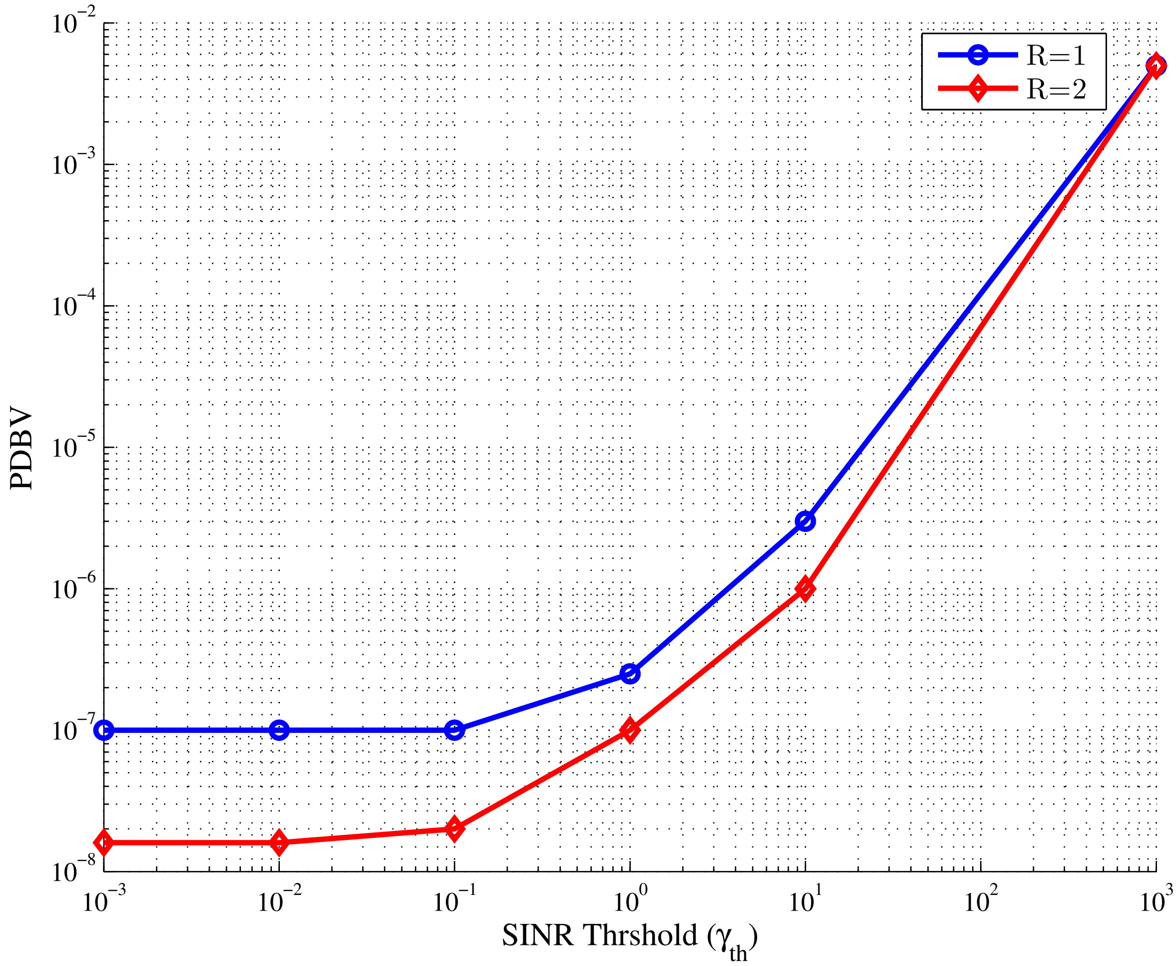


FIGURE 3.4: PDBV (P_F) vs the SINR Threshold (γ_{th}). The packet Arrival Probability (p_a)=10^{-4}, and No Repetitions, (k)=1

3.4.2 Optimal Scheduling for GB eMBB Traffic

In this section, we compare different scheduling algorithms with the optimal grid search technique. In this setup, we assume the number of frequency slots, N_f, equals 6 and the minimum rate requirement for each eMBB user is 2 Mbps with the rest of the parameters adopted from Table 3.2. Unless stated differently, The system parameters are given in Table 3.2. The goal of the scheduling algorithm, after knowing the number of frequency slots allocated for URLLC devices, is to choose the suitable channels that maximize the eMBB rate, while maintaining the minimum rate requirements for eMBB users. As explained previously, the goal of Algorithm 1 is to find the optimal number of resource blocks to satisfy the latency and reliability constraints of the URLLC traffic. Next, the number of resources chosen by Algorithm 1 is fed to the schedulers to choose the optimal allocation for both the eMBB and URLLC traffic. This separation will not affect the resource allocation optimization problem optimality since the PDBV is only affected by the number of resource blocks, R. The GA scheduler operation is explained in the previous section and the parameters adopted are as shown in Table 3.3. As seen in Table 3.3, the population size is 100 to allow the scheduler to choose the best 100 actions. The scheduler must choose integer values, as discussed before, to allocate the channels to the suitable eMBB user and the URLLC traffic. The constraint-dependent function is used to allow only the choice of the integer numbers within the suitable range, from 0 to the number of eMBB users, E. The generations limit is set to 1000 in order to avoid the scheduler searching for suitable allocation more than the intended time. This limit is important especially for environments with high dimensions, a large number of RBs.

Figure 3.5 shows a comparison between the aforementioned algorithms, with a 95% Confidence Interval (CI). In this scenario, 50 simulations runs are done for each solution point and the results are averaged along with calculating the 95% Confidence Interval (CI) for each scheduler. Figure 3.5a shows the accumulated eMBB rate when varying the number of eMBB users, E, and reserving one frequency slot for URLLC traffic, $R = 1$. In Figure 3.5b, the number of allocated URLLC frequencies, R, is changed, while maintaining the number of eMBB users fixed, $E = 3$. The Best CQI algorithm shows a higher accumulative rate than the optimal

TABLE 3.2: system Parameters [4]

Parameter	Value	Parameter	Value	Parameter	Value
BW	100 MHZ	N_a	150	N_t	10
N_f	100	ϵ	10^{-5}	τ	1.4 ms
γ_{th}	0.1	P_e	0.5 W	σ^2	-114 dbm
P_a	10^{-5}	g_m	m	R_e^{min}	12 Mbps

TABLE 3.3: Genetic Algorithm Parameters

Parameter	Value	Parameter	Value
Population Type	Double Vector	Population Size	100
Creation Function	Constraint D ependent	Selection Criteria	Stochastic Uniform
Generations Limit	1000	Scaling Function	Rank Function

search grid since it ignores the minimum rate constraint for eMBB users. The GA performs near-optimal as for a small search space the GA converges to near-optimal results. The proposed mixed scheduler performs slightly lower than the GA to maintain the fairness condition among eMBB users. Since the available resources are limited, the mixed scheduler aims to keep the fairness condition among the eMBB users than maximizing the overall accumulated eMBB rate. The PF performs the least due to the strict fairness condition.

In the next section, an operational scenario is discussed and it will be shown the shortcoming of the GA scheduler in large dimensions problems. In addition, the results show that the proposed mixed scheduler performs better than any of the mentioned techniques as it reaches the highest sub-optimal results.

3.4.3 Operational Scenario Results

In this section, a full operational scenario is discussed. The system parameters are summarized in Table 3.2.

First, PDBV is calculated for different repetition factor, k, and different number of allocated frequencies, R. Figure 3.6 shows that for $k = 1, 2, 3$ and $R = 1$ or $R = 2$, the reliability threshold, accepted by 3GPP [14], for URLLC devices, $\epsilon = 10^{-5}$, is satisfied.

Next, Best CQI, PF, GA, and our proposed mixed scheduler are used for the scheduling step. In addition, to show the drawbacks of Best CQI an error percentage is calculated in each case where the results are taken by averaging several simulation runs, precisely 10 simulations runs for this environment due to the high dimensionality of the problem. In addition, a CI of

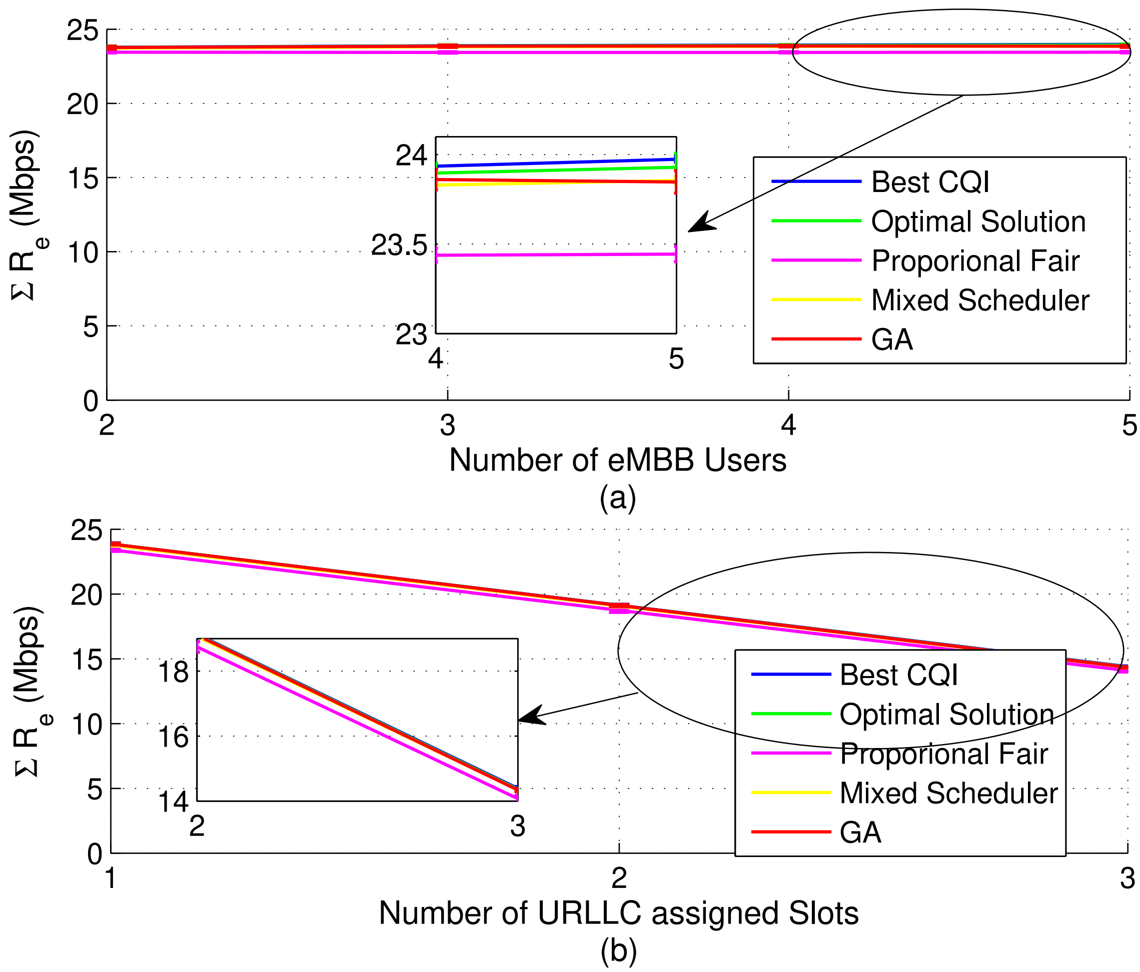


FIGURE 3.5: Comparison among the Different Scheduling Techniques. (a) Fixing the Number of URLLC Allocated RBs, $R = 1$. (b) Fixing the Number of eMBB users in the System, $E = 3$.

