RESOURCE MANAGEMENT AND BACKHAUL ROUTING IN MILLIMETER-WAVE IAB NETWORKS USING DEEP REINFORCEMENT LEARNING

by

Malcolm Makomborero Sande

Submitted in partial fulfillment of the requirements for the degree
Philosophiae Doctor (Electronic Engineering)

in the

Department of Electrical, Electronic and Computer Engineering
Faculty of Engineering, Built Environment and Information Technology

UNIVERSITY OF PRETORIA

February 2023

SUMMARY

RESOURCE MANAGEMENT AND BACKHAUL ROUTING IN MILLIMETER-WAVE IAB NETWORKS USING DEEP REINFORCEMENT LEARNING

by

Malcolm Makomborero Sande

Promoter(s):	Prof. Sunil Maharaj
Department:	Electrical, Electronic and Computer Engineering
University:	University of Pretoria
Degree:	Philosophiae Doctor (Electronic Engineering)
Keywords:	Congestion control, deep reinforcement learning, integrated access and backhaul, millimeter wave, recursive discrete choice model, resource allocation, routing, user satisfaction.

The increased densification of wireless networks presents complex challenges for mobile network operators, which include but are not limited to site acquisition and fiber deployment costs. These challenges, coupled with the maturity of millimeter wave (mm-wave) communication and its suitability for wireless backhaul links have led to the development of integrated access and backhaul (IAB) for the fifth generation (5G) networks. This research work investigated the applicability of machine learning strategies, in particular, deep reinforcement learning (DRL), for solving resource management and backhaul routing problems in mm-wave IAB networks.

Realizing network context and managing network resources based on real-time parameters are attractive approaches to address the challenge of congestion in dense wireless networks. In this work, a resource management solution that aims to avoid congestion for access users in an IAB network is presented. The proposed solution applies DRL to learn an optimal policy that aims to achieve effective resource allocation whilst minimizing congestion and satisfying the user requirements. The novelty of the solution is the exploration and development of two learning algorithms that place emphasis on congestion control and user satisfaction in maximizing throughput performance. With regard to

computational complexity, the individual learning algorithm was shown to have a lower complexity whereas the complexity of the cooperative learning approach was shown to be the same as that of the baseline approach. In terms of throughput performance, the two algorithms were found to have more than 50% better performance when compared with a baseline algorithm. In terms of user satisfaction, the two proposed algorithms provided at least 12% improvement when compared with the baseline solution. From the results, the nearest neighbor cooperative learning approach was found more suitable for the multi-hop mm-wave IAB networks because its throughput has a good correlation with the congestion rate.

Backhaul availability and backhaul scalability are prominent challenges that are experienced in poor wireless channel conditions that are typical for mm-wave frequencies. Such conditions may lead to high energy consumption and packet losses, as such, it is essential that multi-hop IAB networks have increased robustness to backhaul failure. Existing solutions for data routing in mm-wave IAB networks do not consider ways in which resource exhaustion at a serving BS can be minimised. As such, this work proposed a framework that avoids resource exhaustion and is of adapting to different learning mechanisms while satisfying user requirements in real-time. To this end, this research work presents a DRL-based backhaul adaptation scheme that is controlled by the access load. The DRL strategy leverages a recursive discrete choice model (RDCM), which incorporates choice aversion from prospect theory. Simulation results where the proposed DRL-RDCM algorithm was compared to the conventional DRL and the generative model-based learning algorithms showed that the proposed scheme provides better throughput and delay performance. With regard to computational complexity, the proposed DRL-RDCM algorithm was shown to have the same complexity as that of conventional DRL. In terms of cumulative throughput, the proposed DRL-RDCM scheme showed an average of 0.2 Gbps higher achievement when compared with conventional DRL. On the other hand, the delay performance of the DRM-RDCM solution was shown to be significantly higher than the two baselines for a low number of IAB nodes and as the number of IAB nodes increased, the performance matched that of the conventional DRL solution but these two continued to provide more than 20% better delay performance when compared with the generative model-based learning approach. Overally, it was observed that machine learning, in particular DRL, can be leveraged to improve throughput performance in mm-wave IAB networks.

ACKNOWLEDGEMENTS

I would like to acknowledge and express my sincerest gratitude to the following individuals and groups/departments for their invaluable contribution during the course of this research work:

- My supervisor, Prof. Sunil Maharaj, for his expert guidance and support.
- My friend and colleague, Dr. Mduduzi Hlophe, for his unending technical and emotional support.
- The Sentech Chair in Broadband Wireless Multimedia Communications (BWMC) and the Postgraduate Department at University of Pretoria for the resources and financial support.
- My fellow BWMC students for their technical advice and encouragement.
- My wife Ashiella, my mother, my sisters, and my other family and friends for their unending moral support and encouragement.

LIST OF ABBREVIATIONS

3GPP	third generation partnership project
5G	the fifth generation
5G PPP	5G Infrastructure Public Private Partnership
AAS	advanced antenna system
ACM	adaptive coding and modulation
AI	artificial intelligence
AMPS	advanced mobile phone service
AP	aceess point
BAP	backhaul adaptation protocol
C-RAN	cloud-radio access network
CA	carrier aggregation
CAPEX	capital expenditure
CCO	coverage and capacity optimization
CDM	code-division multiplexing
CDMA	code division multiple access
CMDP	constrained Markov decision process
CoMP	coordinated multi-point
CR	cognitive radio
CRN	cognitive radio network
CSI	channel state information
CU	central unit
D2D	device-to-device
DBN	deep belief network
DDQN	double deep Q-network
DL	deep learning
DNN	deep neural network
DQN	deep Q-network
DRL	deep reinforcement learning
DTAC	dynamic traffic admission control
DU	distributed unit

EM	expectation maximization
eMBB	enhanced mobile broadband
EMF	electromagnetic fields
EPC	evolved packet core
GM	Gaussian mixture
GMBL	generative model-based learning
gNB	next-generation NodeB
HBDS	heuristic backhauling and dynamic sleeping
HetNet	heterogeneous network
HQF	highest-quality-first
HMM	hidden Markov model
HSPA	high speed packet access
HSVM	hierarchical support vector machines
IAB	integrated access and backhaul
ICA	independent component analysis
IMT	International Mobile Telephone
ITU	International Telecommunications Union
KDN	knowledge defined networking
KNN	*k*-nearest neighbor
KPI	key performance indicators
LOS	line-of-sight
LSTM	long short-term memory
LTE	long-term evolution
LTE-A	LTE-Advanced
M2M	machine-to-machine
MAB	multi-armed bandit
MBS	master base station
MDP	Markov decision process
MEC	mobile edge computing
MIMO	multiple-input multiple-output
MLB	mobility load balancing
MLP	multi-layer perceptron
MLR	maximum-local-rate

mMTC	massive machine-type communications
MNO	mobile network operator
MP-MAB	multi-player multi-armed bandit
MRO	mobility robustness optimization
MT	mobile terminations
MU-MIMO	multi-user multiple input multiple output
NFV	network functions virtualization
NLOS	non-line-of-sight
NMT	Nordic mobile telephone
NN	neural network
NOMA	non-orthogonal multiple access
NR	new radio
NSA	non-standalone
OPEX	operating expenditure
OSA	opportunistic spectrum access
PA	position-aware
PCA	principal component analysis
PDCP	packet data convergence protocol
POMDP	partially observable Markov decision process
PU	primary user
QAM	quadrature amplitude modulation
QL	Q-learning
QoE	quality of experience
QoS	quality of service
RA	resource allocation
RAN	radio access network
RAT	radio access technology
ReLU	rectified linear unit
RF	radio-frequency
RL	reinforcement learning
RNN	recurrent neural network
RRC	radio resource control

SA	standalone
SBS	slave base station
SCMA	sparse code multiple access
SCTP	stream control transport protocol
SDAP	service data adaptation protocol
SDN	software defined networking
SGD	stochastic gradient descent
SINR	signal-to-interference-plus-noise ratio
SNR	signal-to-noise ratio
SON	self-organizing network
SU	secondary user
SVM	support vector machines
TAC	traffic admission control
TD	temporal-difference
TDMA	time division multiple access
UAB	unsynchronized access-backhaul
UAV	unmanned aerial vehicle
UE	user equipment
UMTS	universal mobile telecommunications service
UPN	user provided network
URLLC	ultra-reliable low latency communications
VPI	value of perfect information
WF	wired-first
WLAN	wireless local area network
WRAN	wireless regional area network

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

The fifth generation (5G) of mobile and wireless communication networks has been envisioned to support a much broader spectrum of quality of service (QoS) profiles compared to its predecessor, i.e., the fourth generation (4G) or long–term evolution (LTE) and its variants such as the LTE-Advanced. The proposed improvements will be enabled by new features and technologies that affect both the core network and the air interface. The 5G networks come with requirements of high data rates, high reliability and low latency transmissions for mobile and wireless communication systems, and they are motivated by a multitude of wireless standards [1]. With the global data rate demands estimated to continuously increase by an annual rate of more than 30% until 2024 [2], effective solutions for handling such relentless QoS demands are required. Recent reports reveal that the global data traffic increased by 82% between 2018 and 2019 due to the escalating smartphone ownership and the increased average data volume per subscription, owing to the increase in video traffic [3].

The 5G networks are currently under intense deployment all over the world with the objective of satisfying all the use cases. The initial deployment of the 5G new radio (NR) under the non-standalone (NSA) architecture is developmental, that is, it operates by leveraging the existing 4G network infrastructure and 4G core [4]. In the NSA deployment scenario, the network access is overlaid on the existing 4G/LTE network core through a feature known as dual connectivity. The objective here is to continue delivering broadband services with higher data speeds and better coverage while enabling seamless migration from 4G standalone to 5G standalone (SA) [5]. However, despite the many benefits and advantages of having a fully-fledged 5G SA architecture, there is one challenge that cannot be underestimated. The operation in millimetre wave (mm-wave) frequency bands is beneficial in terms of providing more bandwidth and achieving high data rates, as well as improving the quality of communications in complex environments, more especially enabling the 5G use cases; however, higher frequencies have less penetration capabilities, which equate to limited network coverage. As such, to

enhance wireless or cellular network coverage in higher frequency operation, a dense deployment of small cells is required.

The main challenges associated with network densification are site acquisition and fiber deployment costs [6]. According to the laws of physics, fiber technology has significant inherent bandwidth-carrying capabilities and it has always been the future-proof solution in terms of backhaul technology. However, wireless or microwave backhaul is the most used backhaul technology due to a combination of its capability and relative ease of deployment. The Third Generation Partnership Project (3GPP) conducted a study on IAB for 5G NR with the aim of evaluating potential solutions for its efficient operation [7], [8], and since then there has been a great deal of research interest in IAB for 5G mm-wave networks. As a result, IAB was standardized for 5G NR mm-wave operation in 3GPP's Release 16 technical report, which was completed in the first quarter of 2020 [9]. To this effect, the 3GPP has endorsed a multi-hop IAB architecture for operation in mm-wave frequencies. As a result, the increasing maturity of mm-wave communications and their suitability for wireless backhaul links has led to most mobile network operators (MNOs) heavily relying on mm-wave bands for wireless backhaul solutions.

Backhaul infrastructure is an essential component of mobile wireless communication networks, which offers data transportation to and from the core of the network. In traditional networks, that is in the 2G and 3G, backhaul refers to the transport network that connects the radio access network (RAN) to the core of the network. However, with the advent of small cells in 4G networks, the fronthaul concept, which is a transport network connecting the small cell BSs to the macrocell BS has emerged - albeit not in the context of this work [10]. With a densified network, more stringent requirements will be imposed in terms of data delivery, and mobile backhaul will become even more important. Backhaul provides connectivity between small cells, macro cells, core networks, and possibly between numerous gateway nodes. IAB is thus envisaged as the most flexible and cost-effective backhaul technology for wireless and mobile networks in 5G and it has been found to be lucrative for practical implementations, particularly in the mm-wave frequencies, where interference effects are subdued due to large bandwidths and beamforming techniques [11].

1.1 PROBLEM DEFINITION

1.1.1 Problem Statement

Increasing network densification through the deployment of small cells while providing traditional fiber backhaul access to every access point becomes difficult and costly [12]. On the other hand, the availability of large amounts of spectrum in mm-wave frequencies makes it possible to achieve wireless backhaul solutions for small cell access points whilst having sufficient bandwidth for access for the user equipment (UEs) [8]. As a result, the 3GPP has endorsed a multi-hop IAB architecture for operation in mm-wave frequencies. The IAB networks require accurate and adaptive resource management schemes because of the directional transmissions, device heterogeneity, and harsh propagation conditions that are experienced in the 5G mm-wave wireless networks. The latter, in particular, characterizes the operations of such networks, thereby resulting in transmission links with varying link reliability. For this reason, the traditional optimization techniques do not provide the best performance in these conditions. Despite the provision of new spectrum bands in higher frequencies, achieving vast network capacity, low latency, and reliability are still very challenging since these bands have a lack of penetration ability and are prone to blockages. As a result, the deployment of IAB in these bands should be assisted by relevant learning schemes that overcome these drawbacks and prevent service disruption.

1.1.2 Context of the Problem

The typical topology of an IAB network is as illustrated in Fig. 1.1. The wireless backhauling shown in Fig. 1.1 is assumed to be based on mm-wave commmunications, that is frequencies above 10 GHz, and it is constrained to line-of-sight (LOS) propagation conditions. The donor node, also known as the gateway or master base station (MBS), is the node that is connected to the rest of the network in the conventional way (through fiber). The donor node serves both the IAB nodes and the UEs that are connected to it.

The standard for IAB networks that was specified in 3GPP Rel-16 [9] is based on the functional split architecture that was introduced in Rel-15 [13]. In 3GPP Rel-15, the next-generation NodeB (gNB) comprises a distributed unit (DU) and a central unit (CU). The DU terminates the lower layer protocols, that is, the radio link control (RLC), the medium access control (MAC), and the physical layer, while the CU terminates the upper layer protocols, that is, the packet data convergence protocol (PDCP) and the radio resource control (RRC). The user plane and control plane protocol stack of an IAB network, as defined in 3GPP Rel-16, is shown in Fig. 1.2.

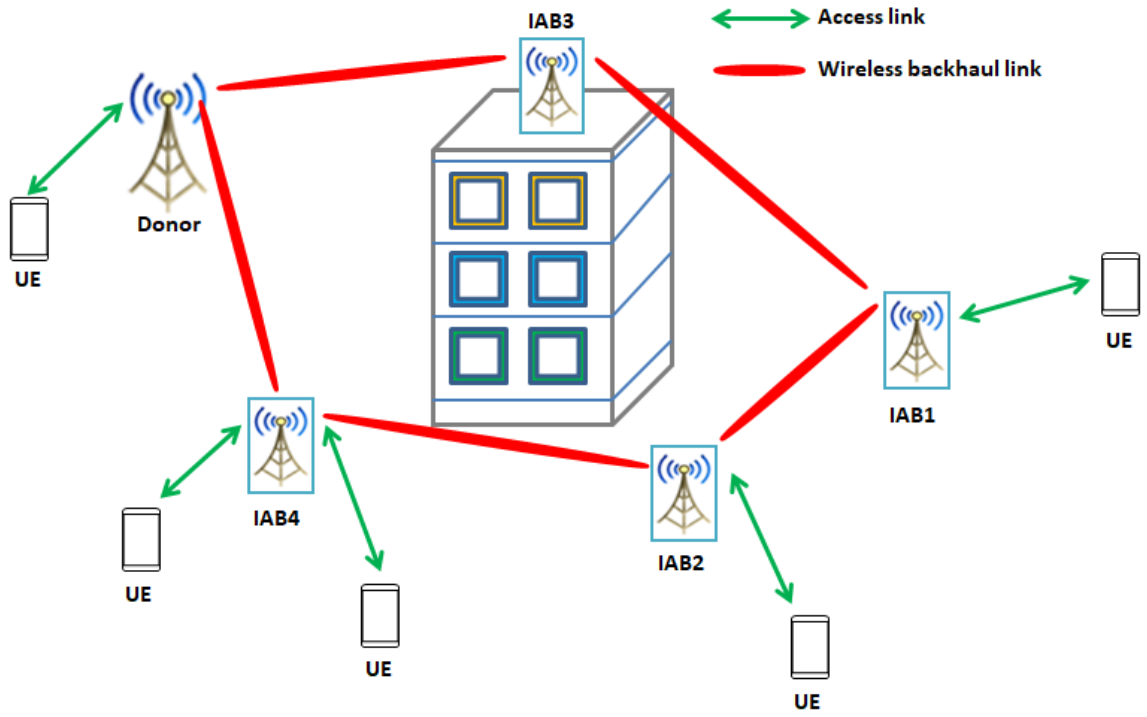

Figure 1.1. Typical topology of IAB network connectivity, illustrating the wireless backhaul among IAB nodes and their access connections with UEs.

In the control plane, the RRC and the PDCP are terminated at the UE and the control plane part of the CU (CU-CP) as shown in Fig. 1.3. The corresponding packets are transported over an F1-C interface. The F1-C is realised via a set of stream control transport protocol (SCTP) associations between the CU-CP and the DU part of the IAB node serving the UE. It is worth noting that the protocol architecture of an IAB node is transparent to the UE, which means that it cannot differentiate between that of a donor or an IAB node. The backhaul adaptation protocol (BAP) is a new protocol in routing, which is responsible for the forwarding of packets in the immediate hops between the donor DU and the access IAB node. In this protocol, the donor node configures each IAB node with a unique BAP ID. Thus, for the downlink packets, the donor DU inserts a BAP routine ID on the packet it is forwarding to the next hop. This is the BAP ID of the access IAB node serving the UE and a path identifier, just in case there are several possible paths to reach the access IAB node.

The multi-hop capabilities of the IAB network help with providing more range, especially when operating in mm-wave frequency bands due to the limited range. In addition, the multi-hop functionality provides the capability to hop around fixed obstacles such as seasonal foliage that may impact the

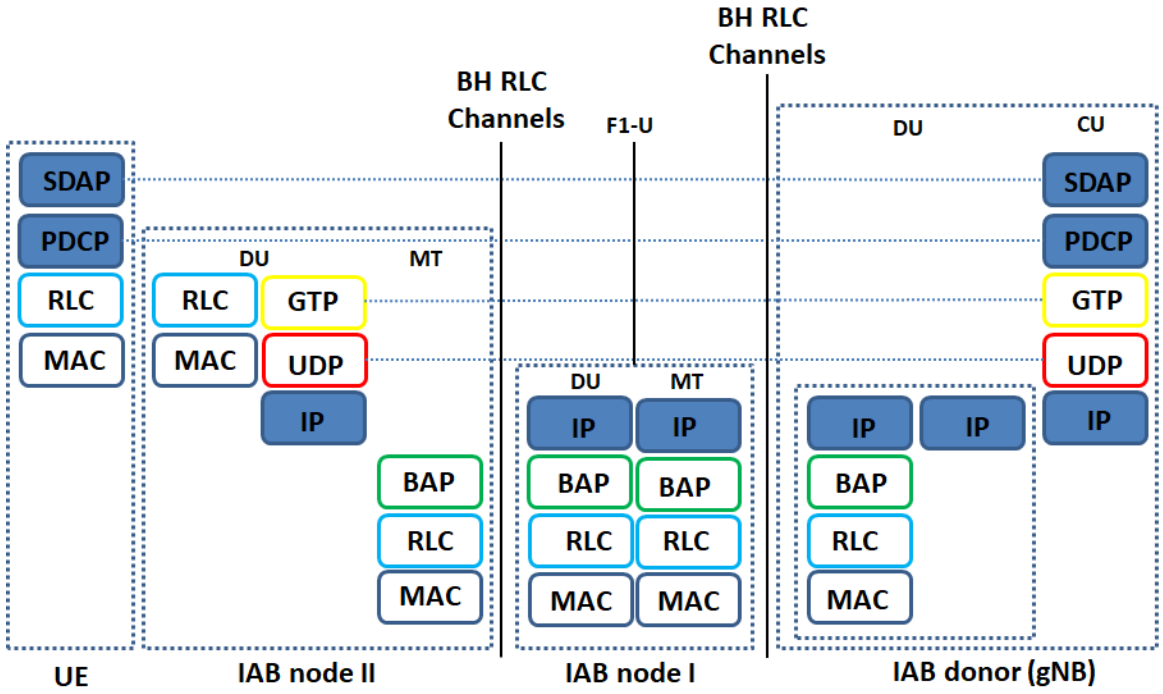

Figure 1.2. The user plane protocol stack of an IAB network, showing the links of the communication protocols between the donor node and the UE with two hops, according to 3GPP Rel-16 [14].

signal. In terms of the technical aspects relating to spectrum usage, the IAB networks are flexible, that is, they support both in-band (using the same frequency bands) and out-of-band [15].

For high spectral efficiency, the in-band approach is desirable; however, it requires interference mitigation mechanisms such as multiple-input multiple-output (MIMO) beamforming. The in-band wireless backhaul approach also offers reduced hardware costs from an implementation perspective. Traditionally, the in-band implementation was restricted to synchronised access-backhaul (SAB), however, since an MBS needs more backhaul slots compared to the SBSs, the SAB approach cannot provide efficient resource allocation in self-backhauled networks [16]. Thus, unsynchronised access-backhaul (UAB), which is also known as IAB, has been developed. In IAB, access and backhaul links need not be scheduled on orthogonal resource blocks, which allows a small cell BS to use unscheduled backhaul slots for its access. Although IAB presents increased interference in the backhaul links, it has been found attractive for practical implementations in 5G networks particularly in the mm-wave frequencies where the interference effects are subdued [11].

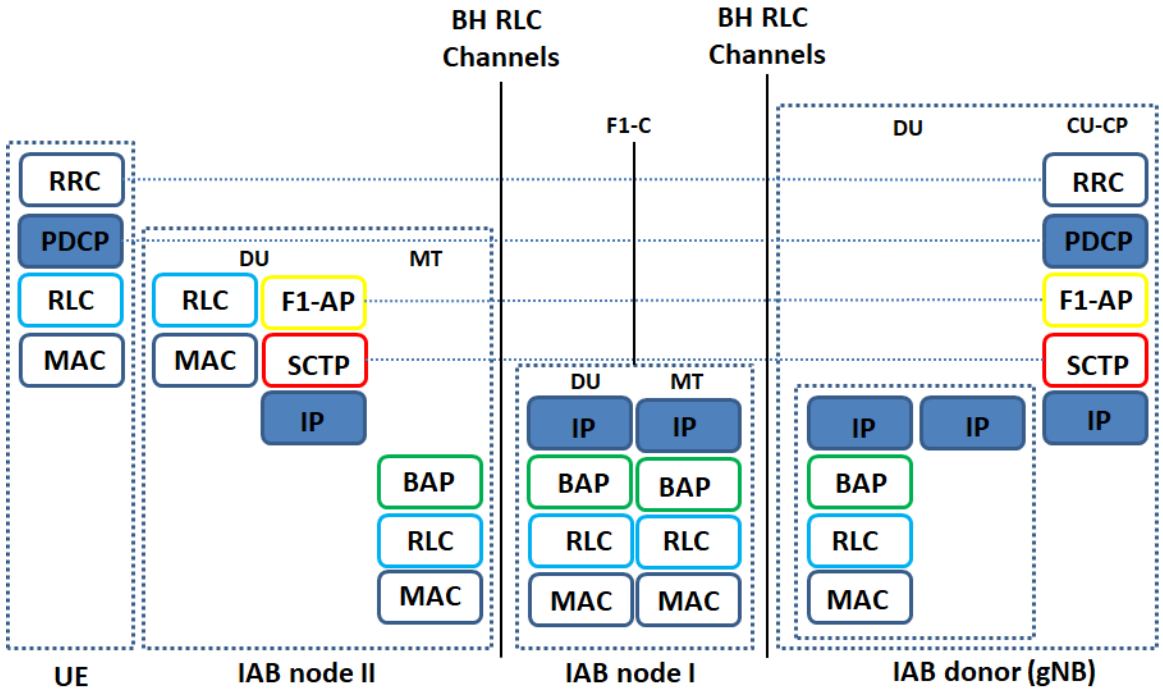

Figure 1.3. The control plane protocol stack of an IAB network for the user plane shown in Fig. 1.2.

1.1.3 Research Gap

Matching the demand for resources, that is the load, to the supply of resources, which is the capacity, is a basic problem across wireless networks. Mobile communication operators face a challenge of providing the required access and backhaul for non-uniform mobile voice and data transmission demands from users with varying quality of experience (QoE) requirements. In addition, the various mobile users may be situated in different environments such as urban, rural, underground, and fast-moving vehicles. Network modelling and optimization for smart devices in 5G networks, which are capable of jointly using many different frequency bands and communication technologies in order to achieve the required rate and latency performance, is very challenging [17]. As a result, the analysis of network characteristics such as user association, resource allocation (RA) and data routing using the received signal-to-interference-plus-noise ratio (SINR) as the base metric is not suitable for the heterogeneous 5G networks.

This research work considered other contextual information such as the load on the serving and the neighboring BSs for machine learning-based user access, resource management, and efficient backhaul routing in mm-wave IAB networks. There are systems that support the collection and dissemination

of context and applications that adapt to changing context that have been developed and proposed in literature. The users of smart devices in current and future networks are faced with diverse interfaces that are used in diverse environments. This is a step towards the realisation of a ubiquitous computing paradigm whose integration into IAB systems is a requisite. However, there are still missing pieces that are necessary to achieve this ubiquitous computing vision. The resource allocation problems in IAB have been extensively studied and solved by formulating them as optimization problems. Most of the optimization methods need accurate or complete network information, such as channel state information (CSI), which are practically challenging to obtain in stochastic and dynamic systems such as in the next-generation mobile and wireless networks. If achieved, the solutions usually come with increased costly computational complexity. Furthermore, the network dynamics are seldom addressed and many solutions to the optimization problem only apply to a snapshot of the network or are only valid in a specific network architecture. Due to their network model dependence, such models are inappropriate for highly complex and time-varying scenarios. As such, in this work, it is proposed to incorporate machine learning models to address these drawbacks of classic optimization approaches.

The user data requirements for current and future generation networks, which are stochastic and bursty in nature, present a challenge of unbalanced network traffic distribution. In addition, the dynamics of wireless cellular environments, such as fading, random user mobility, path loss effects and shadowing make it difficult to rely on a model of the network environment for solving optimization problems [18]. On the other hand, the design of efficient IAB networks that satisfy the 3GPP performance requirements is still an open research challenge. One of the main challenges of 5G wireless backhaul is the realisation of high capacity non-line-of-sight wireless backhaul links, which have a low footprint, low power consumption, and are quick to install [19]. Most of the research contributions on RA problems in the IAB networks sought to optimally allocate a fixed demand for resources by the UEs, whose performance degrades with increasing congestion. This approach overlooks the inseparable coupling of the demand and the cost for a resource. This coupling leads to “the tragedy of the commons” [20], where the performance limitation along a path is controlled by its most congested edge. To address this issue, part of this research work aimed to maximize the access capacity for the associated UEs of a congested SBS in an IAB network, while dynamically reserving some resources for the wireless backhaul traffic.

Despite the provision of new spectrum bands in higher frequencies, achieving vast network capacity,

low latency, and reliability are still very challenging since these bands are prone to blockages and the lack of penetration ability. As a result, the deployment of these bands must be assisted by relevant architectures that overcome these drawbacks and prevent service disruption. To address the problem, there has been a plethora of research contributions from both academia and standardizing bodies that aim to enhance the operation of the 5G NR. However, lifting mobile broadband to the next level still remains a challenge because the UEs as well as user behaviour are some of the problems that need to be addressed to realize a multi-functional 5G NR. As a result, reinforcement learning (RL) and deep reinforcement learning (DRL) have been shown to be the most promising strategies to enable a multi-functional 5G communication system.

Another key problem that has been identified in recent works is that the authors usually envision diverse scenarios for the same system model application, without determining the influence of traffic-related characteristics on the latency. Because the traffic offered by the mobile devices has an effect on the transmission rate, latency and energy efficiency, it should be considered in all the various network optimization solutions.

Solving RA problems in wireless networks has been traditionally carried out using optimization-based approaches, which can provide optimal solutions in quasi-static scenarios. However, in the dynamic mm-wave access network environments, it is practically infeasible to achieve near-optimal solutions on-the-fly using traditional optimization methods because of their time-consuming algorithms. This motivates the consideration of RL strategies, which can provide solutions that learn to capture the intrinsic irregularities of the dynamic environment to provide a robust network model in realistic scenarios. The disadvantage of the RL-based solutions is that they can, however, get stuck into local optima. Thus, a hybrid approach that realises the strengths of both optimization and RL techniques is required.

Prior research works in IAB networks have highlighted that as BSs need to multiplex access and backhaul resources, this may lead to excess buffering. This may result in high latency and low throughput as the main consequences, when sub-optimal resource partitioning is selected, which renders traditional optimization techniques less effective in IAB resource management. Considering that power control extends the battery life of UEs by ensuring that they transmit at the minimum possible power levels while achieving the required QoS, different approaches have been proposed to properly model the power allocation by using the utility functions they seek to maximize. From

observing most research contributions in this area, it is safe to mention that the focus is mainly on UE and MBS power allocation, while small cell BS power allocation is least explored.

1.2 RESEARCH OBJECTIVES AND QUESTIONS

1.2.1 Research Objectives

In light of the aforementioned challenges, it is essential to attempt to solve the challenges faced in IAB networks by combining traditional optimization approaches with learning strategies. To tackle the access and backhaul resource management problem, RA solutions need to be designed with three goals in mind, which serve as the main objectives of this work:

(i) **Flexibility**: To easily adapt RA strategies to different use case requirements that come with different traffic classes.

(ii) **Alignment with 3GPP specifications**: For IAB networks to comply with 3GPP specifications, the RA framework should rely on information that can actually be exchanged and reported in a 3GPP deployment. This means that the proposed IAB solution(s) have to be developed using appropriate multi-hop IAB models that are based on the 3GPP specifications.

(iii) **Low Complexity**: If an algorithm is to be developed, it should have lower complexity whilst achieving equivalent or even better performance when compared to existing algorithms.

Based on the above guidelines, the main aim of the research work was accomplished through the following set of specific objectives:

- To develop an appropriate multi-hop IAB model based on the 3GPP specifications;
- To formulate typical resource allocation and user association problems for the proposed IAB model;
- To investigate the performance of the access network under constant backhaul traffic;
- To investigate the performance of backhaul adaptation under constant and varying access traffic;
- To compare the performance of the solutions developed with other applicable solutions that are available in literature.

1.2.2 Research Questions

The research sought to find answers to the following questions.

1. Which network models are best suited to smart devices in 5G mm-wave IAB networks?
2. What are the main challenges when modelling mm-wave IAB networks, and how can they be addressed?
3. What is the most efficient approach to solve context-aware resource management and user satisfaction problems, as well as performing adaptive traffic admission control and backhauling in mm-wave IAB network scenarios?
4. Which performance evaluation techniques are suitable for the proposed IAB network model?
5. Can machine learning techniques be leveraged for enabling context-aware resource management and user satisfaction satisfaction user QoE requirements in mm-wave IAB network scenarios?
6. Can machine learning techniques also be leveraged to effectively perform traffic admission control and backhaul routing mm-wave IAB networks?
7. Are there other comparable RL-based or DRL-based solutions that are available in literature?

1.3 HYPOTHESIS AND APPROACH

1.3.1 Research Hypothesis

Considering the intelligent solutions that have been proposed to address the challenges faced in mm-wave IAB networks, the application of artificial intelligence (AI) strategies is currently being progressively explored. The optimization for context-aware resource management and adaptive backhaul routing in mm-wave IAB networks using DRL approaches can provide more efficient solutions compared to using RL strategies.

While BSs are becoming more energy efficient, increasing packet arrival rates tend to outweigh this achievement. In addition, the continuous operation of all SBSs in highly dense IAB networks may increase the operational expenditure (OPEX) for MNOs. Recent research has proposed energy harvesting schemes that integrate smart grid deployments and renewable energies such as solar and wind, thus providing green communication networks [21]. However, in as much as these renewable energy solutions may reduce the OPEX for MNOs and contribute towards green communication networks, they generally require high capital expenditure (CAPEX). This research work considered other smart ways of improving energy efficiency in IAB networks through the following hypothetical

questions. How can a system learn how to handle dynamic backhaul traffic without compromising the access QoS? How much of a role do transmission delays and buffer size play in the power management and throughput performance in IAB networks? Can machine learning techniques be leveraged to improve the energy efficiency and throughput performance in constrained IAB networks?

1.3.2 Research Execution

The research work was comprised of the following tasks.

- An extensive literature study on 5G NR, cognitive radio technology, and machine learning techniques was carried out.
- A network model for an IAB network operating in the NSA mode of 5G NR was developed. The developed model incorporated DRL for best action selection, where the SBSs are taken as the learning agents.
- The optimization problems for the appropriate models were formulated as Markov decision processes.
- The DRL algorithms to solve the formulated problems were developed and implemented in the simulation software.
- Lastly, the system performance of the proposed solutions in the developed models was analyzed from the simulations with comparisons to the baseline solutions, which enabled conclusions to be drawn.

1.4 RESEARCH CONTRIBUTIONS AND OUTPUTS

1.4.1 Access Congestion Avoidance and Resource Management in IAB Networks

Since the traditional adaptive traffic signal control cannot resolve the type of congestion that is experienced in dense mm-wave IAB networks, realizing context in the network and adapting content based on real-time parameters is an attractive approach. As part of the contributions of the research work, an optimization problem that aimed to maximize the transmission throughput of all UEs subject to environmental context such as traffic load and transmission power was formulated. Here, the objective was to optimize the policy that enables the system to provide better QoS to the UEs whilst adhering to the power constraints. The problem was solved using a context-aware solution for access resource management in an IAB network model using a DRL strategy, and details of the contributions of this part of the work are as follows:

- **Adaptive congestion control:** The impact of congestion on the link layer behavior of an access IAB network is evaluated by defining the levels of congestion as the exact values of system utilisation. In this work, an SBS's traffic load was modelled using an M/G/1 queuing model with an ergodic arrival process. The optimization problem was then formulated as a QoS maximization problem that emphasized the estimation of the SBS load as context. In the implemented system model, the agent aimed to learn the context-related behaviors that enabled it to take appropriate actions within a reasonable time by efficiently leveraging the feedback from the output.
- **DRL-based adaptive learning scheme:** A DRL algorithm that uses an online RL approach was proposed, where the system continuously monitors the congestion of an SBS. Then, depending on the already allocated resources, the DRL agent generates optimised actions using a defined policy. The aim of the control actions of the DRL agent is to avoid a situation where the system has to increase the packet transmission rate by raising the transmission power, which may result in high and inefficient energy consumption. The proposed learning approach was then evaluated using two algorithms: (i) an individual learning algorithm, and (ii) the nearest neighbor cooperative algorithm. The performance evaluation thereof considers the SBS congestion rate, user throughput, and user satisfaction.

The details of the peer-reviewed publications that resulted from this work are presented as follows.

1.4.1.1 Peer-reviewed conference paper

- **M. M. Sande**, M. C. Hlophe, and B. T. Maharaj, "Instantaneous Load-Based User Association in Multi-Hop IAB Networks using Reinforcement Learning," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2020, pp. 1–6.

1.4.1.2 Peer-reviewed journal article

- **M. M. Sande**, B. T. Maharaj, and M. C. Hlophe, "Access and Radio Resource Management for IAB Networks Using Deep Reinforcement Learning," *IEEE Access*, vol. 9, pp. 114218–114234, Aug. 2021.

1.4.2 Adaptive Backhaul Traffic Routing Scheme in IAB Networks

1.4.2.1 Knowledge-defined networking

With the aim of performing optimal backhaul route selection, an IAB network is modeled as a probabilistic graph. Since each node in an IAB network should perform dynamic bandwidth reservation in a distributed manner, a knowledge defined networking (KDN) scheme was proposed as an architecture for network monitoring and the bandwidth reservation procedure was carried out in the MAC layer to make the reservation process more rapid. The proposed model can be split into two planes, namely (i) the data plane, and (ii) the knowledge plane. The data plane supports distributed traffic admission control (DTAC), where a node in the IAB network allocates bandwidth resources for traffic flows in a distributed manner. The RA problem for maximizing the overall backhaul capacity, subject to a flexible range extension, was formulated as a non-convex programming problem with the objective of ensuring that the QoS requirements of access users remain satisfied. The knowledge plane uses the probabilistic graph model to estimate the Q values and determine the maximum bound latency. That is, a process of learning network information from distributed network states is developed using Q-learning. This is a transfer learning procedure for sharing information with nearest neighbors using the forward-backward exploration technique. A performance prediction scheme that uses the principles of DRL strategies was then proposed with the aim of tackling the complexity of the IAB network, as well as assessing effective throughput, in addition to the latency upper bound.

1.4.2.2 Addressing the tragedy of commons

In order to solve the formulated backhaul routing problem, a QoS-aware routing optimization scheme that uses a rigorous and unified framework based on constrained Markov decision processes (MDPs) is presented. The optimization approach presents the reward and cost functions using implicit and explicit constraints. This approach was followed to simultaneously utilise physical-centric and system-level techniques to maximize the throughput and minimize delays as well as power consumption. A crucial cognitive function, where an agent's learning procedure applies the prediction error to adjust future predictions was incorporated into the proposed DRL strategy. The DRL strategy was used together with a recursive discrete choice model (RDCM) to evaluate the route choices in the presence of stochastic channel and traffic conditions. Regret learning, which exploits historical information about channel and queue states in selecting the optimal route, was leveraged with conditions of choice aversion.

The proposed DRL strategy was used to aggregate link states on paths in a flexible architecture that represents a source-destination routing scheme. A multi-dimensional matrix format is presented to

embed the topological and link reliability information of the IAB network. The incorporation of factors such as traffic arrival distribution, channel state, buffer occupancy status, and power consumption status into a single expression is infeasible with classical techniques. To provide better QoS to UEs in the IAB network, a max-weight algorithm was used together with back-pressure routing. Consequently, a multi-variable cost function that simultaneously accounts for both explicit and implicit constraints, as well as several QoS parameters, is formulated. A post-decision state learning strategy was then employed to deal with the known and unknown components of the cost function.

The details of a peer-reviewed article that has been submitted for publication, which comprises the contributions highlighted in this subsection are as follows:

- **M. M. Sande**, M. C. Hlophe, and B. T. Maharaj, "A Backhaul Adaptation Scheme for IAB Networks using DRL with Recursive Discrete Choice Model," *IEEE Access*, vol. 11, pp. 14181–14201, Feb. 2023. DOI: 10.1109/ACCESS.2023.3243519

1.5 OVERVIEW OF STUDY

The persistent traffic congestion in dense wireless networks, coupled with the propagation effects in mm-wave frequencies, are prominent obstacles towards realizing the performance requirements of the 5G NR. As such, this study comprised two main contributions, which are detailed in section 1.4. The rest of this thesis is in turn organized as follows.

Chapter 2 aims to present an in-depth review of the solutions that have been proposed to address the problems in IAB networks in the context of 5G NR. Here, an extensive discussion and analysis of previous research works in IAB networks together with an analysis of the solution models used to address the pressing problems are presented. The unaddressed problems that were identified from the literature study consolidated the aforementioned research gap and highlighted the motivation for this research work. In chapter 3, a dynamic user association and resource management scheme that depends on instantaneous load-based bandwidth partitioning in multi-hop IAB networks is proposed and presented. Chapter 4 presents a backhaul adaptation scheme that aims to maximize the access capacity for the associated UEs of a congested SBS, while dynamically reserving some resources for the wireless backhaul. Chapter 5 then details the overall concluding remarks of the study and notes some future work aspects that can be explored.

CHAPTER 2 LITERATURE STUDY

2.1 CHAPTER OBJECTIVES

This chapter aims to give an in-depth review of the solutions that have been proposed to address the problems in IAB networks in the context of 5G NR. The introduction and the overview of the 5G NR design principles, with a discussion on the two deployment scenarios, that is, the stand-alone (SA) and the non stand-alone (NSA), as well as their benefits, form the first objective of this chapter. Secondly, the relentless transformation of the 5G NR is discussed from the fronthaul, culminating with the background of the mobile backhaul options that led to the development of IAB. Next, there is a discussion of cognitive radio technology and the AI-enabled networks is presented, with more emphasis on reinforcement learning and the deep reinforcement learning techniques, which have been applied for the solutions proposed in this work. An extensive discussion and an analysis of previous research works in IAB networks together with an analysis of the solution models that were used to address the pressing problems is then presented. The unaddressed problems that were identified from the literature study consolidated the aforementioned research gap and signified the motivation for this research work.

2.2 5G NR SYSTEM SOLUTIONS

The design of the solutions for 5G NR should support the standards-based specifications to ensure that the integrated circuits, the hardware, the software systems and the UEs all perform within the set industry specifications. The test instruments for performance measurement should be flexible to enable the easy upgrading of hardware and software for each use case and the respective specifications. It is essential to test the operation and the performance of a 5G solution with a BS simulator that mimics the near-to-real operating conditions. In addition, the evaluation of radio-frequency (RF) performance requires measuring instruments that can connect to, communicate with, and control the hardware for both RF testing and protocol testing.

2.2.1 5G Enabling Technologies

The aforementioned technical objectives and the key performance indicators (KPIs) for 5G NR have led to the development of various technologies that enable the existence of the 5G networks. This subsection gives an overview and a brief description of the main technologies and/or the technological concepts that enable the realisation of the 5G NR objectives.

2.2.1.1 Flexible spectrum management

The enhanced mobile broadband (eMBB) leg of the 5G use cases presents a large demand for wireless capacity and traffic. To meet this demand, there was a need to provision more spectrum, and to develop techniques for flexible and more efficient spectrum utilisation for 5G NR [22]. "Incorporating more spectrum would most probably take into consideration the cm-Wave licensed bands (such as the 3.4–3.8 GHz band, which is emerging as a new small-cell band), unlicensed bands (such as the 5 GHz Wi-Fi band, which is also considered for the unlicensed band LTE), and even the unallocated, higher frequency bands in the mm-wave region (28, 38, 70 GHz)" [23]. The overall spectrum that has been identified to be useful for 5G NR is classified into three main bands, that is,

- low band (< 2 GHz), which will be useful in suburban and rural areas to provide efficient wide area coverage,
- mid-band (2-8 GHz), which is useful in less dense urban environments where there is need to cater for the capacity/coverage trade-off, and
- high band (> 8 GHz), which aims to provide ultra experience for users with high quality of experience (QoE) demands in urban hotspots.

In line with the concepts of the 5G NR, the network operators have the ability to allocate spectrum in a flexible manner to the various radio access technologies that would be able to operate at various frequency bands across the three 5G spectrum bands [24]. The MNOs can also implement carrier aggregation to circumvent the challenge of coverage that is posed by the use of mm-wave frequency bands. Carrier aggregation is the combination of frequency carriers/channels to increase transmission bandwidth, and this technique has been used in LTE-A [25]. The aggregated transmission bandwidth may be comprised of contiguous channels or it may consist of non-contiguous chunks. For the 5G NR, Ericsson [5] proposed an inter-band NR carrier aggregation technique to expand the coverage and capacity. Unlike the in-band aggregation of LTE-A, here the mid-band and the high band frequency channels are combined with NR on lower frequency bands. As a result, the indoor speeds and the areas

with poor coverage are improved. This is a complement of the 5G platform that ensures a smooth and cost-efficient way to expand the 5G network whereby the existing 4G low-band carrier can be upgraded using software only to operate both the NR and the LTE-A simultaneously.

2.2.1.2 Emerging radio access technologies

The evolution from 3G to 4G saw the development of radio access technologies (RATs) from TDMA/FDMA and CDMA to orthogonal frequency division multiple access (OFDMA) schemes. The emerging RATs for the 5G NR aim to improve the use of radio resources in order to maximize spectral efficiency [23], [26], and these RATs include:

- technologies that relax orthogonality in the air interface design, for example, the generalized frequency division multiplexing (GFDM),
- schemes that provide asynchronous and non-orthogonal multiple-access, for example sparse code multiple access (SCMA) and non-orthogonal multiple access (NOMA),
- enhanced inter-cell interference coordination (eICIC) schemes such as interference alignment and coordinated multi-point (CoMP),
- scaling up multi-user MIMO (MU-MIMO) to exploit massive antenna arrays that apply spatial diversity techniques, and
- incorporating MU-MIMO into Wi-Fi standards.

When deployed in isolation and under the idealised conditions of signal propagation, backhauling, and processing power, these RATs offer substantial network performance gains. However, the main challenges of 5G NR arise from including the technologies as integral parts of the full 5G architecture.

2.2.1.3 Network densification

In a traditional macrocell network setup, the high throughput demand of modern wireless communications would make it difficult to satisfy the user QoE, particularly for the cell-edge users. The use of small cells with low-power access points (APs) enhances energy efficiency and it provides better SINR performance [27]. In addition, the dense deployment of small cells enhances spatial reuse of the limited spectrum, which leads to increased spectral efficiency [28]. However, the heterogeneous network (HetNet) structure has been developed for 4G networks, to reap the coverage benefits of both the macrocells and the small cells. In this structure, a macrocell that serves a set of macrocell

user equipment, is overlaid by a number of small cells that serve a varying number of small-cell user equipment.

The term *small cells* is an umbrella term that has been defined by the Small Cells Forum as "operator-controlled, low-powered radio access nodes, including those that operate in licensed spectrum and unlicensed carrier-grade Wi-Fi" [29]. These small cells typically vary in radius ranging from 10m to hundreds of metres. The various small cells are categorised as either metrocells, microcells, picocells or femtocells depending on their radius and the cell range as summarised in Table 2.1.