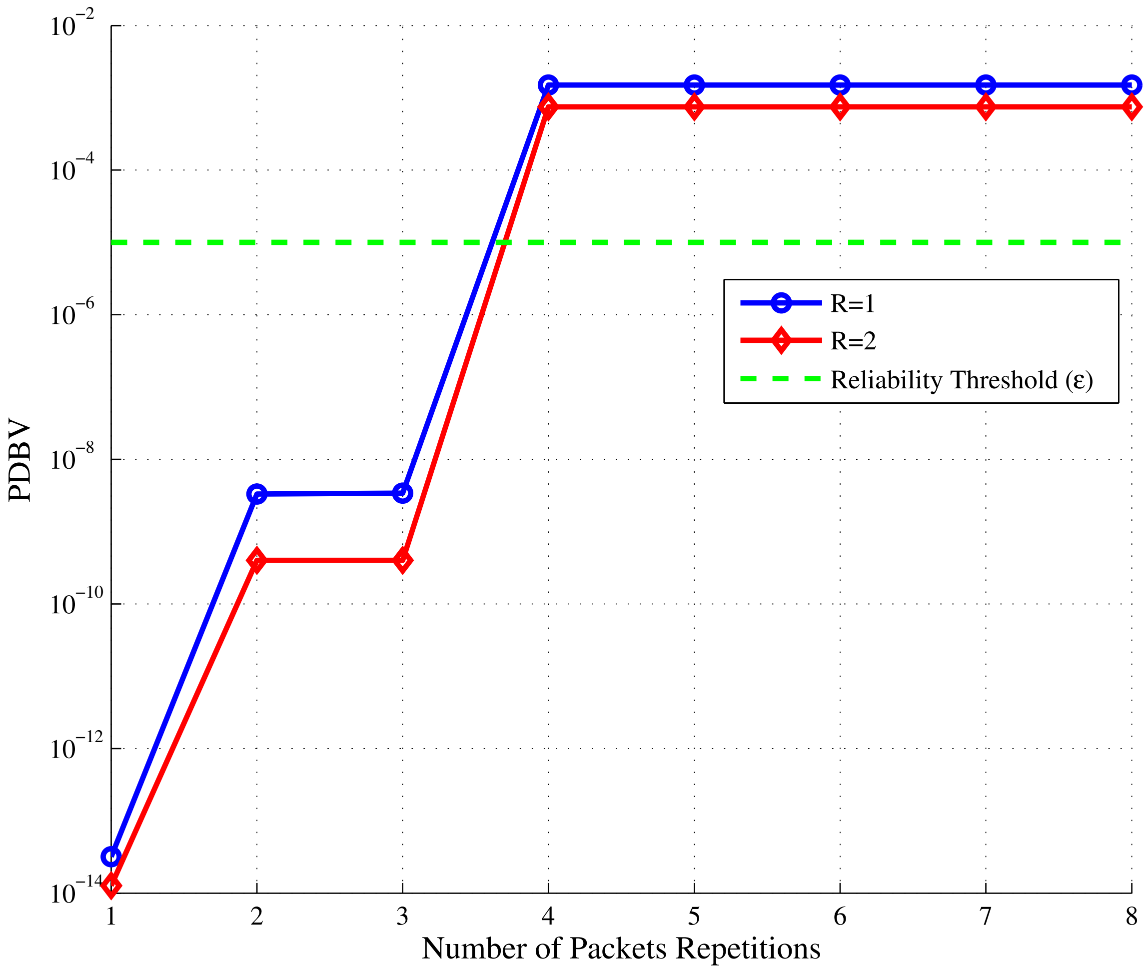


FIGURE 3.6: PDBV (P_F) vs the Number of Packet Repetitions (k)

95% is calculated for each case. The same setup for the GA is adopted, as shown in Table 3.3. Figure 3.7a shows the accumulated eMBB rate when varying the number of eMBB users, E, and reserving one frequency slot for URLLC traffic, $R = 1$. While Figure 3.8a shows the accumulated eMBB rate when varying the number of allocated URLLC frequencies, R, and a fixed number of eMBB users, $E = 15$.

As shown in Figures 3.7 and 3.8, The best CQI algorithm results in the highest data rate. Our proposed mixed scheduler comes second in terms of data rate, with all the requirements satisfied. The GA approach produces results that are lower than our approach, due to the high dimension of the problem. The PF algorithm remains the least one in terms of the overall achieved data rate due to its strict fairness condition. In addition, all the schedulers are tested for satisfying the eMBB minimum rate constraint and it is evident from Figures 3.7b and 3.8b that the best CQI scheduler is the only scheduler that violates the minimum rate requirements, and this violation increases as the number of eMBB users increases.

3.5 Chapter Summary

In this chapter, our system model is presented and each of the GF and GB scheduling models is analyzed. The optimization problem defining the scheduling task is introduced and discussed. The effect of different systems parameters on the GF traffic is presented and analyzed. Different scheduling techniques are discussed and compared to our proposed techniques in several environments with different dimensions and compared with the optimal solution whenever possible. The results show that the proposed mixed scheduler outperforms the other algorithms in the literature without violating the system's constraints. In addition, at high dimensions the mixed scheduler results remains consistent and outperforms other schedulers in terms of performance, accumulated data rate and providing the eMBB traffic minimum rate guarantee.

In the next chapter, the ML approaches are discussed. The transformation from our resource allocation optimization problem domain to different ML domains is done based on the

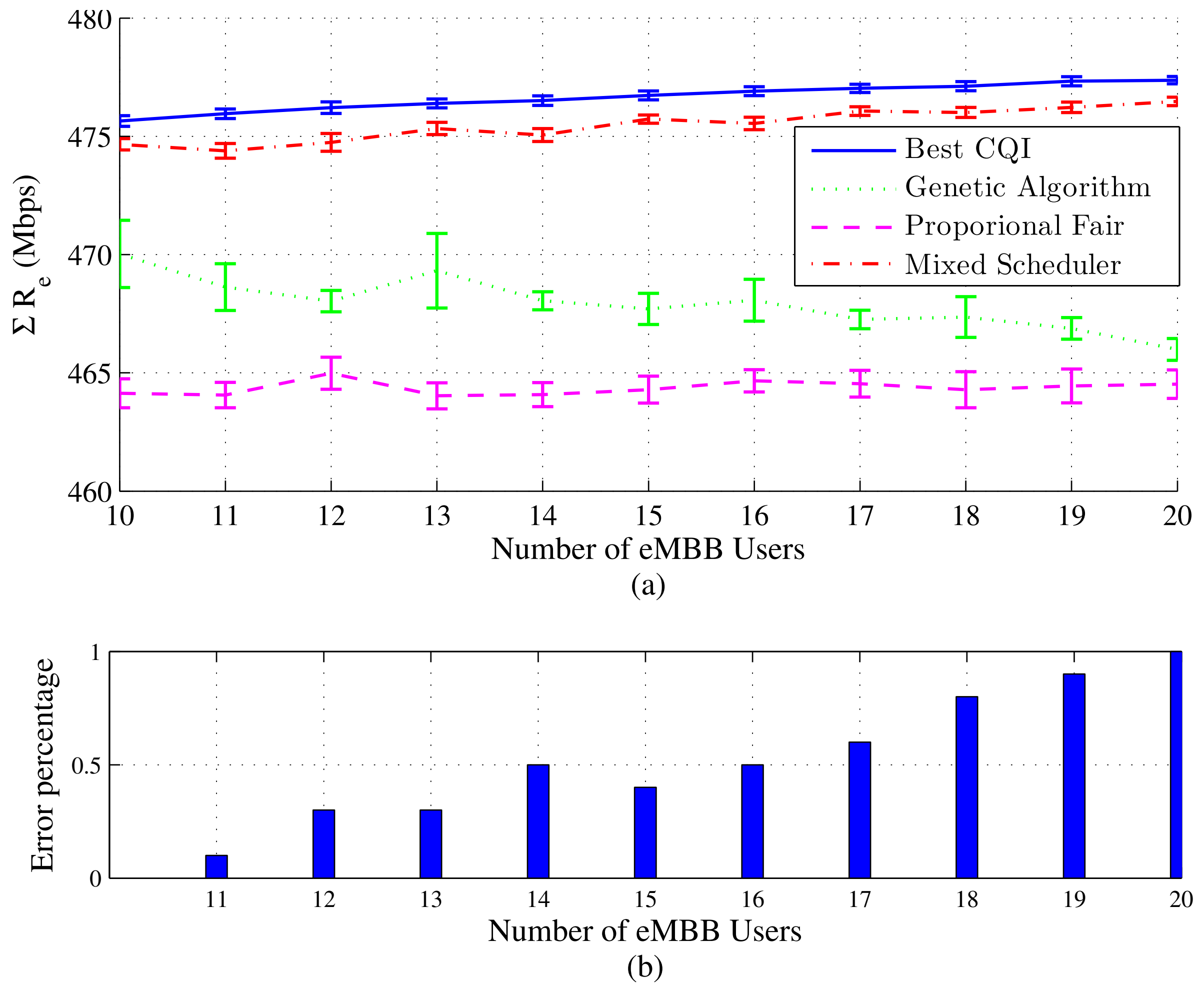


FIGURE 3.7: (a) Comparison among the Different scheduling Algorithms. (b) Error percentage in satisfying the Minimum Rate Requirements.

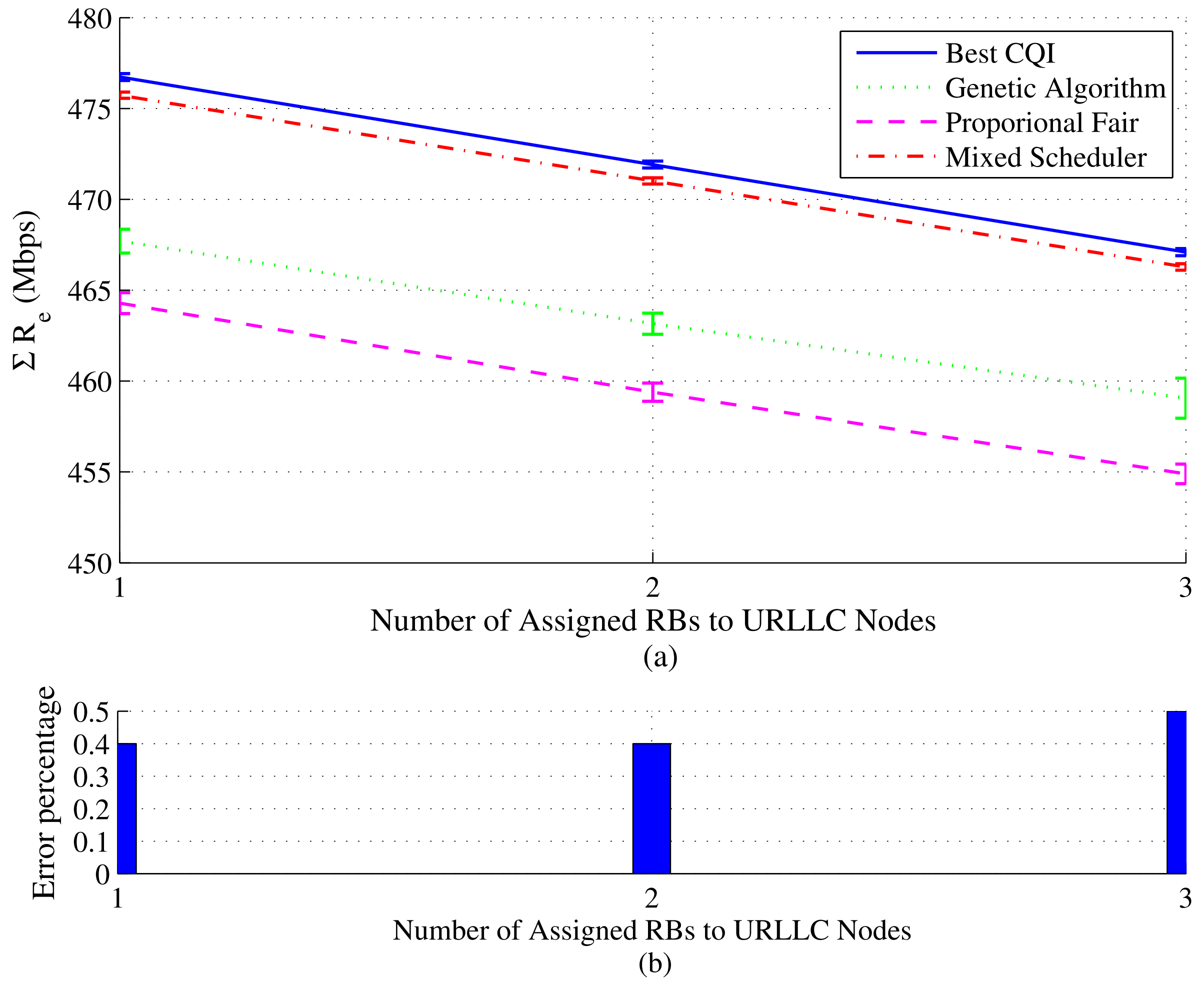


FIGURE 3.8: (a) Comparison among the GA, PF, Best CQI and the Proposed mixed scheduler. (b) Error Percentage in Satisfying the Minimum Rate Requirements.

approach adopted, reinforcement learning or neural networks. The limitations of reinforcement learning are discussed. Different environments for reinforcement learning are simulated to show the validity and the potential of the approach. Next, the neural network approach is considered to mitigate the shortcoming of the reinforcement learning approach. The solution approach and the problem transformation are discussed. Different environments are considered with different dimensions to show the soundness of the approach. In addition, all the aforementioned schedulers are compared with both approaches.