Table 2.1. Summary of Types of Small Cells

Small Cell Type	**Description**
Metrocells	Small cells that are designed for high capacity metropolitan areas, which have BSs that are typically installed on building walls or street lamp-posts.
Microcells	Cells with an outdoor short-range base station, which aim to enhance coverage for users who are insufficiently serviced by macrocells. This goes for users located both indoors and outdoors.
Picocells	Cells with compact and low-power BSs, used in enterprise or public indoor areas. The term may also refer to some outdoor small cells.
Femtocells	Self-contained small cells with low-power, short range BSs. The term was initially used to refer to small cells intended for urban residential homes, but it has been extended to encompass higher capacity infrastructure for enterprise, rural and metropolitan areas.

Some advantages of small cells are:

- improved network coverage, particularly for indoor connections;
- enhanced spectrum efficiency;
- improved network capacity through providing better SINR and user offloading opportunities;
- maintaining visual unobtrusiveness by being artistically incorporated in existing structures such as buildings and monuments;

- improved safety, since the BSs use low transmission power, which in turn reduces the impact on health facilities;
- improved energy efficiency through low power requirements, which reduces the carbon footprint of networks.

"The need for small cells will be even more critical in 5G networks due to the introduction of higher spectrum bands, which necessitate denser network deployments to support larger traffic volumes per unit area" [30]. To identify when small cells become necessary to supplement macrocells, some experts use the traffic volume density per allocated unit of bandwidth (Gbps/km^2/Hz) metric [28]. Some operators in Asia would start deploying small cells when this metric surpassed 0.02 Gbps/km^2/Hz.

Small cell deployment scenarios

Small cell deployment scenarios will be mostly targeted at outdoor urban areas, outdoor rural areas, and indoor urban enterprise spaces. The outdoor urban scenarios arise from the need to provide spot coverage in places where there are outdoor coverage holes in existing macro coverage areas. The rural outdoor scenarios are typically motivated by the need to serve localised hotspots in remote areas, whereas the enterprise deployment scenarios are typically indoor, premises-centric deployments. Figure 2.1 illustrates some examples of the small cell deployment scenarios under these three categories, as well as other applications found elsewhere.

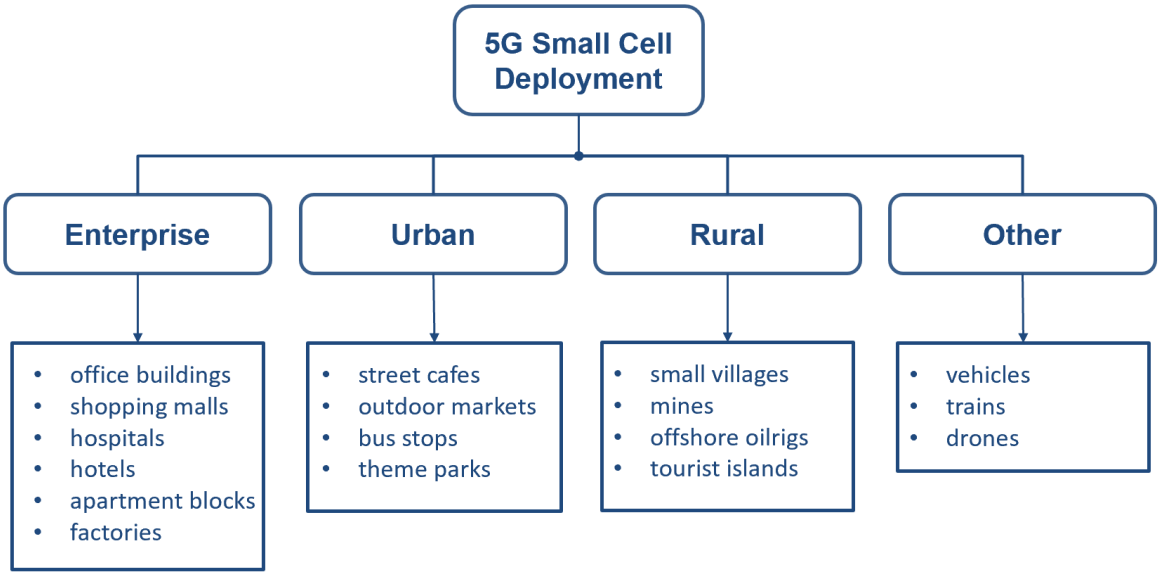

Figure 2.1. 5G small cell deployment scenarios and examples of the various deployment environments.

Small cell deployment considerations

The following are some key aspects that need to be taken into consideration for the (dense) deployment of small cells.

- **Spectrum**: Large amounts of spectrum will be required to support the densified small cell deployment that is envisaged for the 5G NR systems. To this end, there has been allocation of new spectrum bands and there are other targeted spectrum bands that are being envisioned for 5G NR [30].
- **Backhaul and fronthaul**: The design and implementation of backhaul links, which are the fixed links that connect the cellular BSs to the core network, are critical for the overall network performance.
- **BS/AP powering**: Although the small cell APs consume much less power compared to the macrocell BSs, the increased network densification in the 5G NR networks means that there would be much more sites that require powering. The required power will cater for signal transmission power, processing power and the cooling systems. Hence, there is need for power-efficient designs to overcome the powering barrier to network densification, especially for the NSA 5G NR.
- **Sharing of facilities**: The contractual agreements between the mobile network operators with respect to the sharing of small cell infrastructure need to be carefully considered.
- **Safe operation**: The deployment of small cells and the operation in the mm-wave frequencies has brought up some health concerns with regard to human exposure to electromagnetic fields (EMF). The International Commission on Non-Ionizing Radiation Protection (ICNIRP) has proposed safety guidelines for radio frequency (RF) electromagnetic fields (EMF). The guidelines, which are recognised by the World Health Organisation, state that “Extensive research has been conducted into possible health effects of exposure to many parts of the frequency spectrum including mobile phones and base stations. All reviews conducted so far have indicated that exposures below the limits recommended in the ICNIRP EMF guidelines, covering the full frequency range from 0-300 GHz, do not produce any known adverse health effect. However, there are gaps in knowledge still needing to be filled before better health risk assessments can be made” [31].
- **Environmental considerations**: The antennas mounted on buildings and monuments may be perceived as visual pollution, thus there is need for their artistic integration into these structures

in order to preserve the historical and environmental aspects.

2.2.1.4 D2D and M2M communications

The Device-to-Device (D2D) communications and the Machine-to-Machine (M2M) communications are two technological trends that have been envisaged to improve system performance whilst supporting new services for the 5G networks. These two wireless technologies contribute the most part towards the massive machine-type communications leg of the 5G use cases, which is also well known as the internet of things (IoT). D2D communication refers to the direct communication between devices, that is the direct links between the devices using the same spectrum and the radio interfaces for wireless cellular communications [32]. On the other hand, the M2M communications aim to connect a large number of low-rate, low-power devices through wireless links using cellular communication technologies.

The concept behind the D2D and the M2M communications is that the end-user equipment that are in close proximity with each other communicate directly using wireless network resources, unlike in traditional cellular networks, where this communication would go through the network infrastructure such as the BSs [33]. This direct link communication reduces the data transfer and related signaling load on the backhaul and the core network. Thus, a wireless user device with D2D capability can act as a relay node and/or as an end-user device. This direct link communication has been viewed as a necessary feature for supporting real-time services with good reliability in 5G networks.

The rationale behind the D2D and the M2M communications in the 5G networks context is the need to improve the area spectral efficiency, and to enable new cellular services that can operate with the direct links between the devices. The increase in spectral efficiency that is provided by this concept of direct connectivity between nearby devices within the same cell, known as *link densification*, is similar to the improved spectral efficiency that is contributed by cell densification [34]. The other benefits that result from the D2D and the M2M communications include extended coverage, reduced latency, reduced OPEX) and power efficiency.

2.2.1.5 Cloud networking, SDN and NFV

The concepts of cloud networking and network functions virtualization (NFV) aim to reduce the CAPEX and OPEX costs for the MNOs [26]. The CAPEX and OPEX savings are achieved through sharing the storage or computing resources, which results in the reduction of the redundancies in repositories for network functionalities.

C-RAN

The wireless networks are comprised of the core network and the radio access network (RAN), and it has been shown that in terms of computation and real-time requirements, the RAN presents more challenges for the MNOs. The basic concept of cloud-RAN (C-RAN) is to centralize the location of the different baseband units (BBUs), which would be geographically scattered in a traditional cellular network. "Once centralized, it would be possible to make different BBUs to communicate with each other in a more timely way by connecting them with high-speed switch networks, therefore allowing implementation of the cooperative algorithms to improve system performance" [35]. Coupled with virtualization, the centralization allows for the sharing of resources among the different physical BBUs, which would not be supported with simple centralization. In addition to the C-RAN concept, the other cloud computing concepts such as green cloud computing [36], are envisaged to gain traction for applications in the 5G networks.

SDN

Software-defined networking (SDN) is an emerging network architectural technology, where a centralized network controller that is situated in the control plane is responsible for allocating traffic to the network elements in the data plane of the network [37]. In SDN, the network is vertically layered, with the network layers separated by open application programming interfaces (APIs) such as OpenFlow [26], [38]. The APIs enable the implementation of network services such as data routing, bandwidth management and data security.

NFV

Network function virtualization (NFV) is a technology initiative that was started with an aim to consolidate the various network equipment onto standardized high-capacity servers [39]. The servers are typically located at different network nodes or at various end-user equipment. The architecture of a network with virtualized network functions consists of a number of virtual machines that run different software application processes, which simulate the functions of the network devices. These application processes are meant to replace the custom hardware devices.

Towards achieving the objectives of the 5G NR, virtualization is viewed as a key technology enabler for enhancing network efficiency, scalability and flexibility to accommodate various applications and scenarios. The operation of carriers in a network with virtualized functions leads to reduced equipment costs and more effective energy consumption [26]. Although NFV and SDN do not rely on each

other, they can be seen as complementary technologies. NFV provides flexible infrastructure where SDN applications can run, whereas SDN enables the flow-based configuration of virtualized network functions.

2.2.1.6 Advanced antenna systems

The advancement in technology to develop advanced antenna systems (AASs) is driven by the need to provide superior uplink and downlink performances for the end users. However, the development of the AASs needs to also consider the feasibility of implementing cost-effective systems [40]. This is supported by the integrating the baseband processing, the RF-unit and the antenna into one unit. This results in a reduction in the digital processing cost of performing advanced beamforming and MIMO. To meet the rapidly increasing capacity demands, the MNOs typically focus on maximizing the use of existing sites before the acquisition of additional sites. In doing so, they reduce the cost and time for acquiring new sites. The AASs are key enablers for both the lower-frequency and the higher-frequency bands. For the lower-frequency bands, that is the sub-6 GHz, capacity enhancement is essential due to the highly interference-limited environment. For the higher-frequency bands, coverage is crucial due to propagation and path loss challenges.

2.2.2 5G Architecture

A mobile network architecture aims to define the interaction between the network elements that ensures efficient system operation [38]. The network architecture considers the integration of different technologies, the proper inter-working of multi-vendor equipment and the cost effective design of the physical networks. The overall 5G network architecture must be able to integrate the aforementioned technology enablers and communication paradigms whilst supporting the coexistence of human-centric and machine type applications [41]. Figure 2.2 shows the overall 5G architecture as envisioned by the 5G Infrastructure Public Private Partnership (5G PPP). The 5G PPP is a consortium of collaborative research projects that is working to develop the specifications of the main elements of the 5G architecture.

The 5G network topology will consist of network slicing, service-based architecture, SDN, and NFV as the fundamental pillars that support the KPIs of the heterogeneous 5G architecture with reduced CAPEX and OPEX. Network slicing is a technique where multiple logical mobile network instances are executed on a shared infrastructure [42]. This requires a continuous reconciliation of customer-centric service level agreements among the MNOs. At the network level of the 5G architecture, each network slice is a virtualized independent platform and an application only accesses the network slice

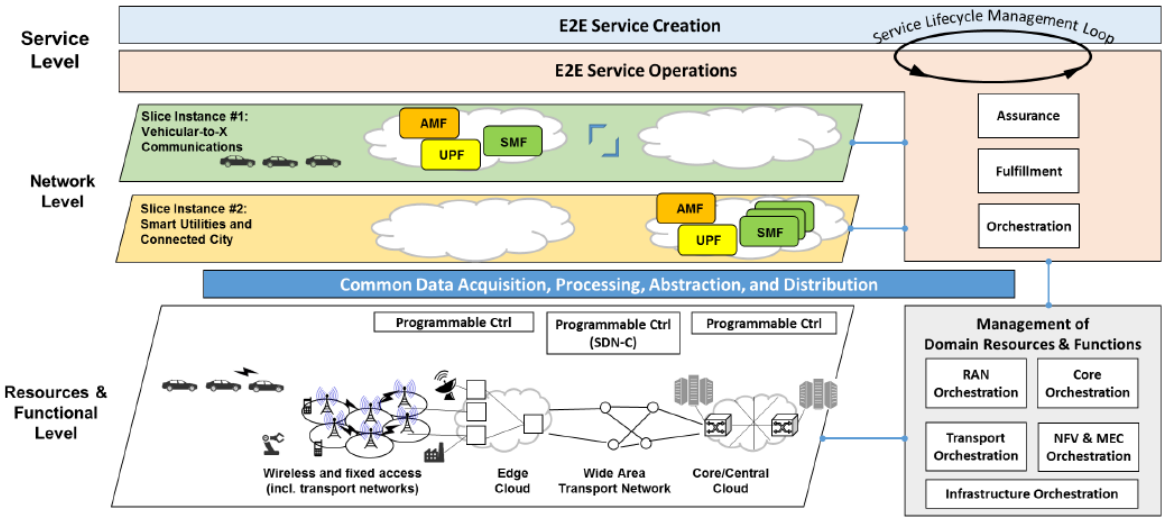

Figure 2.2. 5G overall architecture, illustrating the concept of network slicing. (Taken from [41], © 2017 5G PPP).

that it is subscribed to [43]. The proposed architecture in Fig. 2.2 forms a recursive structure, where in the context of the 5G networks, it is defined as the ability to build a service out of existing services, including repetition of the same service [41]. This recursive structure improves scalability and it allows 5G networks to handle more complex workloads compared to prior network generations.

2.2.3 5G Deployment

Although the 5G NR is a totally new interface, it is scheduled to be deployed in two modes, that is, the NSA mode and the SA mode. The initial wave of the 5G NR deployment focused on the NSA deployment scenario, whose operation leverages the existing 4G/LTE network infrastructure and the core. In this NSA deployment scenario of the 5G NR, most solutions have aimed to deliver eMBB services with high data speeds and better coverage. In the NSA mode, a 5G NR device requires an LTE BS to connect to the Evolved Packet Core (EPC) for access to the control plane [4]. In the SA mode, the 5G network operates independently of the LTE core network, thus making use of its own infrastructure. Figure 2.3 illustrates both the NSA and the SA system architectures.

2.2.3.1 5G NSA - An integrated architecture

The integrated architecture of the NSA mode shown in Fig. 2.3 allows the service providers to deliver high-speed connectivity using the already existing 4G evolved packet core (EPC). Thus, the current 5G roll-out has been a smooth transition from LTE because the NR base stations are integrated into the existing LTE network infrastructure in the NSA mode. The carrier aggregation (CA) technique

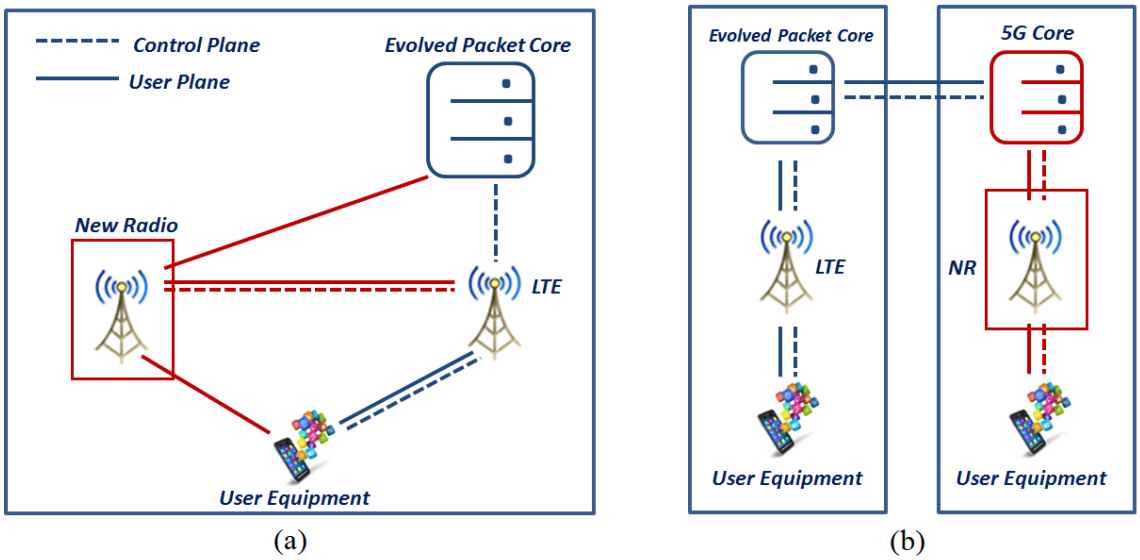

Figure 2.3. System architectures of (a) the NSA and (b) the SA modes of 5G NR from the concepts in the GSMA 5G deployment guide, where the NSA deployment runs independently on its own evolved core [44].

proposed in [5] is a complement of the 5G platform that ensures a smooth and cost-efficient way to expand the 5G network, where the existing 4G low-band carriers can be upgraded using software only to operate both the NR and the LTE simultaneously. This CA technique, which improves the indoor speeds and the areas with poor coverage, in its own right can be thought of as the 5G NSA deployment scenario. However, for the mid and high-band frequency bands network coverage is limited by the radio uplink signal quality, as a result, the effect of CA on the system capacity is severely limited. In as much as CA is supposed to improve system capacity, its impact is limited due to the different SINRs of the different spectrum bands. Thus, achieving this vision has become a very big challenge, owing to the fact that different SINRs of the different frequency spectrum bands suffer from the exiting radio nature of component carriers. This brings about the challenging implementation issues in terms of delivering a multi-functional 5G wireless communication.

2.2.3.2 5G SA architecture

To achieve a broader digitization impact and access new revenue streams, the next wave of 5G is of an SA network that operates independently from the 4G core. The 5G SA uses only the 5G NR technologies and a new 5G packet core architecture to provide full support for the crucial capabilities

designed to work only in this new architecture. Since the beginning of 2021, a rising number of operators are promoting their 5G SA initiatives. The 5G SA means that the radio part is 5G NR and the core part is a cloud-native 5G core [45]. The relentless evolution of the wireless networks will see the LTE core being replaced by a complete stand alone network that facilitates the 5G SA deployment, improves the throughput performance at the network edge, and also assists the development of new cellular use cases such as the ultra-reliable low latency communications (URLLC). Thus, at the time a fully stand-alone operation is realized, the 5G wireless network is expected to deliver multi Gigabits per second data rates to improve businesses, education and different technologies around the globe. However, due to the shortage of the radio spectrum, this vision requires that the new spectrum bands should be released in the millimeter wave frequencies to support better access to information.

2.2.3.3 Deployment timeline

To give a sense of the development and deployment of IAB in the progressive deployment of 5G technologies, Fig. 2.4 shows the timeline for the development of standards and their respective deployment, according to the 3GPP standard releases. The 5G NR deployment consists of two main phases, which are

Phase 1: 5G NR Release 15, with NSA and SA options that lay the foundation for eMBB and URLLC use cases.

Phase 2: 5G NR optimization with the introduction of new use cases specified in Release 16.

There are seven connectivity options that have been identified for the 5G NR deployment, and Fig. 2.4 defines the relevant options and it illustrates the stages at which each option is deployed.

Phase 1 (Release 15)

The main capabilities of the 5G NR as given by the 3GPP's Release 15 specifications include [13]:

- An architecture that enables various user services with different access systems.
- 5 Gbps peak downlink throughput initially, which increases to 50 Gbps in later releases.
- Scalable OFDM, with numerology for $2^N \times$ 15 kHz subcarrier spacing.
- Massive MIMO and beamforming
- Application support with edge computing.

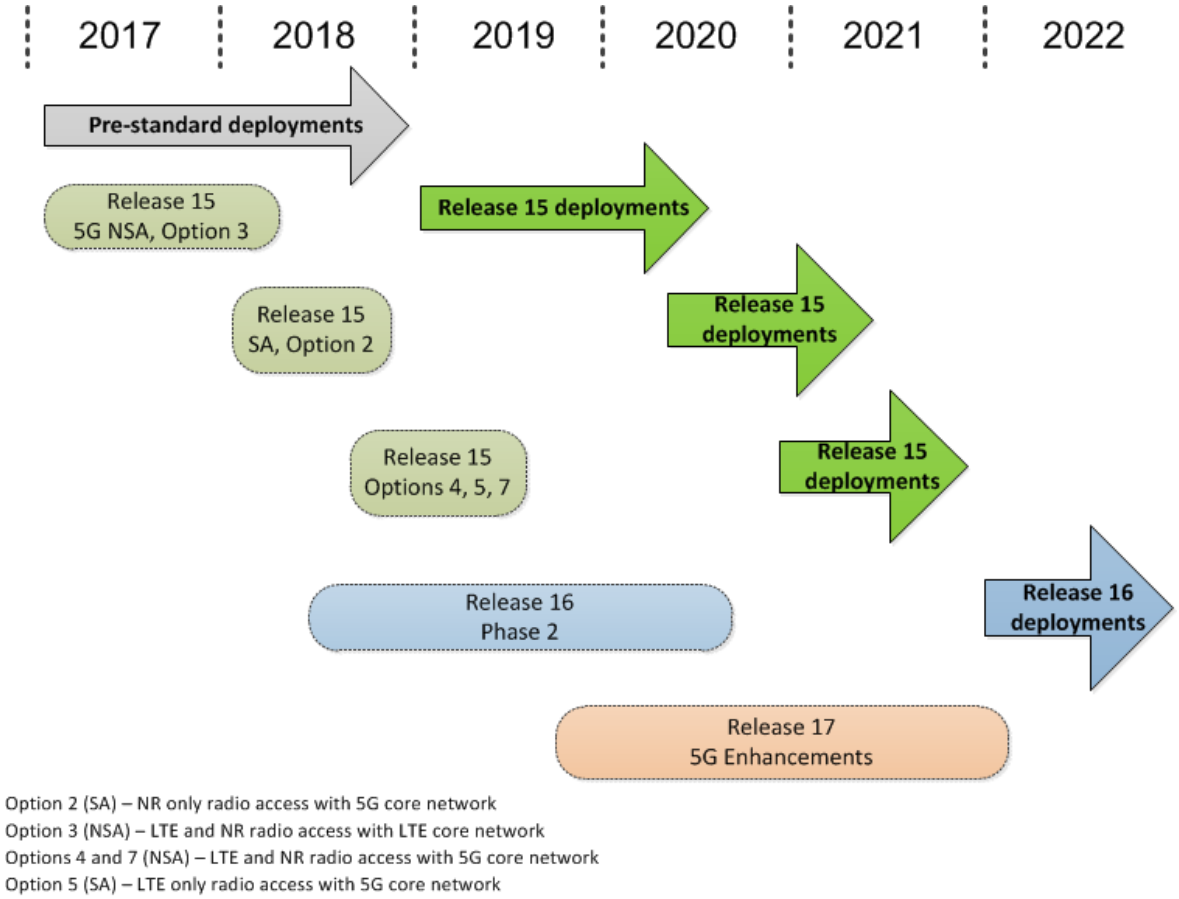

Figure 2.4. 5G deployment timeline. (Adapted from [13]).

Phase 2 (Release 16)

Some of the main specifications of the Release 16 will include support for:

- URLLC, with industrial IoT support;
- operation in unlicensed spectrum below 7 GHz;
- IAB;
- V2X communications ;
- study on support for radio bands above 52.6 GHz;
- study on non-orthogonal multiple access;
- enhancement of network slicing.

Release 17

The specifications of Release 17 include the improvements of the 5G NR, just as the LTE continued to be improved through the LTE-A. These improvements will include:

- NR-light to support wearable devices and energy efficient IoT;
- operation in frequencies above 52.6 GHz, including unlicensed bands;
- support for non-terrestrial networks, for example the operation of drones;
- mobile IAB applications such as on buses or trains.

Release 17 was completed in March 2022, "with its scope largely intact despite the fact that the entire release was developed in the midst of a pandemic that hit the world, including 3GPP, right after the scope of the Release was approved in December 2019" [46].

2.2.4 Challenges for 5G NR

The 5G NR is promising to provide a radical transformation for future wireless communications. However, the implementation of the 5G NR enabling technologies and the realisation of the proposed architecture faces a multitude of challenges. The following list highlights some of the high-level challenges as identified in [35], [43], [47].

- Integration of edge computing and cloud services, and the need to accommodate a number of different operator architectures;
- The requirement of greater real-time processing for the operation of higher frequency components and virtualized network functions;
- Propagation limitation in Tetrahertz frequencies, which would be much worse than that of mm-wave transmissions;
- More heat that is generated by increased processing power limits the compactness of devices;
- Dense deployment of small cells may face resistance from communities and regulatory authorities, in addition to the delays in deployment due to site acquisition and approval;
- Data security and privacy are significant concerns for 5G networks;
- There is potential of the global technology fragmentation that could occur as a result of political or social tensions, such as the conflict between USA and China;

- There is a need for the accurate acquisition of instantaneous CSI for the efficient implementation of massive MIMO.

2.3 5G WIRELESS BACKHAUL

Although fiber-optic is emerging as a preferable backhaul choice for 4G and 5G systems, the MNOs are looking into backhaul solutions in the microwave frequencies (i.e. 7 GHz to 40 GHz) and mm-wave bands (i.e. $>$ 40 GHz). The microwave and mm-wave mobile backhaul is a flexible low-cost option that is well suited for 5G applications. The solution implementations for both frequency bands can be deployed in a few days and they are capable of supporting distances of up to several miles. There has been advancement in the development of technologies that enable microwave and mm-wave backhaul in 5G networks such as

- adaptive coding and modulation (ACM);
- higher-order quadrature amplitude modulation (QAM), e.g. 64-QAM, 256-QAM, and 1024-QAM;
- cross polarization interference cancellation;
- massive MIMO and beamforming;
- compression accelerators;
- multi-carrier bonding.

The microwave and mm-wave backhaul has been found attractive for practical implementations in 5G networks due to its capability of providing large bandwidths. A drawback of the microwave and mm-wave backhaul is that it requires a license for operation in the respective frequencies, unless if the operation is in the unlicensed bands. Although the mm-wave frequencies suffer from high signal attenuation and are thus range limited, they are finding popular and practical applications as a solution for high bandwidth, short distance links, which are found in the densified 5G networks.

Table 2.2 highlights the trade-offs that are usually considered for the various backhaul solutions in the context of the 5G network architecture. The simulation and measurement results showed that the mm-wave frequencies provide a viable high-performance wireless backhaul solution in non-line-of-sight (NLOS) small cell networks compared to microwave frequencies [48]. In addition, the high-performance NLOS mm-wave backhaul links can provide higher antenna gain than the microwave

backhaul links for similar antenna sizes. This enables the realisation of small, compact, point-to-point fixed backhaul links that can achieve up to hundreds of Gbps throughput.

Table 2.2. Mobile Backhaul Options Trade-Offs in the context of 5G networks

Characteristic	**Satellite**	**Microwave**	**mm-wave**	**Fiber-optic**	**Copper-line (Bonded)**
Available bandwidth	Low	Medium	High	High	Very low
Deployment cost	High	Low	Low	Medium	Medium/High
Support for Heterogeneous Networks	Rural only	Outdoor cell-site/access network	Outdoor cell-site/access network	Outdoor cell-site/access network	Indoor access network
Interference immunity	Medium	Medium	High	Very high	Very high
Range (km)	Unlimited	5-30+	1 - 3	<80	<15
Time to deploy	Months	Weeks	Days	Months	Months
License required?	No	Yes	Light license/ unlicensed	No	No

2.3.1 Integrated Access and Backhaul

Wireless self-backhaul is when a slave base station (SBS) sends/receives backhaul data through a single direct wireless link or over multiple links (or hops) to/from a master base station (MBS), which is also known as an anchor BS [49]. Figure 2.5 illustrates the concept of wireless self-backhaul, where the RF spectrum is shared by both access and backhaul. As shown in the figure, the wireless backhaul can be a single-hop, as in the case of SBS_1, or it can be multi-hop, as in the case of SBS_3. In SAB, during a scheduled backhaul slot between the MBS and SBS_1, SBS_2 would be silent. Thus, this approach does not make efficient use of resources especially as the number of SBSs that are associated with an MBS grows. On the other hand, the IAB approach allows SBS_2 to utilise the unscheduled backhaul slots between the MBS and SBS_1 for access communication with its UEs.

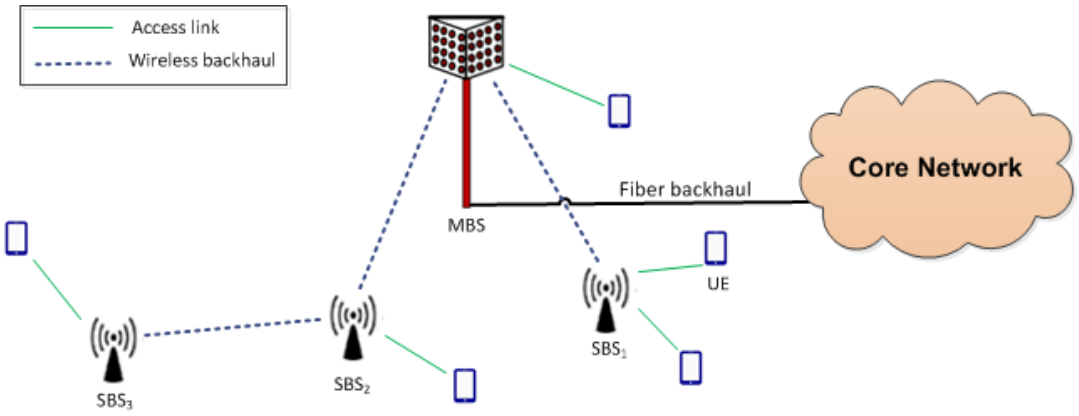

Figure 2.5. Illustration of wireless backhaul among an MBS and SBSs in 5G NR, where the MBS is connected to the core network via fiber backhaul.

2.3.2 Benefits of IAB

Cost reduction

As aforementioned in Chapter 1, IAB is envisaged to be a flexible and cost-effective backhaul technology for 5G NR operation in mm-wave frequencies. Reducing the need for fiber backhaul to each cell site in dense networks, which is very costly and time-consuming, is a capability that will be welcomed by the MNOs of the 5G NR.

Remediating coverage gaps

In dense urban environments with high-rise buildings, IAB will be useful for the remediation of coverage gaps through the use of small cell APs on but not limited to street lamp posts and building walls. In addition, IAB will contribute towards bridging the outdoor to indoor access especially in mm-wave frequencies, which have low penetration capabilities.

Enhancing capacity

"High-capacity wireless backhaul will enable mobile operators to keep up with capacity demands and maintain excellent quality of experience for their customers" [19]. Leveraging wider channel spacing, higher modulation schemes such as 1024 QAM, as well as spectral efficiency techniques such as massive MIMO and beamforming in the microwave and mm-wave frequency bands will enable high-capacity provision in the 5G networks.

2.3.3 IAB in Millimeter-Wave Frequencies

A reliable, cost-effective, and scalable wireless backhaul solution that is capable of providing gigahertz bandwidth is a prerequisite for enabling effective communications in hyper-dense networks [50]. The mm-wave technology has found attraction for applications in 5G wireless communications,

and numerous research works have investigated the features of mm-wave frequencies and electronic components. “Furthermore, standardization bodies devote great efforts to establish a norm for mm-wave communication applications. For example, European Computer Manufacturers Association (ECMA)-387, IEEE 802.15.3c, IEEE 802.11ad and IEEE 802.11aj are all existing 60-GHz industrial standards” [51]. Although the mm-wave bands had been deemed unsuitable for wireless communications for many years due to their physical barrier limitations, which include high path loss and other degradation losses, mm-wave transmission in the small cells of 4G networks in various urban environments was shown to be feasible. Results based on extensive measurement data have supported the feasibility of wireless communications in the 28-GHz, 38-GHz, 60-GHz, and the 73-GHz bands [52], [53]. In addition, the small wavelengths of these bands make it feasible to produce low-cost compact antenna packages [54].

2.3.3.1 Overview of mm-wave frequencies

Table 2.3 summarises the advantages and disadvantages of the mm-wave frequency bands in the context of 5G NR applications. The quantitative research results showed that transmission in the 28-GHz and the 38-GHz bands has less losses, but it has smaller bandwidth, while the 60-GHz and the 73-GHz bands suffer from more losses, but have larger bandwidth and they use smaller antenna size. Considering the coverage, the attenuation effects due to atmospheric absorption and rain are small compared to the path loss effects in all the mm-wave bands.

A channel model that is based on real measurements in the 28 GHz and the 73 GHz bands has been proposed [55]. Spatial statistical channel models in these frequency bands were derived to provide a realistic analysis for channel parameters such as path loss, angular dispersion, and outage. It was shown that even in the NLOS scenarios, relatively strong signals can be detected up to 200m away from a transmitter, and as such, the simulations have shown that the mm-wave systems can provide up to an order of magnitude higher capacity compared to the 4G networks for the same cell density in urban environments.

2.3.3.2 IAB system architecture

IAB is expected to support both the indoor and the outdoor 5G NR deployment scenarios. The initial deployments would typically be stationary SBS nodes and future deployments will extend to mobile scenarios such as on buses or trains. The 3GPP study on IAB investigated multiple IAB architectures

Table 2.3. Advantages and Disadvantages of mm-Wave Bands

Frequency Band	**Advantages**	**Disadvantages**
28GHz (K_a band)	Low propagation losses; Low oxygen absorption and rain attenuation	License requirement not stringent; Relatively small bandwidth
38GHz (Q band)	Relatively less attenuation caused by oxygen absorption and rain	No much research and applications
60GHz (V band)	Falls under unlicensed bands; Large bandwidth that provides multi-gigabit rates	Peak point of oxygen absorption; Relatively large rain attenuation
73GHz (E band)	Small effects of absorption	Largest rain attenuation; Very large path losses due to high frequency point

for both the SA and the NSA modes of 5G NR [7]. The architecture requirements are summarised as follows:

- The SA and the NSA should be supported for the access link, and both SA and NSA should be studied for the backhaul link;
- The IAB architecture design should support the multi-hop backhaul and should not impose limits on the number of hops. A single hop is considered a special case of multi-hop;
- Fixed relays should support topology adaptation to enable robust operation, which aims to mitigate blockages and consider the load variation on backhaul links;
- The IAB design should aim to minimize effects on the core network specifications and signalling load;
- IAB should be backward compatible with the Release 15 specifications of 5G NR.

The multi-hop capability allows flexible communication, with some nodes backhauling over one hop and others over as many as three hops as shown in Fig. 2.5. The results from the practical field trials have also shown that compared to the single-hop only deployment, the multi-hop deployment provides better performance in terms of coverage improvement, the UE rates, and cost reduction in a typical

urban environment [56], [57]. Figure 2.6 shows a block diagram of the architecture 1a that has been recommended from the architectures considered in the study by 3GPP. Although the recommended architecture has no limit on the number of hops, the maximum practical number depends on a number of factors such as the frequency band, the cell density, the channel conditions, and the traffic load. Another consideration that increases with the number of hops is latency.

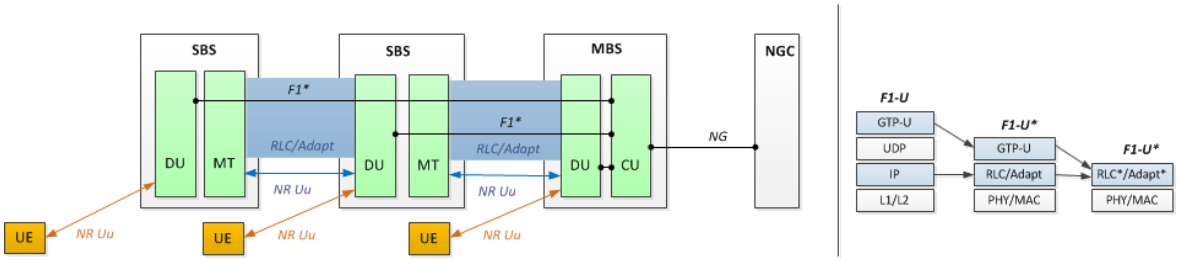

Figure 2.6. Block diagram for IAB architecture 1a showing the connections among the components of the SBSs and the MBS (Taken from [7], ©2018 3GPP).

The overall IAB architecture that is illustrated in Fig. 2.6 is based on the functionality split between the user plane and the control plane [58]. In this architecture, the MBS comprises a CU and no less than one DU. The CU hosts the packet data convergence protocol, and the service data adaptation protocol for the control plane, or the radio resource control protocol for the user plane. The F1 interface that is indicated in Fig. 2.6 is a standardized interface between the CU and the DU, which defines the higher layer protocols. The SBSs consist of mobile terminations (MTs) and the DU functionalities. Here, the MTs act as normal devices and they associate with the DU of the MBS. The message transmission is based on the lower layer functionality provided by the link between the MT of the SBS and the DU of the MBS.

In addition to the mm-wave frequencies and their hybrid use with the sub-6 GHz frequency technologies, the other two key requirements for the successful implementation of IAB are flexibility and programmability. This constitutes the incorporation of SDN where an operator or controller should be able to re-configure the data plane when new small cell APs are installed or are switched off [59]. The controllers should be able to enforce per-user policies and in case of congestion or link failures, the control plane should quickly re-route traffic. SODALITE, an SDN-based architecture for the backhaul control of IAB nodes was presented in [59]. The architecture was implemented and validated on an LTE network testbed using standard open-source solutions to illustrate its feasibility. Stochastic evaluations using traffic traces that were obtained for the LTE network validated the scalability of SODALITE, and the projection of this scalability study to future 5G networks was also presented.

Two IAB system architectures for the 5G wireless backhaul were considered in [60], which are, a centric approach and a distributed approach. In the centric approach, the MBS is centrally located, with the SBSs uniformly distributed around it. The SBSs get access to the core network through the central MBS and direct links between the SBSs are not allowed. In the distributed approach, the backhaul data is relayed to the MBS through mm-wave links between adjacent SBSs. "System-level simulations have shown that the distributed solution achieves higher energy efficiency and throughput gain, mainly due to sharing cooperative traffic among multiple wireless SBSs" [60]. In addition, it can be seen that the distributed architecture complies with the 3GPP requirement to support multi-hop backhaul.

Most works on IAB usually ignore the effect of network configuration and its interaction with user traffic, which collectively affect the SINR of both access and backhaul links as well as the loads on the various BSs [61]. Stochastic geometry is a mathematical tool that can be applied for accurate interference modeling and the performance analysis of wireless networks with random topologies [62]. A stochastic geometry-based modeling approach has been adopted for multi-tier networks, which aims to capture the independent random deployment of small cell APs [63], [64]. The approach not only captures the randomness in the network arrangement and configuration, but it also provides tractable analysis results. "These stochastic geometry-based models, initially applied to sub-6 GHz networks, have been extended to the coverage analysis for mm-wave networks. However, none of these works consider the impact of limited backhaul capacity (of which IAB is a special case)" [61].

2.3.3.3 Bandwidth partition and resource allocation

Full spectrum reuse in ultra-dense networks employing IAB is susceptible to severe inter-tier and intra-tier interference, and to address this challenge, the legacy resource allocation optimization problems require accurate CSI acquisition or computationally complex solutions [65], [66]. In addition, many solutions for the optimization problem are solved in only a snapshot of the network, which does not address the network dynamics. The rate that is allocated to user equipment is determined by the minimum achievable rates of the wireless links and the number of UEs and SBSs that would be sharing the available bandwidth.

The authors of [67] developed a tractable framework for analyzing the performance of three backhaul bandwidth partition strategies in a mm-wave IAB network using tools from stochastic geometry. The model assumes that the total downlink bandwidth B, for an SBS is partitioned into two parts, $B_b = \eta B$ for wireless backhaul and $B_a = (1-\eta)B$ for access, where $\eta \in [0,1]$ represents the bandwidth partition

factor. The three bandwidth partition schemes that were considered are

(i) equal partition, where all the SBSs share the total bandwidth equally, irrespective of their load,

(ii) instantaneous load-based partition, where the MBS partitions the backhaul bandwidth among the SBSs based on load information that it frequently collects from the SBSs, and

(iii) average load-based partition, where the MBS collects the average load information from the SBSs and the backhaul bandwidth allocated to each SBS is proportional to its average load.

The performance of the three partition strategies was evaluated by determining the downlink rate coverage probability in each scenario, which is the probability that the downlink data rate experienced by a randomly selected user will exceed a target data rate.

In [61], the authors consider the impact of limited backhaul capacity in mm-wave IAB networks by analysing the rate coverage probability for two types of RA at the MBS, considering the same network model in [67]. The two RA schemes are

1. **Integrated resource allocation (IRA)**: In this scheme, the total bandwidth is dynamically split between access and backhaul by considering the load on each SBS.
2. **Orthogonal resource allocation (ORA)**: Here, a fixed access-backhaul partition of the bandwidth is defined a-priori.

From the results of [61], it was observed that there is a near-optimal access-backhaul split for which the probability of ORA is maximized. As expected, the optimal value of the optimal access-backhaul split decreased as the density of the SBSs increased, since more bandwidth has to be reserved for the backhaul to support a given data rate. It was also shown that offloading traffic from the MBS to the SBSs in two-tier networks is more effective in fiber-backhauled SBSs than in mm-wave IAB networks. However, the cost of installing fiber backhaul for every SBS outweighs the traffic load handling advantage.

Th optimization for link scheduling and RA by applying binary interference classification, which is an interference coordination scheme that prevents two links from being active concurrently, was proposed in [68]. By using a conflict graph model, the authors proposed a maximum independent set-based scheduling algorithm, which makes fixed routing decisions. For this proposed solution, the

simulation results showed that it gives improved performance compared to common multiplexing and interference mitigation schemes in terms of the achievable data rate and latency performance. Another RA solution that performs joint power-carrier resource assignment in multi-hop IAB networks is proposed in [69]. Here, the authors presented low-complexity optimal and sub-optimal RA algorithms. The proposed solution formulation, however, does not consider the traffic load as was done in [68], but instead it considers the dynamic channel effects that may be caused by the time lag between the CSI measurement and the RA decision implementation.

2.3.3.4 Backhaul path selection

Wireless communications in mm-wave frequencies typically use direct links to reduce path loss effects, through the application of steerable antenna arrays and beamforming techniques. This enables multi-stream data support for multi-user scenarios and it paves way for spatial reuse, which should be fully exploited for overall backhaul performance improvement [51]. In addition to the scheduling and RA solution, the authors of [68] also presented a dynamic backhaul routing algorithm for optimal path selection in multi-hop IAB networks. The proposed dynamic routing algorithm considers real-time network context such as the BS load to generate the optimal backhaul path for each UE through a greedy approach. Although the proposed dynamic algorithm considers context for optimal routing, it is worth noting that incorporating machine learning (ML) techniques can help improve the system latency and cost performance.

The authors of [51] solved the path selection and the time allocation problems in 5G mm-wave backhaul by proposing an optimization model of the tractable hybrid system architecture. In this hybrid model, more than one BS is connected to the core network via wired fiber links. The SBSs in the resulting backhaul mesh network communicate with each other using pure mm-wave links or the backhaul data can be routed via a wired SBS. This hybrid technique aims to improve reliability by addressing blockage in mm-wave links, since mm-wave communication suffers from severe penetration loss. The tractable system architecture combines performance enhancing physical layer techniques with routing and scheduling schemes in higher layers to combat two types of blockages, namely temporary blockage and permanent blockage. An example of temporary blockage is a moving vehicle, and in such instances an SBS adopts a smart beamtracking method that enables it to switch between alternative links to avoid the blockage. On the other hand, the permanent blockage may be due to a building between two SBSs, and in this case transmitting data to adjacent SBSs as relays may be a more suitable option for high capacity applications compared to using the NLOS paths. Ultimately, the simulation results showed

that the physical layer techniques such as hybrid beamforming and full-duplex transmission provide improved capacity and latency performance in mm-wave backhaul networks.

The D2D communication has been exploited for providing networks access to the UE experiencing poor QoE via a technique known as the user provided network (UPN) [70]. Through the UPN, the users with high channel quality are able to serve as relay operators to provide network connection to the users with weak channel conditions using D2D links such as Wi-Fi [71]. In [70], the authors analysed the UPN service in a 5G mm-wave IAB network framework. The modelled framework considered a two-tier, multi-hop, multi-path hybrid mesh network, where the energy efficiency and monetary data download cost are parameterized for each user whilst considering the interference between the backhaul links and the capacity constraints. The formulated problem is solved using a Nash bargaining solution, with the aim of establishing fair cooperation among the users and improving the SBSs' resource utilisation efficiency. The Nash bargaining solution was developed for two algorithms, namely the centralized algorithm and the distributed algorithm. The distributed algorithm addresses the UE privacy issue, which is a drawback of the centralized algorithm.

A joint transmission and scheduling scheme for the mm-wave IAB networks was proposed in [72], where a path selection criterion was designed to enable the D2D communications. The authors analyzed the impact of path selection on performance improvement in the D2D network under various scenarios. Similar to the work of [70], if the direct link between two users that wish to communicate has high channel quality, direct transmission would be adopted instead of transmission via the backhaul network. In [72], the direct transmissions between the devices was formulated as a mixed integer linear program, with the aim of minimizing the number of time slots to accommodate the traffic demand of all the flows. The proposed optimization solution was shown to achieve near-optimal delay and throughput performance in some cases of varying path-selection parameter.

Backhaul path selection in a mm-wave IAB setup that applies beamforming and sectorized deployment was investigated in [73]. The authors analyzed four path-selection techniques using a distributed approach and they compared the performance of the path-selection schemes in terms of hop count and bottleneck SINR in the mm-wave channel model of [55]. The four path selection policies that were considered differ in the metric that is used to measure the link quality (using either the SINR or the data rate), and the priority factors used to rank the different link that would be available at each hop. These four policies are described as follows:

(i) *Highest-quality-first (HQF) policy*: In this policy, the SBS at each hop selects the backhaul link with the highest signal-to-noise ratio (SNR), without considering any other additional information.

(ii) *Wired-first (WF) policy*: The WF policy aims to reduce the number of hops that are required to reach the wired MBS, by selecting the link to a wired MBS even if it is not associated to the connection with the highest SNR. The HQF policy is applied if there is no link to a wired MBS at the current hop.

(iii) *Position-aware (PA) policy*: This scheme considers context-awareness for the SBS that has to perform link selection in relation to the position of the MBS. The aim here is to select the path to the closest MBS from the current SBS.

(iv) *Maximum-local-rate (MLR) policy*: In this policy, each hop selects the path to the MBS that has the highest achievable Shannon rate. It is worth noting that the MLR policy does not consider the SNR as a metric, but instead it considers the achievable data rate.

The simulation results showed that the number of hops required to connect to an MBS can be decreased by designing aggressive bias functions which do not affect the average bottleneck SNR. It was shown that the WF policy is ineffective in cases of sparsely deployed networks and for low SNR scenarios. On the other hand, the PA scheme, which applies context-awareness to perform the path selection, significantly improves the overall performance in terms of both the number of hops and the QoS.

2.3.3.5 Research motivation

From the reviewed literature on contributions that have been made to solve problems in the IAB networks, it is apparent that dynamic resource management and data routing still remain underinvestigated problems in the mm-wave IAB networks. The existing solutions focus on finding low-cost routes for traffic to reach the MBS, without considering the ways in which resource exhaustion at a serving BS can be minimised. Resource exhaustion is the unavailability of communication resources to satisfy all the associated user requirements at a BS, mainly due to congestion. A framework that avoids resource exhaustion should be capable of adapting to different learning mechanisms while adhering to system requirements in real-time effectively. Considering that the operational spaces of the next generation wireless networks will be very diverse and that the future wireless environments will be highly unpredictable, rule-based decision-making, that is learning directly from training, may not be ideal. As a result, designing apriori cost functions and then solving optimal control problems in real-time may be ineffective. For this purpose, the decision maker of an IAB system can incorporate a

deep neural network (DNN) to provide action choices for any given state of the system. The DNNs are essential for optimal solutions in highly stochastic and dynamic environments such as mm-wave IAB networks, where an action's reward value depends on future actions and states. In such cases, the DRL-based schemes become an attractive alternative. As such, the subsequent section gives an overview of literature with respect to machine learning applications in wireless networks, leading up to DRL applications in 5G networks and consequently IAB networks in particular.

2.4 COGNITIVE RADIO AND AI-ENABLED NETWORKS

Modelling and optimising for smart devices in 5G networks, which would be capable of jointly using many different frequency bands and communication technologies to achieve the required rate and latency performance, is very challenging [17]. The evaluation of network performance using metrics such as peak data rates and spectral efficiency is not suitable for the HetNet architecture of 5G networks. The performance metrics for the 5G networks are centered around enhancing the QoE of the UEs [18]. Thus, modeling the wireless networks and the analysis of network characteristics such as user satisfaction, RA and backhaul routing using the received SINR as the base metric for resource allocation is inadequate for 5G scenarios [74]. To address this limitation, there has been considerable research that looks at the application of CR technology and AI for smart devices in 5G networks. The application of AI techniques in 5G networks enables the optimization of the network in real-time through the control of BS association, the routing configurations for wireless multi-hop backhaul, or load balancing [40].

2.4.1 Context-Awareness in Wireless Networks

The term *context-awareness* was first introduced in the 1990s with a focus on computer applications for context-aware computing [75]. The term was redefined for systems to state that "a system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task" [76], where context is any information that can be used to characterise the state of a device/element. From the wireless communications perspective, context awareness is a mobile networking paradigm in which devices or applications can discover and take advantage of wireless contextual information such as user association, BS load, and user behaviour. This phenomenon has been studied extensively in mobile computing to build diverse context-aware applications [77]. It has been established that there is need for integration of multiple parameters in the optimization of network functionalities in designing models for 5G networks smart devices [78], and as such, a number of models that consider context-awareness for network elements in 5G networks have been proposed. To provide high QoE to end users, "5G systems will need to be context-aware,

utilizing context information in a real-time manner based on network, devices, applications, and the user and his/her environment" [18]. Hence, there is need to develop new ways of obtaining context information as well as new ways of sharing context information between applications, network layers, and devices. The context that need to be considered by 5G systems include:

- device-level context, which includes the battery state, the CPU load, and the device characteristics;
- user applications such as video streaming, web browsing, gaming, or interactive cloud-based applications;
- user environment context, which considers factors such as user mobility, user location, and the proximity of other users;
- network context, which considers factors such as congestion/load, wireless channel state, backhaul quality, and the availability of alternative spectrum or routes.

An ideal solution for a context-aware user association problem for 5G HetNets is found at the intersection of three perspectives that are usually considered individually in literature [78], which are (i) a load balancing perspective, (ii) a seamless mobility and handover perspective, and (iii) a self-organizing network (SON) perspective that aims to improve network efficiency and ease the management tasks for diverse complex networks.

2.4.2 Cognitive Radio Technology

A key enabler for flexible spectrum management, which is one of the techniques that make it possible to achieve the 5G NR objectives, is cognitive radio (CR) technology [26]. A CR is an intelligent device that makes decisions that enable opportunistic spectrum access by being aware of and understanding its environment [79]. Although the spectrum may be exclusively reserved to be used by licensed primary users (PUs), the PUs may not always make efficient use of the spectrum. Hence, in a cognitive radio network (CRN), the unlicensed secondary users (SUs) are allowed to be temporarily allocated the available spectrum in an opportunistic manner [80], [81].

In most of the literature on CR technology and CRNs, the "cognitive" aspects are focused on spectrum sensing, channel selection and adaptive communication. However, a broader definition of CR, which incorporates cognition aspects such as intelligent observation, learning, and decision-making, has found preference in 5G networks [82]. Hence, a CR should be aware of its environment by being able

to sense and characterize the various RF activities in its surroundings, and it should learn and make reasonable decisions that improve future performance [83], [84].

There are three main constituents of intelligence that should be built into the CRs of future networks, which are **perception**, **learning**, and **reasoning** [85]. Perception can be viewed as sensing the wireless environment to identify ongoing RF activities in the CR's surroundings. The CR then tries to learn from the acquired information to generate knowledge that is used to make decisions and take action that allows the CR to achieve its objectives. This relationship of the three constituents of an intelligent CR is illustrated in Fig. 2.7. It can be seen that learning is an essential tool that allows a CR to acquire knowledge from its observed data, and as such, the succeeding subsection delves more into the applications of machine learning for CRs in 5G networks. The learning capability enables the CRs to autonomously learn to adapt to the dynamic wireless environment in an optimal manner.

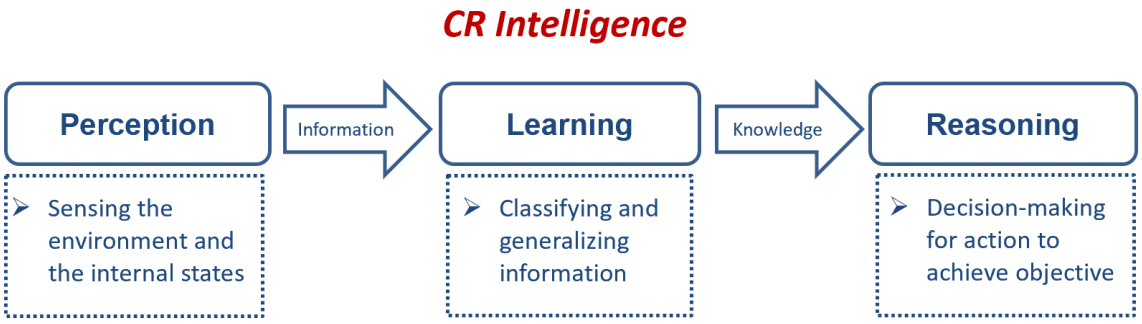

Figure 2.7. Illustration of an intelligent design of a CR, showing the connection of the three components of a CR.

The need for optimised resource usage and management in wireless networks has led to the introduction and application of AI in the learning and reasoning stages of CRs [86]. The reasoning stage of CR technology comprises inference and decision-making, as shown in Fig. 2.7. Cognitive inference is responsible for making the choice of actions that lead to efficient decisions. In addition, cognitive reasoning involves the complex interaction between current and learned information, and this can be well-handled by applying ML.

2.5 INTRODUCTION TO MACHINE LEARNING

Machine learning is when a machine learns the execution of a particular task with the aim to maintain a certain performance metric, based on some experience [87]. ML is an AI tool that is envisaged to address a number of technical challenges that are presented by the need for accurate modelling in

implementing some of the aforementioned technologies that enable 5G networks [88]. In addition to being an AI tool, ML is also viewed as a modelling technique that draws upon ideas from a wide range of disciplines such as information theory, probability and stochastics, as well as control theory [89].

2.5.1 Machine Learning in 5G Networks

Machine learning-based models have found applications in gaming, data mining, telecommunications, bio-sciences and control automation. ML algorithms can be classified into three main categories, which have all found applications in 5G networks. Figure 2.8 shows an overview of the 5G NR use cases where the various ML solution algorithms are typically applied. To supplement this, Fig. 2.9 shows the family-tree of the ML algorithms and it summarises the potential applications of the respective ML techniques in 5G NR problems according to [88].

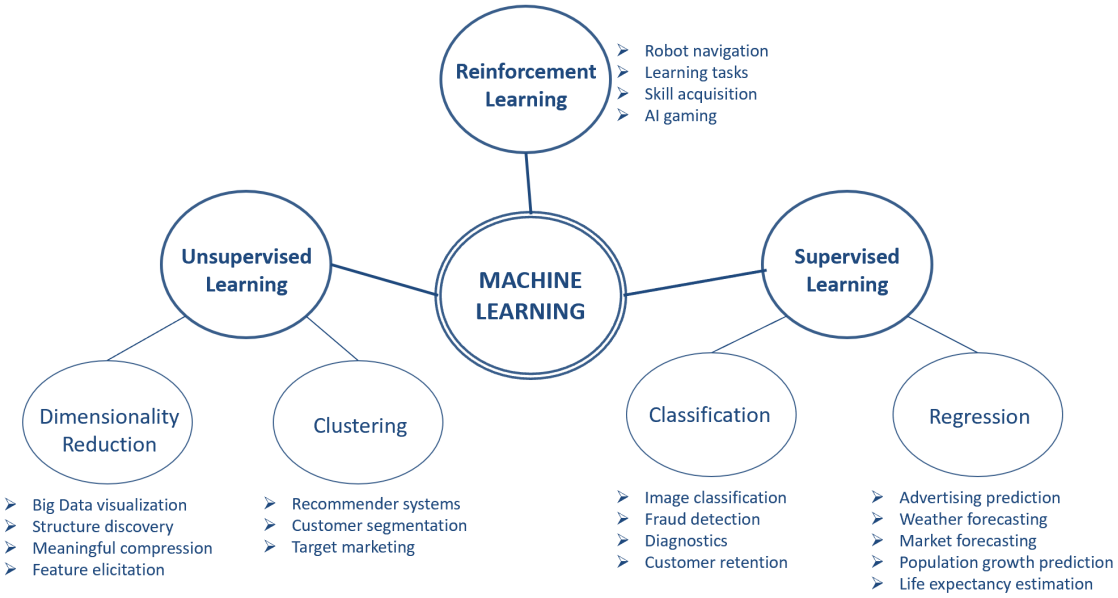

Figure 2.8. Applications of the various ML solution algorithms in 5G use cases.

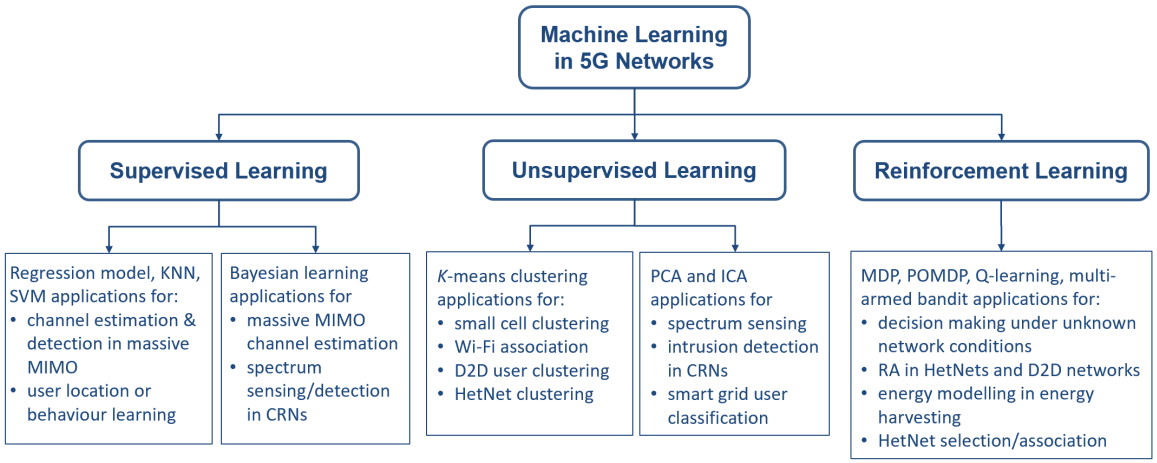

Figure 2.9. Some applications of the three branches of machine learning in 5G networks, showing the applicability of RL for resource management in IAB networks.

2.5.2 Supervised Learning

In supervised learning, the tasks are accomplished by learning from examples provided by an external supervisor, where each training example consists of an input/output pair of data that is provided during the training [90]. The input/output data is typically labelled and classified to provide a reference for future learning processes. Thus, supervised learning provides learning algorithms that use known quantities of examples and their respective labels to provide future judgement [91].

Considering a typical input denoted by x and its label y that form the input/output pair $(x, y) \in D$, where D is an existing data set. The main goal in supervised learning is to learn a function that accurately predicts the output y for an input x, or alternatively estimate its conditional distribution $p(x|y)$. The common supervised learning models and their associated data classification algorithms are

(i) *Support vector machines*: Support vector machines (SVMs), are supervised learning schemes that were initially applied for binary classification, but were then extended for use in regression and multi-class classification [87], [92]. The SVMs are non-parametric models, that is, the number of parameters for an SVM model is not fixed when the model is constructed. The SVM algorithm depends on nonlinear mapping, which transforms the input training data to make it separable. The algorithm then searches for the optimal linear separating hyperplane that

classifies the separable data, which relies on the family of non-linear kernel methods such as a radial basis function [93].

(ii) *K-nearest neighbour (KNN) algorithm*: In the KNN technique, an input is classified into a specific class, which would be the most common class among its k nearest neighbours [87]. The output is determined by a certain property of the input, such as the average value of its k nearest neighbours.

(iii) *Decision trees*: The decision tree algorithms offer a lightweight solution to classification problems by assuming that the inputs that lie close to each other in the data space should be similar [94]. A decision tree is basically an aggregation of conditions that enable a new input to traverse all nodes in a network until the terminal node, which gives the output.

(iv) *Bayesian learning*: The premise of Bayesian learning is computing the conditional a-posteriori probability distribution on all of the training instances. The models that apply the Bayesian learning techniques in wireless networks include the Gaussian mixture (GM) model, the expectation maximization (EM), and the hidden Markov models (HMM) [87]. In the GM model, each input belongs to one of several Gaussian distributed clusters; the EM approach is a form of maximum likelihood estimation, whereas the HMM approach aims to represent the probability distributions of the sequences of observations that contain hidden variables.

In wireless networks, supervised learning has found applications in regression, classification and in clustering problems. In regression and classification problems, supervised learning is applied to predict continuous values for the current input, and to identify the class to which the prediction belongs. An example of such problems in the 5G NR is channel estimation for the users in massive MIMO systems. The KNN and SVM techniques have been applied for determining optimal handover solutions in HetNets, where they were used for learning user contexts that were used in a location-specific interface [95]. A hierarchical version of SVM, known as hierarchical support vector machines (HSVM) was proposed for use in the data classification problems for channel estimation in MIMO networks [96]. The HSVM model was applied for learning the estimation of the distribution of channel noise.

Bayesian learning has found applications for learning spectral characteristics and channel conditions in wireless networks. To address the challenge of pilot contamination in massive MIMO systems, the sparse Bayesian learning techniques were applied in [97]. Here, the authors estimated the channel parameters of a desired link as well as those of the interfering links. Bayesian learning has also been widely applied for modelling channel occupancy in CRNs [98], [99], [100], [101]. In [102], a

cooperative wideband spectrum sensing scheme that applies the EM algorithm was proposed. The iterative technique first created a log-likelihood function for the unknown spectrum occupancy, the channel state information, and the noise before maximizing the log-likelihood function to jointly detect the PU signal and the channel information. A two-state HMM was developed in [103], where the EM algorithm was applied for estimating channel parameters such as "the sojourn time of the available channels, the inactive states of the PUs, and the PUs' signal strength." Another application of Bayesian learning in CRNs is found in [104], where a tomography model, which is a Bayesian inference framework, was proposed for modelling at both the link and the network layers. The model aimed to conceive and to statistically characterize the techniques that are capable of extracting the essential channel parameters and traffic/interference patterns in CRNs.

2.5.3 Unsupervised Learning

Unsupervised learning is a ML technique where the learning model aims to discover the required information from unlabelled datasets. Thus, unlike supervised learning, the unsupervised learning algorithms aim to discover a hidden structure in the input, without assuming a labelled example. This allows unsupervised learning to perform more complex processing tasks compared to supervised learning. The common unsupervised learning models that are applied in wireless communications are:

(i) *K-means clustering*: The *k*-means clustering unsupervised learning approach aims to partition *n* input observations into *k* clusters, where each input is assigned to the closest cluster. The clustering algorithm is designed to minimize the intra-cluster Euclidean distance by iteratively updating the cluster-centroid, until convergence. Here, convergence is achieved when the cluster assignment reaches steady state, that is, when the clusters formed in two consecutive rounds are the same [88].

(ii) *Principal component analysis*: Principal component analysis (PCA) is a technique where a set of potentially correlated variables is transformed into a set of uncorrelated variables, which are known as principal components. The number of principal components should be less than or equal to the number of original variables, and the principal components are orthogonal since they are the eigenvectors of the symmetric covariance matrix.

(iii) *Independent component analysis*: "Independent component analysis (ICA) is a statistical technique conceived to reveal hidden factors that underlie sets of random variables, measurements, or signals" [88]. In an ICA model, the input variables are assumed to be linear mixtures of some

unknown latent variables. The latent variables are known as the independent components of the input data and they are assumed to be mutually independent and non-Gaussian. The process of determining these latent variables is known as ICA [87].

In wireless networks, the *k*-means clustering approach is prevalently applied in clustering and association problems, which are common problems in the 5G networks, particularly in HetNets and D2D networks. Examples of such applications include but are not limited to:

- small cell clustering in Hetnets for interference management using CoMP,
- clustering of mobile users to obey an optimal offloading policy,
- clustering of devices in D2D networks to improve energy efficiency, and
- Wi-Fi users clustering for optimal access point association.

Because of their ability to recover statistically independent source signals from mixtures, both PCA and ICA are considered as powerful statistical signal processing techniques that have found applications in traffic monitoring problems in wireless networks. These applications include

- anomaly detection, fault detection and intruder detection in wireless sensor networks and mesh networks,
- classification of PU behaviours in CRNs, and
- recovery of wireless transmissions of smart utilities in smart grids.

2.6 THE REINFORCEMENT LEARNING STRATEGY

The RL strategy aims to acquire the ability to generalise the learning process in a similar fashion to supervised learning. However, the supervision in the reinforcement approach of RL means that the supervision comes in the form of feedback from evaluation of the learner's behaviour, in contrast to supervised learning. In RL, a decision-making agent in a certain state takes action based on a defined policy and it receives a reward based on the action it takes [105]. In selecting the next actions, the agent aims to strike a balance between exploration, that is choosing untested actions, and exploitation, which is the selection of actions already identified as beneficial. In doing this, the agent's overall aim is to reach an optimal policy for maximizing the reward. A generic illustration of the operation of the RL strategy is shown in Fig. 2.10.

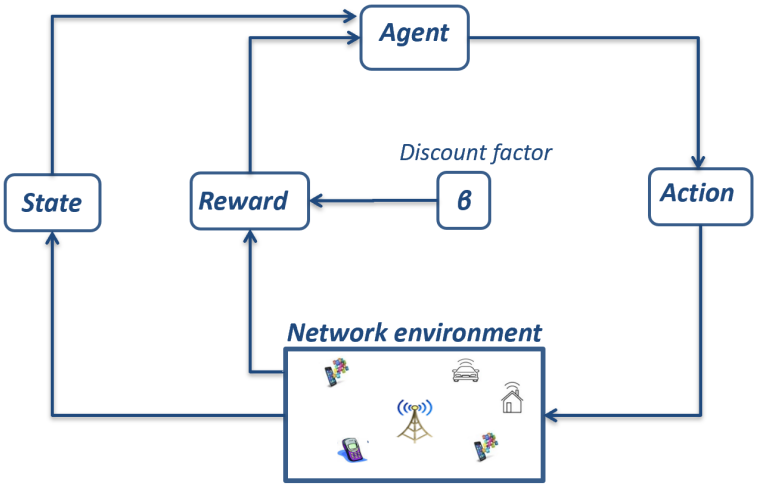

Figure 2.10. The concept of RL in the context of wireless networks, showing the interaction between the agent and the network environment.

Figure 2.10 shows the generic operation of the RL strategy, which can easily be understood by using the concepts of agent, environment, states, actions, as well as the rewards as discussed below:

- **Agent**: The agent can be defined as an entity that interacts with the environment and performs the learning process by transitioning between the states as a result of taking a actions and obtaining associated rewards.
- **Environment**: The environment can be defined as the space through which the agent transitions. By taking an action a_t in a current state s_t, the agent obtains a reward r_t and it transitions to the next state s_{t+1}. The transfer function of the environment is not always known and it is typically modelled as a black box, where only the inputs and outputs can be observed.
- **State**: The determined immediate conditions that the agent can be in at any given time.
- **Action**: The action can be defined a s the output of the agent's learning process when in a given state, which is performed in the environment.
- **Reward**: The reward is the feedback that is a result of evaluating the agent's actions. The reward function defines how the goal of a RL problem is achieved. As aforementioned, the main objective of the RL agent is to maximize the accumulated reward it receives. The reward function is generally unalterable by the agent, however, the achieved reward may serve as a basis

for changing the policy. That is, if an action selected by the policy generates a low reward, the policy may be changed to select a different action in the same state in the future.

- **Policy**: A policy defines how an agent maps the perceived states of the environment to actions that are taken in the respective states, which is the RL agent's behaviour. The policy may be a simple function, a lookup table, or a complex computation.
- **Value function**: A value function determines the total accumulated reward that an agent can expect when starting from a given state. Whereas a reward function indicates the immediate reward of a given state, the value function specifies the long-term rewards. As an example, a state might always yield low immediate rewards yet it may have a high value because it would be followed by states that achieve high rewards, or vice-versa.

2.6.1 Reinforcement Learning Methods

There are three main methods of classifying solutions for the RL problem namely, dynamic programming, Monte Carlo solutions, and temporal difference learning [105], [106]. Each class of solution methods has its strengths and weaknesses.

(i) ***Dynamic programming***: Dynamic programming algorithms are used to determine the optimal policies given an accurate model of the environment, which would be modelled as a Markov decision process (MDP). The main idea behind dynamic programming is to use value functions to organise and structure the search for good policies. An advantage of dynamic programming methods is that they are mathematically well-developed. However, the drawback is that they require a complete and accurate model of the environment and they can be computationally intense.

(ii) ***Monte Carlo methods***: The Monte Carlo methods are learning methods for discovering optimal policies and for estimating value functions using experience only. That is, the Monte Carlo methods do not require a model. They only require sample sequences of states, actions and rewards that are obtained on-line or from simulations of the environment. The Monte Carlo methods are based on solving the RL problem by averaging complete sample returns. In addition to not requiring a perfect model, the Monte Carlo methods present simpler solutions, which is desirable, but they are not suitable for step-by-step incremental computations.

(iii) ***Temporal difference (TD) learning***: TD learning is a combination of the Monte Carlo methods and the dynamic programming ideas. The TD methods can learn from raw experience without a model of the environment's dynamics, and similar to the Monte Carlo methods, and they update

their learned estimate values based on other estimates without waiting for a final outcome, just like the dynamic programming methods. The main TD-based solution approaches are the Sarsa algorithm, Q-learning and the actor-critic methods. Although the TD learning methods do not require a model and are incremental, they are generally more complex to analyse compared to the other RL methods. On the other hand, the advantages of TD learning approaches are:

- they do not require a model of the environment, nor distributions of the reward or next state probability;
- they are implemented in an online and fully incremental fashion, instead of waiting for an episode to end as in Monte Carlo methods;
- they have been found to converge faster than Monte Carlo methods when solving stochastic tasks;
- TD learning is optimal in a way that is more relevant to predicting returns compared to Monte Carlo methods. TD gives lower error on future data, whereas Monte Carlo methods are better on the given data;
- in large state-space operations, TD methods may be the only feasible way to approximate the certainty-equivalence solution.

The TD learning methods are categorised as either on-policy or off-policy learning methods. “On-policy methods attempt to evaluate or improve the policy that is used to make decisions, whereas off-policy methods evaluate or improve a policy different from that used to generate the data” [107]. When using trajectories that may not be obtained using the current policy, learning with off-policy methods is straightforward, whereas on-policy methods usually introduce a bias when used with a replay buffer [106]. This makes off-policy methods sample-efficient when using the trajectories that may not be solely obtained under the current policy, π, since they are able to make use of any experience.

2.6.1.1 Actor-Critic Methods

Actor-critic methods are on-policy TD learning methods that have a separate memory structure for explicitly representing the policy, independent of the value function. The *actor* represents the policy structure, which is used for selecting actions when in a given state. The *critic* refers to the estimated value function, which criticises the actions that are selected by the actor. As aforementioned, the learning is on-policy, where the critic learns about and criticises the current policy being implemented by the actor. The sole output of the critic, that is, the value function, is a TD error, which is a scalar

signal that drives the learning in both the actor and the critic. The architecture of the actor-critic learning operation is illustrated in Fig. 2.11.

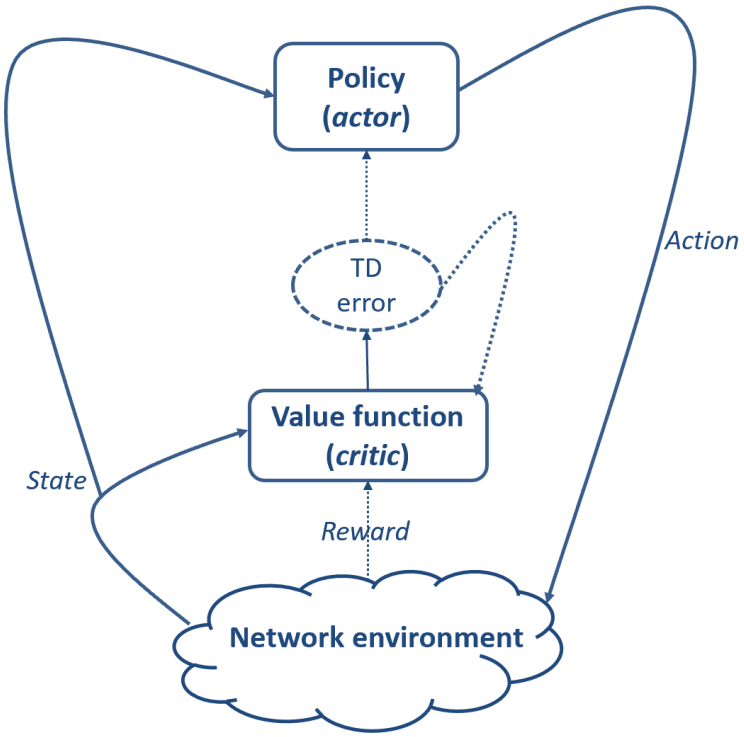

Figure 2.11. The actor-critic operational architecture, illustrating the interaction between the main components. The solid lines represent learning inputs, whereas the dashed lines represent the outputs.

The main advantages of the actor-critic methods over the other TD learning methods are

- they require minimal computation for action selection, which is useful in cases where there are infinite possible actions such as continuous-valued actions, and
- "they can learn an explicitly stochastic policy; that is they can learn the optimal probabilities of selecting various actions" [105].

Although many of the earliest TD-based RL methods were actor-critic, recent attention has been devoted to the methods that learn the value function and determine the policy to implement using only the estimated values.

As illustrated in Fig. 2.11, after each action selection by the actor, the critic, which is typically a state-value function, evaluates the new state to determine whether there has been an improvement. This evaluation is the computation of the TD error and it is given as

$$\delta_t = r_{t+1} + \beta V(s_{t+1}) - V(s_t), \tag{2.1}$$

where $V(s_t)$ is the value-function of the current state s_t, which is implemented by the critic at time t. This TD error gives the basis of evaluating the selected action a_t when in state s_t. If the error is positive, it shows that selecting the action a_t should be favourable for the future, however, if it is negative, the tendency to select a_t should be weakened.

2.6.1.2 Sarsa Algorithm

The Sarsa algorithm is another on-policy TD method. Here, the first step is to learn an action value function. That is, the considered transitions are from the state-action pair to state-action pair, unlike in state-value function learning where the considered transitions are from state to state. In this method, the action-values $Q^{\pi}(s,a)$ for the current behavioural policy, π, are estimated and updated for all state-action pairs according to the following theorem.

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \beta Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)], \tag{2.2}$$

where $\alpha \in (0,1]$ is the learning rate, and $\beta \in (0,1)$ is a discount factor. This update is carried out after every transition from a non-terminal state s_t. The general form of the Sarsa algorithm is given in Algorithm 1.

Algorithm 1 Sarsa TD learning algorithm

Initialize $Q(s,a)$ arbitrarily
For each episode do
  Initialize s_t
  Choose a_t from s_t using the selected policy for Q
  For each step of episode (until s_t terminal) do
    Take action a_t, determine reward r_{t+1} & next state s_{t+1}
    Choose a_{t+1} from s_{t+1} using the selected policy for Q
    $Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \beta Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)]$
    $s_t \leftarrow s_{t+1}$; $a_t \leftarrow a_{t+1}$
  End for
End

2.6.1.3 Q-learning

Q-learning is an off-policy TD approach that has enabled early convergence proofs. As mentioned in Section 2.6, it is a RL technique that does not require the exact transition formulation of the system. The basic idea of Q-learning is to maintain a Q-table consisting of Q values, which represent a measure of goodness resulting from taking an action a when in state s. It can be seen as a category of ML techniques that are useful for learning the policies to achieve a desired objective. The basic idea of Q-learning is that a learning agent determines the current state of the environment, s_t, from a set of all possible states, and it takes an action $a_t \in \mathcal{A}$, where $\mathcal{A}$ is the set of all possible actions. The agent receives a reward, r_t, for the action taken and the environment transitions to the next state s_{t+1}. In choosing the state actions, the agent aims to maximise its discounted cumulative rewards over time. In Q-learning, the learned action-value function, Q, directly approximates the optimal action-value function, Q^*, independent of the policy being followed. The Q-learning algorithm makes use of the Bellman equation to determine the one-step action-value, which is given by

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \beta \max_{a_{t+1}} Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)]. \tag{2.3}$$

The learning rate, α, balances between using new information and using previous knowledge, whereas β balances between the immediate and the future rewards.

Q-learning maintains a table that consists of estimates of the Q values, which are updated based on the rewards received using the TD approach. The estimate at time $t+1$ is updated as follows.

$$\begin{aligned} \hat{Q}_{t+1}(s_t,a_t) &= \hat{Q}_t(s_t,a_t) + \alpha[Q(s_t,a_t) - \hat{Q}_t(s_t,a_t)] \\ &= (1-\alpha)\hat{Q}_t(s_t,a_t) + \alpha[r_{t+1} + \beta \max_{a_{t+1}} Q(s_{t+1},a_{t+1})]. \end{aligned} \tag{2.4}$$

The Q-learning algorithm operates based on the generic RL strategy shown in Figure 2.10. The basic idea is to maintain estimates of Q values that represent a measure of goodness, which results from obtaining a reward for taking a certain action when in a given state.

Exploration and exploitation

Exploration plays a vital role in RL. "Of particular interest is the manner in which exploration strategies balance the need to learn more about the environment with the agent's desire to perform well based on its current knowledge of the environment" [108]. Although pure exploration might seem to be the best learning approach, it may waste a lot of time and computing resources in exploring areas of the environment that are irrelevant to the task. On the other hand, although exploitation might be the better

approach for maximizing the expected reward in one episode, exploration may produce the greater total accumulated reward in the long run [107]. There are a number of various approaches for finding the compromise between exploration and exploitation of the state-action space that exists, since it is not possible to both exploit and explore with any single action selection. In the case of wireless networks, knowing whether it is best to exploit or to explore depends on a complex relationship between the precise values of the estimates, channel uncertainties, and the number of remaining episodes at a given instant.

The main objective for exploration and exploitation in RL approaches is to minimize costs and the learning time. Typically, the smaller the learning time, the larger the cost, hence the poorer the agent's learning performance. By exploiting its current knowledge of the environment, the agent can identify areas of the environment that are worth exploring. In addition, the learning cost cannot be minimized without some degree of exploration of the environment to discover effective behaviours or actions. It is worth noting that an effective exploration strategy in one environment may not be best suited in another environment and it was observed that the negative impact of an ineffective or unsuitable exploration technique on both learning time and learning costs can be "enormous" [109]. The exploration methods for RL are classified into two categories namely, undirected exploration and directed exploration [108]. Undirected exploration techniques randomly explore the environment without considering the previous history of the learning process, whereas directed exploration methods consider the agent's history of experienced states to influence the parts of the environment that the agent will further explore. Following is a discussion of the common exploration and exploitation methods that are applied in Q-learning.

The ε-greedy exploration

The ε-greedy exploration method is the most popular undirected policy selection method [105]. Considering that Q-learning maintains estimates of action values, then at any given time instant there would be at least one action with the highest estimated value, and this is known as the *greedy* action. Selecting the greedy action amounts to the exploitation of the current knowledge of the environment, whereas selecting a non-greedy action means exploration since this enables the agent to improve the estimate value of the non-greedy action. The ε-greedy exploration method behaves greedily most of the time, that is, it selects the action with the highest estimated action value. However, in some time instances, with a small probability ε that is uniformly distributed, it selects an action at random. An advantage of this method is that in the limit as the number of episodes approaches infinity, it is ensured

that the Q value estimates converge to the actual value. A disadvantage of the ε-greedy method is that it equally selects the action to take among all the possible actions when exploring the action space and this does not perform well in learning problems where some actions incur large negative rewards or severe penalties.

Boltzmann-Gibbs exploration

The Boltzmann-Gibbs exploration, also known as the softmax exploration, is another undirected exploration method that is similar to the ε-greedy method, except that it uses a Gibbs or Boltzmann distribution for the random selection of actions, to enhance performance [105], [110]. The softmax method aims to address the drawback of the ε-greedy method by varying the action probabilities as a graded function of the estimated value. Here, the actions with the highest value estimates are assigned the greatest probabilities, and the remaining actions are weighted and ranked based on their estimated action values. When in state s_t, an action a is selected with probability $p(s_t,a)$ given by

$$p(s_t,a) = \frac{e^{\frac{Q(s_t,a)}{T}}}{\sum_{i=1}^{n} e^{\frac{Q(s,a_i)}{T}}}, \tag{2.5}$$

where T is a positive parameter known as the *temperature*, which controls the exploration process. High temperature values causes all the actions to be equiprobable, whereas low temperature values increases the probability of selecting one of the actions with the highest estimated values. Thus, as $T \to 0$, the softmax action selection method becomes the same as the greedy selection method. It is generally difficult to discern the better method between the softmax and the ε-greedy action selection methods, and this may depend on the task. Since both methods have only one parameter that must be set, that is, either ε or T, determining the values for these parameters depends on the confidence or acquired knowledge.

Bayesian Q-learning

Bayesian Q-learning is a directed action-selection method that uses probability distributions to strike a balance between exploitation and exploration by applying a Bayesian approach to Q-learning [111]. Unlike in undirected exploration methods that use point estimates of Q-values, the idea here is to keep and propagate distributions over the Q-values to make more informed decisions. There are two techniques that are used in Bayesian Q-learning, namely Q-value sampling and the myopic value of perfect information (myopic-VPI).

Q-value sampling is an exploration method that was initially proposed for solving bandit problems

[112], which was then extended to multi-state RL problems [111]. In Q-value sampling, the learning agent's knowledge of the rewards is represented as probability distributions. "These probability distributions depend on both the current expected reward and the current level of uncertainty about the actual reward" [110]. Using the calculated probability distributions, the action with the current optimal probability distribution is then selected.

Myopic-VPI is similar to Q-value sampling in that it is a direct way of evaluating the trade-off between exploration and exploitation using the current probability distributions over the rewards to control exploratory behaviour [111]. By using new information for information-gathering, the myopic-VPI provides an approximation of the action's utility in terms of the expected improvement in decision quality.

Directed vs undirected methods

The major drawback of the directed action-selection methods is that they require complex and time-consuming computations compared to undirected methods. On the other hand, the undirected action-selection methods can be easily implemented and are suitable for applications that require real-time decision-making; however, the values of the exploration parameters in undirected methods are usually determined correctly after many iterations. Another drawback of the undirected methods is that they do not consider the uncertainty of each action's expected reward.

The Q-learning algorithm

Although Q-learning is an off-policy TD approach, the policy still plays a role in determining which state-action pairs are visited and updated. The general procedure of the Q-learning algorithm is given in Algorithm 2.

In practice, the convergence to the optimal value function is attainable under the following conditions [106]:

- the state-action pairs are represented in a discrete manner, and
- repeated sampling of all actions is carried out in all states to ensure sufficient exploration.

These conditions are often infeasible, especially in scenarios with (possibly continuous) large state-action spaces. In such cases, a parameterized value function, $Q(s_t, a_t; \theta)$, is needed. Here, θ represents

Algorithm 2 Generic Q-learning algorithm

Initialise $Q(s,a)$ arbitrarily

For each episode do

Initialise s_t

For each step of episode (until s_t terminal) do

Choose a_t from s_t using the selected policy for Q

Take action a_t, determine reward r_{t+1} & next state s_{t+1}

$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \beta \max_{a_{t+1}} Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)]$

$s_t \leftarrow s_{t+1}$

End for

End

some parameters that define the Q-values.

2.6.2 Model-free Reinforcement Learning Strategies

"In model-free RL, the value function is estimated first, then the policy can be determined based on the estimated value function [113]."

2.6.2.1 MDP Solution Models

For both model-based and model-free RL methods, the extensions of the basic RL algorithms differ mainly in the way in which the feedback from the environment is used for improving the learning process [114]. These extensions have been shown to be efficient in scaling for problems with large state-action spaces and depending on the the problem, the decision-making process usually follows any of the variants of MDPs. According to [115], MDPs have become the de facto standard for learning sequential decision-making in most RL applications. As such, this section gives an introduction and an overview of the MDP models and algorithms that are predominantly applied in literature, some of which were used in the contributions of this research work. The MDPs have been utilised in providing solutions for a wide range of problems in the 5G networks. These include handling resource requests and resource scheduling to maximize the throughput/reward. In a MDP model, an agent makes a sequence of decisions with the goal of optimizing a given objective such as improving the performance or the accuracy [116]. Each decision is dependent on the current state s_t, and typically leads to a new state s_{t+1}. This is a purely Markovian process because the system's current state completely determines the future events. With the use of dynamic programming in combinatorial problems, it is generally difficult to obtain exact solutions for most MDP-based schemes.

2.6.2.2 Optimality criteria and discounted rewards

For an MDP, the learning process aims to gather rewards, and there are basically three approaches of obtaining optimality for MDPs, which are:

- *finite horizon model* - The agent optimizes its expected reward over a finite horizon of a given length.
- *discounted, infinite horizon model* - The rewards that are received in future are discounted according to how far away in time they will be received. That is, the earlier rewards are discounted less compared to the rewards obtained later. Here, a discount factor, β, $0 \leq \beta \leq 1$ is selected at each time instant and it ensures that the sum of discounted rewards is finite, even though the horizon may be infinite.
- *average reward* - This aims to maximize the long-run average reward. Here the averaged rewards can be discounted rewards and as the discount factor approaches 1, it becomes the same as the infinite-horizon discounted model.

"Choosing between these optimality criteria can be related to the learning problem. If the length of the episode is known, the finite-horizon model is best. However, often this is not known, or the task is continuing, the infinite-horizon model is more suitable" [115]. In addition, learning optimality is also concerned with the assurance that the learning process reaches a global or local optimum, the speed of convergence to the optimum solution as well as the complexity of the solution algorithm.

2.6.2.3 Partially observable Markov decision processes

The partially observable MDPs (POMDPs) are basically a combination of MDPs and the hidden Markov models. They find applications in modelling dynamic systems by connecting unobservable system states to observations. Here, the agent would not able to observe the system state directly and at a given discrete time it can only make observations that help it create a *belief state*. The belief state is determined by the probability distribution over the system states and the resulting solution of the POMDP is a policy that determines the optimal action for each belief state. There are also decentralized POMDP models that are usually implemented in complex scenarios considering the system uncertainty such as wireless sensor networks, autonomous navigation, and e-commerce. While the decentralized POMDP models provide elaborate frameworks for cooperative sequential decision-making under uncertainty, they have increased computational complexity.

2.6.2.4 Multi-agent Markov decision processes

The multi-agent MDPs are MDPs where the selected action at any given state consists of individual action components performed by cooperative agents. They find applications in multi-agent planning, where it is typically assumed that the heterogeneous agents may have their own set of actions for a given task to be solved. Although each agent might have its own objectives, the system utility for any given state is the same for all the agents [117]. The multi-agent MDPs are thus usually considered to be a general form of stochastic games, where repeated interactions between a number of participants whose environmental states change stochastically and the result depends on each participant's decisions.

2.6.2.5 Constrained Markov decision processes (cost criteria)

The constrained MDPs are a class of MDPs that are applied in cases where the agent or system has more than one objective [118]. Instead of maximizing one utility or minimizing a single cost that would be a function of the different objectives, in these models a single type of cost is minimized while keeping the other types of costs below some given bounds. As such, the constrained MDPs are also well-known as cost criteria approaches. The cost criteria approach uses value functions and discounted rewards during the action selection process. In the context of wireless communications, this enables the efficient allocation of resources to maximize the long-term reward in a cost-effective manner.

The cost criteria can be concisely defined by the tuple $\{\mathcal{S}, \mathcal{A}, \mathcal{P}, c, \mathbf{d}\}$, where $\mathcal{S}$ is the set of all possible states, $\mathcal{A}$ is the set of all possible actions, $c : \mathcal{K} \rightarrow \mathcal{R}$ is the immediate cost, which is related to the cost function that should be minimized, and $\mathbf{d} : \mathcal{K} \rightarrow \mathcal{R}^K$ is a K-dimensional vector of immediate costs related to a set of K constraints. For any policy π, and initial distributions ∂, the finite horizon cost is given by

$$C^T(\partial, \pi) = \sum_{t=1}^{T} \mathbb{E}_{\partial}^{\pi} c(s_t, a_t), \tag{2.6}$$

where T is the finite horizon and $\mathbb{E}$ represents the mathematical expectation. For the case of having discounted rewards, is an associated discounted cost, where for a given discount factor, β^t, it is given by

$$C_{\beta}^{T}(\partial, \pi) = (1 - \beta^t) \sum_{t=1}^{T} \alpha^{t-1} \mathbb{E}_{\partial}^{\pi} c(s_t, a_t). \tag{2.7}$$

For the finite horizon model, the cost function that is related to immediate cost d_k is then defined in a similar way as

$$D^T(\partial, \pi) = \sum_{t=1}^{T} \mathbb{E}_{\partial}^{\pi} d_k(s_t, a_t). \tag{2.8}$$

The constrained problem thus aims to find a policy that minimizes $C(\partial,\pi)$ subject to $D(\partial,\pi) \leq V$, where V is a predetermined threshold.

2.6.3 Applications of the RL Strategy in 5G Networks

This section focuses on the applications of RL in the 5G networks, particularly in the areas of network modelling and network solutions for the CRs in 5G networks. Efficient and scalable online learning algorithms are of particular interest and they have been successfully applied in various domains, including CRNs. In CRNs, RL has been found applicable for maximizing the long-term system performance by striking a balance between exploration and exploitation [90]. By virtue of not being affected by emotions during the entire process of perception, learning and reasoning, the application of RL in CRs is thus lucrative.

RL has been applied for solutions of various problems in CR-enabled networks and this section discusses the 5G networks-related problems. The stochastic and bursty nature of user data requirements in 5G NR presents a challenge of unbalanced traffic distribution. In addition, dynamics of wireless cellular environments, such as random user mobility, fading, shadowing and lesses due to channel effects, make it difficult to rely on a model of the network environment for solving optimization problems [18]. As such, it is desirable to take advantage of the TD-based RL approaches in cognitive wireless networks [119].