Chapter 4

Machine Learning Scheduling Approaches

4.1 Introduction

Machine Learning (ML) is proven to be one of the vital tools for solving different communications systems problems [9]. It is applied in many domains like signal processing, network security and control, mobility analysis, and many other topics. Scheduling is one of the topics whose limits and applicability with ML techniques are still vague. To that extent, we aim to solve our scheduling problem defined in the previous chapters in different ML domains. First, we use reinforcement learning techniques to solve our scheduling problem in different environments with different policies. We discuss the limitations imposed by the reinforcement learning techniques. To mitigate the limitation imposed, we use neural network techniques to solve the same problem. Different simulation environments are simulated and discussed to show the validity of our approach in different scenarios.

4.2 Reinforcement Learning Approach

In this section, the reinforcement learning approach is introduced and analyzed. The scheduling problem is transformed to fit the RL approach. The motivation behind the choice of the reward function and action space is tackled in order to satisfy all the resource allocation optimization problem constraints and reach the highest possible rates. The shortcomings of the

RL approach are discussed and the limitations of the approach are analyzed.

As discussed in the previous chapters, RL showed great promise in solving different scheduling problems and combinatorial optimization problems in general [56, 57, 58, 59]. The RL model contains two building blocks; the agent and the environment. The agent takes decisions, action, at every instant and the environment responds with feedback with its state to take proper action in the next time instant. The policy the agent tracks, for selecting an action, is based on maximizing the reward function defined for the system. The three main ingredients of any reinforcement learning are the action space, the state space, and the reward function. The action space is the set of all the possible actions the agent could take in a specific environment. The state space is the space with all the environment feedback depending on the action take in the previous time instant. Finally, the reward function is the function that defines to the agent how to give a proper action to maximize the system output.

In this chapter, the transformation of the resource allocation optimization problem to the RL problem is discussed. In addition, the limitations of the RL in solving scheduling problems are discussed. Finally, various environments are simulated for different learning policies and their results are discussed.

4.2.1 Problem Transformation

The main three parameters of any RL model are the state space, $s(t)$, the reward function, $r(t)$, and the action space, $a(t)$. In the RL approach, there is no explicit way to define our scheduling problem constraints. It needs to be defined implicitly in the reward function in order to avoid violating these constraints. The reward function, $r(t)$ should be designed in a way that increases when the rate sum of eMBB users increases and decreases when the PDBV or the problem's constraints are not satisfied.

$$r(t) = \sum_{e} R_e - c_1\zeta_1 - c_2\zeta_2 \tag{4.1}$$

where

$$\zeta_1 = \begin{cases} 1 & \text{if } R < R_{assigned} \\ 0 & \text{otherwise} \end{cases}$$

and

$$\zeta_2 = \begin{cases} 1 & \text{if } R_e < R_e^{min} \\ 0 & \text{otherwise} \end{cases}$$

where $R_{assigned}$ is the assigned number of resources that provide the latency and reliability requirements of the URLLC traffic as estimated by Algorithm 1. In addition, ζ_1 and ζ_2 implicitly define the reliability and latency requirements of the URLLC traffic as well as the minimum rate constraints for each eMBB user. It is important to note that a good weighting is needed to balance the constraints variables, ζ_1 and ζ_2, with the accumulated rate.

The action space, $a(t)$, is the space of all the decisions the agent, gNB, can take. In our formulation, it is the scheduling for eMBB users, and the frequency slots allocated for the uRLLC traffic. In this setup, the agent will function as the scheduler in the gNB. The agent must choose the best possible allocation that maximizes the reward function, i.e., maximizing the eMBB rate and satisfying the constraints. If we assume we have 4 eMBB users and one RB assigned to the URLLC traffic with 10 RBs in total, we can find the action space vary from all the resources to URLLC traffic to all resources for user 4 with all the different possible allocations in between, i.e., from

$$S = [0, 0, 0, 0, ..., 0]$$

to

$$S = [4, 4, 4, ..., 4, 4]$$

The state space, $s(t)$, is the environment that gives feedback for learning and building the Markov decision process and transition probabilities. The feedback is the channel state information, the individual rate of each eMBB user, the overall rate of the eMBB users, and the

P_F.

$$s(t) = [h_i^{1XN_F*E}] \tag{4.2}$$

In the Markov Decision Process (MDP), the agent chooses the policy, π that will increase both the present and the future rewards based on the transition probabilities. The policy can be considered as a mapping from the state space to action space; $\pi(s) : S \rightarrow A$. The discounted accumulative reward function using a policy π and a starting state s is defined by:

$$V^{\pi}(s) = \{\sum_{t=1}^{\infty} \gamma^t r_t(s_t, a_t) | s_0 = s, \pi\} \tag{4.3}$$

The goal is to find the optimal accumulative reward function

$$V^*(s) = \max_{\pi} V^{\pi}(s) \tag{4.4}$$

The optimal policy can then be calculated as

$$\pi^*(s_t) = \arg\max_{a_t \in A} \bar{\text{U}}_t(s_t, a_t) + \sum_{s_{t+1}} p(s_{t+1}|s_t, a_t) V^*(s_{t+1}) \tag{4.5}$$

Since the state space is continuous, infinite, deep RL is adopted. A Neural Network (NN) layer is added to find the optimal policy based on the reward function.

There are several learning policies in the RL and each policy differs in the exploration-exploitation factor. The greedy policy [60], aims to find the highest rewards with no extra exploration. Another variation is the epsilon-greedy policy [61] which modifies the exploration from the greedy policy to avoid reaching a sub-optimal result. Boltzmann and max-Boltzmann policies [62] balance exploration and exploitation using statistical distributions to reach better results.

The greedy action at any time instant, t, can be defined mathematically by the following equation

$$a_{greedy}(t) = \arg\max Q(a(t)), \tag{4.6}$$

where $Q(.)$ is the action-value function. As discussed, the greedy policy aims to maximize the reward at any time instant. This might affect the agent by choosing a sub-optimal path due to the absence of the exploration factor.

The ϵ-greedy action modified the greedy policy by adding an exploration factor in the training policy. It can be defined mathematically as

$$a_{\epsilon-greedy}(t) = \begin{cases} \arg\max Q(a(t)) & \text{with probability } 1-\epsilon \\ \text{any random action } a(t) & \text{with probability } \epsilon \end{cases}. \tag{4.7}$$

Based on Equation(4.7), the ϵ-greedy chooses the optimal action with probability $1-\epsilon$. On the other hand, there is a probability of choosing a random action that enables the agent to explore the action space.

In the Boltzmann policy, the aim is to exploit all the available information from the Q-table. Instead of choosing the optimal action or a random action, a distribution for all available action is designed based on Boltzmann distribution, to choose the most probable action to produce better results. The main difference in this policy compared to the previous ones is that it construct a belief table. The information of other actions in the belief table can be taken into consideration. The max-Boltzmann policy offers a slight modification compared to the Boltzmann policy. It allows to control the exploration parameter for the high probable actions, producing good results, gathered from the belief table.

4.2.2 Dimensionality Problem

The main problem in the RL approach is the dimensions of the problem. If we consider a small setup with only 3 eMBB users and 6 RBs with one reserved for URLLC traffic, with a simple calculation, the number of possible actions is 4096 actions. In general the number of actions for any number of eMBB users R with N_f RBs and R of them assigned to URLLC traffic equals $(E+1)^{N_f}$. The number of actions increases rapidly with any of the aforementioned parameters and for a moderate or large environment, the training becomes cumbersome and time-consuming.

TABLE 4.1: Scenario 1 System and Tuning Parameters

Parameter	Value	Parameter	Value
BW	1.4 MHz	N_a	150
N_f	6	ϵ	10^{-5}
γ_{th}	0.1	P_e	0.5 W
P_a	10^{-5}	g_m	m
learning rate	10^{-4}	no. of layers	2
c_1	10^8	c_2	10^8
$R_{assigned}$	1	Activation Function	RELU
N_t	10	τ	1.4 ms
σ^2	-114 dbm	R_e^{min}	3 Mbps
no. of neurons/ layer	64	loss function	MAE

4.2.3 Simulation Results and Discussions

As stated in the previous sections, different policies are tested in order to find the best policy to fit each situation. Google Colab notebook is used with keras-rl package. Adam optimizer is used, with a learning rate 10^{-4} and the Mean Average Error (MAE) is used for error calculations. Two hidden layers are used to train the model with Rectified Linear Unit (RELU) as an activation function and 64 neurons each. To the best of our knowledge, this setup and the tuning parameters in Table 4.1, produced the best achievable results. Following the work of the previous chapters, two different scenarios are discussed with different dimensions to check the validity of our approach.

4.2.3.1 Scenario 1 Results

In this system, different policies are tested in a small environment with 6 RBs. This system is adopted by the 3GPP [63]. The system and the deep RL parameters are shown in Table 4.1.

The results of scheduling of this system from the Deep RL are compared with 2 different algorithms, the optimal algorithm and the Best CQI algorithm as shown in Figure 4.1. To show the robustness of the results, a 95% CI is done for different numbers of trials, 50 trials, for each algorithm. Figure 4.1b shows the percentage of times the schedulers did not satisfy the minimum eMBB rate constraint. The Best CQI is the only scheduler that does not satisfy the eMBB minimum rate constraint, however, all the adopted policies satisfy the minimum rate constraint for each eMBB user. In Figure 4.1c, the deviation percentage between different policies rate with the optimal rate is plotted. It is evident that all the RL policies produce near-optimal results with only a 2% deviation. However, it is clear that the Epsilon Greedy policy is the best choice amongst the other policies. The Epsilon Greedy policy has the highest robustness as its deviation from the 2% line is minimal. So, in small environments, the Epsilon Greedy is the best choice due to its high robustness compared with other policies.