A distributed framework for the RAT selection in a CRN, which is based on RL, has been proposed [78]. The modular framework aims to satisfy the three user association perspectives mentioned in section 2.4.1 by applying RL for effective user association learning. In addition, it also applies unsupervised machine learning to determine the state of the user and network model.

Two RL approaches that aim to improve the learning efficiency for dynamic channel selection in CRs were investigated in [120]. The first approach, the pre-partition scheme, is a fast and efficient exploration scheme that reduces the interruption overhead that is presented by applying RL in communication systems. In this scheme, the UEs reserve spectrum resources randomly before transmission, which reduces the action space that the CR needs to explore, thus reducing the exploration phase significantly. In addition, the UEs in a local area will have a higher probability of avoiding each other. In the other scheme, the weight-driven exploration scheme, weights are attached to a used resource to create a

weight-driven probability distribution that shows the historical successful usage of a resource. The assigned weights correspond to the information that would have been learned by the cognitive users and they are constantly updated. This addresses the drawback of uniform random exploration that has been shown to be less efficient [121]. It was shown that by employing the pre-partitioning scheme, approximately 25% more activations are accepted during the exploitation stage, whereas with the weight-driven scheme 40% more activations are accepted, compared to using the uniform random exploration scheme.

2.6.3.1 MDP-based solutions

The decision making for CRs in 5G networks has been modelled by a family of MDPs or POMDPs [88]. The typical applications of MDP modelling that are found in the literature include energy harvesting, BS association for users in HetNets, and spectrum sensing as well as RA in CRNs. A POMDP model was applied in [122] to investigate the transmission power control problems in energy harvesting systems. Here, the state space comprised the limited battery state, the time-variant channels, and the packet transmission/reception states. The action was when a node sent a packet at a certain power level. From the partially observable MDP formulation, two near-optimum solutions were presented.

The authors in [123] proposed a cross-layer 5G NR scheduling with component carrier aggregation that compares different algorithms such as: (i) Markov decision process (MDP)-based cost reward packet selection, (ii) adaptive packet scheduling, and (iii) adaptive component carrier scheduling. The results obtained indicate that their cross-layer carrier aggregation scheme outperforms the other approaches in terms of system capacity, network reward and packet failure rate. However, it should be noted that the greedy solutions eagerly maximize the reward at each step and might be sub-optimal in the long run. Since the analytical solutions to such scheduling problems usually scale poorly in the time horizon, the TD-based RL methods have been shown to provide the best solutions.

A RL-based cooperative sensing method for SUs in a CRN operating in licensed frequency bands has been proposed [124]. The RL approach aims to address the challenge of the overhead that is usually incurred in the CRNs due to sensing delays for decision-making and increased control traffic. In the proposed scheme, the cooperative sensing problem is formulated as a finite-horizon MDP and an SU aims to:

- determine the optimal number of cooperative neighbours with minimum control traffic and under correlated shadowing,
- minimize the delay overhead, and
- improve the energy efficiency.

"Simulation results showed that the proposed reinforcement learning-based cooperative sensing method reduces the overhead of cooperative sensing while effectively improving the detection performance to combat correlated shadowing" [124].

2.6.3.2 Multi-arm bandit problems

The multi-armed bandits (MAB) and multi-player, and the multi-armed bandit game (MP-MAB) are signal processing tools that are capable of solving RA problems in wireless networks by exploring the channel environment and exploiting the known channels [88]. These reinforcement learning tools have been used to model RA problems for the D2D communication systems operating in an underlay to a cellular network [125]. Here, a distributed channel selection scheme for a MP-MAB model of the D2D users is proposed, where no-regret learning and calibrated forecasting are applied in the solution. The analytical results showed that the proposed approach gives vanishing regret compared to the global optimal solution and the empirical joint frequencies of the game converge to the set of correlated equilibria. In addition, the proposed solution is not only limited to channel selection problems, but it can be applied to a broad class of multi-player stochastic learning problems.

2.6.3.3 Q-learning solutions

A number of Q-learning-based frameworks for enhancing SON functions, which include mobility robustness optimization (MRO), mobility load balancing (MLB) and coverage and capacity optimization (CCO) for various distributed user scenarios, have been proposed in literature [78], [126]. In [126], Q-learning was applied to manage spectrum access and to control the interference in a CRN setup. The proposed framework was applied to solve the MRO and MLB problems in cognitive cellular networks. The MRO approach learns to optimize handover performance, whereas the MLB approach learns to optimize the instantaneous load among cells. Load balancing for MRO was also addressed in [127]. Here, the optimization problem was split into many small optimization problems and tasks, which were then solved by applying machine learning algorithms.

A multi-agent RL approach for spectrum management in the CRNs operating in licensed bands was proposed in [128]. The approach applies Q-learning to evaluate the choice of different transmission

parameters and it enables spectrum allocation and energy efficiency improvement by maximizing the long term reward. The authors also applied the Kanerva-based function approximation to improve the approach's ability to handle large CRNs and to evaluate the approach's effect on network performance. It was concluded that the RL-based spectrum management reduces interference to the PUs, while maintaining a high probability of successful transmission for the cognitive users. On the other hand, in [129], a channel access problem was formulated as a non-cooperative game where a single channel is used by only one user at a time. Considering the transmission delays, the channel switching distance for the SUs was limited to a certain scope, and the optimal access policy depended on the long-term behaviour of primary users as well as the actions of other secondary systems. The solution for the non-cooperative game was obtained through a multi-agent Q-learning algorithm, which requires neither the prior knowledge of channel dynamics nor negotiations among players.

Q-learning has been applied for interference management and cell outage compensation management in dense small cell networks [90], [119], [130]. In [119], a cell outage management framework that is comprised of solutions for outage detection and outage compensation for a HetNet with separate data and control planes is presented. By exploiting a large-scale collection of minimization of drive test reports for many UEs in control cells, the authors applied ML-based anomaly detection schemes for outage detection. In [131], the heterogeneous RL algorithms were applied for opportunistic spectrum access (OSA) in LTE-based HetNets. The RL model for the self-organization capability that was proposed enables the small cells in HetNets to use different learning strategies for autonomous OSA whilst reducing the interference between the two network tiers, with the aim of satisfying the QoS requirements.

Broadband access for mobile terminals with high mobility such as users in high-speed trains requires frequent seamless cell switching in harsh and dynamic environments. Bayesian Q-learning and the multi-agent theory were applied for spectrum management in high-speed railway wireless networks in [132]. The proposed multi-agent coordination Bayesian Q-learning algorithm proved to improve the probability of successful data transmission whilst reducing wireless spectrum handovers in a model of cascaded base station groups. In [133], the authors applied multi-agent RL for optimal channel selection for cognitive base stations in a network model similar to that of [132]. The selection of channels in [133] is done by fusing the interaction information between the different cognitive BSs, and this was shown to reduce the handover failure in typical high-speed train scenarios.

A cooperative Q-learning based spectrum sensing technique for SUs in an ad-hoc network was proposed in [134]. In this technique, each SU maintains a dynamic priority list of channels that is determined by the Q-value estimates from its own action-observation history, as well as from the spatial channel information that would have been shared by local neighbours. The simulation results showed that the proposed Q-learning approach improves the response time and the call block/drop rate with significantly less computing and scanning overhead compared to other contemporary reinforcement learning based approaches.

Most of the works on user association in wireless networks usually assume that the users instantaneously switch between the BSs, which is not realistically feasible considering that the prospective BS might not have free channels, hence transmission delays may be incurred. In [135], the problem of aggregated interference that is generated by multiple CRs at the receivers of PUs is addressed. The CR system considered by the authors is based on the IEEE 802.22 standard for wireless regional area networks (WRANs), and was modeled as a multi-agent system. Here, the multiple agents acted as the different secondary BSs that are responsible for controlling the secondary cells. A real-time multi-agent RL technique known as decentralised Q-learning was then proposed to manage the aggregated interference generated by the multiple WRAN systems. Two scenarios were considered: (i) users having complete information, or (ii) users having partial information about the environment. The multi-agent system then learned through direct interactions with the surrounding environment in a distributed fashion.

A cognitive spectrum management framework that aims to improve spectrum utilisation efficiency and the energy efficiency of each sensor node in a CR sensor network was proposed in [136]. Considering that the balancing of spectral efficiency and energy efficiency has become a critical challenge in today's resource-constrained networks, the channel characteristics and the energy efficiency of the CR sensor networks were analyzed from a RL perspective. A joint channel selection and power control spectrum decision algorithm that is based on distributed Q-learning was then proposed. In order to evaluate the performance of the proposed framework, the authors formulated a problem to determine an optimal Q-value subject to communication efficiency index. The RL action selection scheme was designed to solve the optimization problem. In the proposed learning model, each node can get the policy implemented by the other nodes to select the optimal policy by introducing distributed strategy estimation.

2.6.3.4 Overview of contributions in the literature

A summary and overview of the RL solutions and the corresponding problems that have been addressed for the 5G networks in the literature is given in Table 2.4.

Table 2.4. Overview of some applications of RL for network optimization in 5G networks.

Network Problem	5G Network Application	RL Method	References
User association and Coverage optimization	Interference and cell outage compensation management in dense networks	Q-learning	[90], [119], [130]
	User association in CR systems	Decentralized Q-learning	[135]
	Spectrum sensing in CRNs	Q-learning	[134]
	Switch migration in distributed SDN controllers	Multi-agent RL	[137], [138]
Resource management	RAT selection in CRNs	Q-learning	[78]
	Dynamic channel selection in CRs	Weight-driven action selection	[120]
	RA in D2D networks	MAB, MP-MAB	[125]
	Spectrum access and interference control in CRNs	Q-learning	[126], [128], [129]
	Spectrum efficiency and energy efficiency maximization in CRNs	Distributed Q-learning	[136]
	Routing in VoIP systems	e-RLRP	[139]
	Spectrum management in high-speed railway networks	Bayesian Q-learning, multi-agent RL	[132], [133]
Cost minimization/ Revenue maximization	User scheduling and cooperative sensing in CRNs	MDP-based cost minimization	[123], [124]

2.7 DEEP LEARNING

Deep learning (DL) is a form of machine learning that utilises neural networks (NNs) to transform a set of inputs into a set of outputs. DL has shown great potential in solving tasks involving complex and high-dimensional data, thus enabling significant progress in the fields of computer vision and natural language processing. Although the RL methods have been shown to provide efficient and scalable online learning algorithms for solutions to problems in communication systems, "the advent of deep learning has had a significant impact on many areas in machine learning, dramatically improving the state of the art in tasks such as object detection, speech recognition, and language translation" [140]. An essential property of DL is that using deep neural networks (DNNs), compact low-dimensional representations/features of high-dimensional data such as images, text, and audio, can be found autonomously [141]. The statistical flexibility and computational scalability properties of DL enable it to harness information in large and rich datasets.

2.7.1 Deep Learning Concepts

Deep learning is basically defined as a process that learns the relationship among several variables as well as the knowledge governing the relationship [142]. DL is a ML technique that employs DNNs, where a DNN is a multi-layer neural network that contains two or more hidden layers. Each hidden layer consists of a non-linear transformation of the output of the previous layer, and the sequence of these transformations leads to learning different levels of abstraction. All these layers are trained to minimise a loss function, which represents the empirical error. Thus, there is need for a training data set that is required for the initial training phase of a neural network. DL also consists of networks that are capable of performing unsupervised learning from unstructured and unlabelled data sets by utilizing a hierarchical level of artificial neural networks as was shown in [143].

An example of a neural network is the fully connected feed-forward network shown in Fig. 2.12. The first layer is given the input values/features in the form of a column vector $\mathbf{x}_0$. The values of the first hidden layer are a transformation of the input values through a non-linear parametric function given by

$$\hat{f}(\mathbf{x}_{l-1}, \theta_l) = \sigma(\mathbf{W}_l \mathbf{x}_{l-1} + \mathbf{b}_l), \tag{2.9}$$

where σ is the activation function, $\mathbf{W}_l$ and $\mathbf{b}_l$ are the weights matrix and bias vector, respectively, at the l^{th} hidden layer.

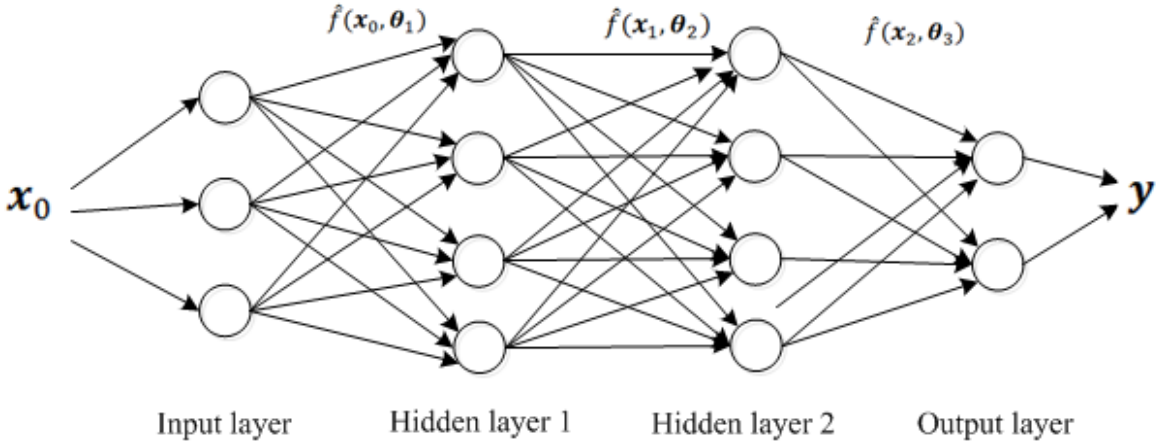

Figure 2.12. Example of a fully connected feed-forward neural network with two hidden layers in addition to the input and output layers.

In addition to the feed-forward neural networks, the current applications of DL in wireless communications employ many other various types of neural networks. Each variation has its own advantages, depending on the application. For example, one can provide a good trade-off between bias and overfitting in a supervised learning scenario. The number of layers in a neural network can be arbitrarily large, and the current trend is to have an ever-growing number of layers with some tasks having more than 100 layers [106]. The two types of neural networks that are mainly used in DRL are convolutional neural networks (CNNs) and recurrent neural networks (RNNs). The translation invariance property of CNNs makes them well suitable for image data. A convolutional layer can be seen as a special type of feed-forward layer, where many weights are set to be zero, that is, they are not learnable, and the other weights are shared. On the other hand, the RNNs are well suited for sequential data. Here, several variants are advantageous in different scenarios.

The most common method for optimizing the parameters of a neural network is the stochastic gradient descent (SGD) method, which employs the backpropagation algorithm. In the backpropagation process of the SGD method, a gradient based on randomly selected samples is initially calculated before updating the model parameter along the negative gradient direction of the current iterate. This update of the algorithm's internal parameters θ is given as follows:

$$\theta \leftarrow \theta - \alpha \nabla_\theta I_s[f], \tag{2.10}$$

where α is the learning rate, and $I_s[f]$ is the empirical error. Due to its low computational complexity per iteration and its ease of implementation, the SGD method is used in solutions for many optimization

problems in ML and signal processing.

2.7.2 Activation Functions

After selecting the appropriate DL model to use, one challenge to consider is how to handle gradient flow in a network. Some gradients tend to be steep in certain directions while they are gradual or even zero in other directions. This makes the optimal selection of learning parameters a complex challenge. An activation function is a transfer function that determines the state of a node in a NN as a function of its bias and the weighted sum of inputs from other nodes [144]. The output of an activation function determines whether a neuron or a neural node is fired or not and this is achieved by manipulating the input to a neural node through some gradient processing such as the SGD method to produce an output [145]. Table 2.5 gives the different types of activation functions that are used in DL methods. As

Table 2.5. List of common activation functions used in DL applications.

Function Name	**Activation Function**	**Range**
Linear function	$f(x) = x$	$(-\infty;\infty)$
Rectified linear unit (ReLU)	$f(x) = max(0,x)$	$(0;\infty)$
Softplus function	$f(x) = log(1+e^x)$	$(-\infty;\infty)$
Sigmoid function	$f(x) = \frac{1}{1+e^{-x}}$	$(0;1)$
Softsign function	$f(x) = \frac{x}{1+\lvert x\rvert}$	$(0;1)$
Gibbs Softmax function	$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$	$(0;1)$

shown in Table 2.5, the activation function can be either linear or non linear, and this depends on the application domain. The typical DL architectures use linear activation functions in the hidden layers and a non-linear function at the output layer. For the linear models, the mapping of inputs to outputs at each node is given by

$$f(x) = \mathbf{w}^T\mathbf{x} + b, \tag{2.11}$$

where $\mathbf{w}$ is the matrix of weights, $\mathbf{x}$ represents the input vector, and b is the bias value of the neuron node. The output neural node takes the sum of the weighted inputs and applies it to the non-linear activation function ϕ and the output is given by

$$f(x) = y = \phi\left(\sum_{i=1}^{n} w_i x_i + b\right). \tag{2.12}$$

In multilayered networks such as in the DNNs, these outputs are fed into the subsequent layer until the final output is obtained. The non-linear activation function enables the learning of high order polynomials in deeper networks.

2.7.3 Loss Functions

The SGD method that is used in training the NNs requires that ones chooses a loss function during the design of the model. In the context of an optimization algorithm, the objective function is either minimized or maximized; when it is minimized, it is also called the cost function, the loss function, or the error function [146]. A loss or cost function is used to evaluate the performance of an ML model by determining the difference between the actual value and the estimated value. This difference is estimated by running the ML model iteratively. "In each iteration, the estimated prediction (i.e., unknown value of y) is compared with the ground truth (i.e., the known value of y), with the objective of finding parameters and structures or weights that minimize the cost function" [114]. In other words, a loss function maps the ML decisions to their associated costs. In determining the error during the optimization process, it is essential that the loss function accurately represents the design goal because a poor error function leads to unsatisfactory learning results.

The loss functions are categorised as either *regression loss* or *classification loss*. The regression loss functions predict values that are continuous in nature, such as predicting stock prices [147]. On the other hand, the classification functions predict a discrete class output by dividing the dataset into different and unique classes based on the different parameters, for example, classifying a mail as spam or not spam. Table 2.6 gives the loss functions that are mainly used in ML algorithms for communication network problems. For the functions in the table, y_i represents a datapoint and $\hat{y}_i$ is its predicted value, $h_\theta(x_i)$ is the predicted value post hypothesis, and $P(x)$ and $Q(x)$ are distributions for the Kullback Leibler Divergence loss. These distributions can be either discrete or continuous, and the function expression for the two cases is shown in Table 2.6.

Table 2.6. Loss functions mainly used in ML algorithms for wireless communications.

Category	**Function Name**	**Loss Function**
Regression	Mean Squared Error (MSE)	$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
	Mean Squared Logarithmic Error (MSLE)	$MSLE = \frac{1}{n}\sum_{i=1}^{n}(log(y_i) - log(\hat{y}_i))^2$
Classification	Cross Entropy Loss	$J = -\sum_{i=1}^{N} y_i log(h_\theta(x_i))$
	Kullback Leibler Divergence	$D_{KL}(P\|Q) = \left\{ \begin{array}{l} -\sum_x P(x).log\frac{Q(x)}{P(x)} \\ -\int \sum_x P(x).log\frac{Q(x)}{P(x)}.dx \end{array} \right\}$

2.7.4 Deep Learning Applications in 5G networks

The heterogeneous data that is generated by 5G mobile networks leads to a wide range of problems that have been shown to be challenging to solve with traditional machine learning tools. As aforementioned, DL is capable of handling big data by employing hierarchical feature extraction and eliminating domain expertise [148]. DL thus enables efficient information distillation and abstract correlations with reduced pre-processing overhead. This section highlights some of the novel applications of DL in 5G networks that have been presented in literature, with the main focus being on the network control schemes. Neural network-based control, also known as analysis-based control is gaining traction in mobile network control. Contrast to RL, which learns through interaction with the environment and acts directly on the environment as illustrated in Fig. 2.10, the analysis-based control extracts useful information and passes it to the agent to execute the actions. Figure 2.13 illustrates the operation of the analysis-based control approach in the context of wireless mobile network environments.

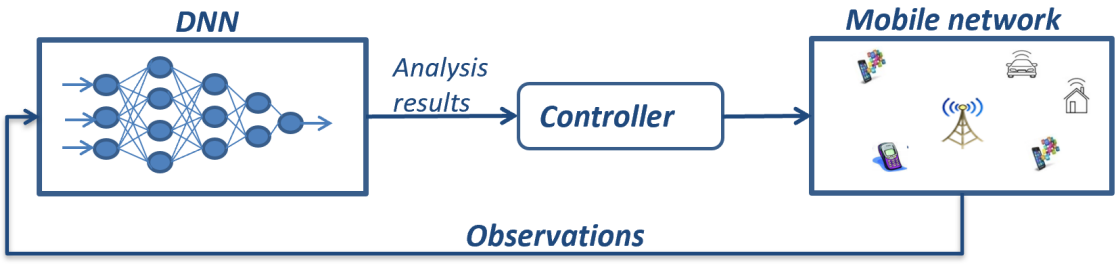

Figure 2.13. Illustration of analysis-based control in wireless networks, where the analysis results from the DNN are used by the controller to determine appropriate actions to take.

There are many examples of iterative algorithms that asymptotically reach an optimum or near-optimum solutions to optimization problems in wireless communications, but converging after many iterations or complex operations in each iteration. Such algorithms might not be practically useful in wireless networks that have very low latency or real-time implementation requirements. In such scenarios, DL has been applied to approximate known but computationally complex algorithms [149], [150]. Such applications of DL enable the neural network to learn how to make algorithmic shortcuts to obtain a good trade-off between accuracy and computational complexity.

Table 2.7 summarizes some of the contributions that apply pure DL for the network control problems in 5G networks, as was identified from [148] and [151].

Table 2.7. Summary of main applications of DL in 5G network control problems.

Network Domain	Application	NN model	References
Resource allocation	5G broadband TV service	RNN	[152]
	RA in 5G heterogeneous D2D networks	CNN	[153]
	Load balancing in 5G ultra-dense networks	Deep LSTM	[154]
Routing	Virtual routing in wireless ad-hoc networks	NDC	[155]
Network optimization	Channel estimation, SE maximization, and power control in massive MIMO	CNN, RNN	[156], [157], [158], [159]
	Load balancing for IoT	DBN	[160]
	NOMA	RNN	[161]

2.8 THE DEEP REINFORCEMENT LEARNING STRATEGY

The DRL technique is a sub-field of ML that combines both DL techniques and the RL strategies [106]. Since the RL part considers the problem of a computational agent learning to make decisions by trial and error, the incorporation of the DL technique allows the DRL agents to take actions based on learning from unstructured input data. One of the greatest achievements of the DRL strategy is in sequential decision-making under uncertainty, even with larger dimensions of the state and action spaces. Here, a learning agent can perform sequential decision-making tasks under uncertainty with the aim of taking actions inside an environment to maximize some cumulative reward. “In particular, during real-time learning, the obtained experiences will be stored and used as data to train the neural network” [162]. The trained neural network then aids the agent to make optimal decisions in real-time. Thus, unlike the DL technique which typically performs once-off training of the neural network, in DRL the neural network is trained frequently based on the new experiences obtained from real-time interactions with the environment.

Due to its ability to learn different levels of abstractions from data, DRL can handle complicated tasks with low or no prior knowledge. As such, DRL has found applications in video gaming, robotic control and navigation, finance, and smart grids [106], [140]. Although the combination of DL and RL algorithms has been shown to improve performance by providing simple and efficient solutions

[156], there are a number of challenges that arise in applying the DRL algorithms. These include, among others, the exploration of the environment or generalising a good behaviour in a (slightly) different context is very complex. The general architecture that shows the elements of DRL that are implemented in DRL algorithms is shown in Fig. 2.14.

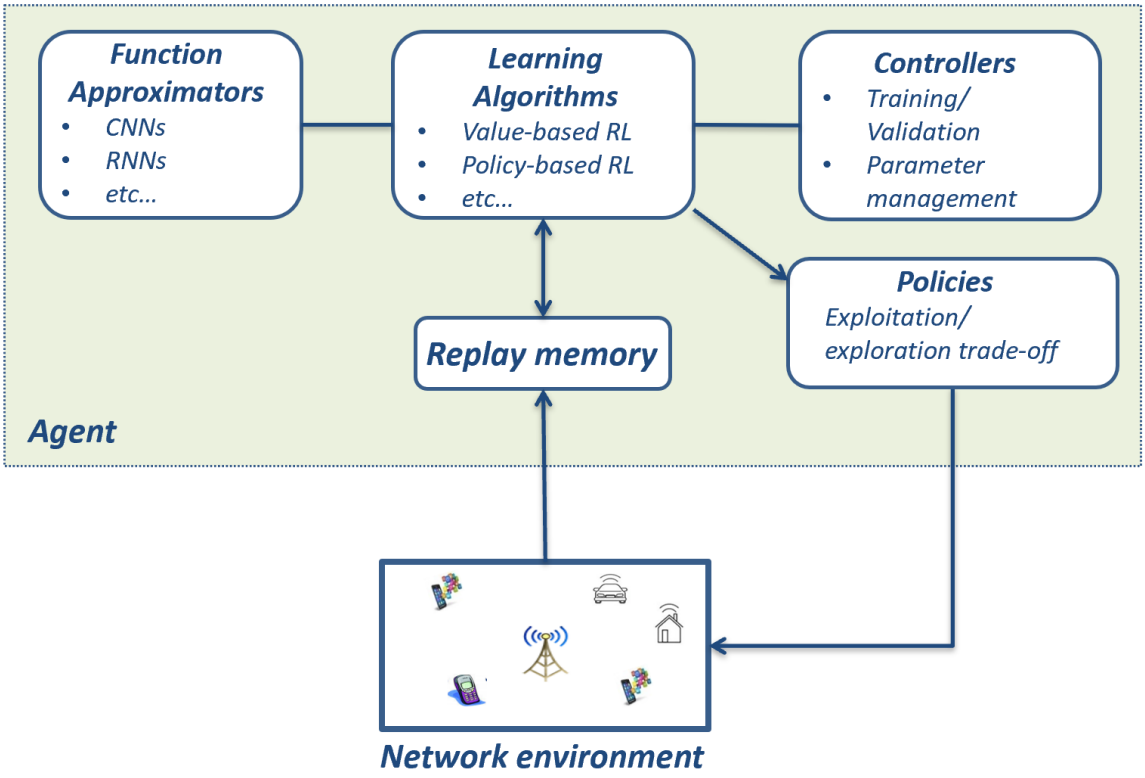

Figure 2.14. Elements of DRL in a general wireless network architecture showing the components of the agent (in green shading) and how they interact with the network environment.

The incorporation of DL techniques in RL to create the field of DRL algorithms has been used to develop intelligent autonomous agents in wireless networks. Consequently, powerful DRL models that are capable of scaling up to solve problems that could not be previously solved have been created. As shown in Figure 2.14, the learning agent makes use of the relevant function approximators in the DNN, whilst it gathers experience via a selected exploration/exploitation policy in implementing the associated RL algorithm. The agent can also use a replay memory to store its experience, which can be accessed for processing at a later stage. The data training and validation is carried out by the controllers, which also control the policy selection for the RL algorithm. As such, DRL is suitable for problems where decisions should not be solely based on the state of the network environment. Here,

an NN is used to extract additional information such as traffic forecasts, which subsequently aids the decision-making process [148].

2.8.1 DRL Methods

This section discusses the RL methods that scale to deep neural network function approximators. These methods enable learning in many various challenging sequential decision-making tasks by learning directly from rich high-dimensional inputs [106]. In particular, the focus is on value-based algorithms, which aim to build a value function that defines a policy. Here, the most popular variants of the Q-learning algorithm that have been applied for learning in wireless network problems, which use parameterised function approximators, are considered.

2.8.1.1 Fitted Q-Learning

The fitted Q-learning algorithm starts with randomly initialised Q-values, $Q(s_t, a_t; \theta_0)$, where θ_0 represents the initial parameters. The initial parameters are set such that the initial Q-values are relatively close to 0 to avoid slow learning. The optimal Q-value estimate at the k^{th} iteration is then given by

$$\hat{Q}_k(s_t, a_t; \theta_k) = (1-\alpha)\hat{Q}(s_t, a_t; \theta_k) + \alpha[r_{t+1} + \beta \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_k)]. \tag{2.13}$$

Neural fitted Q-learning [163] is a technique where the state can be provided as an input to the Q-network, and for each of the possible actions, a different output is given. This enables the computation of $\max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_k)$ in a single forward pass in the neural network for a given state, which makes the structure relatively more efficient. The Q-values are parameterised with a neural network and the parameters θ_k are updated using the SGD method through minimizing the square loss. Thus, the Q-learning update amounts to an update of the parameters as follows:

$$\theta_{k+1} = \theta_k + \alpha(\hat{Q}_k(s_t, a_t; \theta_k) - Q(s_{t+1}, a_{t+1}; \theta_k))\nabla_{\theta_k} Q(s_{t+1}, a_{t+1}; \theta_k). \tag{2.14}$$

This ensures that $Q(s_t, a_t; \theta_k)$ approaches $Q^*(s_t, a_t)$ after many iterations, considering that the neural network is well-suited for the task and sufficient data has been gathered. However, as the weights are updated, the optimal estimate changes, and due to the generalisation and the extrapolation abilities of the NNs, the fitted Q-learning approach can accumulate large errors at various points in the state-action space [106]. Thus, the mapping of the action-values via the Bellman operator in (2.3) does not guarantee convergence. In addition, the use of function approximators tends to overestimate the Q-values due to the max operator. These risks of experiencing instabilities or value overestimation require special care to ensure proper learning.

2.8.1.2 Deep Q-Networks

The deep Q-network (DQN) is an algorithm that leverages ideas from neural fitted Q-learning, which aimed to improve performance in an online setting for a variety of ATARI games by learning successful policies directly from high-dimensional inputs [164]. To limit the instabilities that are experienced when a nonlinear function approximator is used to represent the Q-function in neural networks, the DQNs apply the following two heuristics:

(i) Introduction of experience replay, which randomises over the data to remove correlations in the observation sequence and smooth over the changes in the data distribution.

(ii) Iterative updating of the Q-value estimates to approach the optimal values that are only updated periodically, thus reducing correlations with the target. This enables a relatively larger update of the parameters, while having an efficient parallelization of the algorithm.

2.8.1.3 Double Deep Q-Networks

The use of the same values for action-selection and action-evaluation in the max operation in equations (2.3) and (2.13) make Q-learning to be more likely to select overestimated values in case of inaccuracies or noise [106]. This results in overoptimistic value estimates and an upward bias that is induced by the DQN algorithm. The double deep Q-network (DDQN) approach is a double estimator method, where two estimates are used for each variable [165]. This allows for uncoupling of the selection and the evaluation of an estimator, and regardless of the source of errors in the estimated values, the positive bias in estimation of the action values is removed.

In DDQN, the optimal Q-value estimate is determined by [166]

$$\hat{Q}_k^{DDQN}(s_t, a_t; \theta_k) = (1-\alpha)\hat{Q}(s_t, a_t; \theta_k) + \alpha[r_{t+1} + \beta Q(s_{t+1}, \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta_k); \theta_k^-)], \quad (2.15)$$

which results in two sets of parameter weights, θ_k and θ_k^-, being used for each update. One set of parameter weights is used to determine the greedy policy and the other determines its value. It can be noted that the action-selection is still due to the online parameter weights, θ_k; however, the second set of weights, θ_k^-, is used to evaluate the value of the action-selection policy. This leads to a reduced overestimation of the Q-values and improved stability, hence providing better performance compared to DQN.

2.8.2 Applications of Deep Reinforcement Learning in 5G networks

In DRL, the NNs are the agents that learn to maximize the reward from the state-action transitions by assigning weights to approximate a function that relates the input and the output [140]. The learning comprises finding the appropriate weights by iteratively adjusting them along the gradients that minimise the error. In wireless communication networks, an agent can use CNNs to establish the state of the network environment, that is the channel conditions, through signal recognition. In such cases, the action to be taken in a current state depends on the future states and the actions, and here the CNN ranks the possible actions before selecting the best action. The policy agent then maps a state to the best action.

At the beginning of the DRL process, the neural network coefficients can be initialised randomly. Then, using the feedback from the environment, the neural network uses the difference between its expected reward and the actual reward to adjust its weights. This feedback process is similar to the backpropagation of error in supervised DL. However, unlike supervised DL, which begins with the knowledge of the ground-truth labels that the neural network aims to predict, DRL relies on the environment to send it a reward in response to each selected action. These rewards can be varied, delayed or affected by the unknown variables, which introduces noise in the feedback loop. Having no need for hand-crafted features to train the algorithm makes DRL promising in applications where the agents need to act on sequential data for time series modeling, where the RNNs are the widely used models.

A DRL algorithm for deriving an optimal time scheduling policy for an SU gateway in an RF-powered backscatter CRN was proposed in [167]. To deal with large state and action spaces, the authors used a double deep Q-network to enable the gateway to learn optimal policies for scheduling the transmissions of multiple SUs. The simulation results showed that the proposed DRL algorithm enables the gateway to learn an optimal time scheduling policy, which gives throughput that is significantly higher compared to non-learning algorithms. "However, the proposed DRL algorithm still requires each secondary transmitter to report its status to the gateway, which introduces communication overhead."

The control of the trajectories of the unmanned aerial vehicles (UAVs) that provide coverage to vehicles in a cell-free network was investigated in [168]. Here, the authors proposed a DRL framework for the optimization of network performance with the aim of maximizing the coverage to highly mobile vehicles with the minimum number of UAVs and with minimum energy consumption. The DRL

framework applies an actor-critic method, where a central agent is trained to observe the network environment and make decisions that determine the minimum number of UAVs and their trajectories. The optimization constraints include maintaining an acceptable QoS for each vehicle, and limited energy for each UAV. Another DRL-based solution for an MDP model RA problem in V2V networks was proposed in [169]. In this decentralised resource management scheme, the agents, which are the V2V links aim to select the resource block and transmission power to ensure satisfaction of their latency constraints while minimizing the interference.

Another DRL framework for a cell-less network architecture was proposed in [170], where the distributed APs are dynamically clustered. Each cluster acts as a single virtual AP within a distributed antenna system with a successive interference cancellation-enabled signal detection scheme and an inter-user-interference-aware receive diversity combining scheme. The joint optimization problem of maximizing the UE rate and AP clustering is then solved by applying a hybrid DRL scheme, namely, the hybrid deep deterministic policy gradient double deep Q-network.

Table 2.8 gives a summary and an overview of some of the DRL-based solutions for the 5G NR applications that have been proposed and presented in literature. Here, the focus is on the contributions on the network optimization solutions as aforementioned as well as those identified in [162].

Table 2.8. Summary of some applications of DRL for network optimization in 5G networks.

Network Problem	**5G Network Application**	**DRL Algorithm**	**References**
Throughput maximization	HetNets	DDQN	[171]
	CR-based IoT	DQN	[172], [173], [174]
	HD video streaming	DQN	[175], [176]
	Cache-enabled interference alignment networks	DQN	[177]
Resource management	D2D communications	DQN	[168], [169]
	Cognitive satellite communications	DQN	[178], [179]
	Vehicular networks with MEC	DDQN	[180], [181]
Cost minimization	MEC and caching	DQN	[182]
	Fog computing	DQN	[183]

2.9 REINFORCEMENT LEARNING-BASED IAB SYSTEM SOLUTIONS

Having given an overview of the ML schemes and their applications in 5G networks, this section focuses on the application of ML particularly in IAB networks. This aims to further reveal the research gap that was stated in Chapter 1. It should also be noted that some of the published contributions of this research are discussed in this section.

2.9.1 User Association

In [184], the RL strategy was applied for association of UAVs at the network edge in a mm-wave IAB architecture. The authors considered the dynamic environment of a UAV in motion to develop a context-aware reinforcement learning-based solution for performance optimization. In addition to the fixed SBSs, the considered model also leveraged the use of mobile UAVs as relay SBS nodes. It was shown that with beamforming employed for both access and backhaul, a RL scheme based on reference signal receive power (RSRP) provided an improvement in performance in terms of the achievable data rate compared to alternative association algorithms.

As part of the contributions of this research work, a fully autonomous and distributed RL scheme that aims to improve user satisfaction in a congested IAB network that uses instantaneous load-based bandwidth partitioning was proposed in [185]. Here, a cognitive engine judiciously determines network congestion to determine when to initiate bandwidth split. By considering network context in the form of the neighbouring SBSs' loads, reactive load balancing was then applied using exponentially weighted moving average. The results showed that even at higher neighbouring load intervals, the load at an SBS can be kept below a predetermined threshold. This gives insight on how much extra bandwidth needs to be shared among different SBSs according to the instantaneous load-based partitioning scheme.

A fully autonomous and distributed DRL scheme for user association in a congested IAB network was proposed in [186], where the model applies instantaneous load-based bandwidth partitioning. The framework of the proposed DRL scheme that has been applied in this work for context access resource management in a mm-wave multi-hop IAB network architecture, which builds on the model from [185], is shown in Fig. 2.15. In the model, the decision maker that continuously monitors the environment in an IAB system implements a DNN, coupled with a RL strategy, to make the best decision of action selection in any given state of the system. Such work has shown that the DNNs are crucial for optimal action selection in highly stochastic and dynamic environments such as mm-wave IAB networks, where the reward value of taking an action depends on future actions and states.

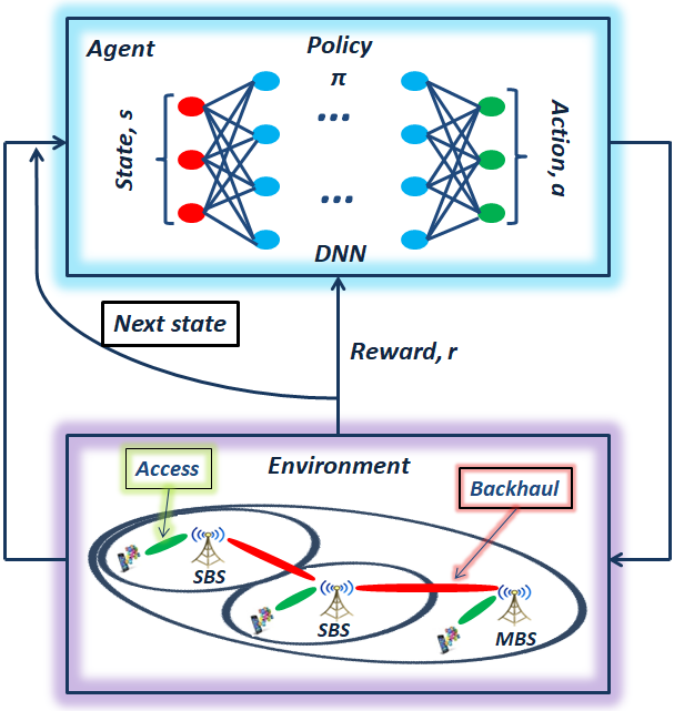

Figure 2.15. Illustration of the DRL model for UE association in a multi-hop IAB network showing the components of the agent and how they interact with the IAB environment.

2.9.2 Link Scheduling

To address the problem of flow allocation and link scheduling in mm-wave IAB networks, a DRL-based solution was proposed in [187]. The authors focused on tackling the link unreliability challenge that is posed by realistic harsh environments in the mm-wave transmission frequencies. The approach aimed to handle the dynamics of the network links and learn the best RA method in networks with intermittent links. This was motivated by the principle that applying traditional optimization algorithms in dynamic mm-wave environments would require that the optimization be performed each time the network undergoes a change, which make the time consuming solutions infeasible. For their solution, they proposed an online and an offline version of an advantage actor critic (A2C)-based approach to solve a two-part problem. The proposed approach first solves a link pattern generation problem that considers hardware constraints. The actor-critic method is then applied for the flow allocation and pattern scheduling problem, where a RL agent activates sequences of per-slot patterns for data transfer from the MBS to the UEs. The results showed that the proposed algorithm can automatically and promptly adapt to environmental changes, thus recovering from failure more efficiently compared to the traditional optimization approaches.

Another DRL approach for addressing the link scheduling problem in multi-hop IAB networks was proposed in [188]. Here, the authors aimed to minimise the end-to-end delay experienced by a typical packet, where the system is modelled as a network of queues with a centralised scheduler. The problem was then formulated as an MDP over a continuous action space and it was solved by applying the deep deterministic policy gradient algorithm. The results of system-level simulations showed that the proposed scheduling scheme performed better than the backpressure scheduler and the max-min delay round-robin scheduler in terms of packet delay.

With the aim to deal with the blockages that are typically experienced in mm-wave IAB networks, the authors in [189] proposed a hybrid DRL-based framework to solve a joint routing and scheduling problem. The DRL framework applied LSTM to implicitly capture the irregularities of the environmental dynamics of the mm-wave transmissions. This was applied to solve the optimization problem, which was formulated using column generation. The authors implemented both offline and online LSTM-based DRL approaches, where the online version of this approach was developed to increase robustness and account for system dynamics on-the-fly by adjusting the training to adapt to new changes. The results obtained indicated that the proposed strategy outperforms the optimization approaches that are typically used in multi-hop networks by automatically adapting to environmental changes.

2.9.3 Resource Allocation

A scalable and model-free framework for solving the spectrum allocation optimization problem in mm-wave IAB networks was developed in [190]. The model applied RL and DNN techniques to address the challenge of dealing with very large solution spaces when solving RA optimization problems in dense IAB networks. Q-learning, was shown to provide good performance in small-sized networks but it becomes less efficient as the network scales up. Owing to this limitation, the RL technique was combined with the DNN techniques to leverage the DNN capability of providing more accurate Q-value estimates in large state spaces. This deep reinforcement learning technique is capable of handling dynamic systems with varying network setups and UE requirements. The scalable and model-free framework developed in [190] performs dynamic spectrum allocation to maximize the sum log-rate while satisfying the UE demands. The authors applied two DRL-based algorithms to solve the optimization problem, namely the double deep Q-network and the actor-critic spectrum allocation (ACSA) algorithms. It was shown that the ACSA algorithm gave better performance in terms of convergence efficiency and system sum log-rate.

To address the problem of joint spectrum and power allocation in the IAB networks, the authors of [191] proposed an advanced DRL approach, namely spectrum allocation and power management scheme by leveraging DDQN (SAPM-DDQN). The joint problem was formulated as a mixed integer non-linear optimization problem and the SAPM-DDQN is applied in a decentralized manner, which reduces the transmission overhead compared to the centralised approach. The strengths of this approach are that (i) the users' privacy is enhanced since no information exchange is required, and (ii) training of the model can be performed offline, which eases communication overheads. The performance of the SAPM-DDQN scheme was evaluated in terms of the average rate of the UEs for various network scenarios and this was compared to a DQN approach. The simulation results showed that the proposed SAPM-DDQN significantly outperforms the other method.

2.9.4 Summary and Overview

A summary and overview of the RL solutions that have been proposed for various problems in mm-wave IAB networks is given in Table 2.9.

Table 2.9. Summary of RL-based solutions for problems in mm-wave IAB networks.

Network Problem	**Objective**	**RL/DRL Method**	**References**
Congestion control	Instantaneous load balancing	Q-learning	[185]
	Maximizing throughput and satisfying QoS requirements of UEs	Individual and Cooperative learning	[186]
Link scheduling	Maximize throughput considering fairness	A2C-based	[187]
	Minimise end-to-end packet delay	DDPG	[188]
	Joint coordination between access and backhaul to maximize throughput	LSTM-based DRL	[189]
Resource allocation	Maximize throughput for all the UEs	ACSA algorithm	[187], [189], [190]
	Joint spectrum and power allocation	SAPM-DDQN	[191]

2.10 DYNAMIC BACKHAUL ROUTING USING MACHINE LEARNING

As the MNOs extend their 5G NR capacity beyond the initial 5G market launches, they need to implement high-bandwidth backhaul solutions to the AP sites in a fast and cost effective manner. The

5G NR deployment in mm-wave frequencies increases the challenge of securing optimum backhaul solutions to the AP sites as the demand for access to the wireless network grows exponentially. In high-throughput mm-wave IAB networks, from a reliability perspective, it is important to ensure that each IAB node is be able to continually provide access coverage and service even when the active backhaul routes are temporarily unavailable. In this regard, the 3GPP has endorsed the use of topology adaptation for the IAB networks, which aims to autonomously reconfigure the backhaul network without service disruption nor packet losses during the reconfiguration. The IAB topology adaptation is comprised of integrating a new IAB node into the topology, removing a node from the topology, detecting backhaul link overload, worsening of backhaul link quality, link failure, or other related events. This technique enables the provision of a reliable, cost-effective, and scalable wireless backhaul solution that is capable of providing enhanced mobile broadband in mm-wave IAB networks. This makes IAB topology adaptation attractive for use in the 5G networks.

There are many research contributions that have focused on the optimization of network resources with the objective of efficiently distributing the scarce radio resources between access and backhaul. However, if the NR network is fully optimized to guarantee QoS, and if the transport network gets congested, the backhaul might create large unfairness among the backhauled traffic, regardless of the UEs' QoS class. The application of ML strategies for backhaul optimization has been considered, especially for context switching in mm-wave IAB networks.

2.10.1 Supervised Learning-based Approaches

The application of ML techniques in traffic handling problems in wireless networks is not new, especially the supervised learning techniques. Supervised learning models for directly learning paths for high-throughput dynamic packet routing have been proposed in [192] and [138]. The approach in [138] first predicts future traffic before optimizing the routing plan using the predicted values. "However, simulation results show that this approach might be ineffective" [193]. On the other hand, the approach in [192] assumes a central controller, which uses the information gathered from the whole network to train a different model for each source and destination pair in the network. The solution for the congestion optimization problem is then provided by a heuristic algorithm.

The main challenge that these aforementioned protocols addresses is deciding the best path to be taken by traffic from its source to the destination, under certain constraints. This is the routing problem.

When more than one path between source and destination exists, the decision-making process should follow some criteria. Routing decisions can be made in various ways depending on the desired objective and the objective function thereof. As a result, various optimization techniques can be applied, such as the mixed integer linear programming. However, obtaining such optimal solutions is usually computationally complex and it cannot be typically performed in real time. Instead, heuristics for these problems are often created but designing them is non-trivial in many cases. As such, in addition to the supervised learning approaches, DL and DRL approaches have also been proposed for solving routing problems.

2.10.2 Reinforcement Learning-based Approaches

Since network traffic in 5G wireless networks varies with time and space, there is need for dynamic resource management to avoid uneven load distribution among nodes. The dynamic AI-based switching of backhaul routes can provide a balanced load distribution among SBSs in IAB networks, which can improve the QoE that is provided to the UEs. When an SBS becomes congested on the access and it cannot admit-to-serve any more traffic, the traffic can be admitted to the backhaul portion of the bandwidth and routed to the MBS. This process requires that the SBSs check if the traffic flow to be admitted meets the admission control conditions. The RL-based decision-making for context switching generally involves training the model adaptively in various network environments [194], and this technique has been applied for switch migration in the distributed SDN controllers [137]. In addition to the supervised learning approach, [138] also proposed a RL-based approach for the routing problem.

With the view that the conventional routing algorithms do not consider the network data history such as the overloaded routes and route failure, the authors in [139] used the advantages of network data to present a RL-based routing strategy. However, a RL-based routing algorithm requires additional control message headers. To address this, the authors proposed an enhanced protocol named enhanced reinforcement learning routing protocol (e-RLRP), which aims to reduce the network overheads. Here, different network scenarios were implemented, where the number of nodes, routes, traffic flows and the degree of mobility were varied. From this analysis, network parameters such as packet loss, delay, throughput, and overhead were then obtained. In addition, a voice over internet protocol (VoIP) communication scenario is implemented, in which the E-model algorithm is used to predict the communication quality. The performance of the e-RLRP scheme is compared to that of the OLSR, BATMAN and the RLRP protocols. The experimental results showed that the e-RLRP protocol provides

reduced network overhead in most network scenarios compared to all the other protocols.

2.10.3 Deep Learning-based Approaches

A DL model for directly learning paths for high-throughput packet transmission was proposed in [195]. The authors in [193] also proposed a DL-based solution for a traffic flow routing problem in computer networks. Unlike the decentralised approach in [195], where each network node trains a model for each destination and the output is the next hop, the authors of [193] presented a centralized model similar to that of [192]. In the model in [193], the controller uses the knowledge from the whole network as inputs and it gives full paths from source to destination as output. This routing framework presents an alternative to the design of heuristics, whilst still achieving good performance. Here, a DL model was trained on the optimal decisions over flows from the known traffic demands. The performance of the solution was evaluated considering network congestion, and the experimental results, which used publicly available datasets of networks with real traffic demands, showed that the solution achieves near-optimal network congestion values.

2.10.4 Deep Reinforcement Learning-based Approaches

The conventional Q-learning methods are not capable of representing the load of each network node as a continuous value due to the limited size of the Q-table [196]. This makes it difficult to incorporate the dynamic change of the load among the APs. To address this problem, the authors of [137] and [194] adopt a neural network to approximate the action values in continuous state space. This DRL-based solution provides a better action-selection strategy and the results of the scheme proposed in [194] achieves better performance compared to the conventional schemes. The DRL-based dynamic load balancing scheme that was proposed in [137] applies multi-agent RL, whereas the one proposed in [194] applies a DQN. In addition to the controllers being the DRL agents as was assumed in [137], the authors of [194] also consider the switches as DRL agents. In IAB network models, the DRL strategy has been applied for solving link scheduling problems as was discussed in Section 2.9.2. As part of the contributions of this research work, Chapter 4 presents a DRL-based model and technique to optimize the backhaul routing in IAB networks subject to SBS load, transmission power, and backhaul bandwidth constraints. This provides a realistic model that is applicable for the 5G NR wireless environment.

2.11 CONCLUDING REMARKS

From the reviewed research contributions, it transpired that the dynamic RA and resource configuration problem, as well as the backhaul route adaptation problem, still remain less investigated problems in the IAB networks. Despite the aforementioned intelligent developments in solution schemes for

challenges in IAB networks, it is believed that the modeling for IAB network problems requires the inclusion of AI strategies. It can be seen that including ML strategies in mm-wave IAB network models, in particular RL, can provide smart and efficient solutions for some of the challenges faced therein.

In terms of the solutions to resource management and backhaul routing problems in wireless networks, the existing algorithms tend to focus on finding low-cost routes for traffic to reach the MBS, without finding ways to minimize resource exhaustion at the current BS. With the view that future mobile and wireless networks' operational spaces will be very diverse and will vary significantly, rule-based decision-making, where decisions are made directly from training, may not be ideal. Consequently, it may not be effective to design apriori cost functions and solve the optimization problems in real-time. It is thus proposed that the decision maker in an IAB system be implemented using a DNN in order to provide action choices for any given state of the system. This is because DNNs enable optimal action choices in highly stochastic and dynamic environments such as IAB networks. As such, it is at this point where the DRL strategies become attractive alternative solutions.

From the literature review, it can be seen that most contributions for the IAB networks focus on user association and RA without considering much of the environmental context such as the SBS load. It is on this note that this research work developed an access resource management scheme that considers context whilst leveraging DRL for the solution formulation of the problem. To provide an end-to-end solution, the work then goes further to explore the optimization of backhaul transmission by applying DRL-based context-switching for backhaul traffic. These contributions are presented in the subsequent two chapters.

CHAPTER 3 DRL-BASED ACCESS RESOURCE MANAGEMENT IN IAB NETWORKS

3.1 CHAPTER OVERVIEW

This chapter presents a dynamic user association and resource management scheme that depends on instantaneous load-based bandwidth partitioning in multi-hop IAB networks. The proposed framework is expected to have the capability of scaling to large IAB network topologies effectively. In addition, the framework should be capable of adapting to the network topology changes and the different real-time IAB system requirements. Most user association and RA methods in the literature use the SINR as the base metric for decision-making. However, supplementing this with the combination of AI and network traffic filtration such as deep packet inspection (DPI) might improve the performance by avoiding resource exhaustion. Thus, in this work's model, the decision maker in an IAB system implements a DRL algorithm to provide action choices for any given state of the system. The proposed approach relates more to the UEs' needs than the global form of the network. The rest of the chapter is organised as follows: Section 3.2 gives the background motivation and a review of the related work on congestion control and dynamic resource management in IAB networks; Section 3.3 describes the IAB multi-hop network model, and Section 3.4 defines the problem and presents the mathematical formulation thereof. Section 3.5 details the proposed solution approach and algorithmic analysis. In Section 3.6, the simulation results are presented together with their performance analysis, and Section 3.7 concludes the chapter.

3.2 BACKGROUND AND RELATED WORK

To address the problem of traffic balancing in IAB networks, advanced node coordination is a requirement. Node coordination in IAB networks requires efficient signaling exchange between the MAC layers of different IAB nodes whilst considering the throughput and latency constraints of wireless backhaul links. Thus, intelligent algorithms that make adaptive decisions on the link and UE/BS

scheduling with fairness and half-duplex constraints need to be implemented at the MAC layer of IAB nodes. However, this requires centralized training procedures to be distributed among the local IAB nodes.

Considering that the CR technology will eventually be integrated into every future wireless communication technology, each AP in a wireless network should have a cognitive engine that is capable of performing automated handover. The cognitive engine will be responsible for assisting with the BS-UE association, where the UEs can instantaneously switch between the BSs for better QoE. However, if the prospective BS does not have enough resources available to satisfy the QoE requirements of new UE arrivals at that instant, the resulting handover delays may lead to transmission delays. This leads to a deterioration of the users' QoE. Most contributions on user association in 5G networks focus on addressing the problem of aggregated interference that is experienced by users [135]. Such problems are usually modeled as multi-agent systems where the BSs are usually the agents that control user admission. Real-time multi-agent RL strategies such as decentralized Q-learning are then applied to manage the aggregated interference generated by multiple users. In this regard, two scenarios have been considered to enable the multi-agent systems' learning. In the first scenario, the agents have complete or partial knowledge of state of the environment, and in the second scenario, the agents interact with the environment directly in a distributed manner. As a result, the developed spectrum management framework was shown to improve the spectral efficiency as well as the energy efficiency performance, as reported in [136]. However, since balancing the spectral and energy efficiencies is becoming a critical challenge in current heterogeneous and resource-constrained networks, the energy efficiency and channel characteristics are analyzed using joint channel selection and power control spectrum decision algorithms based on distributed Q-learning. This leads to distributed strategy estimation in the selection of the learning strategy that is designed to solve the optimization problem.

3.2.1 Congestion Control in 5G Networks

Autonomous congestion avoidance and resource management are desirable features for 5G networks to continuously satisfy user requirements. The MNOs are able to enforce per-UE or even per-BS policies, and when there is congestion or link failure, the control plane can promptly reconfigure itself to manage the available traffic by applying SDN [59]. A number of various ML-based resource provisioning schemes for handling traffic in 5G networks, particularly through network slicing using SDN, have been proposed [197], [198], [199]. By considering varying user requirements, system-level simulation results of the shape-based heuristic algorithms that were proposed in [198] and in [199]

showed that resource utilization and user satisfaction were both improved.

The authors in [200] proposed a topology formation algorithm that aims to enable efficient traffic flow to minimise congestion and increase backhaul link reliability in multi-hop IAB networks. In addition to a dynamic programming-based algorithm, the authors also proposed a less computationally complex topology formation approach. Here, the IAB nodes are incorporated into the network in an "ideal sequence-based topology formation". This practical solution, which is based on the broadcast signaling of received power thresholds, was shown to yield near-optimal performance and achieved 26% better performance compared to a baseline approach. This chapter proposes a resource management scheme that aims to improve user satisfaction through congestion avoidance in the access environment of mm-wave IAB networks by applying DRL. Here, the resource management problem was formulated as a constrained MDP and the objective of the solution approach was to obtain an optimal policy that maximizes the transmission throughput of all the UEs while considering the environmental context such as the traffic load.

3.2.2 Resource Management in IAB Networks using Reinforcement Learning

In ultra-dense environments such as the IAB mm-wave networks, the achievable thoughput of the UEs that are associated with an SBS is regulated by how the total bandwidth is split between access and backhaul. This makes the achievable throughput to be sensitive to the applied RA schemes and it has been shown that the solution space for the RA increases exponentially with increasing BS load or spectrum resources. Since in IAB networks radio resources are shared between access and backhaul, a different RA approach from the typical wireless standards is required. This means that end-to-end RA algorithms are most suitable for IAB networks and the contributions from the literature on RA in IAB networks, including those that proposed end-to-end models, were explored in Section 2.9.3.

Despite the previously discussed intelligent developments in IAB schemes, it is believed that the application of ML strategies in mm-wave IAB networks has not been extensively explored. The RNNs are one of the state-of-the-art models that are favorable for such applications due to their capability of storing information over extended time intervals. By using RNNs, historical information can potentially be used to predict traffic from the various user groups and to enable the optimization of future network configurations. However, with the RNN approach, even if all the relevant statistics are known, solving the RA problem accurately in mm-wave IAB networks results in a POMDP with large state and action spaces. Thus, the problem's complexity is increased because of the lack of prior knowledge of

the stochasticity of the traffic as well as the unobservable channel statistics at each IAB node. This makes the RA problem generally intractable. As such, the dynamic transmission resource management problem still remains a less investigated problem in mm-wave IAB networks.

The existing solution algorithms for the dynamic RA problem in IAB networks do not consider ways to minimize resource exhaustion at a serving BS but instead they focus on finding alternate low-cost routes for traffic to reach the MBS. In addition, a framework that avoids resource exhaustion should be capable of efficiently adapting to different learning mechanisms as well as real-time system requirements. The operational spaces of next generation wireless mobile networks are envisaged to be very diverse and dynamic, which leads to scenarios that are not postulated during the design or training phases of an algorithm. Because of the unpredictability of future wireless environments, action selection based on rules that learn directly from training is not ideal. Thus, designing a-priori cost functions and then optimally solving the allocation problems in real-time may not be effective. In this regard, the decision maker of an IAB system should be implemented using a DNN to provide action choices based on more accurate estimates within any given state of the system. The DNNs enable optimal performance in highly stochastic and dynamic environments such as in mm-wave IAB networks, where the reward for taking an action depends on future actions and states. Hence, in such cases the DRL strategies become an attractive alternative.

3.3 SYSTEM MODEL

The proposed network model considers the uplink transmission of a two-tier 3GPP heterogeneous broadband access network for multi-hop IAB networks [7]. In the IAB network structure, the MBS has a single omni-directional antenna and it is connected to the core network through fiber backhaul. The MBS is overlaid by a set $\mathcal{M}^- = \{1, 2, \cdots, M\}$ of SBS nodes that are uniformly deployed around its area of coverage. The MBS-SBS and the SBS-SBS wireless backhaul links make use of mm-wave frequencies. It was assumed that for each IAB node there is a group of associated UEs such that $\mathcal{K} = \{1, 2, \cdots, K\}$ denotes the set of associated UEs according to a call admission scheme [201], as shown in the environment block of Fig. 3.1.

Assuming the NSA deployment setup of the 5G NR and in line with the IAB network model, the nodes are assumed to be full-duplex capable and all SBS-SBS links are assumed to be symmetrical. The BSs are modelled to employ the instantaneous bandwidth partitioning scheme of [67]. Each SBS is assumed to have two antennas: one for access to serve its associated UEs, and the other for the wireless

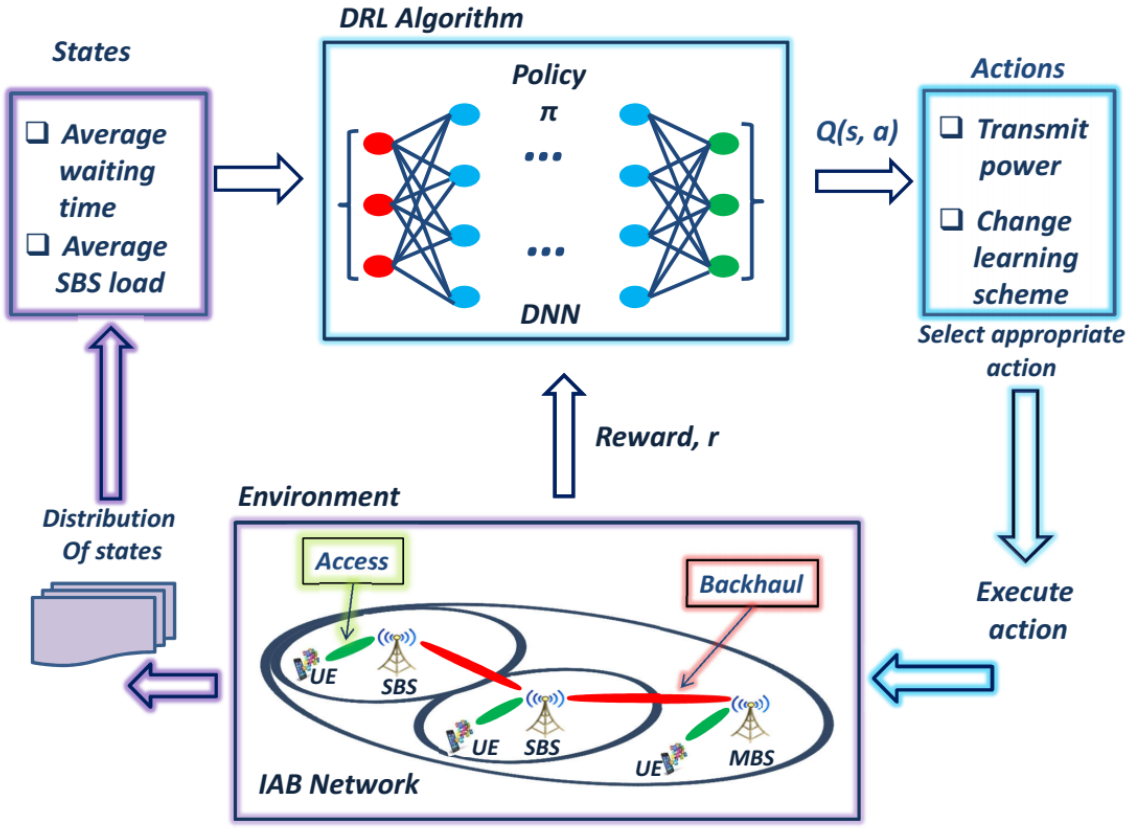

Figure 3.1. A multi-hop IAB network model that implements a DRL algorithm, where the inputs to the DRL that come from the environment as well as the resulting outputs (actions) are shown.

backhaul. The analysis considers the effects of UE congestion on an SBS's link layer behaviour, and this is used to evaluate the performance in terms of achievable throughput and user satisfaction.

Figure 3.1 also shows the states that are taken as inputs by the DRL algorithm, that is, the average waiting time, and the average SBS load. Based on the DNN agent's evaluation, the output, which is the resulting action, is given as the optimal transmit power or change of learning scheme. The resulting reward, which comes from implementing the optimal action, leads to the next environmental state.

3.3.1 Queuing Model and Traffic Load Evaluation

Considering that each BS has a fixed capacity, the available capacity and the achievable throughput are inversely proportional. Thus, to avoid packet losses, the traffic load should be monitored and kept balanced. The average SBS load is a result of the traffic that is injected by the K associated UEs, at a

rate, $\lambda(t)$, and arriving at the SBS in a collective arrival process, $\Lambda(t)$, denoted as

$$\Lambda(t_0) = \sum_{t_0=1}^{t} \lambda(t_0), \quad t_0 = 0, 1, \cdots, T-1. \tag{3.1}$$

This indicates the arrival flow, whose bivariate extension for the range $0 \leq t_0 \leq t$ is defined as $\Lambda(t_0, t) = \Lambda(t) - \Lambda(t_0)$, up to a time horizon T. When monitoring the system at discrete time intervals, $t = 0, 1, 2, \cdots$, a discrete-time $M/G/1$ queuing system with a stationary and ergodic arrival process is assumed to guarantee the existence of stationary limits [202]. Assuming that the $M/G/1$ queue is handled by a transmission buffer that is capable of handling C packets, the system utilization factor is defined as

$$u(t) = \frac{\mathbb{E}[\lambda(t)]}{C}, \tag{3.2}$$

where $\mathbb{E}[\lambda(t)]$ is the long-term expectation of the packet arrival process, and ρ is the SBS load. The capacity that is consumed is monitored and its trend is periodically tracked using an exponentially weighted moving average, which is defined as [203]:

$$\overline{\Delta\rho} = \omega \cdot \Delta\rho(t) + (1-\omega)(\rho(t) - \rho(t-1)), \tag{3.3}$$

where $\rho(t)$ is the current load, $\overline{\Delta\rho}$ is the average load value and $0 < \omega < 1$ represents the decay factor, which is selected through adaptive weights. To avoid congestion, the condition $(\rho(t) + \Delta\rho(t)) = \psi \cdot B_a$ should be satisfied, where ψ is the load/capacity threshold and B_a is the access bandwidth. This approach gives the flexibility to compare the current SBS load with the load of its neighbours in order to prevent premature bandwidth split.

3.3.2 State and Action Spaces

The state space of the system comprises the average waiting time and the average SBS load, ρ. This is in response to the variation in the UE request generation rate, the varying distribution of the requests as well as in the SBS processing speed. In this subsection, the state and action vectors as well as the reward are defined, which are all essential for the implementation of SBS traffic load evaluation. Since the computation of control variables is done at the initial time slot, $t = 0$ is regarded as the time when the initial state, s_0, is defined. Assuming that the delays and service rate are independent and generally distributed with respect to the service rate, the state space is given by

$$\mathbf{s} = \{C(t), \rho(t)\}, \tag{3.4}$$

where $C(t)$ is the average number of packets in the system from $t = 0$ up to time t, which approaches a steady state value as $t \to \infty$. With the state space indicating the amount of pressure that is endured by the SBS's transmission buffer, the DNN agent's action is either a change in the transmission power or a change of learning mechanism. The action space can be thus represented in vector form as

follows:

$$\mathbf{a} \triangleq < p(t), z(t) >, \tag{3.5}$$

where $p(t)$ is the selected transmission power and $z(t)$ is the required transmission throughput. However, to achieve the required throughput, the selection of the optimal transmission power takes precedence over changing the learning scheme.

3.4 MATHEMATICAL PROBLEM FORMULATION

Let $k \in \mathcal{K}$ represent the generic progression of UE admission such that the range of admitted UEs can be represented as $[1, K]$. Assuming uniformity of channel fading within one sub-channel, which may be different on the other sub-channels, the SINR between the k^{th} UE and the m^{th} SBS is expressed as

$$\gamma_{k,m} = \frac{p_k g_{k,m}}{\sum_{j \in \mathcal{M}^-} p_k g_{kj} + N_0}, \quad \forall k, m \tag{3.6}$$

where p_k is the transmission power, $g_{k,m} = \tau_{k,m} h_{k,m}$. Here, $\tau_{k,m}$ is the distance-dependent fading coefficient and $h_{k,m} = \exp(1)$ is the frequency-dependent small-scale fading [190]. The first term in the denominator represents the co-tier interference whereas N_0 represents the white Gaussian noise spectral density. Assuming that both the UEs and the SBSs have self-interference cancellation abilities, the achievable instantaneous transmission rate between UE k and SBS m is determined by

$$\Gamma(\gamma_{k,m}) = B_a \log_2 (1 + \gamma_{k,m}), \tag{3.7}$$

where B_a denotes the access bandwidth.

The SBS aims to determine the optimal transmission power that corresponds to the throughput that satisfies the QoE requirements for all the admitted UEs. In this case, the buffer occupancy status and the transmission power allocation are inseparably related because transmission power should be increased to increase the service rate and avoid congestion. However, proper power control is required for the system to operate with good energy efficiency. As a result, the current load at an SBS can be defined as

$$\rho(t) = \frac{C_i(t)}{\sum_{k=1}^{K} B^* \log_2(1 + \gamma_{k,m})}, \tag{3.8}$$

where $C_i(t) \leq C$ is the capacity required by the admitted UEs, and B^* the sub-channel width. To avoid congestion, the average service time of all the UE requests should be minimized, hence the admission and processing delays should be at their minimum. The average service time minimization problem then becomes a QoS or rate maximization problem and the sum rate of the admitted flows can be

defined as

$$Q_k(\Gamma(\gamma^*,[0,T])) = \sum_{t=1}^{T}\sum_{k=1}^{K}\Gamma(\gamma_{k,m}), \tag{3.9}$$

where $[0,T]$ is the time interval over which the performance of the system is monitored. The objective then becomes obtaining the optimal policy that maximizes the transmission rate of the UEs that are associated with the SBS of interest. The optimization problem is given as

$$\mathbf{F} = \arg\max_{\gamma} Q_k(\Gamma(\gamma^*,[0,T])), \quad \forall k \in \mathcal{K} \tag{3.10}$$

subject to

$$\begin{aligned}
&\mathbf{C1}: \sum_{k=1}^{K}\Gamma_{k,m}^{req} \leq \Gamma(\gamma_{k,m}),\\
&\mathbf{C2}: \Gamma(\gamma_{k,m}) \geq \Omega(\gamma^*,[0,T]), \quad \forall k \in \mathcal{K}\\
&\mathbf{C3}: \rho(t) < \psi \cdot B_a, \qquad \forall k \in \mathcal{K}\\
&\mathbf{C4}: \zeta_k(\gamma^*,[0,T]) \leq 1-\varepsilon, \quad \forall k \in \mathcal{K}
\end{aligned} \tag{3.11}$$

where $\mathbf{F}$ represents the maximum network utility and γ^* represents the target SINR that is required to achieve proportional transmission fairness, $\Gamma(\gamma^*,[0,T])$. This is supported by constraint **C2**, which indicates that the QoS requirements of the admitted UEs should be met. The constraint **C1** ensures that the sum of the transmission rates required by all the UEs is achieved by the allocated access bandwidth B_a, with $\Gamma_{k,m}^{req}$ being the required data rate. The constraint **C3** ensures that the traffic load, ρ, remains below the maximum capacity to avoid congestion, which would result in a decline in the QoS [203]. In the proposed model, an SBS is allowed to admit UEs as long as it is still capable of providing the requirements, until it reaches the threshold. At this point, the SBS becomes overloaded and goodput begins to diminish. Lastly, the constraint **C4** ensures that every action exploration is kept within acceptable powers, where $\zeta_i(\gamma^*,[0,T]) = \frac{\gamma}{\gamma+1}$ is the power allocation condition, and ε is the exploration parameter. If congestion is reached when **C3** is violated, the exploration of transmission powers is initiated. However, this should be carried out within the allocated power budget and it should not upset the system's energy consumption.

3.5 DRL-BASED SOLUTION

To design an efficient solution for solving the optimization problem in (3.10), a finite-source traffic model that is based on both the thinning process and the fading conditions is assumed [204]. The input to the DNN thus originates from $[5:5:K]$ sources that inject packets into the SBS transmission buffer. For the IAB network's access, the DRL algorithm's input, output, and feedback processes determine the appropriate learning scheme that is used. For each learning scheme, the network status is

relatively fixed and the application environment is assumed to be known. This makes the distribution of states, the actions, and the rewards determine the dynamics of the IAB network. In this case, the node behaviour can be anticipated based on the availability of and constraints on the transmission resources. Thus, the proposed DRL algorithm computes the solution for $Q(\gamma^*, a) \approx \sigma/T_s$ using a hierarchical approach and for simplification and ease of application of the DRL strategy, it is assumed that the state and observation overlap perfectly.

3.5.1 Reward Maximization

For any UE that is admitted by an SBS, there should be adequate computational resources that are allocated to it such that the required QoS is met. By applying ACM and using the peak data rate and the gains from other co-cell devices, the spectral efficiency evaluation that was used in [205] is adopted. The reward function can be formulated as

$$r(t) \triangleq \sigma = \lceil z(t) L T_s / \Delta t \rceil, \tag{3.12}$$

where $\lceil \cdot \rceil$ indicates that the throughput should be evaluated using the minimum power required to transmit $\lceil x \rceil$ bits per second, $z(t) \in Z$ represents the throughput in packets per time slot, and L is the packet length. Given the channel state and the throughput, the system evaluates the transmission power, which is used to determine the reward in (3.12). The DNN agent thus generates an optimal action $a^* \in \mathcal{A}$, which is the power allocation action that is used to obtain the solution to (3.10). An action-value function, which is stored in the Q-table at time t, is used to select the appropriate action depending on the current state through a policy, π^*, as follows:

$$\pi^*(a|s) = \left\{ \begin{array}{ll} 1, & \text{if} \quad a = \arg\max_{a \in \mathcal{A}} Q^{\pi}(s,a) \\ 0, & \text{otherwise,} \end{array} \right\}. \tag{3.13}$$

Using the obtained Q-value estimates, an off-policy greedy search is used to determine the optimal stochastic policy function $\pi^*(a|s)$ [105]. A random variable, $Q(s_t, a_t)$, is then considered for the estimation of $Q(\gamma^*, a)$ as follows:

$$Q(s_t, a_t) = r_t + \gamma^t \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}), \tag{3.14}$$

where r_t is the reward that is evaluated using (3.12), $\beta^t \in [0,1]$ is the discount factor, s_{t+1} represents the next state, and a_{t+1} is the resulting action. At each time instant, the agent determines its state using the information that is obtained from the queue dynamics. Using the information that relates to the QoS of each UE, the system evaluates the throughput of each UE according to (3.7) and (3.10). In this regard, a sequence of transition probabilities that describe the transitions from the current state s_t to the next state s_{t+1}, the current actions a_t to the next actions, and the current rewards to the future

rewards, is established and it is given by

$$P(s_t|a_t) = p_p(p_{tx}|[g_{k,m}, x_t], h_{k,m}) p_g(g_{t-1}|g_t), \tag{3.15}$$

where x_t is the current power management. This enables the system to have a Markov property, which results in a Markov model representation of the IAB network with the tuple $(s_t, a_t, r_t, p_t, s_{t+1}), t \in T$. Here, the current channel state, $h_{k,m}$, which is determined using measurements from the received signals, is expressed using the channel transition distribution, $p_g(g_{t-1}|g_t)$. The reward function can then be given as

$$r_t(s_t, a_t)) = \theta \cdot Q(s_t, a_t), \tag{3.16}$$

where θ represents the interference power constraint, which is essential for both power management as well as the estimation of $Q(\gamma^*, a)$, and it is controlled by the $p_p(p_{tx}|[g_{k,m}, x_t])$ term in (3.15). Using (3.14) and assuming that the value function of the state-action pair is independent of the reward features of the current state and the current action, (3.16) can be reformulated as

$$r_t(\gamma^*, \hat{p})) = \theta \cdot Q(\gamma^*, \hat{p}), \tag{3.17}$$

such that the solution to (3.10) can be summed into a state-value function over a finite horizon as follows:

$$Q_t^{\pi}(\gamma^*, \hat{p}) = \mathbb{E}\left[\sum_{t=1}^{T} \beta^t r_t(\gamma^*, \hat{p})\right], \tag{3.18}$$

which is the expected value of the reward based on the transmission power. Here, π is the optimal policy for all the state-action pairs that is used by a value function, which is evaluated using the Bellman optimality equation [105]. The expression $\beta^t r_t(\gamma^*, \hat{p})$ is the discounted reward at time step t and the sum of the discounted rewards is obtained over a finite horizon, T. Here, β^t is kept less than unity because the agent is interested in long-term returns. Since the reward optimal values depend on the physical conditions of the environment as well as the policy that is implemented by the DRL approach, the objective of maximizing (3.18) can be defined as follows:

$$Q_t^{\pi}(\gamma^*, \hat{p}) = \max_{\pi} \mathbb{E}\left[\sum_{t=1}^{T} \beta^t r_t(\gamma^*, \hat{p})\right]. \tag{3.19}$$

By applying the intuitive definition of the Bellmann optimality equation, where the value of a state under an optimal policy should be equal to the expected return for the best action taken from that state, Q^* can be expressed as

$$\begin{aligned} Q^*(s,a) &= \mathbb{E}\left[r_{t+1} + \beta \max_{a_{t+1}} Q^*(s_{t+1}, a_{t+1})\right] \\ &= \sum_{s_{t+1}, r_t} p(s_{t+1}, r_t|s_t, a_t)[r_t + \beta \max_{a_{t+1}} Q^*(s_{t+1}, a_{t+1}], \end{aligned} \tag{3.20}$$

where the expression $\sum_{s_{t+1}, r_t} p(s_{t+1}, r_t|s_t, a_t)$ represents the transition probability. From the Bellman optimality equation, the reward value of a given state can be split into the immediate reward and the

discounted reward of the subsequent state [206], and the Bellman optimality equivalent of Q^* can be derived by assuming accurate estimates from (3.20). With this assumption as well as assuming that the deterministic policy is optimal, (3.13) holds. From the state-value function in (3.18), the Bellman optimality equation for Q^π is given by

$$\begin{aligned} Q^\pi(s,a) &= \mathbb{E}_\pi\left[\sum_{t=1}^{\infty}\beta^t r_{t+1}|s_t=s\right] \\ &= \mathbb{E}_\pi\left[r_{t+1}+\beta\sum_{t=1}^{\infty}\beta^t r_{t+2}|s_t=s\right] \\ &= \sum_a \pi(a|s)\sum_{s_{t+1}}\sum_{r_t} p(s_{t+1},r_t|s_t,a_t)\left[r_t+\beta\mathbb{E}_\pi\left[\sum_{t=1}^{\infty}\beta^t r_{t+2}|s_{t+1}=s_{t+1}\right]\right] \\ &= \sum_a \pi(a|s)\sum_{s_{t+1},r_t} p(s_{t+1},r_t|s_t,a_t)\left[r_t+\beta v_\pi(s_{t+1})\right]. \end{aligned} \tag{3.21}$$

Defining the optimal Q-value as $V^*(s)=\max_{a\in\mathcal{A}} Q^{\pi*}(s_t,a_t)$ the objective to maximize (3.18) can be computed as follows:

$$\begin{aligned} V^*(s) &= \max_{a\in\mathcal{A}} Q^{\pi*}(s_t,a_t) \\ &= \max_a \mathbb{E}_{\pi^*}\left[\sum_{t=0}^{\infty}\beta^t r_{t+1}|s_t=s,a_t=a\right] \\ &= \max_a \mathbb{E}_{\pi^*}\left[r_{t+1}+\beta\sum_{t=0}^{\infty}\beta^t r_{t+2}|s_t,a_t\right] \\ &= \max_a \mathbb{E}_{\pi^*}\left[r_{t+1}+\beta V^*(s_{t+1})|s_t,a_t\right] \\ &= \max_{a\in\mathcal{A}} \sum_{s_{t+1},r_t} p(s_{t+1},r_t|s_t,a_t)\left[r_t+\beta V^*(s)\right], \end{aligned} \tag{3.22}$$

which is the Bellman optimality equation for V^*.

For small-scale wireless network problems such as spectrum sensing problems, arbitrary assumptions for the Q-value estimates can be made and the values are updated through trial-and-error as the policy progresses towards convergence. When the Q-value updates and the action-selection are performed randomly in such cases, the optimal policy may not give a global optimum [207]. However, in 5G and Beyond 5G wireless networks, the problem can grow extensively with discrete states and actions. The discrete state transition probabilities in such scenarios should be explicitly defined depending on the large state space. Construction and storage of Q-tables for large-scale problems in dynamic wireless transmission environments can become computationally complex since the state space grows exponentially and the respective actions need to be determined. As a result, the memory requirement for storing and updating the Q-values increases proportionally with the number of possible states, and

the duration of exploring each state to create the required Q-table becomes unrealistic [208]. Due to the challenges that are introduced by this scaling and increased computational complexity in the traditional RL strategies, advanced strategies such as DRL have been proposed to address this [208]. Thus, to solve the RA problem in IAB networks, it is proposed that the RL strategy be combined with the DL technique.

Since the states in wireless communications are generally discrete, discrete state transition probabilities should thus be defined based on the state space. As such, a conditional a priori probability vector $\mathbf{P}$ is defined as a sufficient statistic substitute for s_t, such that (3.19) can be reformulated as

$$Q_t^{\pi}(\mathbf{P}, a_t) = \max_{\pi} \mathbb{E}\left[\sum_{t=1}^{T} \beta_t r_t(P_{ij}, a_t)\right], \tag{3.23}$$

where P_{ij} represents the inter-state transition probability. By applying dynamic programming, the solution can then be obtained by finding the state-value function, $Q_t^{\pi}(\mathbf{P})$, as follows:

$$Q_t^{\pi}(\mathbf{P}) = \max_{a_t}(r_t(P_t, a_t) + \beta \mathbb{E}\{Q_{t+1}(P_{t+1}|P_t)\}), \tag{3.24}$$

where P_t and P_{t+1} represent successive transition probabilities. Assuming that a limit on T exists, the optimal transmission rate can be reformulated as follows:

$$\bar{\Gamma}_t(\gamma^*, \cdot) = \lim_{t\to\infty} \sum_{t=1}^{T} \Gamma_t(\gamma^*, \cdot). \tag{3.25}$$

Furthermore, with the assumption that the interference caused by the other nodes is stationary and ergodic, and that $\hat{P}$ is a function of I, this limit exists with a probability of one and

$$\mathbb{E}[\bar{\Gamma}] = \mathbb{E}[\Gamma(I), P(I)]. \tag{3.26}$$

Using the Lebesgue-Stieljies integral [209], (3.26) can be converted to a constrained problem given by

$$\int_0^{\infty} \Gamma(\gamma^*, P(\gamma^*)) \cdot dP(I \leq \gamma^*) \geq \Gamma, \tag{3.27}$$

where the function P solves the problem into a relatively unconstrained power management function as follows:

$$\min_{P\geq 0} \int_0^{\infty} P(\gamma^*) \cdot dP(I \leq \gamma^*). \tag{3.28}$$

This is, however, applicable if the signal processing at the SBS is sufficient to provide accurate estimates of the interference power from the received signal power.

3.5.2 Training of the DNN Agent

During each discrete time instant, the SBS should provide transmission throughput that satisfies the QoS requirements of its associated UEs. Using the current state of the system, $s_t \in \mathcal{S}$, the SBS, which

is the DNN agent, should minimize the power that is required to transmit $\lceil z \rceil$ bits/symbol. Given a DNN that consists of $N : i = 0, 1, 2, \cdots, n$ layers, $i = 0$ refers to the input layer whereas $i = n+1$ refers to the output layer; the other intermediate layers are known as the hidden layers. For the proposed solution, the DNN consists of four hidden layers, and the DNN training process is comprised of the two stages that are described below.

Stage 1

Firstly, the DNN's parameter, θ, is initialized with a zero-mean normal distribution. The inputs to the DNN are then transferred from the input layer to the first hidden layer as a sample vector, $\mathbf{x}$. The feed-forward network then goes through a process of determining the values of the hidden layers, which are defined by $\mathbf{h} = f(\mathbf{x}, \theta)$. In this work, the value of the j^{th} computational unit of the i^{th} layer of the DNN architecture is denoted by $h_j^i(\mathbf{a})$. Each link between two successive hidden layers is assigned a weight, W_{jk}, while the respective node is assigned a default activation internal bias, b_j^i. When the weight and bias are assigned the node then computes the loss function using the input sample from the previous layer by applying a rectified linear unit (ReLU) activation, $ReLU(x) = \max(0, x)$. This computational pattern continues throughout the hidden layers of the DNN. The ReLU activation function is applied in this case because of its delinquency in solving the vanishing gradient problem since it has better error transmission capabilities compared to the prevalent sigmoid function. The ReLU activation function is more applicable in DNN-based power control solutions, especially when employed in the hidden layers.

Stage 2

Considering that the output layer provides additional transformation to the hidden layer outputs, for the DNN to complete its task, it should satisfy the design constraints. With y being the output of the DNN, the probability of the last hidden layer is given by

$$P(y = 1|x) = \max\{0, \min\{1, a_j^i(\mathbf{a})\}\}, \tag{3.29}$$

where $a_j^i(\mathbf{a}) = \sum_k W_{jk}^i \hat{h}_k^{i-1} + b_j^i$, such that when $W_{jk}^i \hat{h}_k^{i-1} + b_j^i$ strays outside the unit interval, the slope of the DNN output with respect to its parameters would not be zero. At the output layer, a logistic sigmoid activation function is applied as follows:

$$\hat{h}_j^i(\mathbf{a}) = \Phi(a_j^i), \quad \text{where} \quad \Phi(a) = (1 + e^{-a})^{-1}. \tag{3.30}$$

During each step of the feed-forward procession after stage 1, the delta rule is used to determine updates of the weights and the biases. This is done minimize the loss value, which depends on the loss

function that is used. Here, an efficient SGD algorithm that derives the gradient by a running average of its recent magnitude [210] is employed. Depending on the current policy, π_{θ_t}, the DNN gives a relaxed action $\hat{\mathbf{x}}_t$ as the output, which can be represented by a parameterized function

$$\hat{\mathbf{x}}_t = \Phi(\mathbf{h}_t), \quad \text{where} \quad \hat{\mathbf{x}}_t = \hat{x}_{t,i}. \tag{3.31}$$

This represents the candidate action $\mathbf{x}_k$, which represents the i^{th} entry of $\hat{x}_t$, such that the action selection of the transmission power at the output layer should satisfy $\hat{\mathbf{x}}_t \in (0,1)$. Thus, using the SGD algorithm, the resulting output is given by

$$\mathbf{o}(\mathbf{x}) = \hat{\mathbf{h}}^{n+1}(\mathbf{x}) = \Phi(a^{n+1}(\mathbf{x})), \tag{3.32}$$

which is the required power allocation that the DRL algorithm uses to improve the QoS as well as to support context awareness in the IAB network.

3.5.3 The Learning Strategy

Using the output of the DNN, the proposed DRL framework adopts a learning policy that is parameterized by θ, i.e. π_{θ_t}, to guide the DNN agent in learning the best power allocation solution. This policy is determined by

$$\pi_{\theta_t} : p_j \rightarrow a^*, \tag{3.33}$$

which represents the agent's behavioural guide on how to perform the best action selection, that is, the optimal power that gives best solution of (3.10). Here, the ε-greedy exploration approach, which was described in Chapter 2 (Section 2.4.3.6), is employed to perform the best action selection among the candidate actions to be given as the output by the DNN. The greedy action selection that was defined in (3.13), which can be given as

$$\mathbf{a}_t^* = \arg\max_{\mathbf{a}_i \in \mathbf{a}_k} Q^*(\mathbf{h}_t, \mathbf{a}_i), \tag{3.34}$$

gives the learned action $\mathbf{a}_\mathbf{t}^*$ that has the highest Q-value to achieve the maximum reward, and the system transitions to the next state s_{t+1}. After computing the resulting Q-value, $Q(s,a)$, the packet throughput, which depends on the transmission power, is then evaluated. The Q-learning strategy was proven to converge effectively to an optimal solution for this problem as follows [185]:

$$Q^*(\gamma^*, \cdot) = r_t(\gamma^*, \cdot) + \beta^t \sum_{s \in \mathcal{S}} P(s_{t+1}|s_t, a_t) \max_{a_{t+1}} Q^*(s_{t+1}, a_{t+1}), \tag{3.35}$$

where $P(: \mid :, :)$ is the probability measure that governs the state transitions, with state-action pairs that update the Q-table at each instant to approach an optimal Q-value. After converging to the optimal Q-value, the agent then transits to the next state s_{t+1} and updates the corresponding new Q-value as

follows:

$$Q(\gamma^*,\cdot) \leftarrow \underbrace{Q(\gamma^*,\cdot)}_{\text{old value}} + \underbrace{\alpha_t(r_t(\gamma^*,\cdot) + \beta^t \max_{a_{t+1}} \underbrace{Q(s_{t+1},a_{t+1}) - Q(\gamma^*,\cdot)}_{\text{temporal difference}})}_{\text{learned value}}, \tag{3.36}$$

where $\alpha_t \in [0,1]$ is the learning rate of the Q-learning algorithm. The reward is then awarded to the best action that leads to an optimal Q-value, $Q(\gamma^*,\cdot)$. The Q-values are updated at every time slot according to the TD approach that was outlined in Section 2.6.1. Using the TD updating rule, the learned value at the next time step, which is the immediate reward, is given by

$$r(\gamma^*,\cdot) + \beta^t \max_{a_{t+1}} Q(s_{t+1},a_{t+1}). \tag{3.37}$$

It is worth noting that the analysis in this work focuses on cases when the IAB system learns to deliver the best possible association and guaranteed QoS to the associated UEs. This means that the agent uses the status of the traffic load and its buffer status to learn the optimal actions that would deliver successful transmissions with high reliability. Thus, the objective of the proposed algorithm is to operate at equilibrium and to at least grant each flow the transmission resources that are enough to meet its QoS requirements, without increasing the cost of transmission. The proposed algorithm that performs transmission resource management using individual learning is outlined in Algorithm 3.

Algorithm 3 Proposed individual learning DRL algorithm

Input: Bandwidth, B; Exploration policy, π; Learning rate, α_t; Discount factor, β^t

Output: Reward, $r(\gamma^*, \cdot)$

Initialize θ_t with zero-mean normal distribution
Populate the Poisson arrival distribution
Initialise the network state s_t to s_0
For $t = 1 : T$ **do**
  Input the sample vector of channel gains to the DNN
  Train the DNN model to obtain the output vector **y**
  Assign **a** $\leftarrow$ **y**
  Take a greedy action according to (3.13)
  Populate the state-action pair (s_t, a_t)
  Arbitrarily set Q(s_t, a_t) and solve (3.10) and observe the reward $r(\gamma^*, \cdot)$
  If condition **C3** is true **then**
    Update system using (3.36)
  Else
    Switch to cooperative learning
  End If
  Populate current transition probabilities and observe tuple $(s_t, a_t, p_t, r_t, s_{t+1})$
End For

When an SBS cannot accurately observe and learn from its environment, exchanging information with its neighbours helps to improve its learning processes. In this context, the neighbouring SBSs teach and learn from each others' experiences via extensive information exchange. Here, the nearest SBS for a given DRL agent is located using the Euclidean distance because the nearest neighbour algorithm critically depends on metric spaces. The proposed nearest neighbour cooperative algorithm is outlined in Algorithm 4.