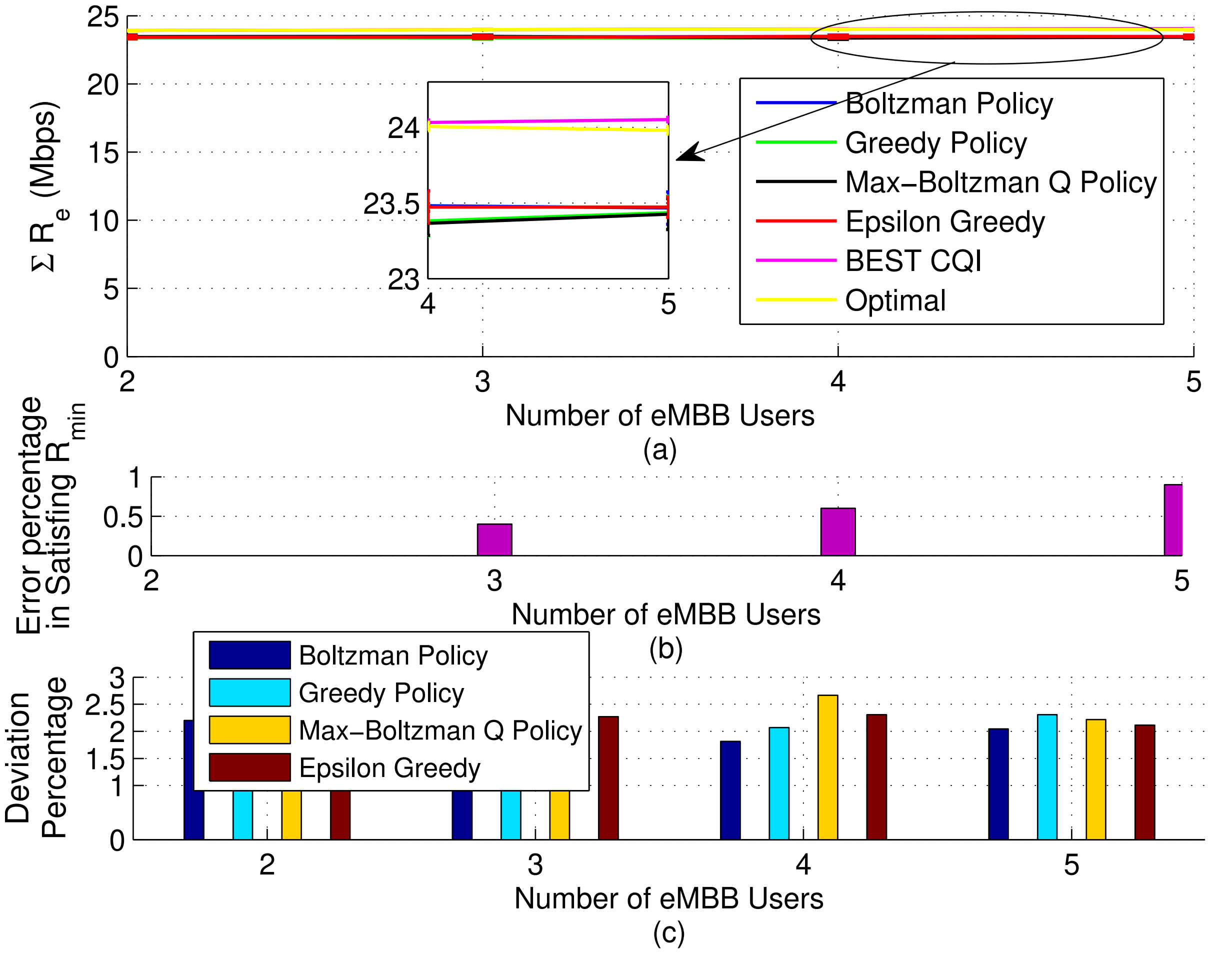


FIGURE 4.1: Comparison among Different RL Policies for Scenario 1

4.2.3.2 Scenario 2 Results

In this section, a larger environment is adopted in order to understand the influence of dimensionality on the system. The system and the RL parameters are the same as scenario 1, for comparison purposes, except the BW = 20 MHz and the RBs $N_f = 10$. In this scenario, the optimal solution is dropped due to the high processing power required for larger dimensions, and all the comparisons are done with the Best CQI algorithm. In addition, 50 trails are used to check the robustness of all the policies and to produce the 95% confidence interval.

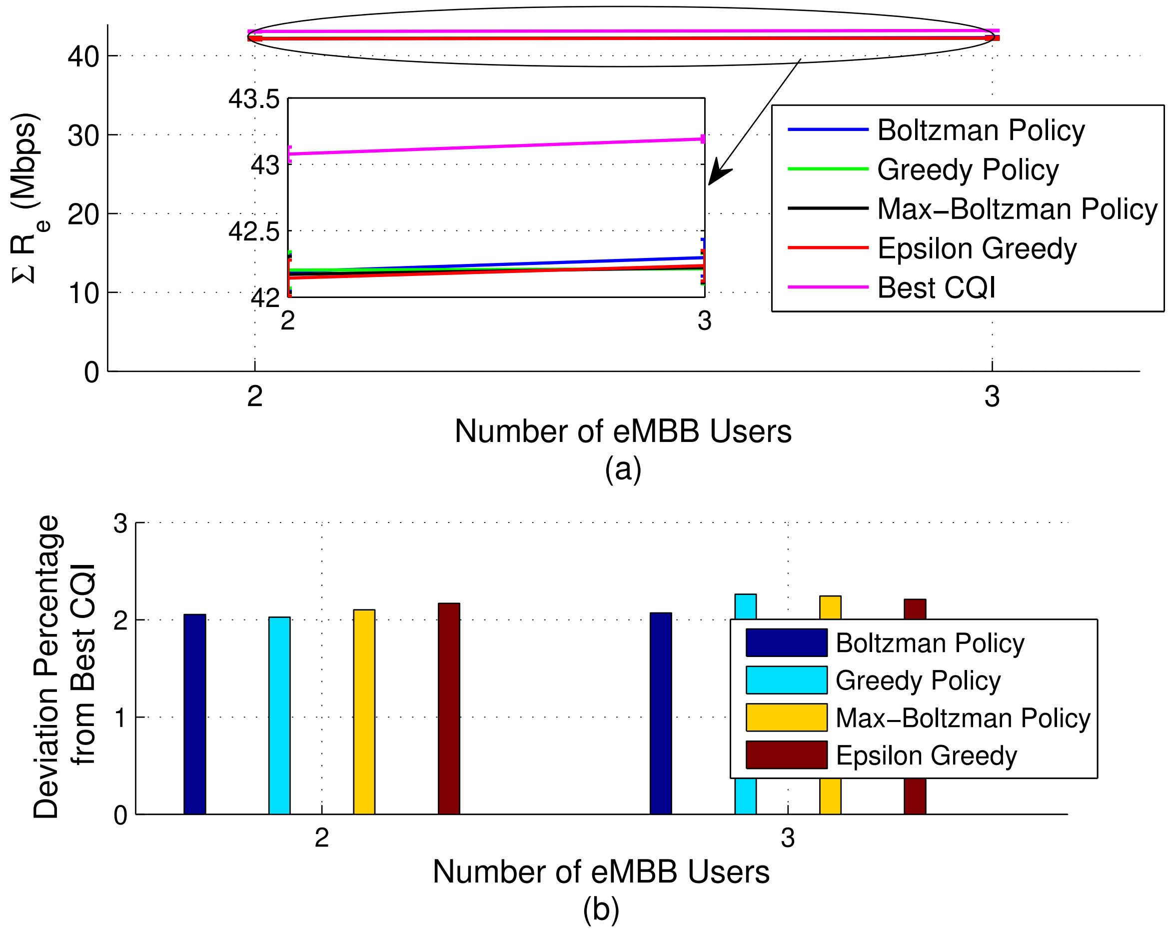


FIGURE 4.2: Comparison among Different RL Policies for Scenario 2

As shown in Figure 4.2, the 2% deviation is maintained for all RL policies. However, as in the previous scenario, the Epsilon Greedy policy has the highest robustness.

In the next section, the neural network-based algorithm is discussed. The transformation to the NN is analyzed to fit our problem. Finally, the results for different operating scenarios are simulated and compared to the RL-based techniques.

4.3 Neural Network Approach

In this section, the neural network approach using the multi-layer perceptron architecture is introduced and analyzed. The transformation of our main resource allocation optimization problem to the NN domain is discussed and the shortcoming of using the modified resource allocation optimization problem is analyzed. Next, the tuning of the NN parameters are discussed and the effect of each parameter on the results is simulated and analyzed. Finally, three different environments with different dimensions are tested using the NN-based scheduler after training with the optimized parameters.

From an engineering standpoint, neural network treats the system as a black box with inputs and outputs. The interactions between different system elements are imitated by the neurons in the hidden layers. The main advantage of the neural network in our problem is bypassing the action space expansion. In this section, the problem transformation to the neural network domain is discussed. In addition, simulation showing the insights on how to tune the neural network parameter is done separately before training, where various parameters of the neural network system are manipulated to show the effect on the network output. Next, environments with large dimensions are simulated that cannot be tackled by the RL approach.

4.3.1 Problem Setup

Unlike the RL approach, Neural Network (NN) needs a different treatment for scheduling problems. One of the biggest limitations of the NN approach is the integers output problems [64]. Due to this issue, the reduced resource allocation optimization problem defined in 3.17 cannot be used to generate good results, however, the main resource allocation optimization problem defined in 3.14 has the potential to generate good results. The problem in this setup is as follows, the neural network takes the channel gains for all eMBB users as input and outputs the scheduling for each eMBB user for each channel. The problem in this setup is the massive number of inputs and outputs along with the binary output constraint in the NN. The input and output layer is of the same size $E.N_f$ while the number of hidden layers, the number of neurons per layer, the learning rate, and the number of training epochs can be

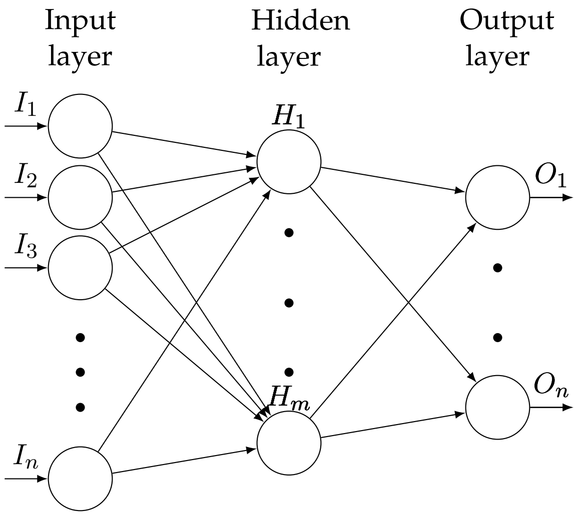


FIGURE 4.3: A NN Diagram with a Hidden Layer and the same size Input/Output layers.

varied to generate good results. The main framework of our approach can be seen in Figure 4.3. The numbers of neurons in the input and output layers are the same and at least one hidden layer, with a different number of neurons, is adopted to understand the interactions between the input and the output. This method is used to overcome the dimensionality problem in the RL approach. If a comparison is done between each approach dimension, it can be seen clearly that if the RL approach is used, the dimension of the problem is exponential to both the number of resource blocks and the eMBB users while it is linear for the NN approach. The mathematical model of the neural network is well established in the literature [65]. The neurons in the hidden layers try to understand the interactions between the input and output through non-linear activation functions. To generate the training samples, MATLAB is used with the Best CQI scheduler. A sample size of 10,000 samples is generated for each training instance on MATLAB and it is converted into a csv file to be used in Google Colab for the training and testing steps. Finally, a split of 70%-30% for training and testing for our neural network scheduler is done.