Algorithm 4 Proposed nearest neighbour cooperative DRL algorithm

01: Initialize θ_t with zero-mean normal distribution
02: Initialize number of chosen neighbours, and
03: Construct discrete state space of neighbouring SBSs
04: Learn distance metric to the nearest SBS
05: **For** $i = 1 : T$ **do**
06: Learn the optimal Q-function using the nearest neighbour regression method
07: Execute steps **06** - **08** of Algorithm 3 to train DNN model
08: Populate the state-action pairs (s_t, a_t)
09: Draw action $a_t \approx \pi(\cdot|\mathcal{S})$ to solve (3.10)
10: Observe the reward $R(\gamma^*, \cdot)$ and generate next state $s_{t+1} \approx p_t(\cdot|s_t, a_t)$
11: **If** condition **C3** is true **then**
12: Update system using (3.36)
13: **Else**
14: Increase transmission power
15: **End If**
16: Populate current parameters and observe the tuple $(s_t, a_t, p_t, r_t, s_{t+1})$
17: **End For**

3.5.4 Algorithmic Computational Complexity Analysis

In the proposed DRL strategy, the DNN contributes towards the efficient estimation of the expected discounted sum of the future rewards when the agent is in a given state. Here, we define an accuracy, δ, which is used as a performance measure for the two proposed algorithms. With the individual learning approach that is illustrated in Algorithm 3, the DRL agent learns from both positive and negative rewards when it executes an action $a \in \mathcal{A}(s)$. When the parameters are initialized, the DNN model is then trained using the input sample of channel gains. By doing so, the agent comes up with a set of possible output actions, which are the allocated powers. To aid with achieving this task, an efficient SGD algorithm that derives the gradient by a running average of recent past magnitude values [210], was considered. The DNN model is then updated to scale with the target variable and in this case, when the size of the target variable is reduced, the gradient that is used to update the weights is also reduced. This realises a more efficient model with a stable training process. As discussed in Section 3.5.2, when the weights of the DNN are represented by probability distributions over the possible observable network states, the uncertainty in the hidden layers allows for the expression of uncertainty

about the outputs [211]. However, for $|a_j| \gg 1$, it can be assumed that $\Phi_{sig}(a_j) \approx \Phi_{0,1}(a_j)$, provided that the weights of the network are not regularized. Hence, the depth of the polynomial networks of such a DNN can be approximated by $\mathcal{O}(\log^2(1/\delta))$, for some fixed accuracy, δ.

3.5.4.1 Complexity of the action selection strategy

By having the output of the DNN as the set of possible transmission powers when in a given state, the Q-learning algorithm is then used to select the best action, that is, the optimal power, using the ε-greedy exploration policy. A persistent exploration learning policy, is used to store the information about how the actions and the states are related in the Q-table. Since this is an undirected exploration, the algorithm thus has no information about the action selection, upon which it bases its decisions. As such, relying on ε-greedy exploration may result in low sample efficiencies because of the undirected exploration. The action-value function, $Q(s,a)$, is then obtained using the Q-learning algorithm and it is iteratively improved by the DNN through the minimization of the loss function [212]. The action selection step in Algorithm 3 (line 08) implements the exploration rule, which determines the next state. With the individual learning approach, the agent only uses the information that is local to the state of that SBS, which includes the Q-values for all the actions, $a \in A(s)$. As stated in [213], the number of executed steps is always bounded by an expression that depends only on the initial and current Q-values. The complexity of action selection, which is well elaborated in [214]. However, for the proposed DRL algorithm, an effective finite horizon power control condition is expected to reduce the sample selection complexity to $\frac{1}{(1-\beta^t)}$. Hence, the expected time and space complexity of action selection in Big O notation is $\mathcal{O}\left(\frac{1}{\delta^2(1-\beta^t)}\log\frac{1}{\delta(1-\beta^t)}\right)$.

3.5.4.2 Updating and reaching the reward state

When the selected action is executed by the learning agent, the state-action pairs are then populated, and then $Q(s_t,a_t)$ is updated. The value of $Q(s_t,a_t)$ is then used to determine the accumulated reward that is received by the agent when it executes an action a_t when in state s_t. If the congestion condition is not met, an update step in Algorithm 3 (line 12) adjusts $Q(s_t,a_t)$. The agent then receives an immediate reward, $r(s_t,a_t) \in \mathcal{R}$, and if the agent starts in $s \in \mathcal{S}$ and executes the selected actions for which it receives the immediate reward r_t at time step t, then the agent's total accumulated reward for this particular behaviour is $r(\gamma^*,\cdot) = \sum_{t=0}^{\infty}\beta^t r_t$. As the agent approaches the reward state, the number of steps can be exponential in the number of states. The Q-value of each state-action pair can be augmented with an estimate of its uncertainty. This helps to guide exploration and to achieve a higher reward during a faster learning process. As an example, when all the Q-values are initialised to zero, and the RL strategy operates on the reward representation, the first Q-value that changes determines

the action that leads to the reward state. For all the other actions, all the Q-values remain zero and no information about the topology of the state space is stored. For the resource management purposes that are considered in this work, because the Q-learning algorithm is admissible, the Q-values remain consistent and monotonically decreasing. Because of this monotonic decrease of the Q-values, the sum of the Q-values also decreases with every step, but it is bounded from below such that the algorithm terminates. Thus, assuming that actions are not duplicated in the state space and that the shortest distance between any two states is bounded by $n-1$, the feasibility result in [215] follows directly. The complexities of a baseline Q-learning algorithm and the proposed DRL algorithms are summarized in Table 3.1.

Table 3.1. Comparison of the computational complexities of the baseline and the proposed algorithms.

Strategy	**Algorithm**	**Action Selection**	**Learning Update**	**Overall**
Baseline	Q-learning	$\mathcal{O}(n\log^2 n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$
Proposed	Individual	$\mathcal{O}(n\log n)$	$\mathcal{O}(n)$	$\mathcal{O}(n\log n)$
	Cooperative	$\mathcal{O}(n\log n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$

As shown in Table 3.1, it is evident that the greedy action selection complexity and the learning update complexity of the proposed individual learning algorithm are both less than those of the baseline algorithm. However, it can be seen that by adding more elements to the DRL algorithm, it becomes more computationally complex to implement. When the agent can no longer learn everything from its own observations and experiences, it switches to the nearest neighbour cooperative learning approach. The nearest neighbour cooperative approach may more complex dynamics in terms of the required operations and memory, which increases the computational complexity with an increase in the observation space. Thus, the worst-case complexity for the cooperative learning update becomes $\mathcal{O}(n^2)$, which is the upper bound on the complexity of the Q-learning algorithm.

3.6 SIMULATION RESULTS

This section presents the results of the performance of the proposed algorithms, which were obtained from the simulations carried out in MATLAB^{TM} software. Table 3.2 gives the values of the simulation parameters that were implemented.

Table 3.2. Simulation parameters used for evaluating performance of the proposed algorithms.

Parameter	**Value**
SBS transmission power	20 dBm
UE transmission power	18 dBm
Shadow fading, SF	4 dBm
UE - SBS Path loss	34.46 +20$\times \log 10(d)$ + SF
Maximum number of UEs, K	40
UE mobility model	Random waypoint
Packet length, L	5000 bits
Traffic type	Constant bitrate
Time slot duration, Δt	10 ms
Fixed symbol rate, $1/T_s$	500 $\times 10^3$ symbols/sec
Achievable data rate, σ/T_s	bits/sec
Capacity threshold, ψ	90%
Exploration rate, ε	0.9
Exploration decay rate	0.995
Learning step sizes, α_t	0.4 - 0.6
Discount factor, β^t	0.98

The performance was evaluated on a 250m$\times$250m grid that simulates an urban environment using a distance-based path loss, with transmission at a carrier frequency of 28 GHz and component system bandwidth of 100 MHz [13]. Here, $M = 10$ SBSs that are randomly deployed are capable of sharing learning information with each other to help with controlling their individual congestion levels. The actions are selected randomly according to an ε-greedy approach with an exploration decay of 0.995. The results also investigate the effect of the learning rate using values of either $\alpha_t = 0.4$ or $\alpha_t = 0.6$, while the reward is discounted with $\beta^t = 0.98$ for better convergence. The performance analysis was conducted in three experiments, which are:(i) congestion rate, (ii) achievable bit rate, and (iii) user satisfaction. Each of these experiments evaluates the performance of both the individual learning scheme, where the SBSs are considered as independent learning agents, and the cooperative learning scheme, where the SBSs are considered to have a multi-agent learning model. In this case, the learning

performance of one SBS is not independent of other SBSs. The performance of the proposed DRL schemes is compared to a baseline DRL algorithm from [190] in each of the experiments.

3.6.1 Experiment 1: Congestion Performance

To evaluate the level of congestion in the serving SBS, channel utilization was used in conjunction with the utilization thresholds. This section presents results that show how the congestion rate of the SBS varies with the number of admitted UEs for the two different learning schemes.

3.6.1.1 Individual learning algorithm

Figure 3.2 shows the congestion performance of the individual learning algorithm when the learning rate is $\alpha_t = 0.4$. In this case, the proposed algorithm and the baseline algorithm exhibit similar performance, having approximately 8% and 21% congestion rates for $K = 5$ and $K = 10$, respectively. It can be observed that both algorithms achieve low congestion rates when the number of admitted UEs is low and there would be a high probability that the SBS will satisfactorily serve all the UEs.

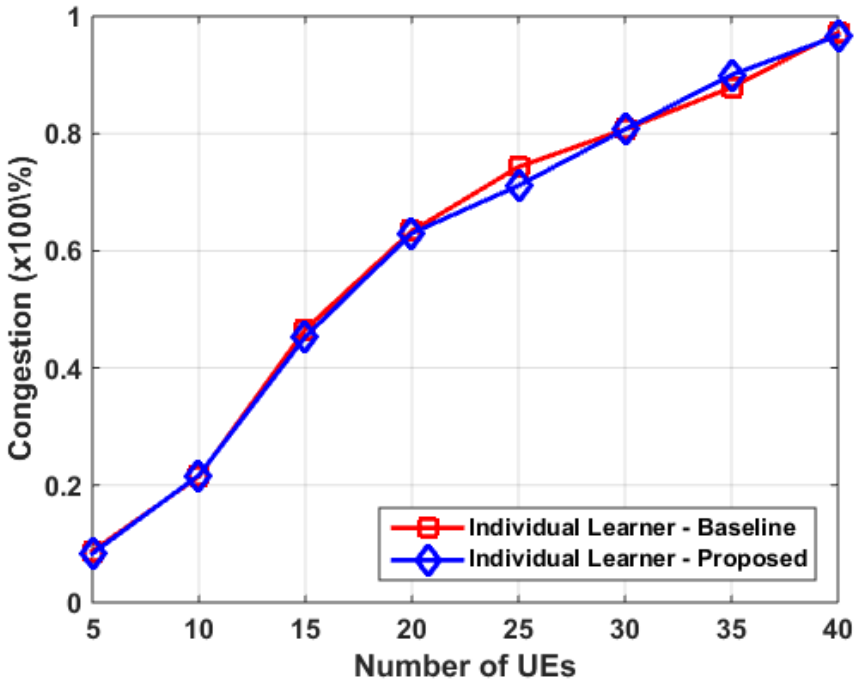

Figure 3.2. Congestion rate at SBS using individual learning with $\alpha_t = 0.4$.

The results of the congestion performance of the individual learning scheme when the learning rate was increased to $\alpha_t = 0.6$ are shown in Fig. 3.3. Figure 3.2 and Fig. 3.3 show that the congestion performance of the proposed individual learning algorithm and the baseline algorithm is more or less

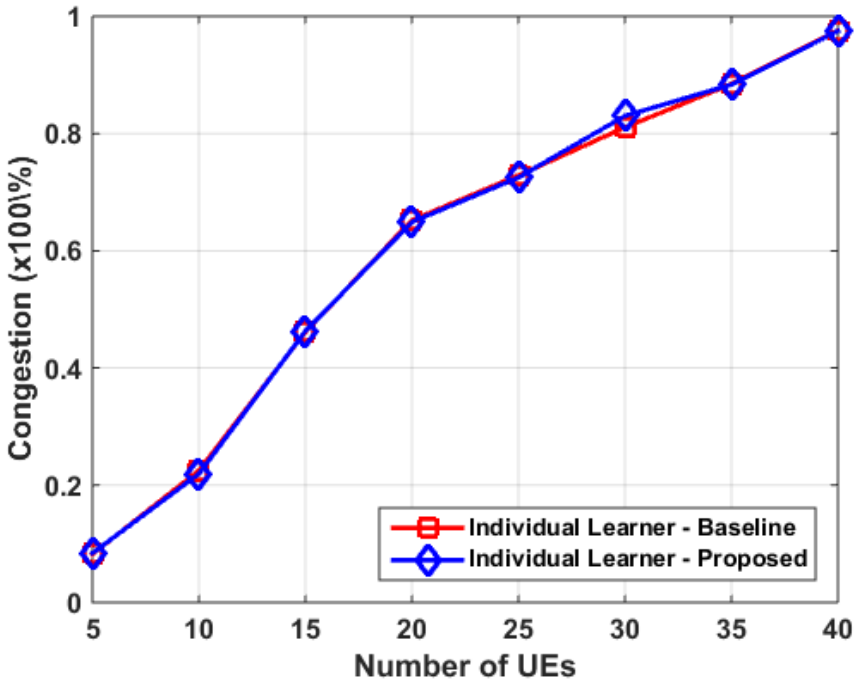

Figure 3.3. Congestion rate at SBS using individual learning with $\alpha_t = 0.6$.

the same. Although the proposed scheme follows the same performance trajectory as the baseline algorithm, it gives a 2.2% better performance compared to the baseline algorithm when $\alpha_t = 0.6$, whereas when $\alpha_t = 0.4$, the performance was only 1.2% better. For $K = 40$, the congestion rate is approximately 93.2%, which is a 3.5% better performance than when $\alpha_t = 0.4$. This means that with $\alpha_t = 0.6$, the SBS has a 3.5% higher probability of satisfying all the admitted UEs than when $\alpha_t = 0.4$. Thus, the value of the learning rate does affect the congestion performance, as was also shown by using different values of the learning rate in the results in [185].

3.6.1.2 Cooperative learning algorithm

For the case of cooperative learning, since a selected nearest neighbour knows the trajectory leading to better rewards, it is then treated as a mentor that performs transfer learning to the current SBS. This assumption resonates with viewing RL as a form of automatic programming [105]. Thus, in this scheme, the current SBS is assumed to learn with the help of its nearest neighbour for better congestion control. The practicality of the nearest neighbour approach was investigated and verified in [190] and in [185].

Figure 3.4 shows that the effect of using the cooperative learning approach is noticeable as the performance of the proposed algorithm begins to improve with an increasing number of UEs compared to the individual learning scheme. The performance of the proposed cooperative algorithm is approximately 7% at $K = 5$, which is about 2.2% better than the baseline algorithm. However, in the range $7 < K < 17$, its performance becomes less than that of the baseline, but it reclaims superiority when $K > 17$. The effect of changing the learning rate while using the cooperative learning algorithm is illustrated in Fig. 3.5.

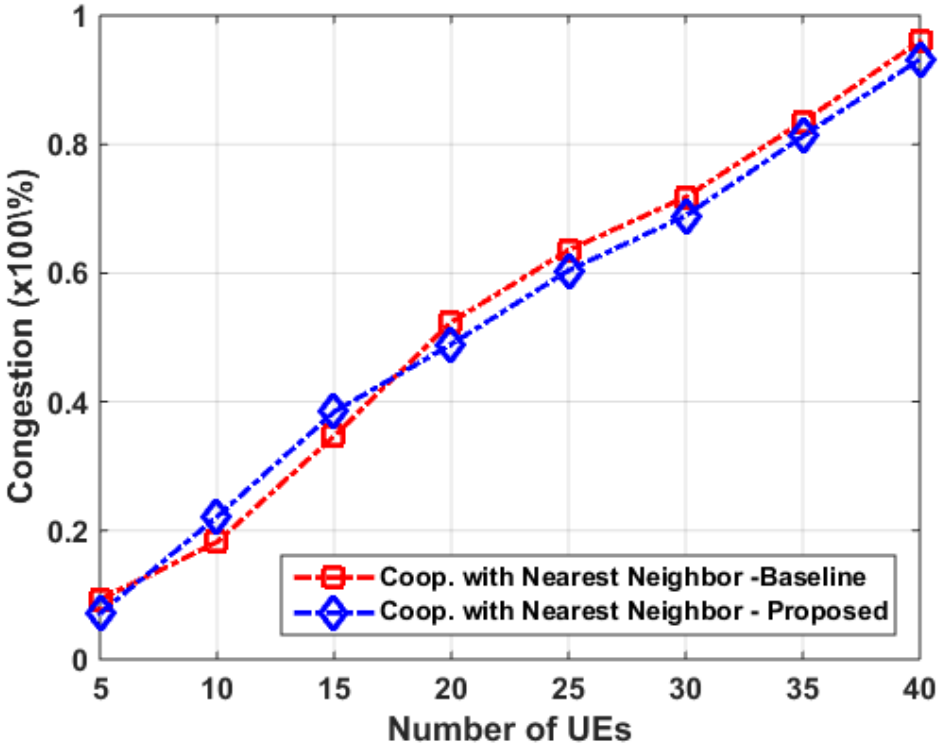

Figure 3.4. Congestion performance for cooperative learning with $\alpha_t = 0.4$.

The performance of the cooperative algorithm in Fig. 3.5 is similar to that observed in Fig. 3.4. Thus, it can be observed that the increase in the learning rate from $\alpha_t = 0.4$ to $\alpha_t = 0.6$ does not have much effect on the cooperative learning approach's congestion performance. This means that the algorithm is insensitive to a change in the learning rate. Such an attribute is valuable for future IAB networks, where the nodes would be capable of self-configuring and adjusting their parameters in to better suit the dynamic network environments.

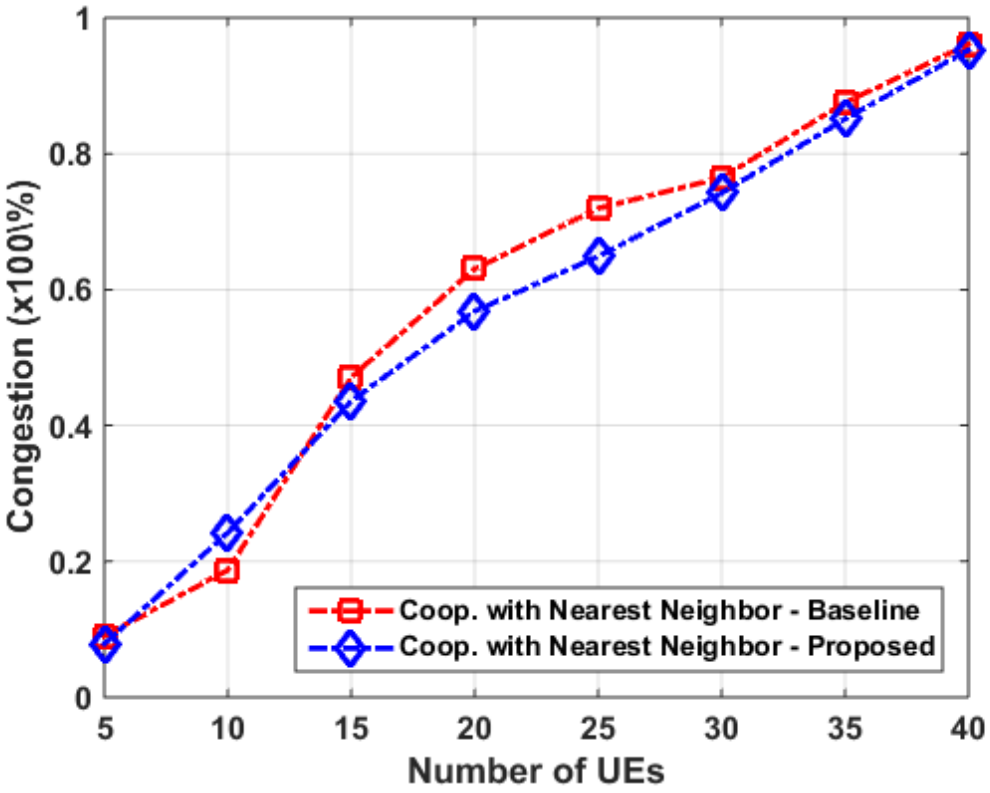

Figure 3.5. Congestion performance for cooperative learning with $\alpha_t = 0.6$.

3.6.2 Experiment 2: Throughput Performance

It was deemed essential to monitor the throughput performance and relate that to the corresponding levels of congestion. This section thus presents results of the performance of the proposed algorithms in terms of the achievable bit rate of the current SBS's admitted UEs. The objective was to maximize the achievable bit rate subject to the two constraints, **C1** and **C2**. Here, the main idea was to ensure that the admission control was based on the satisfaction rate.

3.6.2.1 Individual learning algorithm

The throughput performance of the individual learning approach with a learning rate of $\alpha_t = 0.4$ is shown in Fig. 3.6.

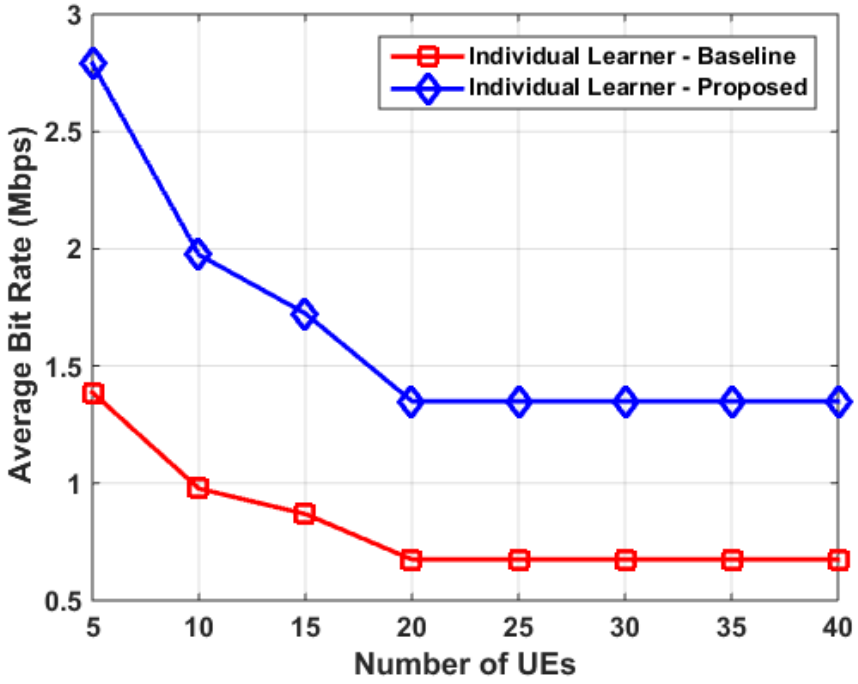

Figure 3.6. Achievable bit rate at SBS using individual learning with $\alpha_t = 0.4$.

Figure 3.6 shows that with $\alpha_t = 0.4$, the proposed individual learning algorithm provides superior throughput performance compared to that of the baseline algorithm. It was observed that the proposed algorithm outperformed the baseline algorithm by 46.79% at $K = 5$; however, in the range $20 \geq K \geq 40$, the difference is approximately 22.50%. It can be noted that the average bit rate that the proposed algorithm achieves in the range $20 \geq K \geq 40$, which is approximately 1.35 Mbps, is what the baseline scheme achieves with $K = 5$. In addition, it is observed that the average bit rate performance for both approaches remains constant in the range $20 \geq K \geq 40$. This is because the solution methods aim to keep satisfying the QoS requirements of all the UEs until the SBS is congested and would not be able to admit any more UEs. From the results in [185], it was observed that when the number of UEs increased above 40 for the same model, the achievable bit rate would significantly drop especially for the cooperative learning approach. This indicates indicates that the systems would have reached a point where there is need for a change in the bandwidth split or a migration of the UEs.

The throughput performance for the individual learning algorithm was also evaluated with a learning rate of $\alpha_t = 0.6$ and the result is shown in Fig. 3.7. From the result in Fig. 3.7, when $K = 5$ the performance of the proposed algorithm is 46.55% higher than that of the baseline algorithm, and

this reduces to 22.50% when $K \geq 20$. Similar to the result in Fig. 3.6, an average bit rate of 1.35 Mbps was achieved by the proposed scheme for the $20 \geq K \geq 40$ range, which is the average bit rate achieved by the baseline algorithm at $K = 5$. There is a 0.0113 Mbps decline in the throughput that was observed after the learning rate was increased, and there is only a 0.24% difference between the results of the two learning rates at $K = 5$. From this, it can be concluded that there is no significant throughput performance sensitivity to the change in the learning rate for the individual learning algorithm. However, the constant throughput performance in the range $20 \geq K \geq 40$ for both the learning rates shows that the throughput performance of the individual learning approach is not affected by the congestion rate at the SBS.

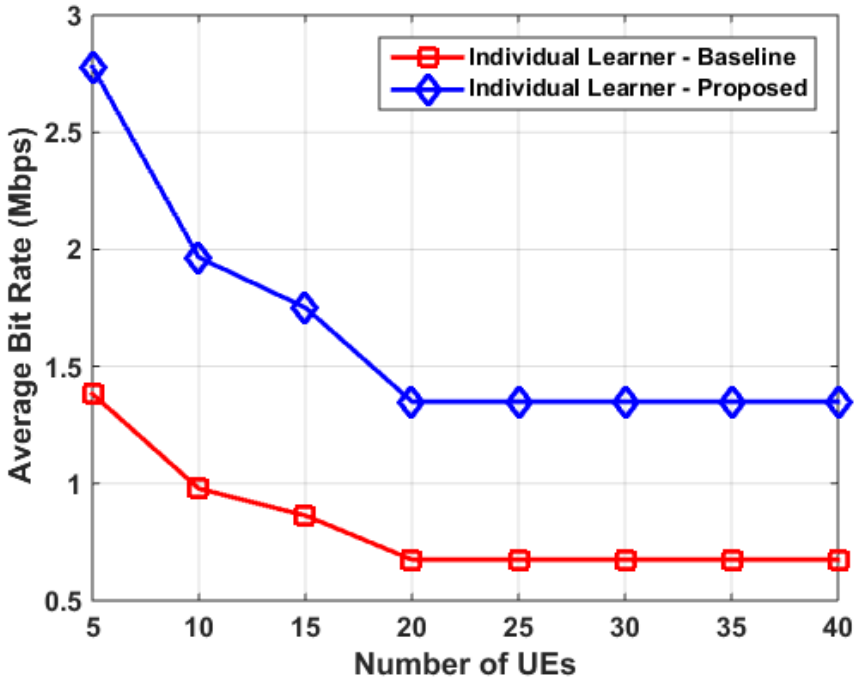

Figure 3.7. Achievable bit rate at SBS using individual learning with $\alpha_t = 0.6$.

3.6.2.2 Cooperative learning algorithm

In the cooperative learning scheme, the current SBS shares the learned policies in the form of value functions with its nearest neighbour. This sharing is consists of the set of states and the system variables that are defined in the state space. The achievable bit rate performance when using the cooperative learning scheme with $\alpha_t = 0.4$ is shown in Fig. 3.8. As shown in Fig. 3.8, when $K = 5$ the proposed cooperative algorithm provides a 46.56% better performance compared to the baseline algorithm and this difference decreases to 21.52% at K = 10. The achievable bitrate decreases steadily, maintaining

almost the same 21.52% better performance compared to the baseline scheme until $K = 40$. In addition, the achievable bit rate performance was also evaluated for $\alpha_t = 0.6$ and the result is shown in Fig. 3.9.

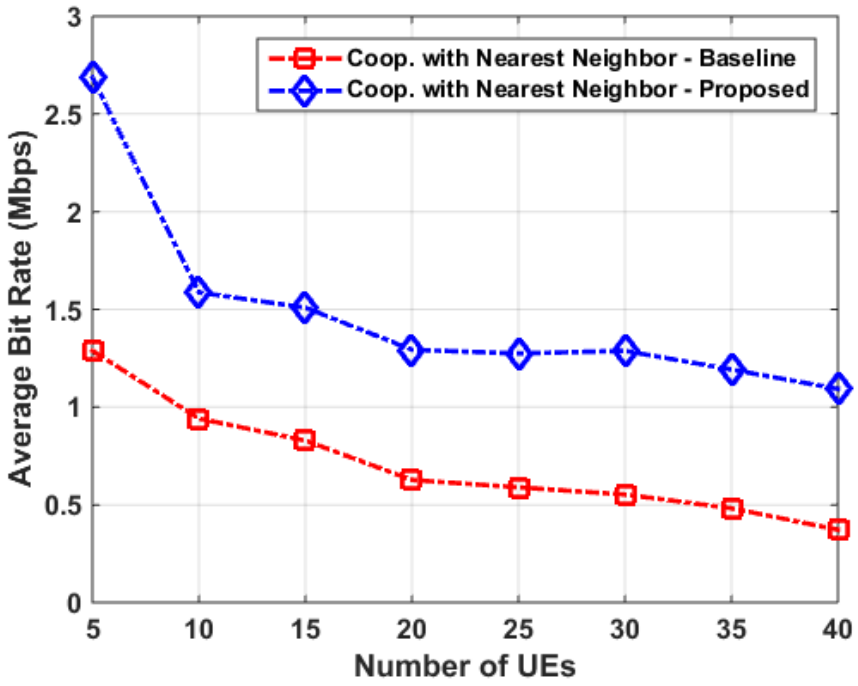

Figure 3.8. Achievable bit rate at SBS using cooperative learning with $\alpha_t = 0.4$.

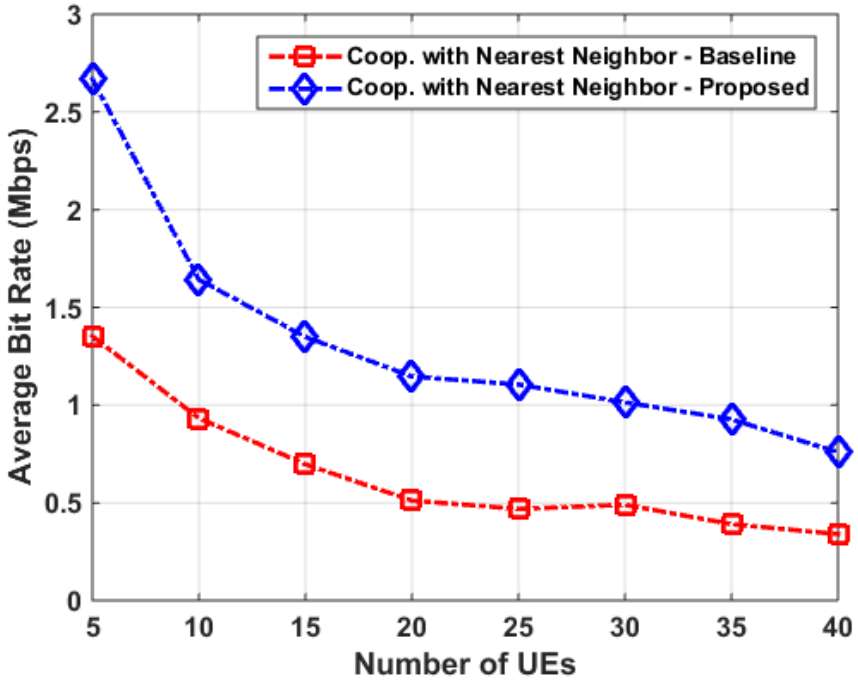

Figure 3.9. Achievable bit rate at SBS using cooperative learning with $\alpha_t = 0.6$.

From the results in Fig. 3.9, the proposed individual learning scheme has 43.84% better performance compared to the baseline algorithm at $K = 5$. This shows a decline of 2.72% in the difference from when $\alpha_t = 0.4$. As the number of UEs increases, the performance continues to decline gradually. This shows that the cooperative learning scheme is sensitive to an increase in the learning rate as well as an increase in the number of UEs on the access network. It can be concluded that with larger values of α_t, as the number of UEs increases, the cooperative learning approach exhibits poorer bit rate performance. As a result, the performance of the nearest neighbour cooperative algorithm is sensitive to a change in the learning rate. These results indicate that the achievable bit rate performance of the nearest neighbour cooperative algorithm is affected by the congestion rate. This is in turn justified by the behaviour shown in Fig. 3.4 at $K = 17$, and in Fig. 3.5 at $K = 13$.

3.6.3 Experiment 3: Quality of Experience

In addition to the evaluation of the congestion and the achievable bit rate performance, user satisfaction was also evaluated to qualify the UEs' perception of the service, that is, the QoE. The user satisfaction is a quality metric that is measured using the mean opinion score (MOS). Some of the system-level

parameters that are related to the throughput performance such as packet losses and the delay are used to measure the QoE using an MOS scale of 1 (worst) - 5 (best). In this experiment, the user satisfaction is evaluated using the regression of the relationship between the required QoS and the available transmission time, while considering the congestion rate. It is then measured based on the MOS, with 5 being the maximum score.

3.6.3.1 Individual learning algorithm

Figure 3.10 and 3.11 show the user satisfaction performance of the individual learning algorithm with $\alpha_t = 0.4$ and $\alpha_t = 0.6$, respectively. The results in Fig. 3.10 and Fig. 3.11 indicate that the degree of satisfaction of the UEs declines uniformly between $5 \leq K \leq 20$, and it tends to remain constant for the remainder of the range. With the learning rate of $\alpha_t = 0.4$, the satisfaction of UEs is as high as 4.5 at $K = 5$, while it is distributed around 4.1 for the range $20 \leq K \leq 40$. On the other hand, when $\alpha_t = 0.6$, at $K = 5$ the satisfaction degree is distributed at approximately 4.8, and distributed at approximately 4.1 for the range $20 \leq K \leq 40$. This showed that the QoE performance of the individual learning scheme has significant sensitivity to a change in the leaning rate.

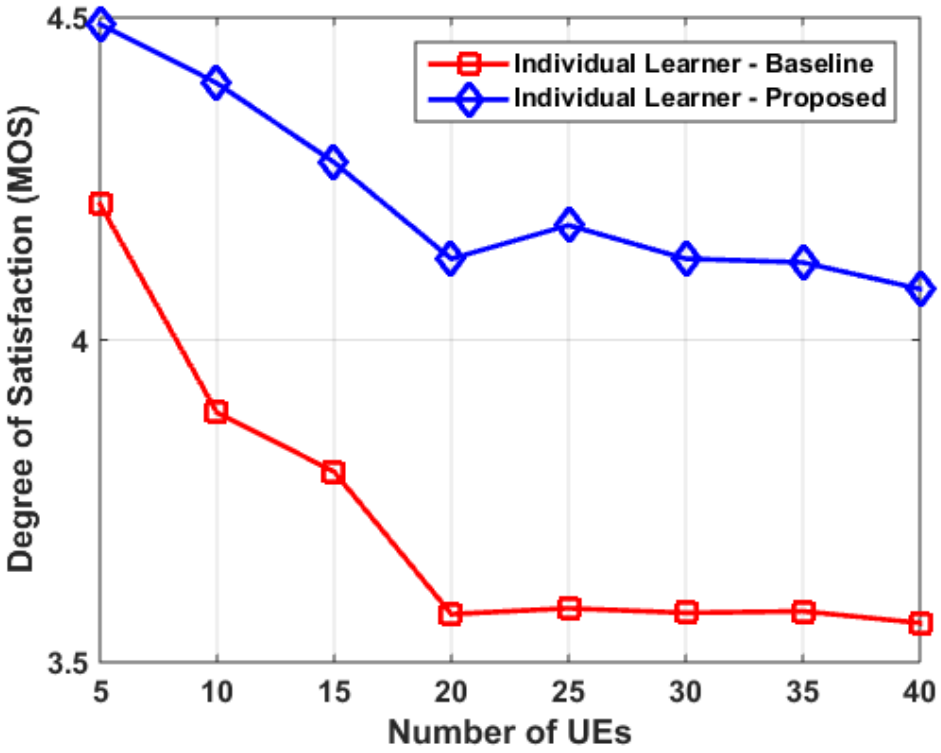

Figure 3.10. UE satisfaction at the current SBS using individual learning with $\alpha_t = 0.4$.

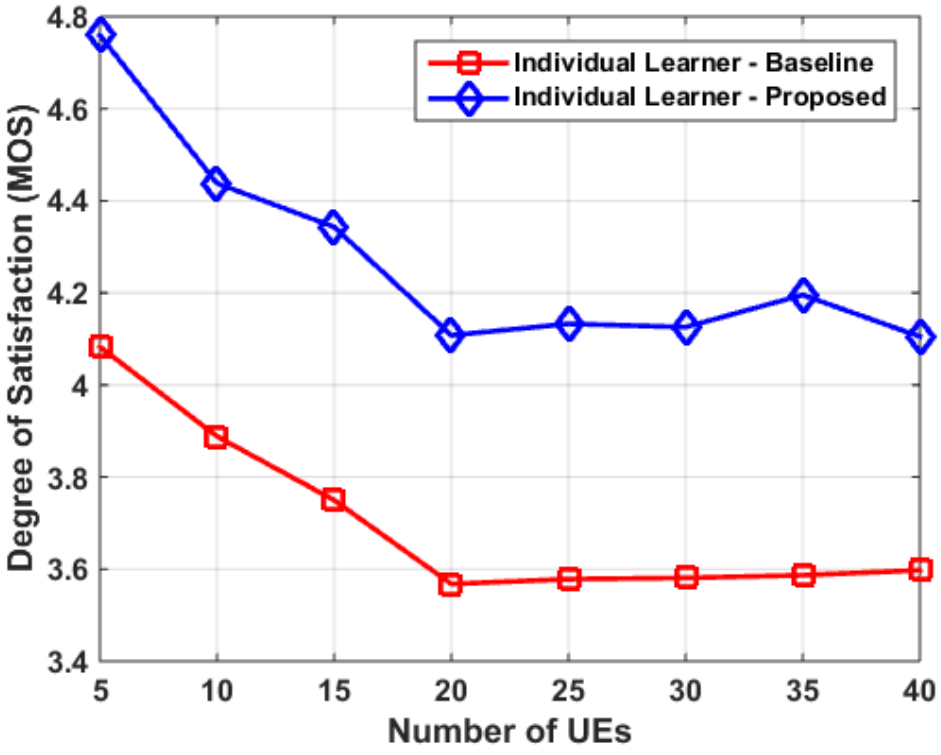

Figure 3.11. UE satisfaction at the current SBS using individual learning with $\alpha_t = 0.6$.

3.6.3.2 Cooperative learning algorithm

Figure 3.12 and Fig. 3.13 show the user satisfaction performance of the cooperative learning algorithm with $\alpha_t = 0.4$ and $\alpha_t = 0.6$, respectively. For the case of the nearest neighbour cooperative learning scheme, the proposed algorithm's good capability of offering better QoE compared to the baseline was similar to the case of individual learning. This showed that user satisfaction performance is not sensitive to a change in the learning scheme. These results are in agreement with recent findings and support the results and analysis in [216]. Even though a discount factor that is closer to 1 requires more time to converge, better performance is guaranteed, which is the objective of this work. Another reason for using a higher discounting factor is the advantage that it has in congestion control. Since congestion is incremental, a higher discount on the rewards maximizes generalization and also avoids over-fitting of earlier learning.

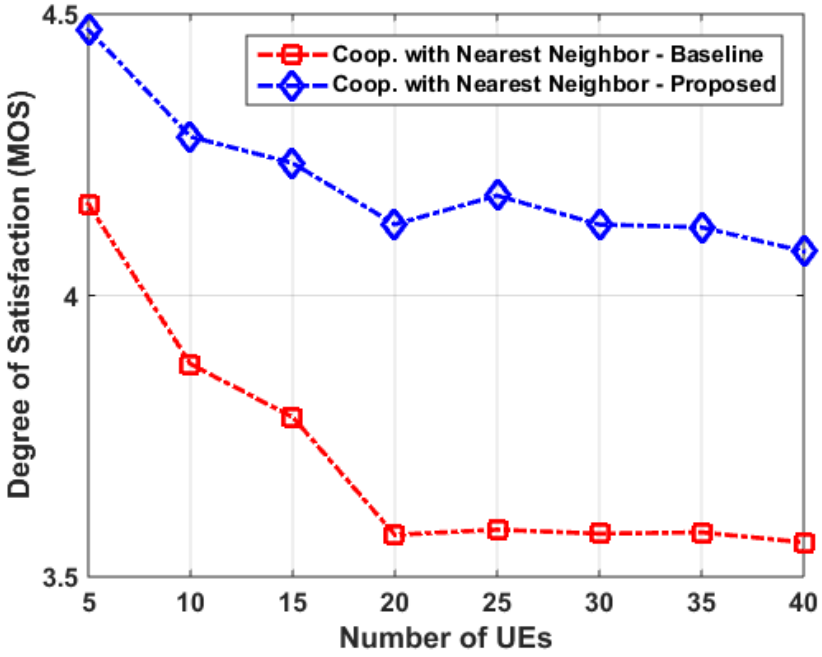

Figure 3.12. UE satisfaction at the current SBS using cooperative learning with $\alpha_t = 0.4$.

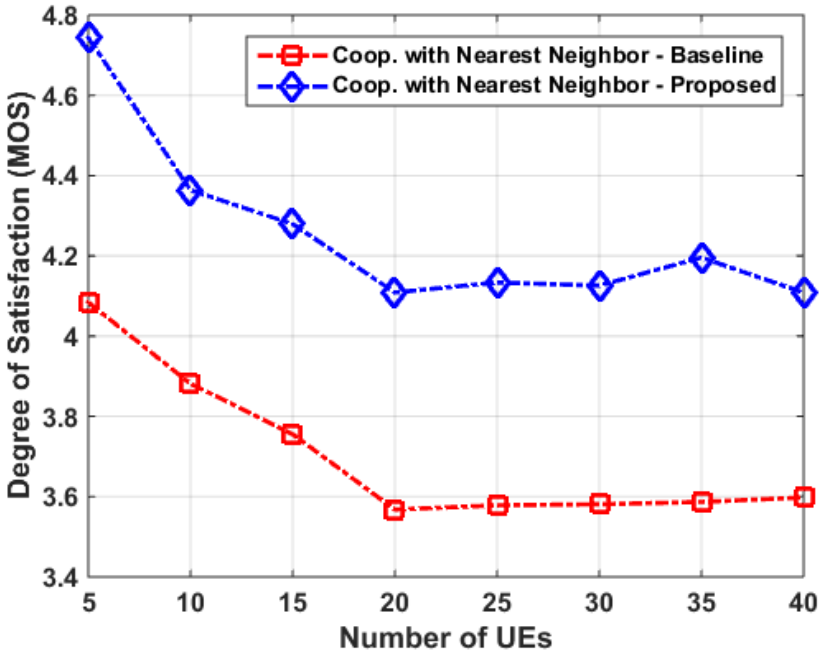

Figure 3.13. UE satisfaction at the current SBS using cooperative learning with $\alpha_t = 0.6$.

3.6.4 Summary of Results

Table 3.3 gives an overview and summary of the performance of the proposed learning algorithms, in comparison to the baseline Q-learning algorithm approach, based on the simulation results.

Table 3.3. Summary of the simulation results of the proposed solution in comparison to the baseline algorithm.

Experiment	**Individual learning**		**Cooperative learning**	
	$\alpha_t = 0.4$	$\alpha_t = 0.6$	$\alpha_t = 0.4$	$\alpha_t = 0.6$
Congestion Performance	Equal performance	Equal performance	More or less same performance	More or less same performance
Throughput Performance	Proposed solution achieving 20% - 50% better performance	Same performance as when $\alpha_t = 0.4$ for both proposed and baseline	Proposed solution achieving 20% - 50% better performance	Same performance as when $\alpha_t = 0.4$
User Satisfaction	Proposed solution achieving 5% - 15% better performance	Proposed solution achieving 10% - 15% better performance	Same performance as for individual learning for both proposed and baseline	Same performance as for individual learning

3.7 CONCLUDING REMARKS

To provide satisfactory RA for users in a mm-wave IAB network model, the congestion rate of an IAB node is monitored by introducing a transmission buffer. Due to the power consumption issues in the RA framework, the problem was converted into a constrained MDP and dynamic power management was also applied. A DRL algorithm was then proposed, where a DNN was trained for optimal power allocation by initializing a power control parameter, θ_t, with zero-mean normal distribution. The DRL algorithm then adopted its output to learn a policy, π, parameterized by θ_t, to achieve the logical allocation of resources by placing more emphasis on congestion control and user satisfaction. The performance of the proposed DRL algorithm was evaluated using two proposed learning schemes, individual learning and the nearest neighbour cooperative learning. It was found that the nearest

neighbour cooperative learning algorithm is suitable for IAB networks because its throughput has good correlation with the congestion rate. From the algorithmic computational complexity analysis, it is evident that the greedy action selection and learning update complexities of the the proposed individual learning algorithm are less compared to the baseline Q-learning algorithm. However, the learning update computational complexity, and consequently the overall complexity of the cooperative learning scheme are the same as that of the baseline algorithm.

CHAPTER 4 DRL-BASED BACKHAUL ADAPTATION IN IAB NETWORKS

4.1 CHAPTER OVERVIEW

Considering the expected high demands for transmission resources and system reliability, especially in urban areas, the IAB systems need to be designed to cope with such requirements. Due to ultra-dense device connectivity, especially during peak hours, the SBSs tend to be congested and their service quality consequently degrades. To address this issue and to answer the questions posed in the hypothesis in Chapter 1, this chapter presents a smart way of achieving robust and efficient backhaul routing in IAB networks by applying DRL. In this work, the effect of packet arrivals and buffer size limitations on throughput and latency requirements in backhaul routing in IAB networks is analysed. The backhaul traffic is routed in a SBS-to-SBS fashion until it reaches the MBS. In the network model, we consider a hybrid access scheme, where the SBS-associated UEs and MBS-associated UEs coexist in the same wireless network. The objective here is to maximize the access capacity subject to the SBS load, the transmission power, and the backhaul bandwidth constraints. The rest of the chapter is organized as follows: Section 4.2 presents the IAB network model, which describes and illustrates the configuration of the data plane and the knowledge plane. Section 4.3 gives the routing optimization problem formulation, whose DRL-RDCM solution is discussed in Section 3.5. A thorough description and computational complexity analysis of the proposed DRL-RDCM algorithm, as well as that of the baseline algorithms, is then given in Section 4.5. The simulation results are presented in Section 4.6 and then Section 4.7 gives the concluding remarks.

4.2 IAB NETWORK MODEL

The uplink transmission of a two-tier multi-hop IAB network, which consists of a single MBS, a set of SBSs, and UEs is considered. The MBS is connected to the 5G core network via fiber backhaul and it is capable of serving some UEs, while the IAB nodes, which also serve the UEs, are connected to each

other or to the MBS through wireless backhaul links as shown in Fig. 4.1. The SBSs are distributed according to a Poisson point process, and the MBS together with the SBSs are assumed to be equipped with multiple antennas and they operate in full-duplex mode.