TABLE 4.2: NN Training Parameters

Parameter	Value	Parameter	Value
BW	20 MHz	$R_{assigned}$	1
N_f	100	loss function	MSE
E	12	P_e	0.5 W
Activation Function	Sigmoid	N_t	10

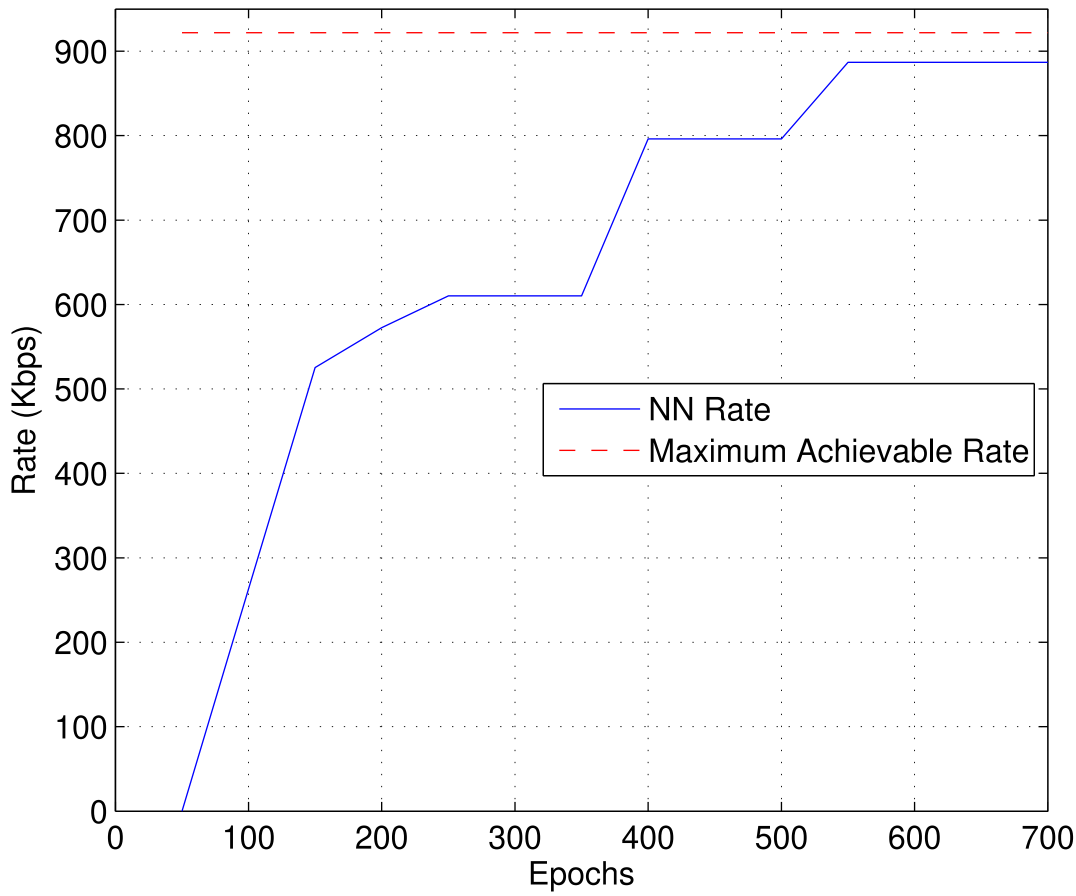
4.3.2 Results and Discussions

In this section, different parameters of training are varied, number of training epochs, number of hidden layers, learning rates, and number of neurons per hidden layer. Each effect of the training parameters is plotted individually against the maximum achievable rate by the Best CQI algorithm for the given scenario. In addition, a simulation is done on the highest dimension environment adopted by the 3GPP in order to show the validity of our claim. Finally, different smaller dimensions environments are simulated to show that the adopted approach is applicable for any dimension. Google Colab notebook is used in training the NN scheduler using the free CPU and GPU resources.

4.3.2.1 Tuning the Neural Network Model

As discussed, several parameters need to be tuned in order to find a model that will generate the best results. To that extent, 10000 samples are generated using the Best CQI algorithm and used for training and validation for the NN scheduler. The system parameters used are in Table 4.2.

To know the best tuning for each parameter, the rest are left fixed and a range of values is plotted versus the rate and compared with the Best CQI algorithm rate. The number of input and output neurons, as stated earlier, is large, it is 1200 in our system. First, the system is tuned for the number of training epochs, as seen in Figure 4.4, it requires at least 550 epochs to begin to converge to a suitable value. A large number of epochs is related to the fact of the large dimensions of the input and output layers. In addition, the system used is stable as it keeps the results to about 250 epochs, as opposed to other systems which diverge quickly from the optimal.



f

FIGURE 4.4: Rate Versus Number of Epochs

Secondly, the number of hidden layers is tested as shown in Figure 4.5. It is clear that the optimal number of layers is 2 since it generates the highest rate. It is clear that for only one hidden layer, the NN did not grasp the system behavior to reach a good performance due to the lack of enough non-linearity. In contrast, when increasing the number of hidden layers, the system will require a larger number of epochs to train and converge, as in the case of the 4 hidden layers.

Next, the number of neurons per layer is tested. A small number of neurons are tested at first, however, it showed inefficiency in training so it is neglected. As shown in Figure 4.6, the optimal number of neurons per hidden layer is 1700 as it generates the highest rate. Both 1500 and 2100 generate near-optimal results however not as high as the 1700. Unlike the previous

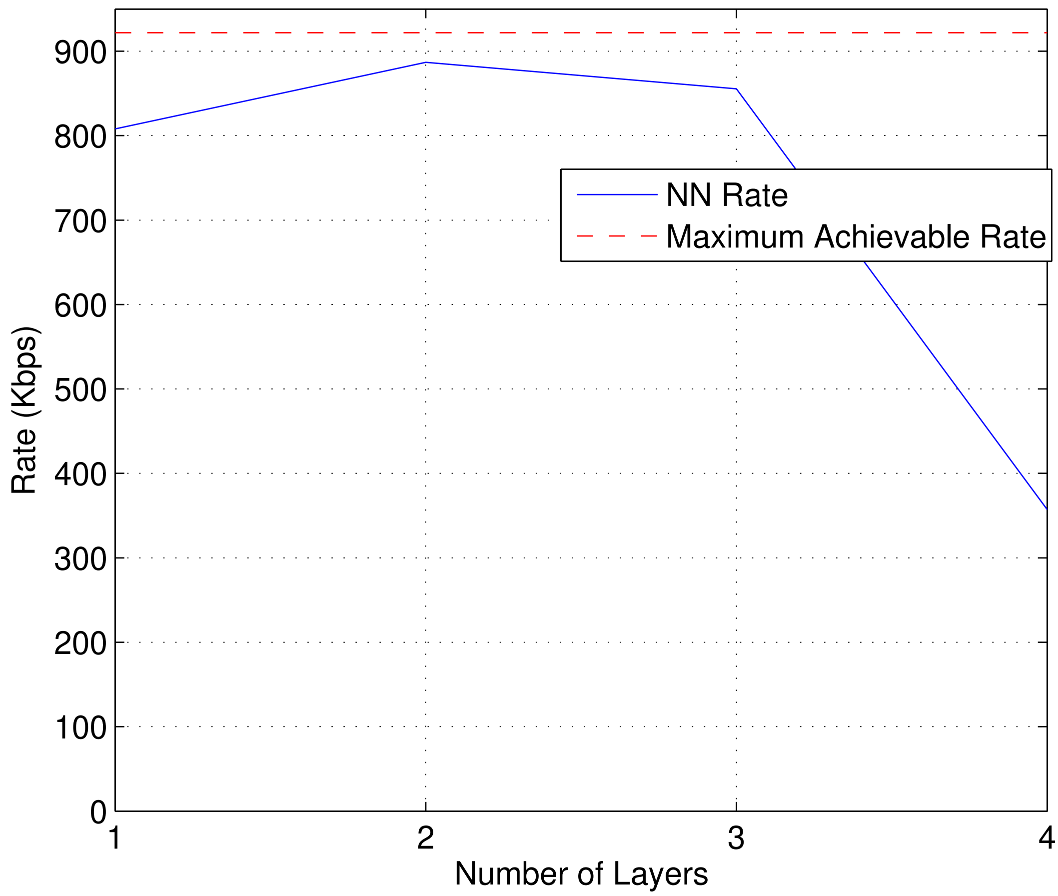


FIGURE 4.5: Rate Versus Number of Hidden Layers

plots, Figure 4.6 shows a fluctuation as opposed to a fixed trend. Due to this fact, a large range is plotted to make sure that our choice is optimal.

Lastly, the rate is plotted versus the neurons learning rate. As shown in Figure 4.7, the optimal learning rate is 10^{-4}. In the case of a low learning rate, the system did not have enough time to learn the optimal policy to reach a solution. In contrast, for high learning rate increases, the results keep fluctuating and do not reach the optimal or near-optimal performance, in addition, fluctuation decreases the robustness and reliability of the system.

Based on the above discussions, the optimal parameters used for training is 600 epochs with 2 hidden layers and 1700 neurons per layer with a learning rate equals 10^{-4}.

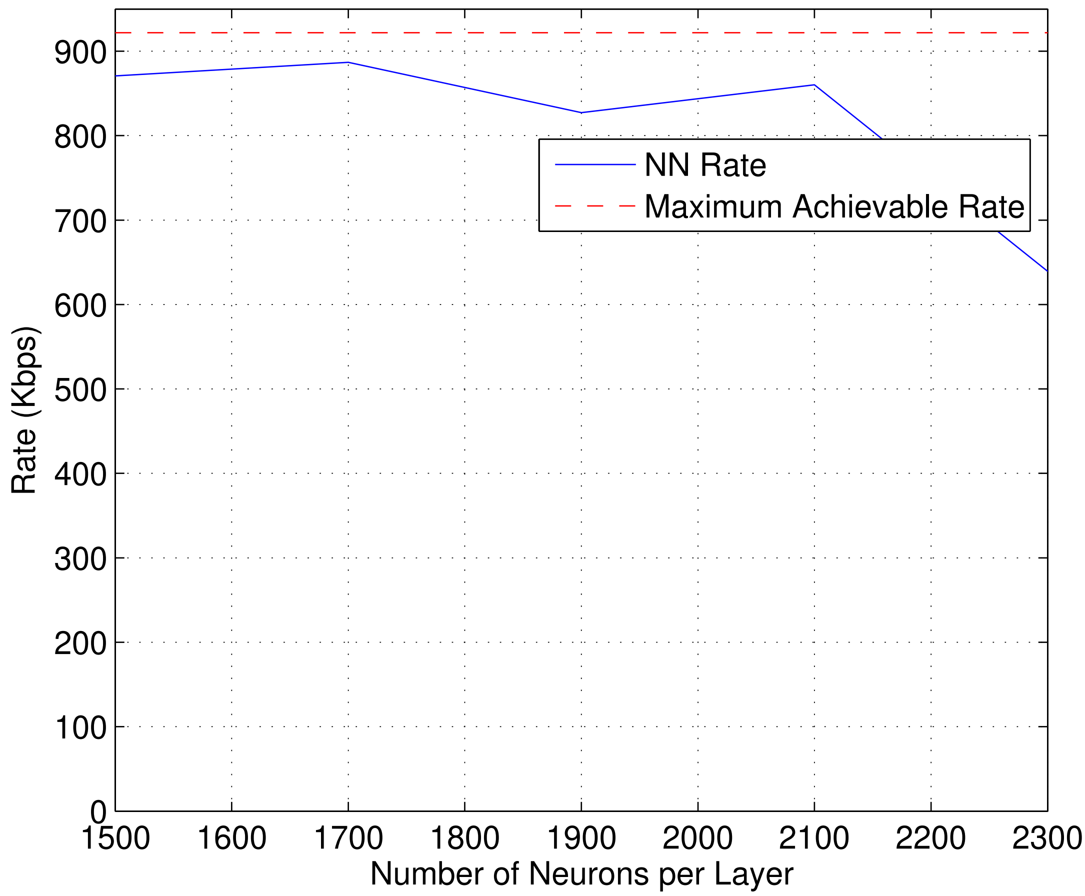


FIGURE 4.6: Rate Versus Number of Neurons per Hidden Layer

4.3.2.2 Scheduling Results

In this section, the neural network results are discussed based on the model defined in in the previous section. As previously discussed, the model parameters are tuned to produce the best sub-optimal results for our system. The system and the NN scheduler parameters are as shown in Table 4.3.

As shown in Figure 4.8, the NN scheduler is compared with the Best CQI results after training for a different number of eMBB users, E. In Figure 4.8b, the deviation between the NN scheduler and the Best CQI algorithm is 6%, which makes sense as there is no reward function or policy to direct the model to the optimal scheduling.