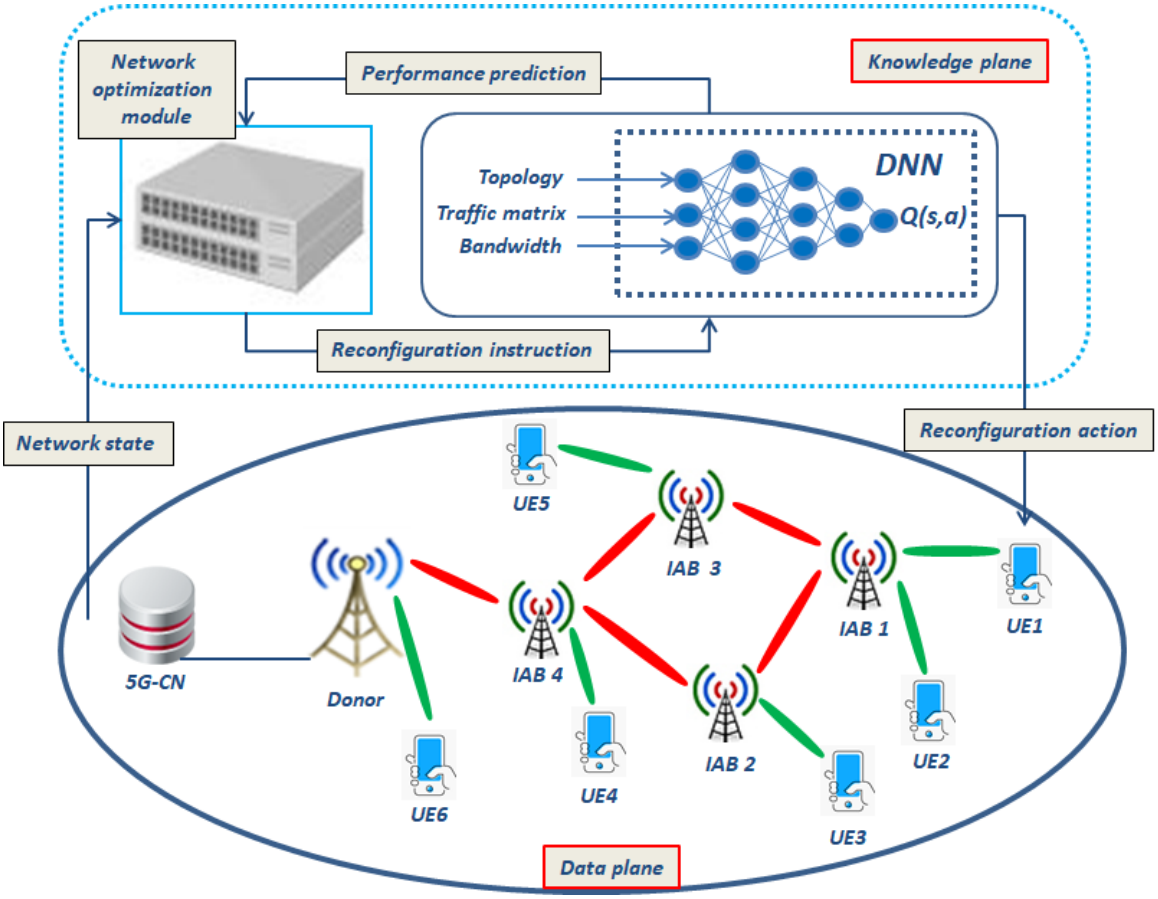

Figure 4.1. IAB network model setup in the 5G standalone deployment scenario showing network components in the knowledge plane and the data plane.

Assuming that the traffic flows are allowed to be scheduled on multiple links, the topology of the IAB network is modeled as an undirected graph. Thus, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes, and $\mathcal{E}$ is the set of edges or links. Here, i and j represent the indices of the transmitting and receiving nodes, respectively, such that $e_{i,j}$ represents the edge/link between the two communicating nodes, and $\mathcal{M} = \mathcal{N} \cup \mathcal{K}$ is the set of all the nodes. In line with KDN as part of the 5G NR requirements, an IAB network that can assemble itself given the high-level instructions, reassemble itself if the requirements change, and autonomously reconfigure itself in the event of an outage, two separate but communicating planes are proposed, which are the data plane, and the knowledge plane.

4.2.1 The Data Plane

As shown in Fig. 4.1, the data plane is where the network nodes, the UEs, and the communication channels are located. It is also where all the signalling and data handling occurs. The data plane of the considered IAB network can be represented using a toy model graph and an example is shown in Fig. 4.2 [58]. Let $|\mathcal{V}|$ and $|\mathcal{E}|$ denote the cardinality of the node and edge sets, respectively, and $\mathcal{E}_i^+$ represent the set of outgoing links from node i. In addition, let $\mathcal{F} = \{1, 2, \cdots, F\}$ represent the finite number of traffic flows, where each flow is assumed to have the attributes that define the source and the destination nodes. Since in a graph tree, a source node or transmitter is defined using index i and the destination node or receiver by index j, $\{(i, j)|j = par(i)\}$ is used to describe the source-destination relationship in the IAB network. Thus, the backhaul links between the IAB nodes and their immediate neighbors, up to the destination, are modeled as edges, $e \in \mathcal{E}$, such that when a route request message from the i-th node reaches the destination node j, the communication link is represented as $e_{i,j}$.

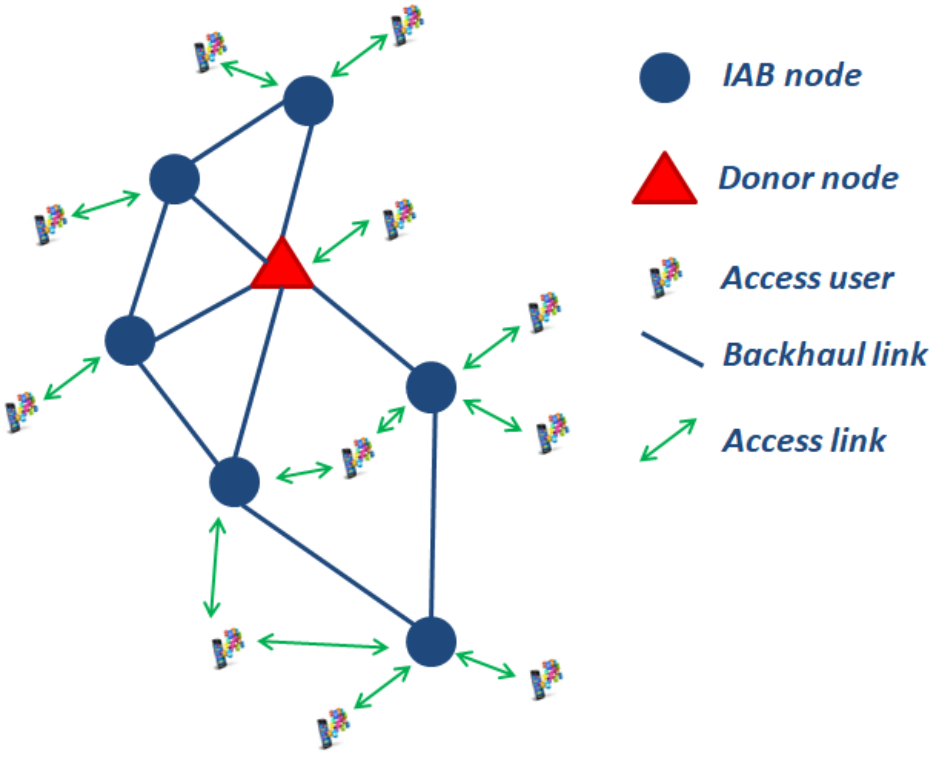

Figure 4.2. A graph model of the data plane of the IAB network showing possible wireless backhaul links between IAB nodes as well as access links with UEs.

4.2.2 The Knowledge Plane and the Network Optimizing Module

The knowledge plane is a distributed network construct that is responsible for gathering, aggregating, and managing information about network behaviour and operation [217]. Similar in operation to the

control plane, it aims to obtain a view of the network topology and it handles all the functions and processes that determine which routes are to be taken by the packets. The ability of the knowledge plane to model the graph-based information of the data plane is made possible through the network optimizing module, which is an OpenFlow device containing the flow table.As such, this plane enables the data plane functions. Using the framework of the knowledge plane in Fig. 4.1, the proposed architecture for this plane's task is illustrated in Fig. 4.3.

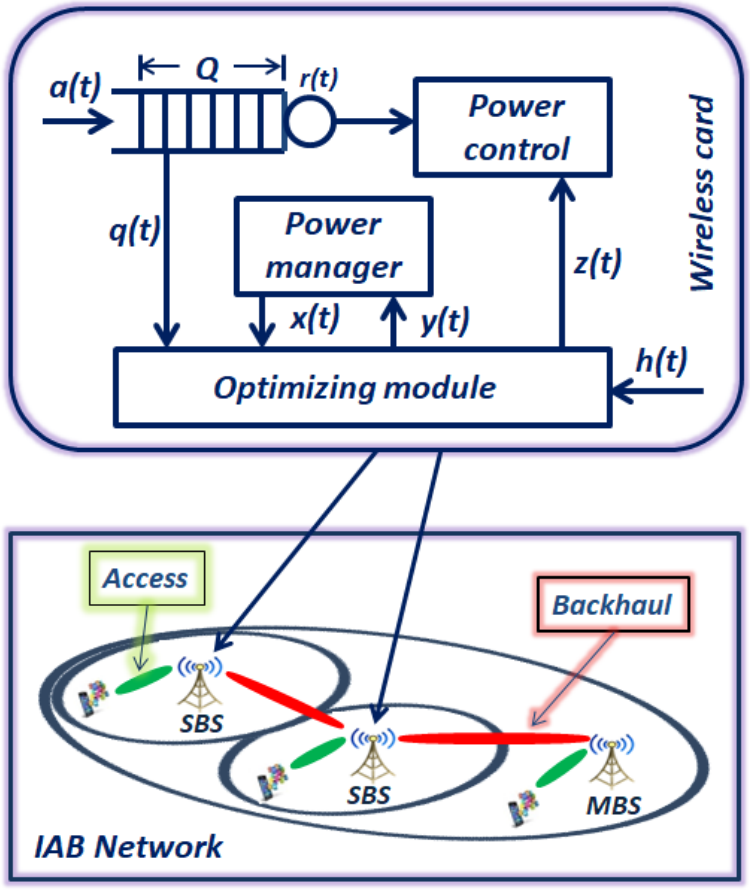

Figure 4.3. Performance prediction and configuration evaluation of the knowledge plane using the wireless card.

In this work, a traffic routing procedure that aims to find good paths between a source and the gateway is developed as part of the framework that includes the support for DTAC. Another objective is to evaluate the reliability of each and every link of nodes that communicate with one another. However, depending on the channel model, the path losses may vary, and the channel gains take into account small fading and interference penalties. To manage the proper resource allocation and routing, each IAB node has to communicate with its neighbors in the network. The task of the optimization module shown in Fig. 4.3 is to keep track of the network operation by monitoring the buffer occupancy status,

that is, the queue state $q(t)$, wireless channel conditions using the feedback obtained as the channel state, $h(t)$, as well as the power consumption status in terms of the power management, $x(t)$. The wireless transmission card shown at the top of Fig. 4.3 consists of a transmission buffer that accepts packets arriving at a rate $\lambda(t)$, with an arrival distribution of $p^{\lambda}(\lambda)$ per time slot. In the evaluation of the network, the following assumptions were made:

(i) the traffic arrival rate, $\lambda(t)$, to each queue is approximated by the Poisson process

(ii) the packet lengths, L, are approximated by an exponential distribution, and

(iii) the queue lengths, Q, are assumed to be independent.

With these assumptions, the analytical results for the buffer occupancy status, that is, the queue load, as well as for the delay distribution, can be derived. The proposed architecture for the knowledge plane exploits the Q-learning technique, with the assumption that all the nodes have the same point of view of the network and run the same algorithm. The exploration of the Q-learning technique in the proposed model is shown in Fig. 4.4.

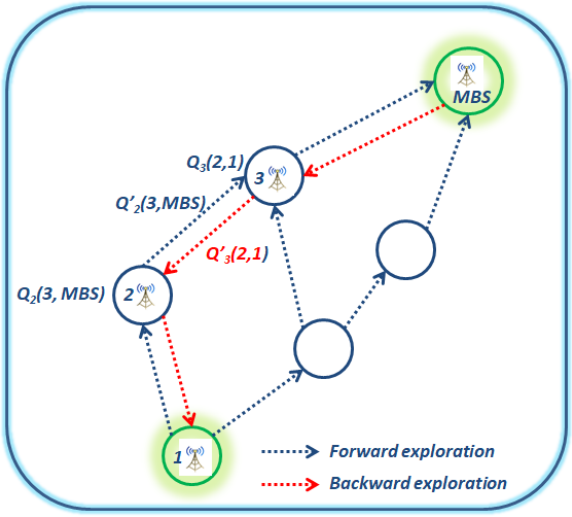

Figure 4.4. The simplified graph model of an IAB environment illustrating the learning exploration procedure.

As part of this framework, the support for DTAC is incorporated using the max-weight scheme, where node congestion is tracked by regularly checking the buffer queue or occupancy status. This provides

a way of evaluating the reliability of the wireless links between the nodes using the backward and forward exploration technique. The process proceeds as follows.

4.2.2.1 Route establishment

According to the process illustrated in Fig. 4.4, let an established transmission pass through the nodes i, j, and j' towards n_0, such that $Q_j(j', n_0)$ is the time that a node j estimates it takes to deliver a packet bound for n_0 via j'. This time estimate includes the time that the packet spends in the queue while being buffered at node j, that is, the holding time. After node j has sent the packet to node j', it immediately receives the estimate of the remaining time for it to reach the destination from node j'. In the network model, each node maintains the information about the Q values for each of the possible next hops. This information represents the delivery time for the packets to reach the MBS. An update regarding the present Q value of each node is sent to the previous node in a process called backward exploration. To keep the Q value as close as possible to the actual values and to also reflect the changes in the state of the network, the estimates of the Q value need to be updated with the minimum possible overhead [58]. Thus, as soon as node j sends a packet $P(i, n_0)$ destined for the MBS to one of the neighboring nodes j', node j' sends its best estimate $Q_{j'}(z, n_0)$ for the destination back to node j, where z is a donor node.

4.2.2.2 Computation of Bounds

It is believed that the maximum bound on the latency can be calculated using the maximum buffer occupancy of each node and the egress link rate [218]. Here, the queue lengths are used to determine the upper bound of latency in terms of the number of nodes deployed to relay traffic to n_0. Upon receiving the estimate, $Q_{j'}(z, n_0)$, node j computes the new estimate using the exploration of Q values. This process is known as the forward and backward exploration, since it involves updating the Q values of the sending node using the information obtained from the receiving node. With every hop of the packet, only one Q value is updated, that is, when node j sends the packet, $P(i, n_0)$, to one of its neighbors, for example, j', the packet can take along information about the Q values of node j. When node j' receives this packet, it can use this information to update its Q values pertaining to its neighbor, that is, node j. Then, when the node j' makes a decision, it uses these updated Q values for node j, then the Q value updates in backward exploration.

4.3 PROBLEM FORMULATION

Considering that the model described in Section 4.2 is discrete time-slotted, the following assumptions were made: (i) the the traffic arrival rate, $\lambda_j^f(t)$, at each node queue is approximated by a Poisson process; (ii) the packet lengths are approximated by an exponential distribution, (iii) the traffic arrival

distribution is unknown; (iv) a wireless transmission card of each node consists of a transmission buffer that can hold a maximum of $\mathcal{Q}$ packets, whose average queue length, $\bar{q}$, can be explained using Little's theorem [219]. Using the number of arrivals and the transmission rate, the evolution of the queue in the transmission buffer can then be represented using the dynamic update equation, which can be expressed as [220]

$$q_i^f(t+1) = \left[q_i^f(t) - \sum_{\forall f \in \mathcal{F}} r_{i,j}^f(t), 0 \right]^+ + \lambda_i^f(t). \tag{4.1}$$

This is the evolution of the queue over time, where $[x]^+ \triangleq \max(x,0)$, and $\lambda_f^i(t) \in A_f(t)$ represents the data arrival rate at node i, with $A_f(t)$ being the set of packets of flow f arriving at the source node, s_f. The point-to-point channel-power states for channel state, $h(t)$, and the transmission power are used to realise the transmission rate as follows:

$$r_{i,j}^f(t) = B_{i,j}(t) \log_2 \left(1 + \gamma_{i,j}^f(t)\right), \tag{4.2}$$

where $B_{i,j}(t)$ represents the backhaul bandwidth, and $\gamma_{i,j}^f(t)$ is the SINR experienced by the traffic flow when transmitted via link (i,j), defined as follows:

$$\gamma_{i,j}^f(t) = \frac{p_{i,j}^f(t) g_{i,j}^f(t)}{\sum_{k \neq j}^K p_{k,j}(t) g_{k,j}(t) + N_0}, \tag{4.3}$$

where $p_{i,j}^f(t)$ represents the transmission power of transmitting flow f from IAB node i to node j, $g_{i,j}^f(t)$ is the distance-dependent channel gain assumed to follow a Rayleigh fading distribution with unitary average power, $g_{i,j}^f(t) \sim \exp(1)$. The first term in the denominator is the aggregated interference from the access users, while the second term, N_0, is the white Gaussian noise spectral density. Since in IAB networks, part of the wireless spectrum is used for the backhaul connection of the SBSs, the SBSs must be able to dynamically reserve resources for backhauling traffic to the gateway, n_0. That is, if $r_{i,j}^f(t)$ in (4.2) is the backhaul rate, then $r_{n,k}(t)$ represents the access rate. Therefore, based on this intuition, the access-backhaul condition can be stated as follows:

$$r_{i,j}^f(t) = B_{i,j}(t) \log_2 \left(1 + \gamma_{i,j}^f(t)\right) \leq r_{n,k}(t). \tag{4.4}$$

4.3.1 The Markov Decision Process

Assuming that the system described follows a Markov process with discrete time intervals, the objective of the agent is to determine an optimal policy, π, that maps a state space, $\mathcal{S}$, onto an action space, $(\pi : \mathcal{S} \rightarrow \mathcal{A})$, which maximizes the expected reward R, while minimizing network delay [221]. An MDP that is represented by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \mathcal{S}')$ can then be formulated. Here, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S}' \rightarrow [0,1]$ is the unknown transition probability function, where $\mathcal{P}(s(t+1)|s(t), as(t))$ is the transition probability from state $s(t)$ to $s(t+1)$ after taking action $a_{s(t)}$. The state set, $\mathcal{S}$, comprises the available bandwidth,

the network load, that is, the number of traffic flows and the traffic demand, and the network status (the channel conditions and interference levels). Generally, the state space can be be summarised into utilisation and the port rate as follows:

$$s_j(t) = \{U_{sw_i}(t), P_{z,sw_i}(t)\} \in \mathcal{S},, \tag{4.5}$$

where $U_{sw_i}(t) \in [0,1]$ represents the current utilization of the flow table of switch i, and $P_{z,sw_i}(t)$ represents the port rate of port z of switch i. On the other hand, the RA decisions constitute the action set, which could be the spectrum and computational resources, as well as the network configurations. In this way, the action set, $\mathcal{A}$, consists of the route choice, the power management, and the throughput, such that the i-th node scheduling action, $a_f(t)$, can be defined as the link to which the flow f is routed, at the allocated transmission power, which can be defined as follows:

$$a(t) = a_j^f(t), \quad j \in \mathcal{V}, \quad f \in \mathcal{F}. \tag{4.6}$$

As such, a scheduling policy π, which maps the system state, $s_f(t)$, to the scheduling action, $a_f(t)$, is defined such that $a_f(t) = \pi(s_f(t))$. The transition function of the MDP is denoted as $\mathcal{P}(j|i,l)$, which is the probability that $s_{i,f}(t+1) = j$ given that $s_{i,f}(t) = i$, and $a_f(t) = l$, is defined as follows:

$$\mathcal{P}(j|i,l) = \begin{cases} 1, & \text{if} \quad l = (j,z) \\ 0, & \text{otherwise.} \quad \forall j \end{cases} \tag{4.7}$$

The reward function can then be represented as

$$R_f^{\pi}(j) = \begin{cases} R(s,\pi(s)), & \text{if} \quad j = d_f \\ 0, & \text{otherwise,} \end{cases} \tag{4.8}$$

where $R(s,\pi(s))$ is the discounted reward for a packet in flow f for being in state j, defined as

$$R(s,\pi(s)) = \mathbb{E}\left[\gamma' \cdot r_{i,j}^f(t)\right], \tag{4.9}$$

where $0 \leq \gamma' \leq 1$ is the discount factor. The cumulative reward expectation of the access-backhaul condition in (4.4) can thus be represented as

$$R_f^{\pi}(j) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=1}^{T}\sum_{f\in\mathcal{F}}\sum_{(i,j)\in\mathcal{E}} R(s,\pi(s))\right], \tag{4.10}$$

where T denotes the horizon for which the system is observed.

4.3.2 Formulating the Constrained Markov Decision Process

Since the action taken to maximize a certain reward always goes with an incurred cost, a cost function, $C_i : \mathcal{S} \times \mathcal{A} \times \mathcal{S}' \to \mathbb{R}$, is defined. Minimizing network delays require that the transmission buffer has to be monitored for queuing delays and packet losses. In this way, a buffer cost is defined to reward the system for minimizing queuing delays, thereby protecting against overflows and subsequent packet losses, as well as penalizing every packet that is lost. The buffer cost is defined as the expected sum of

the holding costs and the overflow costs with respect to the traffic arrival and goodput distributions [222], and can be expressed as follows:

$$g([q,p],\psi,y,z) = \sum_{\lambda=0}^{\infty}\sum_{f=0}^{z} p^{\lambda}(\lambda)p^{f}(f|\psi,z)[q-f] + \eta\max([q-f]+\lambda-Q,0), \tag{4.11}$$

where $[q-f]$ is the holding cost, which represents the number of packets that were in the buffer at the beginning of the time slot. Since a stable buffer is assumed, according to Little's theorem, the holding cost is proportional to the queuing delay [223]. The overflow cost imposes the penalty for each packet that is dropped and it can be given as

$$C_f^{\pi}(j) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T}\sum_{f\in\mathcal{F}}\sum_{(i,j)\in\mathcal{E}}\gamma^{t} g(s,\pi(s))\right]. \tag{4.12}$$

The objective of this formulated constrained Markov decision process (CMDP) is to find a policy, π_θ, which maximizes (4.10), while satisfying (4.12).

4.3.3 The Optimization Problem

A stochastic optimization problem that maximizes the average expected total throughput, subject to the bit rate, queue stability, and airtime consumption constraints is formulated. The optimization problem is expressed as follows:

$$\max_{\mathbf{p},\pi} R_f^{\pi}(j) = \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=1}^{T}\sum_{f\in\mathcal{F}}\sum_{(i,j)\in\mathcal{E}} R(s,\pi(s))\right], \tag{4.13}$$

subject to

$$\begin{aligned} &\mathbf{C1}: C_f^{\pi}(j) \le Q_{th}, \quad \forall j \in \mathcal{V} \\ &\mathbf{C2}: C_f^{\pi}(j) \le C_e, \quad \forall (i,j) \in \mathcal{E}^{+} \end{aligned} \tag{4.14}$$

where the decision vector, $\mathbf{p},\pi$ in (4.13) consists of the transmission power vector, $\mathbf{p}$, whose transmission power range can be defined as follows:

$$p_{i,j}^{f} \ge 0, (i,j) \in \mathcal{V} | \sum_{j\in\mathcal{V}_i}\sum_{f\in\mathcal{F}} p_{i,j}^{f} \le P_i^{max}, \tag{4.15}$$

which ensures that the transmission power assigned to IAB node i does not exceed the maximum allowed transmission power by enforcing a power control condition to the forwarding node. The Q_{th} in **C1** is the threshold on the queue length to prevent buffer overflows and subsequent packet losses. This constraint also puts emphasis on the transmission delay by controlling the packet processing time per node, that is, $0 \le D_j(t) \le \delta_j^f$, where $D_j(t)$ is the instantaneous delay of node j, and δ_j^f is the upper bound on the processing time. Based on the evidence in [221] that the node packet-processing capacity is a very important measure in minimizing delays when the λ_j^f is high, then $D_j(t)$ depends on the node processing capacity, that is,

$$Pr\{D_j(t) \ge d_{j,max}(t)\} \le \delta_{th}, \tag{4.16}$$

where $d_{j,max}(t)$ is the maximum achievable delay of node j, while δ_{th} is the threshold of the probabilistic delay. The constraint **C2** ensures that the required backhaul capacity is always less than the capacity of the link C_e. This constraint means that the average backhaul transmission rate, $\bar{r}^f_{i,j}(t)$, has to be kept below the link capacity thereby ensuring that the long-term arrival rate does not exceed the average transmission rate, which in turn prevents buffer overflows and subsequent packet losses, that is, $\bar{r}^f_{i,j}(t) \geq \lim_{t\to\infty} \sum_i \frac{1}{T}\mathbb{E}\{\lambda_f(t)\}$.

4.4 PROPOSED DRL-RDCM SOLUTION METHOD

The proposed solution algorithm combines the DRL strategy with the RDCM to form a DRL-RDCM scheme that is suited for the next generation routing applications. This proposed approach can provide rapid and accurate route predictions. In this scheme, the mechanism for adjusting the reward value is flexible, that is, if the value of the punishment for choosing a bad route is low, the estimated value of that route will slowly decrease and probably this "bad route" can still be selected in the foreseeable future. On the other hand, if the punishment value is very high, a route may no longer be chosen in future routing events. The proposed framework tries to find a balance between low and high rewards/punishment using an efficient cost model. The proposed framework, which is based on a DRL strategy is illustrated in Fig. 4.5. The proposed DRL-RDCM algorithm computes and updates a policy, π, for the DNN agent to achieve a better level of performance and generality, as shown in Fig. 4.5. The complexity of learning through trial-and-error is reduced using RDCM and this is further discussed in the subsequent section.

In the context of IAB networks, path choice models aim to satisfy requirements such as

- scaling with the increased densification in large urban networks,
- interpreting the behaviour of network elements well, particularly UE behaviour and BS congestion,
- providing accurate predictions under various network conditions.

According to the routing nature and dynamics of the routing problem, a forward-backward exploration technique is used to learn the IAB network attributes such as throughput, average packet losses, and latency. Using this process, both the local and global network attributes update, where the long-term reward is related to the global network performance by looking for routes with the highest success rate. With reference to the graph model in Fig. 4.4, the source node conveys route setup request

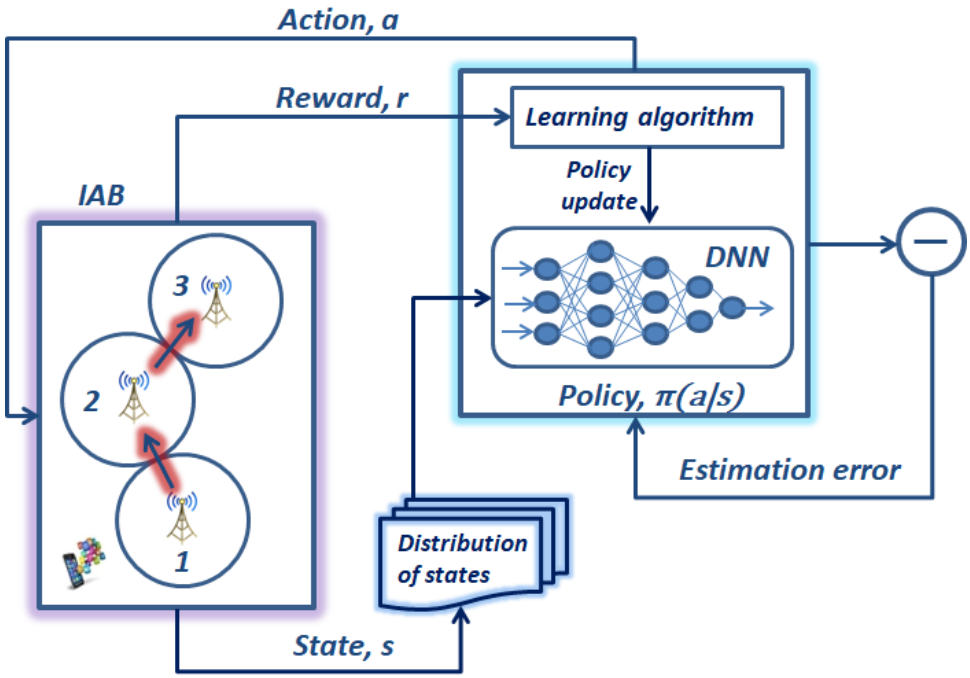

Figure 4.5. Illustration of the interaction between the IAB network and the learning agent in the proposed DRL approach for modeling reward estimation in an IAB network.

messages to the destination, i.e., the MBS. The reward generation and value estimates are presented on a link-by-link basis, and the rewards are propagated based on acknowledgement messaging from other network nodes. In this case, the local reward value is directly related to the receipt of the acknowledgement message for a packet successfully received by the j-th node. A higher value of the local reward means that node j is a good candidate for a link towards the gateway, which increases the probability of being selected for backhaul route establishment in future. Therefore, the set of nodes that are adjacent to node i in this probabilistic graph are referred to as its physical neighbors, and finding the best route between the source and the destination must be rewarded. Since the reward function is intimately tied to the state and action spaces, the Q-learning algorithm is used. In a time-dependent problem such as this one, the distance to the reward is handled using the DRL strategy, where the agent is trained to interact with the environment, and the goal is to maximize the total rewards and to also learn by adjusting its strategy based on the rewards.

4.4.1 The Recursive Discrete Choice Model

The route choice model proposed in this work is based on the assumption that the nodes behave rationally by maximizing a certain utility function, or equivalently by minimizing a certain cost function [224]. In addition, the nodes observe additional parameters that affect their path choice. These

factors vary across nodes and they are unknown to the model. As such, a random term, ε, is added to the cost function. Although the modeller would not know the additional parameters, it knows the family of distributions for ε. The objective is to then infer the probability that a given path is optimal given the current distribution of states.

In this work, a recursive discrete choice model is incorporated in the DRL strategy with the aim of inducing a Markov chain into the graph, $\mathcal{G}$. Considering the graph model in Fig. 4.2, the sets of edges entering and leaving node j are denoted as $\mathcal{E}_j^-$ and $\mathcal{E}_j^+$, respectively. The links or paths passing through node j are represented by e_j, and the route choice model is developed using discrete choice experiment, which incorporates choice overhead by means of a penalty parameter [225]. Choice aversion is computed for each outgoing edge, $e_j \in \mathcal{E}_j^+$, and a collection of i.i.d. random variables, $\{\varepsilon_{e_j}\}_{e_j \in \mathcal{E}}$, are assumed such that the RDCM becomes a recursive logit model [226]. The recursive reward/utility associated with edge e_j is defined as

$$R_f^{\pi}(e_j) = R_{e_j}^f + \mathbb{E}\left(\max_{e' \in \mathcal{E}_j^+} \{V_{e'} - \Omega_{j_e} \log|\mathcal{E}_j^+|\}\right), \tag{4.17}$$

where $R_{e_j}^f$ is the instantaneous reward of edge e_j and the expectation $\mathbb{E}(\cdot)$ is the adjusted continuation value associated with the choice of edge e_j, V_e is the observed realization of random rewards. The factor $\Omega_{j_e} \log|\mathcal{E}_i^+|$ represents the penalty that captures the size of the choice set, that is, $\mathcal{E}_{j_e}^+$, where the parameter $\Omega_{j_e} \geq 0$ is the parameter representing choice aversion [226]. Assuming that the collection of random variables at each node $j \neq n_0$ satisfies the sufficient criterion with a sufficiently scaled distribution as defined in [227], (4.17) can be reformulated as

$$R_f^{\pi}(e_j) = R_{e_j}^f + \log\left(\sum_{e' \in \mathcal{E}_{j_e}^+} e^{V_{e'}}\right) - \Omega_{j_e} \log|\mathcal{E}_{j_e}^+|, \tag{4.18}$$

where the second and third terms represent the closed-form expression of the expectation in (4.17). Since each flow has to find an optimal route to n_0, when the flows reach node $j \neq n_0$ they observe the realization of random utilities, $V_e, \forall e_j \in \mathcal{E}_j^+$, and subsequently choose the edge with the highest utility. This is done by leveraging regret learning, which exploits information about channel states and queue states to choose the optimal route [228]. This intuition is influenced by the learning framework in Fig. 4.4, where the forward and backward exploration are employed in learning the maximization of the long-term utility of traffic flows. This whole process is repeated at each subsequent node, $j' : j \neq n_0$, which results in an RDCM. The expected traffic flow entering a node will then take an outgoing route according to a choice probability defined by

$$\mathcal{P}(e_j|\mathcal{E}_j^+) = \mathcal{P}\left(e_j = \arg\max_{e' \in \mathcal{E}_j^+} V_{e'}\right). \quad \forall j \neq n_0 \tag{4.19}$$

It is worth noting that as the value of the parameter Ω_{j_e} increases, the edge choice probability (4.19) is increasingly penalised by the size of the choice set. This reflects the cost of choice overload onto the edge utility of the user with a large choice set. According to the law of flow conservation, $x_j = \sum_{e \in \mathcal{E}_j^-} f_e$, $\forall j \neq n_0$ is feasible if there exist a unique flow vector that satisfies all the flow constraints. Therefore, the solution of this RDCM can be equivalently written in the form of route choice probabilities, assuming that for each route the utility associated to it is a random variable defined as follows:

$$\begin{aligned} \tilde{R}_f(e_j) &= \sum_{e \in \mathcal{E}} (R_{i,j}^f - \Omega_{j_a} \log |\mathcal{E}_{j_a}^+|) \\ &= \sum_{e \in \mathcal{E}} R_{i,j}^f - \sum_{e \in \mathcal{E}} \Omega_{j_a} \log |\mathcal{E}_{j_a}^+|. \end{aligned} \tag{4.20}$$

Under these conditions, using the choice probability in (4.19), the probability of choosing the route e_j can be defined as

$$\mathcal{P}_e \triangleq \mathcal{P}\left(e_j = \arg\max_{e' \in \mathcal{E}} \tilde{R}_f(e_j)\right), \tag{4.21}$$

which is equivalent to the greedy action selection in [186]. Among the possible multiple routes that the flows can take between the source and the destination, the algorithm selects only one. The flow regulation of rate and delay at ingress can only be ensured along a single path, hence the resource utilisation bounds need to be established.

4.4.2 Formulation of Optimization Bounds

The existence of the non-linear probabilistic constraint (4.16), which ensures that **C1** is satisfied, makes the optimization problem difficult to solve. To circumvent this challenge, the constraint's linear deterministic equivalent is introduced using Markov's inequality such that for a non-negative random variable X and $a > 0$, one can have $Pr\{X \geq a\} \leq \mathbb{E}[X]/a$ [229], which results in

$$Pr\left\{\frac{q_i^f(t)}{\lambda_f} \geq \delta_{th}\right\} \leq \frac{\mathbb{E}[q_i^f(t)]}{\lambda_f \delta_{th}}. \tag{4.22}$$

The mathematical models of queues with deadlines and rewards are used to describe the attributes of the proposed system, such that if the utilization factor follows the accurate stability conditions described in [230], the stability of the system can be guaranteed. Thus, to relax (4.16), the condition of the expected queue length should be satisfied as follows:

$$\mathbb{E}[Q_f^i(t)] \leq \lambda_f \delta_{th} \delta_j^f, \quad \forall f \in \mathcal{F}, \forall t \in T. \tag{4.23}$$

To guarantee that all the flows have a certain minimum level of QoS, a minimum requirement $r_{i,j}^{min}$ is introduced as follows:

$$r_{i,j}^{min}(t) \leq r_{i,j}^f(t) \leq r_{i,j}^{max}(t), \tag{4.24}$$

where $r_{i,j}^{max}$ is the maximum rate constraint which is enforced to avoid the over-allocation of resources when a large number of packets are sent simultaneously, such that $r_{i,j}^f(t) \gg q_f^i(t)$. The optimization

problem can then be rewritten as

$$\mathbf{P}^*: \max_{\bar{\mathbf{r}},\pi} \sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \sum_{(i,j)\in\mathcal{E}} \omega_f R_f^{\pi}(t), \quad \text{s.t.} \quad (4.24), \tag{4.25}$$

where ω_f is a weight assigned to each flow f.

Since the statistical information regarding all the candidate routes is not available, a proper solution to (4.13) is difficult to obtain. Using the reward function in (4.8), the CMDP equivalent of (4.10) can be represented as

$$\max_{\pi} \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \sum_{(i,j)\in\mathcal{S}} \sum_{\tau=0}^{\tau_f} R_f(s_{i,f}^{\pi}(t+\tau))\right], \tag{4.26}$$

subject to

$$\lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \sum_{(i,j)\in\mathcal{S}} \sum_{\tau=0}^{\tau_f} \mathbb{I}\{a_{i,f}^{\pi}(t+\tau)=e\} \le C_e\right], \tag{4.27}$$

where the $\mathbb{E}[\cdot]$ is the expectation taken with respect to the traffic flow arrival process, the transition function, and the optimal policy π. To solve the formulated CMDP in (4.25), the Lagrange duality equivalent of the problem is formulated, where it it assumed that the problem is associated with a Lagrangian, $\mathcal{L}$. The Lagrangian equivalent of (4.26) and (4.27) can be written as follows:

$$\mathcal{L}(\pi,\kappa) = \sum_{e\in\mathcal{E}} \kappa_e C_e + \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \sum_{(i,j)\in\mathcal{E}} \sum_{\tau=0}^{\tau_f} \left(R_f(s_f^{\pi}(t+\tau)) - \sum_{e_j\in\mathcal{E}} \kappa_e \mathbb{I}\{a_f^{\pi}(t+\tau)=e\}\right)\right], \tag{4.28}$$

where $\kappa > 0$ is the Lagrange multiplier. Then, for every feasible policy, $\pi \in \Pi$, it can be observed that (4.26) is bounded below by the formulated $\mathcal{L}(\pi,\kappa)$. Therefore, if the rewards and transition probabilities are the same for every packet in a given traffic flow are the same, then the state-value function can be defined as follows:

$$V_f^{\pi}(\kappa) = \mathbb{E}\left[\sum_{\tau=0}^{\tau_f} \left(R_f(s_f^{\pi}(t+\tau)) - \sum_{e} \kappa_e \mathbb{I}\{a_f^{\pi}(t+\tau)=e\}\right)\right]. \tag{4.29}$$

where $\mathbb{E}[\cdot]$ is the expectation with respect to the underlying transition probability under the policy $\pi_f(\kappa)$. The Lagrangian in (4.28) can be written as follows:

$$\begin{aligned}
\mathcal{L}(\pi,\kappa) &= \sum_{e\in\mathcal{E}} \kappa_e C_e + \sum_{f\in\mathcal{F}} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \sum_{(i,j)\in\mathcal{V}} V_f^{\pi}(\kappa) \\
&= \sum_{e\in\mathcal{E}} \kappa_e C_e + \sum_{f\in\mathcal{F}} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} |\mathcal{E}| V_f^{\pi}(\kappa) \\
&= \sum_{e\in\mathcal{E}} \kappa_e C_e + \sum_{f\in\mathcal{F}} \rho_f V_f^{\pi}(\kappa).
\end{aligned} \tag{4.30}$$

Then, the dual function is obtained as follows:

$$D(\kappa) = \max_{\pi} \mathcal{L}(\pi,\kappa), \tag{4.31}$$

and the dual policy is represented as

$$\pi(\kappa) = \arg\max_{\pi} \mathcal{L}(\pi, \kappa), \tag{4.32}$$

and the optimal dual variable is denoted as

$$d^* = \arg\min_{\kappa \geq 0} D(\kappa). \tag{4.33}$$

Assuming that there is no duality gap, the optimal policy, π^*, of the CMDP is the same as $\pi(d*)$, and that κ and $V_f^{\pi}(\kappa)$ of any flow are independent of all the other flows. In this case, the objective is to obtain the optimal policy, $\pi_f(\kappa)$, for each flow as follows:

$$\begin{aligned} D(\kappa) &= \max_{\pi} \mathcal{L}(\pi, \kappa) \\ &= \sum_{e \in \mathcal{E}} \kappa_e C_e + \max_{\pi} \sum_{f \in \mathcal{F}} \rho_f V_f^{\pi}(\kappa) \\ &= \sum_{e \in \mathcal{E}} \kappa_e C_e + \sum_{f \in \mathcal{F}} \rho_f \max_{\pi_f} V_f^{\pi_f}(\kappa) \\ &= \sum_{e \in \mathcal{E}} \kappa_e C_e + \sum_{f \in \mathcal{F}} \rho_f V_f^{*}(\kappa), \end{aligned} \tag{4.34}$$

where $V_f^*(\kappa) = \max_{\pi_f} V_f^{\pi_f}(\kappa)$, and $\pi_f(\kappa) = \arg\max_{\pi_f} V_f^{\pi_f}(\kappa)$. At this point, $\pi_f(\kappa)$ and $V_f^*(\kappa)$ can be computed using finite horizon dynamic programming.

4.4.3 The Cost Function

The proposed cost function's ability to maintain prior knowledge enables it to accelerate the convergence of the algorithm, which improves the overall delay-power trade-off performance. Since the transition probabilities and the link probabilities cannot be known apriori, a post-decision state-based dynamic programming technique is employed to compute the cost function. Here, the transition probability function is split between the known and the unknown dynamics in order to learn the link probabilities and obtain the optimal policy. Under the assumptions on arrival and processing rates, the analytical results for the buffer occupancy status are used to compute the cost function. The queue load and the delay distribution are taken as the known information and are exploited to develop a more efficient cost function based on the CMDP. In this case, a post-decision state, $\tilde{s}$, is defined, which is related to the current state as follows:

$$\begin{aligned} \tilde{s}(t) &= (\tilde{q}_j(t), h(t), x(t+1)) \\ &= ([q_j(t) - \mu_j(t)], h(t), x(t+1)). \end{aligned} \tag{4.35}$$

The post-decision state in (4.35) represents the state of the transmission buffer after the packets have been transmitted, but just before the new packets arrive, such that the queue length can be represented by $\tilde{q}(t) = q_j(t) - \mu(t)$. Here, the channel state is assumed to be the same as the state at time t, and the

power management post-decision state is the same as the power management at time $t+1$. The state at time $t+1$ can then be represented as

$$\begin{aligned} s(t+1) &= (q(t+1), h(t+1), x(t+1)) \\ &= ([q(t) - \mu(t)] + \lambda(t), h(t+1), x(t+1)). \end{aligned} \tag{4.36}$$

For the next state, $s(t+1)$, the unknown dynamics such as the arrival rate and the channel state, have been incorporated. The introduction of the post-decision state enables the factorization of the transition probability function into known and unknown components. In this case, the known component accounts for the transition from the current state, s, to the post-decision state, $\tilde{s}$. On the other hand, the unknown component accounts for the transition from the post-decision state to the next state, $s(t+1)$. Factorizing the transition probability function results in

$$p(s(t+1)|s(t), a(t)) = \sum_{s} p_u(s(t+1)|\tilde{s}, a) p_k(\tilde{s}|s(t), a(t)), \tag{4.37}$$

where subscript k represents the known component and subscript u represents the unknown component. Since the queue overflow depends on the arrival distribution, which is an unknown component, the queue overflow cost may depend on the action and the post-decision state. Based on the goodput distribution in (4.11), the cost function can similarly be factorised as follows:

$$c(s(t), a(t)) = c_k(s(t), a(t)) + \sum_{\tilde{s}} p_k(\tilde{s}|s(t), a(t)) c_u(\tilde{s}, a(t)). \tag{4.38}$$

Since the goodput distribution has to account for packet losses, the algorithm should penalise the packet overflows. Considering that action exploration is not necessary to learn the optimal policy, the known transition probability function can be defined as

$$p_k(\tilde{s}|s(t), a(t)) = p^x(\tilde{x}|x(t), y) p^f(q(t) - \tilde{q}|\psi, z) I(\tilde{h} = h(t)), \tag{4.39}$$

and the unknown transition probability can be represented as

$$p_u(s(t+1)|\tilde{s}) = p^h(h(t+1)|\tilde{h}) p^\lambda(q(t+1) - \tilde{q}) I(x(t+1) = \tilde{x}), \tag{4.40}$$

where $I(\cdot)$ is the indicator function, which takes a value of 1 if its argument is true, and 0 otherwise. The known and unknown cost functions are then defined as

$$c_k(s(t), a(t)) = \rho([h(t), x(t)], \psi, y, z) + \mu \sum_{\mu=0}^{z} (\mu, \psi, z)[q(t) - \mu] \tag{4.41}$$

and

$$c_u(\tilde{s}) = \mu\eta \sum_{\lambda=0}^{\infty} p^\lambda(\lambda) \max(q(\tilde{t}) + \lambda - \Omega, 0), \tag{4.42}$$

where the parameter η represents the penalty.