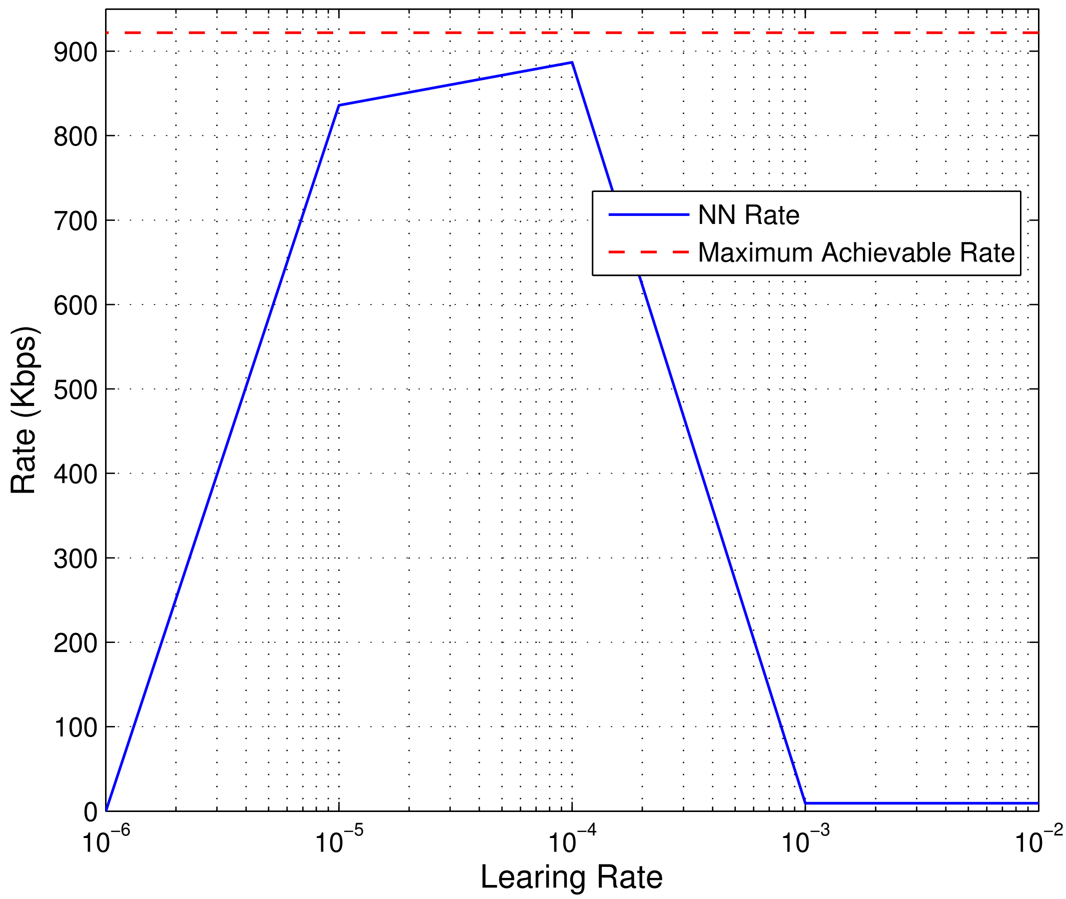


FIGURE 4.7: Rate Versus Learning Rate

4.3.2.3 Neural Network Scheduler Results for Lower Dimensions

To make sure that our proposed technique works better or at least the same for other environments, additional 2 operational scenarios, are tested with different dimensions [63]. For the first scenario, a BW of 5 MHz is used with 25 RBs Several testing is done to tune the parameters as in the previous section, with the goal to change the least number of parameters possible. The adopted system parameters and the NN parameters are summarized in Table 4.4.

It is clear from Table 4.4 that the main difference is using fewer neurons per layer compared to the larger scale scenario. Figure 4.9 shows a comparison among different scheduling techniques along with the NN scheduler. As shown in Figure 4.9(a), the NN scheduler achieves nearly the same as the GA approach with a real-time operation after training. In Figure 4.9(b),

TABLE 4.3: NN Training Parameters for Scenario 3

Parameter	Value	Parameter	Value
BW	20 MHz	$R_{assigned}$	1
N_f	100	loss function	MSE
Learning Rate	10^{-4}	P_e	0.5 W
Activation Function	Sigmoid	N_t	10
Epochs	600	Hidden Layers	2
Neurons/Layer	1700	Number of Trials	10

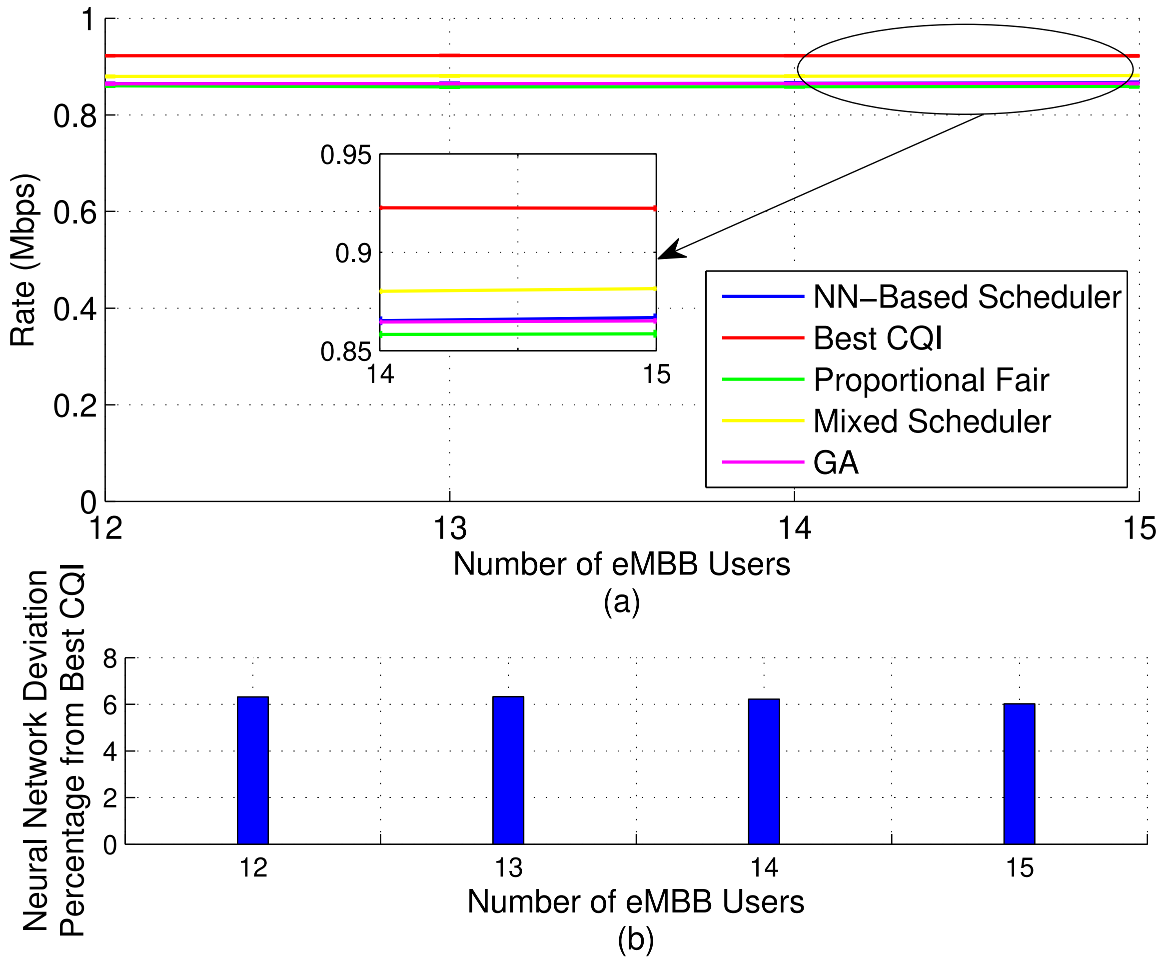


FIGURE 4.8: Comparison Among Different Scheduling Algorithms with the NN Scheduler for 100 RBs.

TABLE 4.4: NN Training Parameters for Scenario 4

Parameter	Value	Parameter	Value
BW	5 MHz	$R_{assigned}$	1
N_f	25	loss function	MSE
Learning Rate	10^{-4}	P_e	0.5 W
Activation Function	Sigmoid	N_t	10
Epochs	600	Hidden Layers	2
Neurons/Layer	400	Number of Trials	30

TABLE 4.5: NN Training Parameters for Scenario 5

Parameter	Value	Parameter	Value
BW	15 MHz	$R_{assigned}$	1
N_f	75	loss function	MSE
Learning Rate	10^{-4}	P_e	0.5 W
Activation Function	Sigmoid	N_t	10
Epochs	600	Hidden Layers	2
Neurons/Layer	1000	Number of Trials	10

it is shown that the maximum deviation from the Best CQI algorithm is 1.2%.

Secondly, a larger environment compared to the previous one is tested and compared to other algorithms. The BW of the adopted system is 15 MHz and the system is composed of 75 RBs. The rest of the parameters along with the NN scheduler parameters are summarized in Table 4.5. It is shown from Table 4.5 that the only NN scheduler parameter changed is the number of neurons per layer which makes sense since the system's dimensions increased significantly.

Figure 4.10 shows a comparison among different scheduling techniques compared to the NN approach. In Figure 4.10(a), the NN approach reaches approximately the same results as the GA approach, the same as the previous scenario. In addition, Figure 4.10(b) shows the deviation of the NN scheduler from the Best CQI algorithm is maintained at 3.4%.

From the previous results, it can be concluded that the NN approach is a powerful tool and applicable for different scenarios and system dimensions. In addition, the NN scheduler reaches nearly the same results as GA with no processing time after training.

The main comparison points between both schedulers, the RL and NN scheduler, is described in Table 4.6. As shown in Table 4.6, the dimensions of the RL scheduler is of exponential complexity as opposed to the linear relationship between each factor for the NN scheduler.

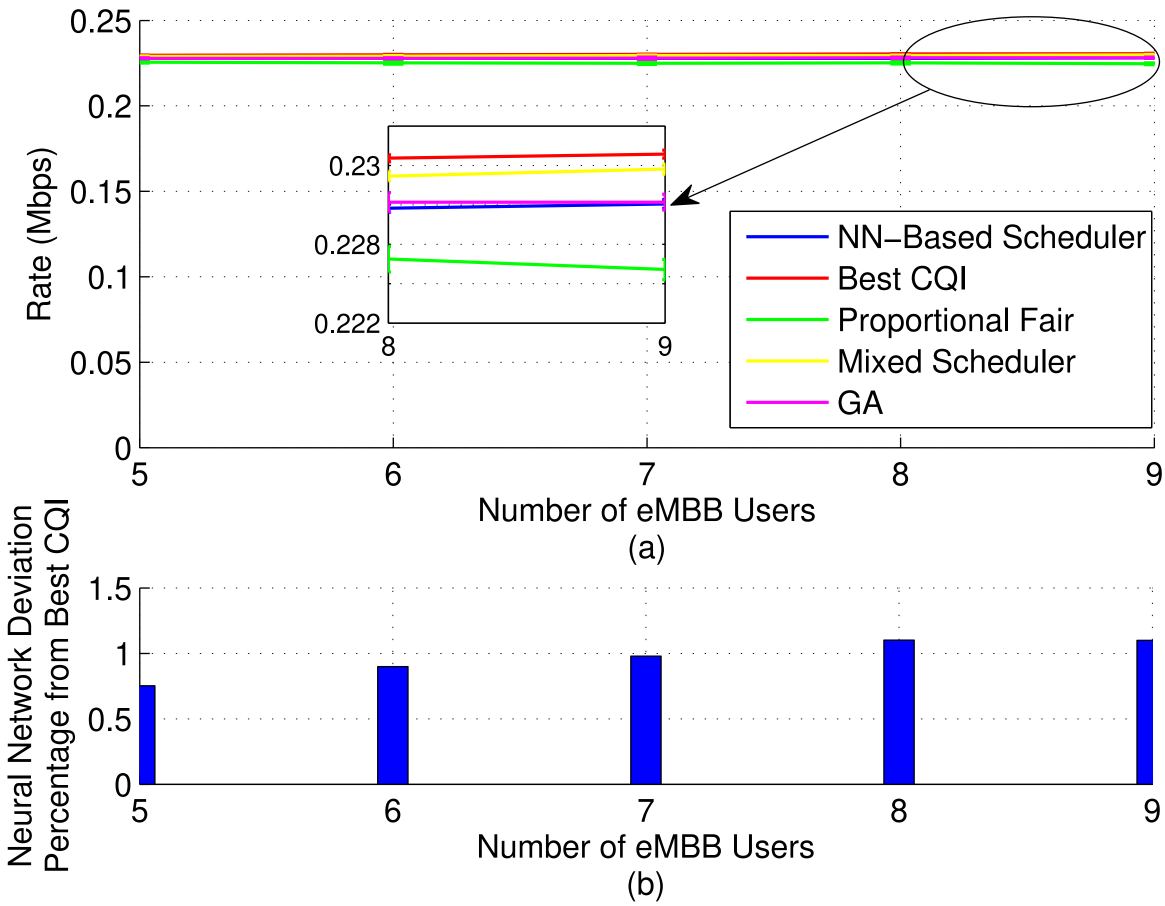


FIGURE 4.9: Comparison among Different Scheduling Algorithms with the NN Scheduler for 25 RBs.

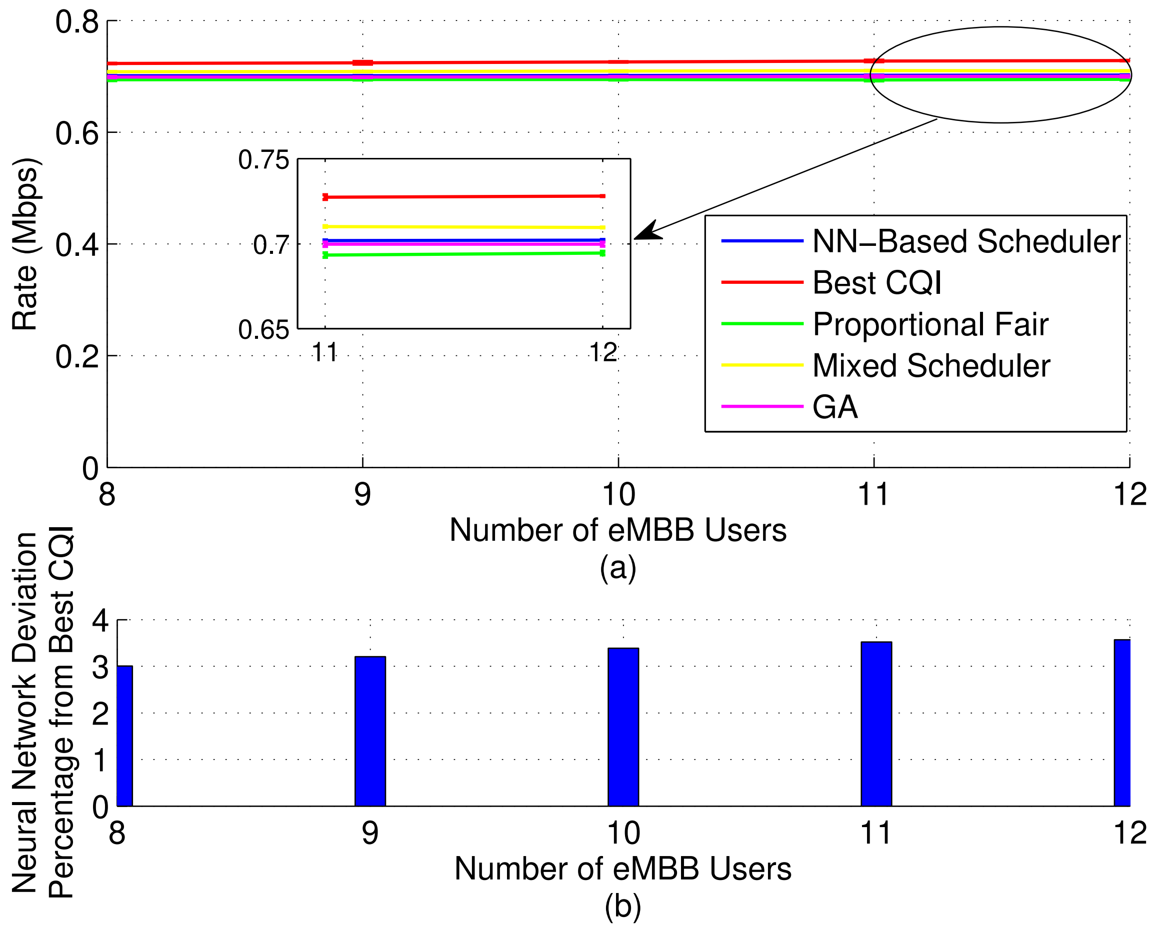


FIGURE 4.10: Comparison Among Different Scheduling Algorithms with the NN Scheduler for 75 RBs.

TABLE 4.6: Comparison between the RL and the NN schedulers

	RL	NN
Dimensions	$(E+1)^{N_f}$	$E.N_f$
Learning Rate	10^{-4}	10^{-4}
Activation Function	RELU	Sigmoid
Training Samples	None	10,000 (70-30 split)
Training Time	100,000 steps	600 epochs
Loss Function	MAE	MSE
Hidden layers	2 with 64 Neurons Each	2 with Massive Number of Neurons
Suitable Optimization Problem	Integer Combinatorial (3.14)	Binary Combinatorial (3.17)
Real-time Operation	Yes	Yes
Applicability	Low Dimensions	Low, Medium and High Dimensions
Deviation	2-2.5%	Low: 1%, Medium: 3% and High: 6%
Loss Function	MAE	MSE

To that extent, the RL scheduler is applicable only in low dimensions environments, 6 RBs till 10 RBs. On the other hand, the NN scheduler is applicable for all dimensions. In our simulated environments, we adopted the 25 RBs, 75 RBs, and 100 RBs for the NN scheduler which generated near-optimal results. It is evident that both schedulers have a real-time operation, which is the main reason for adopting the ML approach. The main advantage of the RL scheduler compared to the NN scheduler is the number of neurons in the hidden layers, however, we were able to train the largest environments using, the open-source, Google Colab GPU. It is clear that the NN scheduler has a lower deviation percentage compared to the RL scheduler. The NN scheduler achieves up to 99% of the performance of the Best CQI scheduler in the 25 RBs environment compared to 98% for the RL scheduler in the 10 RBs environment.

If we compare between the three proposed schedulers in this thesis, the mixed scheduler, the RL-based scheduler and the NN- based scheduler, each of them has certain advantages and disadvantages that need to be considered. The mixed scheduler produces the highest accumulated eMBB rate while satisfying all the system's requirements, however, it lacks the real-time operation of the ML-based schedulers. On the other hand, the RL-based schedulers have a good potential since they produce consistent deviation percentage for the adopted environments and produce good sub-optimal results in real-time. However, the dimensionality problem due to the expansion of the action space remains the main issue in the RL approach. Finally, the NN-based schedulers remains the only adopted scheduler that can produce real-time results in all the dimensions and environments adopted by the 3GPP, however, it requires a massive number of neurons per layers and the highest training time. However, the advantages of the NN-based schedulers overcomes the drawbacks of such approach especially that these drawbacks are before the deployment of such schedulers.

4.4 Chapter Summary

In this chapter, Various ML approaches are discussed; the RL approach and the NN approach. Two types of novel schedulers are implemented based on the RL and NN approaches. The results of each approach are simulated and discussed. The shortcoming of the RL approach is discussed and the NN approach is used to overcome the dimensionality issue. The results show that the RL technique approaches optimal solution with 2-2.5% deviation in any of the simulated environments which shows the robustness and reliability of the approach. However, the dimensionality problem cannot be mitigated to allow such an approach for larger dimensions environments. The NN approach is adopted and showed great promise after tuning the parameters carefully. The results show that the NN maximum deviation from the highest rate is 6% for the largest environment accepted by the 3GPP. The main shortcoming of the NN scheduler, unlike the RL scheduler, is the number of neurons in the hidden layers.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this thesis, the mixed scheduling problem for emerging traffic in 5G systems is discussed. Different solution methods are devised and their advantages and disadvantages are discussed. A comparison between different scheduling techniques in literature is done and the genetic algorithm scheduler and the mixed-scheduler, a combination between the best channel quality indicator and the proportional fair schedulers, are adopted to produce sub-optimal results. Most importantly, a mathematical model for URLLC traffic in a single cell for 5G communication is derived; to the best of our knowledge, this is the first work to mathematically model the URLLC traffic in a single cell. The derived equations aided in formulating a resource allocation optimization problem for the uplink mixed traffic in 5G systems.

Next, different ML techniques are adopted to check their applicability in different types of environments and different dimensions. The reinforcement learning based scheduler is discussed first and the reward function and the action space are defined. The dimensionality problem is discussed and explained. It is shown that the reinforcement learning based scheduler can achieve up to 98% of the performance of the optimal technique. The main drawback of the RL approach is the dimensionality problem; as the system's dimensions increase, the action space explodes and training becomes cumbersome.

Next, a neural network approach is adopted to solve the scheduling problem in a large environment to overcome the shortcomings of the reinforcement learning approach. In the largest environment, in terms of dimensions, the neural network scheduler achieved up to

94% in terms of accumulated throughput. The scheduler is tested for smaller environments, in terms of dimensions, to check its reliability for different systems and achieved up to 99% of the throughput upper bound. However, The neural network based scheduler requires very good tuning before learning, as a slight change in any of the parameters might lead to very bad results. It is important to note that the processing time of the neural network and reinforcement learning schedulers, after training, is negligible compared to the genetic algorithm scheduler and the mixed scheduler. One of the most important aspects of schedulers is the processing time, as discussed in the previous chapters, which strongly justifies our choice of using the ML approaches in general and the neural network approach in specific. It is evident that the neural network based scheduler surpasses all the other tested schedulers in different aspects. It surpasses the reinforcement learning approach in the complexity reduction and performance. In addition, it surpasses both the mixed scheduler and the genetic algorithm scheduler in the processing time.

5.2 Future Directions

In this thesis, many topics and tools are discussed which open multiple research directions. The URLLC model can be reformulated to include puncturing or soft decoding. Another approach is to use dedicated resource blocks for each set of URLLC nodes instead of broadcasting all the allocated resources to all the URLLC nodes. However, this might need extra overhead to use multi-casting instead of broadcasting. In addition, the third HARQ technique, the proactive scheme, does not have a mathematical model for a single gNB in literature till now. Having a concrete mathematical model for it might enhance our understanding of such approach.

On the other hand, the resource allocation optimization problem developed can be modified to overlap different types of traffic on the same resources using puncturing after modifying the equations of the URLLC traffic.

Since the reinforcement learning domain showed great promise in small environments, an approach can be developed to mitigate the dimensionality problem. This approach can be based on the fact that a lot of actions in the action space are useless. The action space contains all the possible actions the scheduler can take including the actions that does not satisfy the resource allocation optimization problem constraints. Allocating all the resource to the URLLC traffic is one on the actions that is included in the action space that is considered non-realistic, allocating resources to a specified set of users instead of all the users are actions that does not satisfy the resource allocation optimization problem constraints as well. Based on that, a good approach is to remove these actions to allow training. For the neural network approach, a lot of experiments can be done to tune the model parameters and find a relation between the system parameters and the learning parameters with less trial and error.

Bibliography

[1] M Carmen Lucas-Estañ, Javier Gozalvez, and Miguel Sepulcre. "On the capacity of 5G NR grant-free scheduling with shared radio resources to support ultra-reliable and low-latency communications". In: *Sensors* 19.16 (2019), p. 3575.

[2] Dave Evans. "The internet of things: How the next evolution of the internet is changing everything. 2011". In: *URL http://www. cisco. com/web/about/ac79/docs/innov/IoT IBSG 0411FINAL. pdf* (2015).

[3] Mehaseb Ahmed Mehaseb et al. "Classification of LTE uplink scheduling techniques: An M2M perspective". In: *IEEE Communications surveys & tutorials* 18.2 (2015), pp. 1310–1335.

[4] Y. Liu et al. "Analyzing Grant-Free Access for URLLC Service". In: *IEEE Journal on Selected Areas in Communications* (2020), pp. 1–1. DOI: 10.1109/JSAC.2020.3018822.

[5] Rashid Ali et al. "URLLC for 5G and Beyond: Requirements, Enabling Incumbent Technologies and Network Intelligence". In: *IEEE Access* 9 (2021), pp. 67064–67095.

[6] Hadeel Elayan et al. "Terahertz band: The last piece of RF spectrum puzzle for communication systems". In: *IEEE Open Journal of the Communications Society* 1 (2019), pp. 1–32.

[7] Richard Li et al. "Towards a new internet for the year 2030 and beyond". In: *Proc. 3rd Annu. ITU IMT-2020/5G Workshop Demo Day*. 2018, pp. 1–21.

[8] Arash Asadi and Vincenzo Mancuso. "A survey on opportunistic scheduling in wireless communications". In: *IEEE Communications surveys & tutorials* 15.4 (2013), pp. 1671–1688.

[9] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. "Deep learning in mobile and wireless networking: A survey". In: *IEEE Communications Surveys & Tutorials* 21.3 (2019), pp. 2224–2287.

[10] Nisha Panwar, Shantanu Sharma, and Awadhesh Kumar Singh. "A survey on 5G: The next generation of mobile communication". In: *Physical Communication* 18 (2016), pp. 64–84.

[11] 3GPP TS 36.213 V14.2.0. "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures". In: (2017).

[12] T. Fehrenbach et al. "URLLC Services in 5G Low Latency Enhancements for LTE". In: *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*. 2018, pp. 1–6. DOI: 10.1109/VTCFall.2018.8690663.

[13] D. Jiang et al. "Principle and Performance of Semi-Persistent Scheduling for VoIP in LTE System". In: *2007 International Conference on Wireless Communications, Networking and Mobile Computing*. 2007, pp. 2861–2864. DOI: 10.1109/WICOM.2007.710.

[14] 3GPP. *NR; NR and NG-RAN Overall description; Stage-2*. Technical Specification (TS) 38.300. Version 15.5.0. 3rd Generation Partnership Project (3GPP), Mar. 2019. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3191.

[15] Liljana Gavrilovska, Valentin Rakovic, and Vladimir Atanasovski. "Visions towards 5G: Technical requirements and potential enablers". In: *Wireless Personal Communications* 87.3 (2016), pp. 731–757.

[16] Jorge Navarro-Ortiz et al. "A survey on 5G usage scenarios and traffic models". In: *IEEE Communications Surveys & Tutorials* 22.2 (2020), pp. 905–929.

[17] Ijaz Ahmad et al. "Overview of 5G security challenges and solutions". In: *IEEE Communications Standards Magazine* 2.1 (2018), pp. 36–43.

[18] ON Yilmaz. "Ultra-reliable and low-latency 5G communication". In: *Proc. Eur. Conf. Netw. Commun.(EuCNC)*. 2016, pp. 1–2.

[19] Arjun Anand, Gustavo De Veciana, and Sanjay Shakkottai. "Joint scheduling of URLLC and eMBB traffic in 5G wireless networks". In: *IEEE/ACM Transactions on Networking* 28.2 (2020), pp. 477–490.

[20] Madyan Alsenwi et al. "eMBB-URLLC resource slicing: A risk-sensitive approach". In: *IEEE Communications Letters* 23.4 (2019), pp. 740–743.

[21] Arjun Anand and Gustavo de Veciana. "Resource allocation and HARQ optimization for URLLC traffic in 5G wireless networks". In: *IEEE Journal on Selected Areas in Communications* 36.11 (2018), pp. 2411–2421.

[22] João Pedro Battistella Nadas et al. "Performance analysis of hybrid ARQ for ultra-reliable low latency communications". In: *IEEE Sensors Journal* 19.9 (2019), pp. 3521–3531.

[23] Tengteng Ma et al. "Slicing Resource Allocation for eMBB and URLLC in 5G RAN". In: *Wireless Communications and Mobile Computing* 2020 (2020).

[24] Cong Xiong et al. "Energy-efficient design for downlink OFDMA with delay-sensitive traffic". In: *IEEE Transactions on Wireless Communications* 12.6 (2013), pp. 3085–3095.

[25] Anupam Kumar Bairagi et al. "A matching based coexistence mechanism between eMBB and uRLLC in 5G wireless networks". In: *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. 2019, pp. 2377–2384.

[26] Anupam Kumar Bairagi et al. "Coexistence Mechanism between eMBB and uRLLC in 5G Wireless Networks". In: *arXiv preprint arXiv:2003.04551* (2020).

[27] M. Y. Abdelsadek, Y. Gadallah, and M. H. Ahmed. "A Critical MTC Resource Allocation Approach for LTE Networks With Finite Blocklength Codes". In: *IEEE Transactions on Vehicular Technology* 69.5 (2020), pp. 5598–5609. DOI: 10.1109/TVT.2020.2982391.

[28] Hamza Khan et al. "Deep Learning Assisted CSI Estimation for Joint URLLC and eMBB Resource Allocation". In: *arXiv preprint arXiv:2003.05685* (2020).

[29] Nesrine Ben-Khalifa, Mohamad Assaad, and Mérouane Debbah. "Risk-sensitive reinforcement learning for URLLC traffic in wireless networks". In: *arXiv preprint arXiv:1811.02341* (2018).

[30] Madyan Alsenwi et al. "Intelligent Resource Slicing for eMBB and URLLC Coexistence in 5G and Beyond: A Deep Reinforcement Learning Based Approach". In: *arXiv preprint arXiv:2003.07651* (2020).

[31] Ali Taleb Zadeh Kasgari et al. "Experienced deep reinforcement learning with generative adversarial networks (GANs) for model-free ultra reliable low latency communication". In: *IEEE Transactions on Communications* (2020).

[32] P. Popovski et al. "5G Wireless Network Slicing for eMBB, URLLC, and mMTC: A Communication-Theoretic View". In: *IEEE Access* 6 (2018), pp. 55765–55779. DOI: 10.1109/ACCESS.2018.2872781.

[33] R. Kassab, O. Simeone, and P. Popovski. "Coexistence of URLLC and eMBB Services in the C-RAN Uplink: An Information-Theoretic Study". In: *2018 IEEE Global Communications Conference (GLOBECOM)*. 2018, pp. 1–6. DOI: 10.1109/GLOCOM.2018.8647460.

[34] Renato Abreu et al. "On the multiplexing of broadband traffic and grant-free ultra-reliable communication in uplink". In: *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. IEEE. 2019, pp. 1–6.

[35] J. Thota and A. Aijaz. "On Performance Evaluation of Random Access Enhancements for 5G uRLLC". In: *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. 2019, pp. 1–7. DOI: 10.1109/WCNC.2019.8885815.

[36] C. Wang et al. "Performance Evaluation of Grant-Free Transmission for Uplink URLLC Services". In: *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. 2017, pp. 1–6. DOI: 10.1109/VTCSpring.2017.8108593.

[37] R. Abreu et al. "Power control optimization for uplink grant-free URLLC". In: *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. 2018, pp. 1–6. DOI: 10.1109/WCNC.2018.8377284.

[38] N. H. Mahmood et al. "Uplink Grant-Free Access Solutions for URLLC services in 5G New Radio". In: *2019 16th International Symposium on Wireless Communication Systems (ISWCS)*. 2019, pp. 607–612. DOI: 10.1109/ISWCS.2019.8877253.

[39] Zhiyi Zhou et al. "Resource allocation for uplink grant-free ultra-reliable and low latency communications". In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. IEEE. 2018, pp. 1–5.

[40] Thomas Jacobsen et al. "System level analysis of K-repetition for uplink grant-free URLLC in 5G NR". In: *European Wireless 2019; 25th European Wireless Conference*. VDE. 2019, pp. 1–5.

[41] Helin Yang et al. "Deep Reinforcement Learning Based Massive Access Management for Ultra-Reliable Low-Latency Communications". In: *arXiv preprint arXiv:2002.08743* (2020).

[42] L. Feng et al. "Dynamic Resource Allocation With RAN Slicing and Scheduling for uRLLC and eMBB Hybrid Services". In: *IEEE Access* 8 (2020), pp. 34538–34551. DOI: 10.1109/ACCESS.2020.2974812.

[43] Mohammed Y Abdelsadek, Yasser Gadallah, and Mohamed H Ahmed. "Matching-Based Resource Allocation for Critical MTC in Massive MIMO LTE Networks". In: *IEEE Access* 7 (2019), pp. 127141–127153.

[44] Renato Abreu et al. "System level analysis of eMBB and grant-free URLLC multiplexing in uplink". In: *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*. IEEE. 2019, pp. 1–5.

[45] Mohammed Y Abdelsadek, Yasser Gadallah, and Mohamed H Ahmed. "Resource Allocation of URLLC and eMBB Mixed Traffic in 5G Networks: A Deep Learning Approach". In: *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE. 2020, pp. 1–6.

[46] Gaith Rjoub et al. "Deep smart scheduling: A deep learning approach for automated big data scheduling over the cloud". In: *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*. IEEE. 2019, pp. 189–196.

[47] Wei Cui, Kaiming Shen, and Wei Yu. "Spatial deep learning for wireless scheduling". In: *IEEE Journal on Selected Areas in Communications* 37.6 (2019), pp. 1248–1261.

[48] Mohan Sharma and Ritu Garg. "An artificial neural network based approach for energy efficient task scheduling in cloud data centers". In: *Sustainable Computing: Informatics and Systems* 26 (2020), p. 100373.

[49] Zubair Khalid et al. "A novel load scheduling mechanism using artificial neural network based customer profiles in smart grid". In: *Energies* 13.5 (2020), p. 1062.

[50] Jinsung Lee et al. "PERCEIVE: Deep learning-based cellular uplink prediction using real-time scheduling patterns". In: *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*. 2020, pp. 377–390.

[51] Shuochao Yao et al. "Scheduling real-time deep learning services as imprecise computations". In: *2020 IEEE 26th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. IEEE. 2020, pp. 1–10.

[52] Mohan Sharma and Ritu Garg. "An artificial neural network based approach for energy efficient task scheduling in cloud data centers". In: *Sustainable Computing: Informatics and Systems* 26 (2020), p. 100373.

[53] Praveena Akki and V Vijayarajan. "Energy Efficient Resource Scheduling Using Optimization Based Neural Network in Mobile Cloud Computing." In: *Wireless Personal Communications* 114.2 (2020).

[54] Mohamed Hadi Habaebi et al. "Comparison between scheduling techniques in long term evolution". In: *IIUM Engineering Journal* 14.1 (2013).

[55] Gianni D'Angelo and Francesco Palmieri. "GGA: A modified genetic algorithm with gradient-based local search for solving constrained optimization problems". In: *Information Sciences* 547 (2021), pp. 136–162.

[56] Maria Amélia Lopes Silva et al. "A reinforcement learning-based multi-agent framework applied for solving routing and scheduling problems". In: *Expert Systems with Applications* 131 (2019), pp. 148–171.

[57] Mohammed Sharafath Abdul Hameed and Andreas Schwung. "Reinforcement learning on job shop scheduling problems using graph networks". In: *arXiv preprint arXiv:2009.03836* (2020).

[58] Nina Mazyavkina et al. “Reinforcement learning for combinatorial optimization: A survey”. In: *Computers & Operations Research* (2021), p. 105400.

[59] Nguyen Cong Luong et al. “Applications of deep reinforcement learning in communications and networking: A survey”. In: *IEEE Communications Surveys & Tutorials* 21.4 (2019), pp. 3133–3174.

[60] Yonathan Efroni et al. “Multiple-step greedy policies in approximate and online reinforcement learning”. In: *NeurIPS*. 2018.

[61] Michael Wunder, Michael L Littman, and Monica Babes. “Classes of multiagent q-learning dynamics with epsilon-greedy exploration”. In: *ICML*. 2010.

[62] Kavosh Asadi and Michael L Littman. “An alternative softmax operator for reinforcement learning”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 243–252.

[63] Jim Zyren and Wes McCoy. “Overview of the 3GPP long term evolution physical layer”. In: *Freescale Semiconductor, Inc., white paper* 7 (2007), pp. 2–22.

[64] Francois Chollet. “The limitations of deep learning”. In: *Deep Learning With Python* (2017).

[65] Andrej Krenker, Janez Bešter, and Andrej Kos. “Introduction to the artificial neural networks”. In: *Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech* (2011), pp. 1–18.