The post-decision value function, $\tilde{V}^*$, which plays a similar role as the action-value function in Q-learning can be used to represent the unknown component of the discounted cost as follows:

$$\tilde{V}^*(\tilde{s}) = c_u(\tilde{s}) + \gamma^t \sum_{s(t+1)} p_u(s(t+1)|\tilde{s}) V^*(s(t+1)). \tag{4.43}$$

The minimization of the cost function can be obtained by substituting the unknown component into the known one as follows:

$$V^*(s) = \min_{a \in \mathcal{A}} \left\{ c_k(s,a) + \sum_{\tilde{s}} p_k(\tilde{s}|s,a) \tilde{V}^*(\tilde{s}) \right\}. \tag{4.44}$$

The optimal policy of the post-decision state-value function can then be computed as follows:

$$\pi^*_{post}(s) = \min_{a \in \mathcal{A}} \left\{ c_k(s,a) + \sum_{\tilde{s}} p_k(\tilde{s}|s,a) \tilde{V}^*(\tilde{s}) \right\}. \tag{4.45}$$

To keep the system at equilibrium, when the queue length approaches its maximum, the system has to quickly generate a policy for an optimal action to reduce the queue length by increasing the transmission rate. As such, the QoS parameters such as the packet goodput and packet holding costs are considered to account for the increase in transmission power as the transmission rate increases.

4.5 ALGORITHM DESCRIPTIONS AND COMPUTATIONAL COMPLEXITIES

In this section, the basic formulation of the proposed DRL approach, which uses a DNN, is introduced. The training and inference phases of the proposed algorithm are separated to improve clarity and the understanding of the analysis of the computational complexity. In addition, the associated computational complexities for the proposed algorithm and other baseline algorithms that were used for comparison are also analysed.

4.5.1 Training and Optimization of the DNN for Action Selection

The topology of the DNN that is implemented by the agent in the DRL strategy is a feedforward MLP neural network with linear hidden neurons and sigmoid output neurons [208]. The feedforward MLP receives input data for routing in the IAB network, which consists of the node's ID of the packet that should be transferred through to the gateway. The interface status or utilisation represents the information about the status of all interfaces for the node/router. As some nodes may fail due to power issues, damage, congestion, as well as environmental interference, this should not affect the overall performance of the IAB network. In terms of node failure due to either of these occurrences, the routing protocol should use the information it has to find alternate links and routes towards the gateway. The procedure for training the DNN is outlined in Algorithm 5.

Algorithm 5 Procedure for training the DNN.

Input: State, $s(t) \in \mathcal{S}$

Initialise environment for IAB network;
Input $s(t)$ into DNN
For each state, $s \in \mathcal{S}$, **do**
  Calculate the step size, i.e., the learning rate
  Randomly pick $w_1, \cdots, w_d$ according to $\mathcal{N}(0, I_d)$
  For each iteration of the training episode **do**
    Find step length and sample minibatch of input data, and
    Run SGD and update weights
  End For
  Determine available action $a(t) \in \mathcal{A}$ and estimate $Q(s(t), a(t))$
End For
Return $Q(s(t), a(t))$

4.5.2 The DNN Optimization and the DRL Algorithm

To train the DNN, information about a known network such as the topology and the link capacities is required. In addition, the time-series, that is, the knowledge of the traffic passing over the network in a certain period of time, is another essential requirement. In this work, a dataset with topology and aggregated information about the traffic, which comes in the form of an $N \times N$ traffic matrix was used, where the element in row i and column j represents the total amount of traffic, that is, the average bandwidth in a certain period of time between nodes i and j. With the state space shown in (4.5), the optimization of the MLP was done using the analysis of the number of neurons in the hidden layer, and using three training sets, which are: training, validation and testing. The training, validation, and testing sets were created based on the topology of the IAB network, and the number of samples for training the model depended on the type of router for which the DNN agent was created. For each DNN layer, a matrix multiplication and an activation function is computed in forward propagation, and the ReLU in the hidden layers computes the transfer function [231]. The procedure for the proposed DRL strategy is outlined in **Algorithm 6**.

Algorithm 6 Procedure for DRL with RDCM.

Input: $\lambda_j^f(t)$; Buffer size, Ω; T, α_t, γ^t

Initialize buffer occupancy as $q(t)$

Initialize post-decision state value function $\tilde{V}^0$

Create candidate set of routes for traffic flow

For each link (i,j) **do**

  Find link to nearest node and observe SINR

  If current SINR $\gamma_{i,j} \geq \gamma_0$ **then**

    Select $a_f(t) = \arg\max_a Q(s_f(t), a_f(t); \theta)$

    Take transmission action, i.e., $a_f(t) = \arg\min_{a \in \mathcal{A}} \left\{ c_k(s,a) + \sum_{\tilde{s}} p_k(\tilde{s}|s(t), a(t)) \tilde{V}(\tilde{s}) \right\}$

    Observe transition to next state, $s(t+1)$

  Else

    Request route on another candidate link

  End If

  Observe the post-decision state experience tuple $\{s(t), a(t), \tilde{s}, c_u, s(t+1)\}$

  Populate transition probabilities, $(s(t), a(t), p(t), r(t), s(t+1))$

End For

Return $\pi^*(q(t))$, $\hat{r}(t)$, $c(q(t), y(t))$

4.5.3 Action Selection, Reward Computation, and Cost Computation Complexities

4.5.3.1 Action selection complexity

Without putting any restrictions on the weights in the DNN, each threshold activated neuron was simulated with a sigmoid activation at the output by computing the transfer function. The max-weight and back-pressure algorithms were used to determine which link to activate. The max-weight checks the maximum queue at any node and gives out the results accordingly, while the back-pressure algorithm checks the difference between two nodes to determine which link should be activated. The whole operation has a run-time complexity of $\mathcal{O}(n)$ in forward propagation as well as in the backward propagation. Then, the run-time computational complexity of both the forward and the backward propagation can be obtained as $\mathcal{O}(n \cdot n) = \mathcal{O}(n^2)$ [232].

4.5.3.2 Reward computation complexity

The outputs of the DNN agent are the actions, which serve as the input to the Q-learning algorithm, whose first task is to select the action that maximises the reward. Since the used evolution metric is the

total reward that is collected by the agent in every training episode, the value of the learning rate is critical. The DRL considers obtaining the long-term reward by performing the choice evaluation/aversion procedure to generate higher rewards and lower costs. This evaluation process is carried out using the defined graph structure and the RDCM, where a number of candidate routes are evaluated in terms of utility and cost. The learning update and the computation of the reward and cost consequently result in a run-time complexity of $\mathcal{O}(n^2)$.

4.5.3.3 Cost computation complexity

The mechanism for adjusting the reward value during the learning process has to be flexible, that is, the adjustment may not be so small that it does not cause changes nor may it be too large to induce sudden change due to a specific events. For example, if the value of the punishment after choosing a bad route is very low, the estimated value of that route will slowly decrease and probably this bad route can still be chosen for a long time. On the other hand, if the punishment value is too high, a route may no longer be chosen because of just one packet loss event. Therefore, a balance should be established between the low and high rewards/punishments. It is thus helpful to evaluate the reliability of backhaul routes in terms of the cost of delays and power.

Unlike the conventional Q-learning algorithm, which uses a sample average of the action-value function to approximate Q^*, the post-decision state learning approach uses a sample average of the post-decision value to approximate $\tilde{V}^*$. Assuming an overlay of application peers and end-to-end-links, which form a data delivery tree, the costs of all the end-to-end links is required. In the post-decision state learning algorithm, the state space is characterised by the buffer state, and the only action is the throughput, which is subject to packet losses. As the algorithm updates the state-action pair, it only provides information about the buffer-throughput pair. The post-decision state learning provides the information about every state-action pair that can potentially lead to all the corresponding buffer-throughput pairs. It is worth noting that here, the experience tuples are updated in parallel, and as such, the post-decision state learning algorithm has the same memory requirements as the DRL algorithm. Thus, the computation of the cost function through the use of the value iteration approach has a sample complexity of $\mathcal{O}(n^2)$, which is similar to the results found in [233]. Overally, the post-decision state learning algorithm does not require more memory than the Q-learning algorithm that is used in the RL strategy, hence the computational complexity of the post-decision state learning algorithm can be determined as $\mathcal{O}(n \cdot n^2) = \mathcal{O}(n^3)$.

4.5.4 Descriptions of Baseline Algorithms

Due to the limitations surrounding the proper utilization of resources in the IAB networks as well as the nature of the route requests and discoveries, a reliable benchmark algorithm has not been identified. To this effect, based on the stochastic nature of the IAB networks, the traditional DRL and the generative model-based learning (GMBL) approaches were selected as benchmark algorithms for this work.

4.5.4.1 Generative model-based learning

The GMBL is a "plug-in" model-based ML technique that is used to build maximum likelihood estimates of the transition model in the MDP from the observations to determine an optimal policy. It operates by preserving a local linear relationship utilizing the Laplacian matrix with the aim of maintaining the graph-based structure of the original data in the Hamming space [234]. In this technique, each link is sampled a predefined number of times to determine its statistics to a desired level of accuracy. The resulting model is then used as an input to the CMDP framework. Moreover, by automatically assigning weights for each view to improve the clustering performance, the method takes distinctive contributions of multiple views into consideration. In this work, an alternating iterative optimization method is designed to solve the resulting optimization problems. It is, however, difficult to implement since all the nodes have to generate packets on their own in order to sample links. The sample complexity of the GMBL is proportional to the number of links in the network topology, which is consistent with the number of unknown parameters such as link success probabilities.

4.5.4.2 Deep reinforcement learning

By incorporating DL into the RL-based solution, the DRL approach allows the agents to make decisions from unstructured input data without manual engineering of the state space. As a result, the equilibria of this strategy differs along three task complexity measures, that is, (i) the cardinality of the choice space, where a state is equivalent to the information set facing the player along the path leading to the equilibrium; (ii) the level of iterative knowledge of rationality, and (iii) the level iterative knowledge of the strategy [235]. The greedy action selection of game theory and RL is illustrated with an almost similar complexity. However, more information is integrated in the RL strategy with the learning update, and as more information is integrated into an algorithm, it becomes more computationally complex to implement. The state-of-the-art RL strategy in resource allocation states that the computation time is lower-bounded by $\mathcal{O}(n^3)$.

4.6 PERFORMANCE EVALUATION

4.6.1 Network Model Setup

In the experimental setup, the IAB network was deployed according to the standards of the 5G standalone deployment, with the 5G core network completely disconnected from the 4G EPC. The radius of the deployment area is 1000 meters, and the donor node is 250 meters from the nearest IAB node, while the IAB nodes are 100 meters apart. A random walk model was adopted in simulating the UE mobilities. Since the wireless connections cannot be connected to the server at the same time, each user activates its wireless connection to the server using 802.11 links. To avoid collision and to provide better QoS to the traffic flows in the dynamic network, time-sharing is employed. The RDCM was used together with the max-weight and the back-pressure routing for route choices, checking link qualities, and handling the routing aspects of the network, respectively.

4.6.2 Simulation Parameters

To evaluate the performance of the proposed DRL strategy, the assumption of network heterogeneity was made. The simulations, which affirm the potential of the proposed algorithm, were conducted using MATLAB ®R2021b software. Here, 100 SBSs were deployed in a randomly distributed manner over a 1000 radius, with the gateway placed towards the end of the network as shown in Fig. 4.1. Small cell connection distances that are indicated in Fig. 4.2, were set to unity, which means that the route lengths are measured in number of hops and delays are measured using queue lengths and waiting times. The rest of the simulation parameters are tabulated in Table 4.1.

The learning parameters stated in Table 4.1 were adopted from the results in [186]. An initial randomized policy was set to a uniform distribution, and the inter-arrival time of a Poisson arrival process is an exponential random variable. In this way, the local reward is given to the route that has the best rate of success in delivering packets within their deadlines. The NetworkX library in python for producing random graphs from a given set of edges [236] was used to set up the network.

Table 4.1. Simulation parameters.

Parameter	Value	Unit
Component carrier frequency	28	GHz
System bandwidth, B	20	MHz
Subcarrier spacing	60	kHz
Maximum transmission power	80	mW
Maximum number of nodes, N	5	-
Finite buffer size, $\mathcal{Q}$	25	packets
Fixed symbol rate, $1/T_s$	500×10^3	symbols/sec
Time slot duration, Δt	0.5	ms
Finite horizon, T	125	time slots
Packet arrival rate	10	packets/slot
Base station processing time	0.6	msec/request
SDN controller processing time	0.2	msec/request
Discounting factor, γ^t	0.75	-

4.6.3 DNN Training Performance

Using the online approach, the testing of the online training framework, where the DNN is continuously being trained while being applied to the IAB network was carried out. The ability of the proposed approach to adopt accuracy in IAB routing as well as the ability of the model to continuously learn from ongoing interactions with the IAB network and automatically re-adapt on-the-fly to changing dynamics was evaluated. The performance evaluation was performed for 40 epochs, which each epoch run over 50,000 iterations, and the training results are as follows.

Figure 4.6 shows the test errors of the SGD as a function of the number of epochs. This is shown using the average MSE per training epoch, and it indicates the good performance of the algorithm in all the three aspects, that is, training, testing, and validation. In the implementation of the SGD, the speed of convergence was enhanced by initializing the weights using heuristics, and by using Nesterov's momentum and dropout [237]. It is also apparent that the algorithm performs well as the distance between the testing and validation curves is minimal.

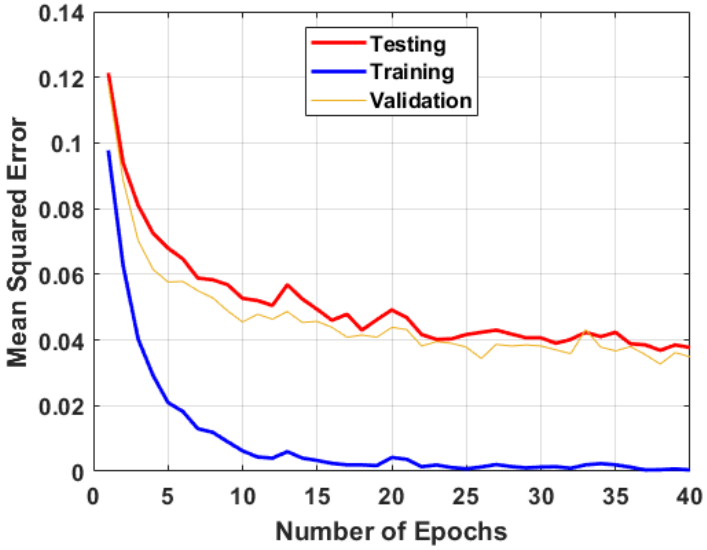

Figure 4.6. Training, testing, and validation loss using the mean squared error.

Figure 4.7 shows the average accuracy as a function of the number of epochs, for both the training and the testing. These results indicate a good performance of the algorithm in terms of learning from the data set, as the training accuracy immediately peaks at $\geq 90\%$, while the testing accuracy reaches 90% accuracy after two epochs. Both these plots are a little noisy, giving the impression that the training algorithm is not making steady progress, however there is an indication that good results will be obtained when the real network data is used to train the system. On overall, these results indicate that the performance loss in terms of training and testing is already low after five training epochs, which suggests that the MLP can be adopted for in the IAB problem under consideration.

4.6.4 Evaluation of Route Choices Using Choice Aversion

In this subsection, the performance of scheduling and the backhaul route selection is evaluated using route prediction probabilities. To improve the understanding of the relationship between two variables against each other, a traffic trace obtained from a 5G standalone testbed at the National Chiao Tung University, China was used. The results in this subsection show how two dependant variables on two different axes vary with a common independent variable. The time-series evaluation of the resource demand and capacity is shown in Fig. 4.8.

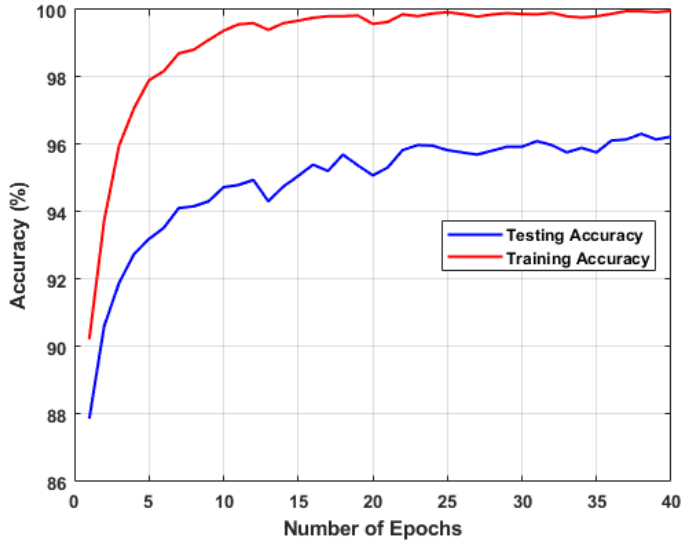

Figure 4.7. Training and testing accuracy.

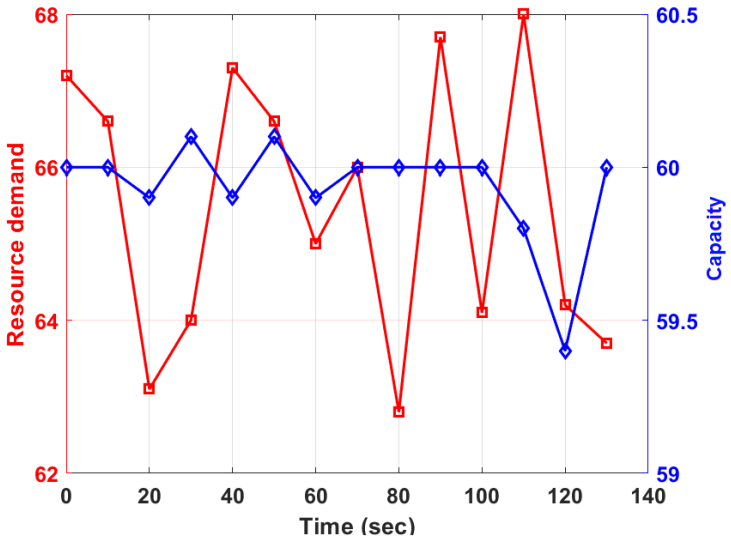

Figure 4.8. Resource demand and capacity vs time.

In Fig. 4.8, it can be seen that in most instances the resource demand is more than the available capacity, which means that the packet arrival rate is more than the departure rate, thus putting the system under pressure. The instability of the system is shown by the lack of proper correlation between the resource demand and the capacity. The time-series evaluation of the number of entities in the node queue and the total delay is shown in Fig. 4.9. As shown in Fig. 4.9, the number of entities in the

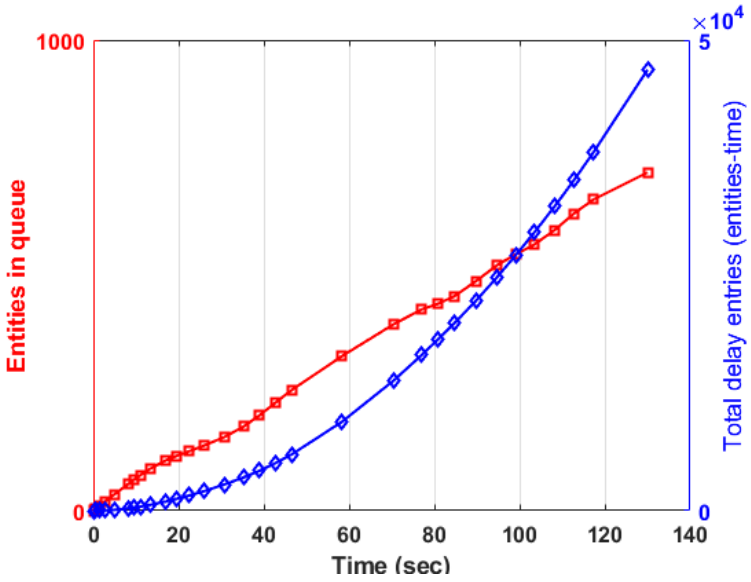

Figure 4.9. Variation of entities in a queue and total delay entities with time.

queue increases almost linearly with time, while the total delay increases exponentially with time. This behaviour is expected since the Poisson arrival rate is modeled as an exponential function. This Poisson-exponential behaviour adequately describes the first order auto-regressive model in a manner that is independent of the average level of the queue. When considered from a classical statistics point of view, each observation would be assumed as a sample of a given random variable, which would usually be i.i.d., thus neglecting the correlation of the evolutionary process.

The route prediction probabilities generated by the RL-RDCM were evaluated as a function of an increasing value of the choice aversion penalization, η as shown in Fig. 4.10. The figure shows that as the value of η increases, the probability of packets being routed on route 1 increases. This means that the choice aversion model assigns more packets to route 1 because it has no alternative comparison in terms of minimum cost. This justifies the basic premise of prospect theory in environments with model uncertainty that the agents tend to explore the route with minimum cost when they are reminded about the incremental cost of their actions.

The cumulative reward, that is, the throughput, of the backhaul traffic was observed in terms of the learning rate, α_t, and the discount factor, γ^t. Firstly, the effect of changing the value of γ^t is evaluated on route 1, that is, the route with the highest choice probability. The performance of route establishment is evaluated using the Q-learning algorithm and the DNN architecture with the learning rate kept constant,

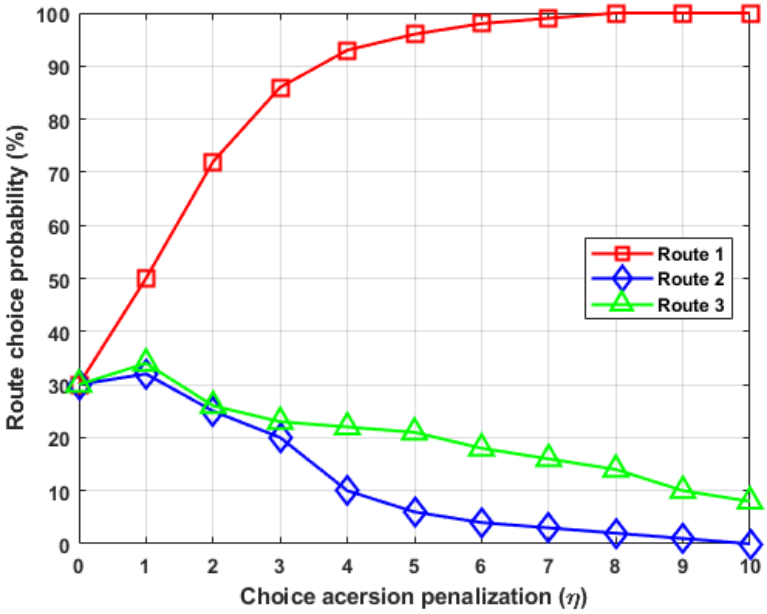

Figure 4.10. Route choice probabilities using the choice aversion model vs the penalization factor.

that is, $\alpha_t = 0.6$, while the value of the discount factor is increased. The diagnostic performance on route 1 is shown in Table 4.2. The results shown in Table 4.2 were run over 1000 iterations, and the role of the discount factor was to determine how much the learning agent cares about the rewards in the distant future relative to those in the immediate future. A higher value of the reward was obtained for $\gamma' = 0.75$ than when γ' is increased. It was observed that when $\gamma' \geq 0.85$, the sums do not converge for the policy, that is, it sums up to infinity. This means that at higher discount rates, the proposed algorithm becomes impulsive in choice behaviour and does not show impulsive responses at lower discount factors. This raises an important aspect that has always been ignored when AI strategies are applied in routing problems, which motivates a more interesting performance-complexity trade-off for the IAB networks design.

Table 4.2. The effect of the discount factor on the reward.

Discount factor, γ'	Overall reward, $\hat{r}$	Most efficient path
0.75	99.99994034	0, 1, 3, 9, 10
0.80	99.99651029	0, 1, 3, 9, 10
0.85	99.98082485	0, 1, 3, 9, 10

An additional analysis on the effect of the learning rate was conducted, and the relationship between the cumulative throughput and the number of iterations is shown in Fig. 4.11. The cumulative throughput shown in Fig. 4.11 above is for different learning algorithms, all using the same transmission budget. The transmission budget was set and the cumulative throughput was evaluated for all the algorithms and the results show an increasing throughput trend for all the algorithms as the number of iterations increase. The proposed DRL-RDCM outperforms the conventional DRL and the GMBL by benefiting from the choice model that is used in its design. It should be noted that all the algorithms have a similar neural training, but they differ in the RL agents they use. To test this relationship even further, the learning rate, α_t, is varied under the same behaviour of the discount factor. The performance for $\alpha_t = 0.4$ is shown in Fig. 4.12.

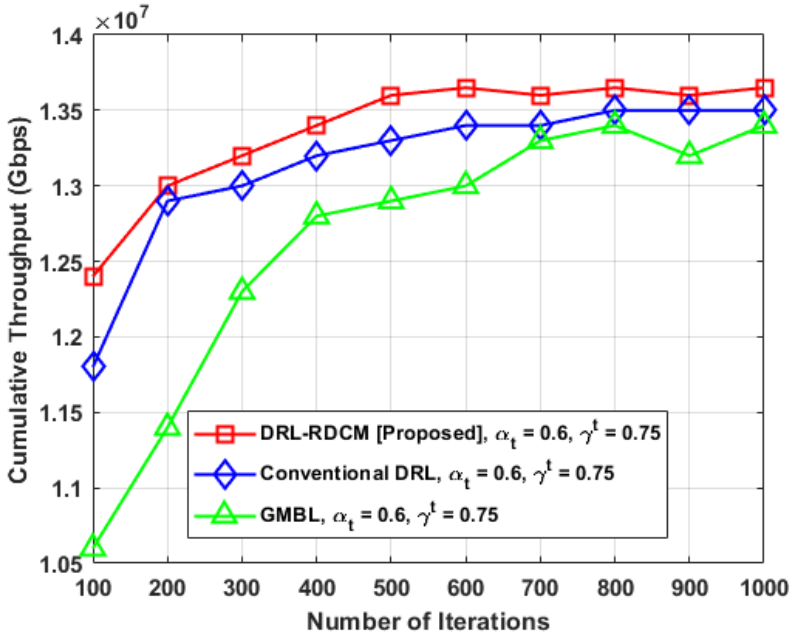

Figure 4.11. Cumulative throughput vs number of iterations with $\alpha_t = 0.6$ and $\gamma^t = 0.75$.

The results shown in Fig. 4.12 above show that the oscillations caused by a large learning rate are reduced. The learning rate was further reduced to 0.2 and the performance is shown in Fig. 4.13. The simulation results suggest that reducing the learning rate helps the algorithm to learn better and prevent the agent from being myopic and only learn actions that would produce immediate rewards. However, this was achieved at the cost of an increased time complexity. The action selection of the DRL strategy was shown to have similar complexity to the GMBL, the only difference being that the GMBL follows a procedure under which each link is sampled a given number of times to determine its statistics to a

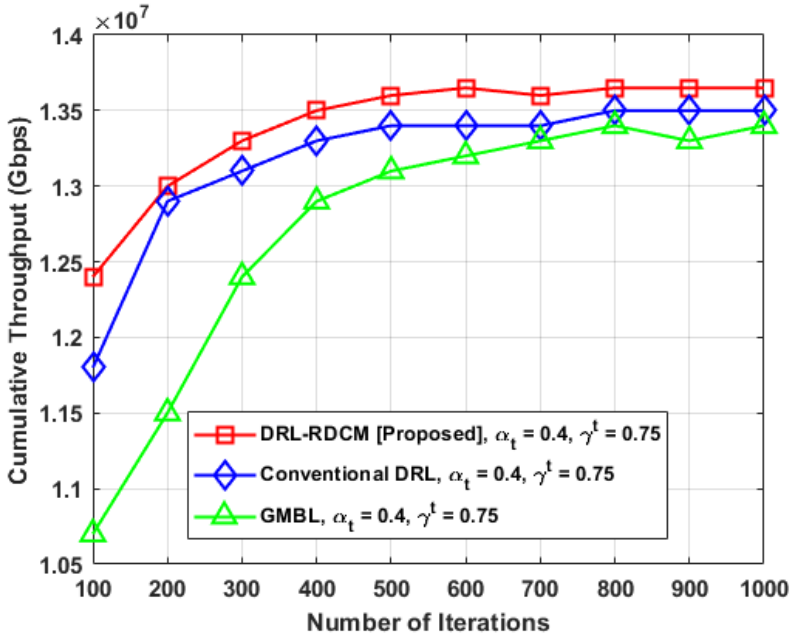

Figure 4.12. Cumulative throughput vs number of iterations with $\alpha_t = 0.4$ and $\gamma^t = 0.75$.

desired level of accuracy. The resulting model is then used as an input to the CMDP framework and how much the prediction error affects this adjustment also depends on the learning rate.

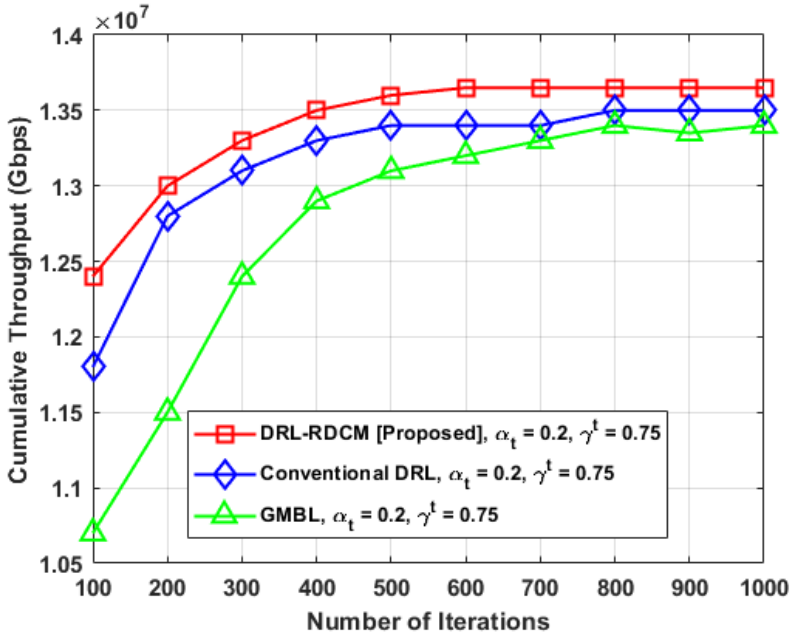

Figure 4.13. Cumulative throughput vs. number of iterations with $\alpha_t = 0.2$ and $\gamma^t = 0.75$.

4.6.5 Evaluating System Stability by Incorporating Delay and Constraints

Up to this point, the system evaluation has exclusively focused on optimizing the network utilities based on the transmission rates. The performance evaluation was extended to incorporate the delay since it is an important performance metric. Both the average throughput and the transmission delay were evaluated as the number of deployed IAB nodes between the traffic source and the donor node was increased. The end-to-end throughput was evaluated against the number of IAB nodes with a source rate of $R = 25$ Mbps and the result is shown in Fig. 4.14.

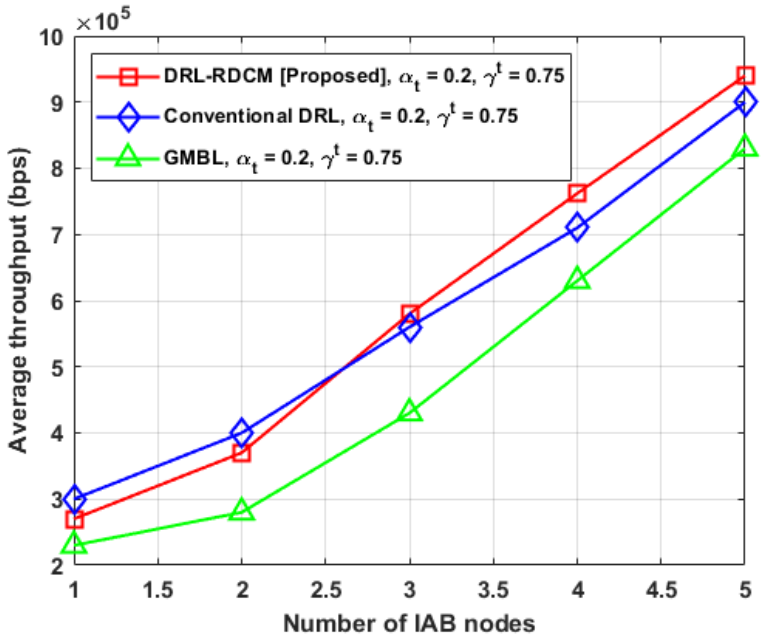

Figure 4.14. Average throughput vs. number of IAB nodes at $R = 25$ Mbps.

In Fig. 4.14, the throughput performance for the three algorithms is evaluated and an increasing trend is observed as the number of IAB nodes increases. The proposed DRL-RDCM strategy is observed to first lag the conventional DRL strategy, but as more spectrum becomes available in the network, it outperforms both baselines. This is because in as much as the RL strategy that is used in DRL is model-free, the use of the RDCM makes it behave more like a model-based strategy. This justifies the fact that the QoE improves as one moves away from the cell edge towards the center as the capacity becomes more guaranteed. The throughput performance is further evaluated with a source rate of $R = 50$ Mbps, and the results are shown in Fig. 4.15.

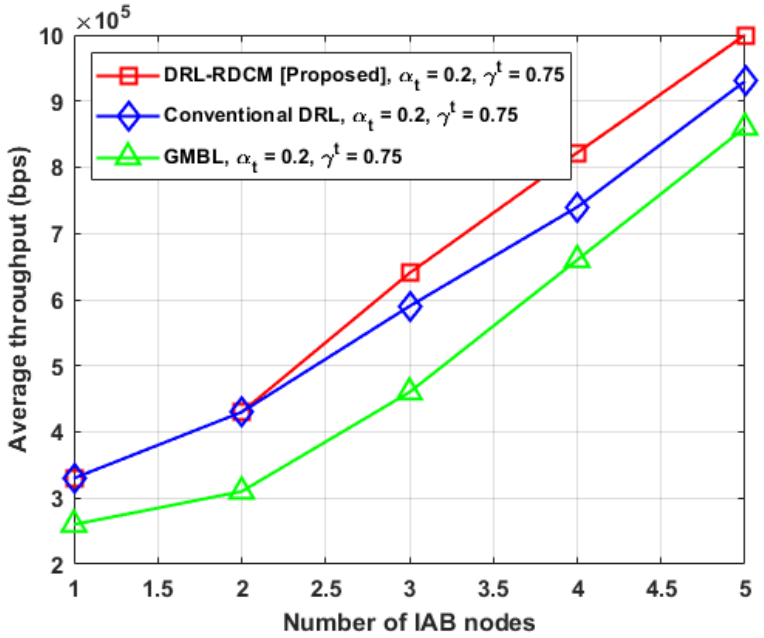

Figure 4.15. Average throughput vs. number of IAB nodes at $R = 50$ Mbps.

Compared to the case of Fig. 4.14, in Fig. 4.15, a congested scenario was created by doubling the source rate and an overall improved performance of the three algorithms is observed, with the proposed DRL-RDCM performing much better than both baselines. The result shows that with a high source rate, the performance of the proposed algorithm progressively improves as the number of IAB nodes increases, which indicates that the proposed algorithm benefits more from coverage enhancement than the other two baselines. This indicates the strength of the proposed solution in terms of maximizing the backhaul link throughput without compromising the access QoS.

The end-to-end latency for the configuration in Fig. 4.15 is shown in Fig. 4.16. As expected, the average end-to-end latency decreases as the number of IAB nodes increases, as shown in Fig. 4.16. This is because as the number of IAB nodes increases, it provides more transmission routes, which decreases the average end-to-end delay of the system.

From Fig. 4.16, it can be seen that the proposed DRL-RDCM provides significantly better end-to-end delay performance compared to the two baselines. This is particularly so for a small number of IAB nodes, and as the number of IAB nodes increases, the performance of the conventional DRL approach improves to closely follow that of the proposed scheme. On the other hand, the average latency for the GMBL approach remains constant at 70 ms as the number of IAB nodes increases to more than 3.

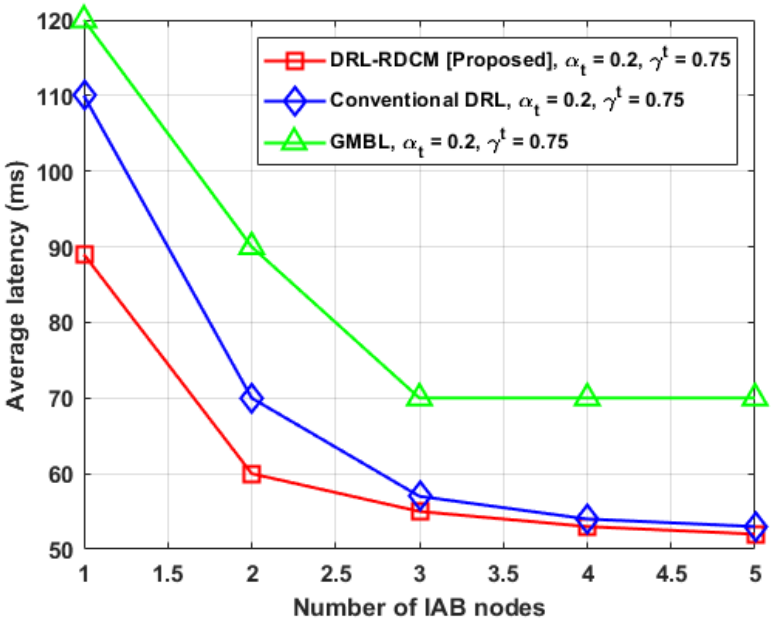

Figure 4.16. End-to-end latency vs. number of nodes for $R = 50$ Mbps.

This shows that the proposed algorithm adheres to reliable communication better than the other two baselines by better imposing the probabilistic delay constraint in (4.16). As expected, high throughput and lower transmission delays are achieved at the cost of high energy consumption and as such, the cost analysis of the proposed algorithm is considered in the following subsection.

4.6.6 Evaluation of the Cost Function

The evaluation of the power-delay trade-off as a function of the number of packets arriving at a node/link is the basic and underlying objective of wireless networks, and it cannot be overstated in the IAB networks. In this work, this trade-off was evaluated in terms of: (i) the time delay of the learning process, (ii) the variation of mean delay and overflow costs with packet arrival rate, (iii) packet holding costs and power points as functions of packet arrivals.

4.6.6.1 Time delay

The time complexity results in terms of populating the useful attributes for the cost function are shown in Table 4.3. From the table, it can be seen that populating the known attributes has higher time complexities than populating the unknown ones. This is because with the transition $s \rightarrow \tilde{s}$ the initial policy is not yet tuned to the specific traffic and channel conditions. The transition from $\tilde{s} \rightarrow s(t+1)$ means the policy has already been tuned, hence less transition time is required.

Table 4.3. Time delay evaluation of complexity.

Populated attribute	Attribute description	Elapsed time
Power cost		0.261440
Known transition probability function	From $s \rightarrow \tilde{s}$	0.573529
Buffer cost		0.447494
Unknown transition probability function	From $\tilde{s} \rightarrow s(t+1)$	0.013129
Unknown cost	From $\tilde{s} \rightarrow s(t+1)$	0.003936

4.6.6.2 Mean delay and overflow costs

In this subsection, the cost function is evaluated using the post-decision state learning scheme. The overflow cost is actually the cost of delay, which is crucial in agile network prioritization as it makes it possible for decision makers to consider the cost of keeping packets in the buffer beyond a single time slot. The performance of the system in terms of the mean delay and the mean overflow cost were evaluated as a function of increasing packet arrival rate and the results are shown in Fig. 4.17.

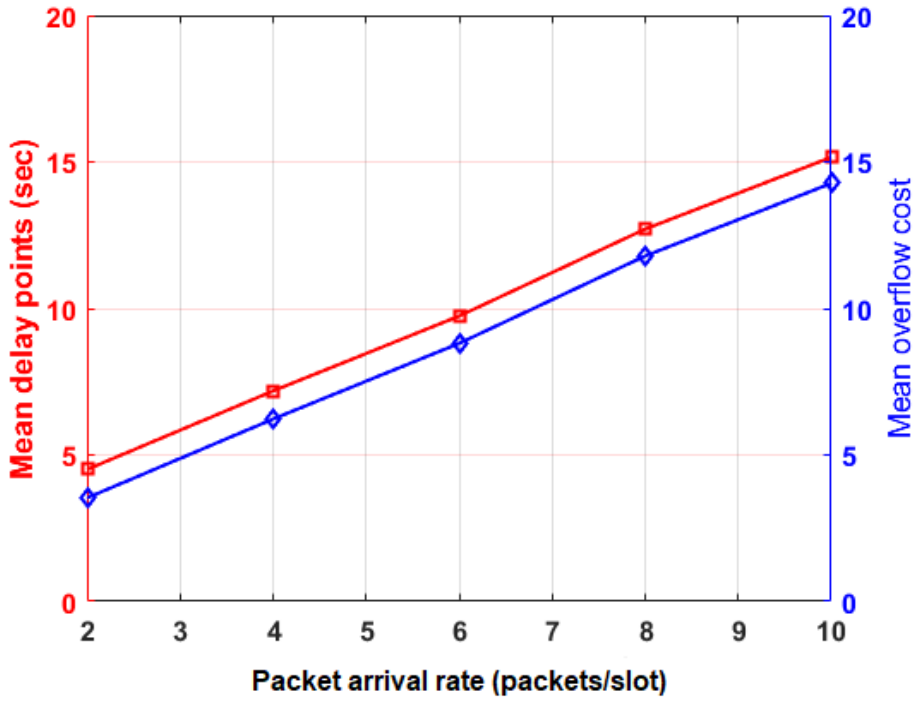

Figure 4.17. Mean delay points and mean buffer overflow costs vs packet arrival rates.

As shown in Fig. 4.17, both the mean delay and the mean buffer overflow cost increase linearly in a similar pattern as the packet arrival rate increases. In addition, the linear rate of increase of the two quantities is the same, which is about 1.25 per unit increase in the packet arrival rate. The increase in the delay points push the Lagrange multiplier to its maximum, which results in the cost function weighting the delays more and the power consumption less. This leads to an increase in the buffer overflow cost as the system begins penalizing every packet held in the buffer more than necessary.

4.6.6.3 Packet holding costs and mean power points

As the increasing delay forces the system to penalize any packet that is held in the buffer beyond a single time slot, the system should increase the transmission power to increase the transmission rate. This is expected to cause a decrease in the packet holding costs, however, at the cost of an increase in the power consumption. This kind of behaviour is verified in Fig. 4.18.

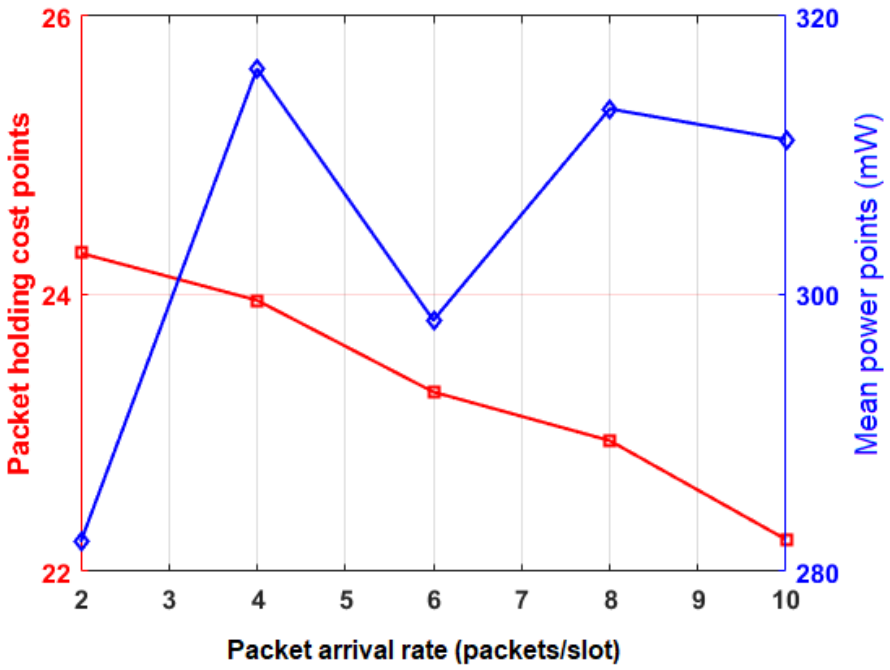

Figure 4.18. Packet holding cost points and mean power points vs packet arrival rate.

In Fig. 4.18, the system performance is evaluated in terms of the packet holding costs and power consumption as a function of an increasing packet arrival rates. It can be observed that as the packet arrival rate increases, the packet holding cost points decrease while the power consumption points increase. This is in line with the hypothesis that was made above, but there is a slight decrease in the

packet holding cost points when the packet arrival rate increases from 4 to 6, which is followed by a sudden dip in the power consumption. However, the power consumption suddenly increases when the packet arrival rate increases above 6 packets/slot. As the delay increases, the Lagrange multiplier is driven to its predefined maximum value, thereby leading to an increase in the power consumption, which then drives the packet holding costs down. This relationship indicates that in a system where the packet losses are penalized, the transmission delays result in a non-Markovian system behaviour.

4.7 CONCLUDING REMARKS

This chapter presented a backhaul adaptation scheme for the IAB networks using DRL with RDCM to address the challenges of backhaul availability and backhaul scalability. The proposed scheme is controlled by the load on the access side of the network as well as the number of traffic flows being routed to the MBS. The problem is formulated as a CMDP and solved using a dual decomposition approach due to the existence of explicit and implicit constraints. A DRL strategy that takes advantage of a RDCM was then proposed. The advantage of the RDCM for this problem is that it incorporates choice aversion from prospect theory and the reward is not the only factor affecting the learning rates, but also the punishment.

Optimal flow allocations in the network topology and the degree of aversion were derived using graph theory and RL. The Lagrange equivalent of the CMDP with respect to the policy and the punishment, and a cost function was also derived from the goodput distribution, and post-decision state learning was used to evaluate the power-delay cost trade-offs. The proposed algorithm was compared with the conventional DRL, that is, without the RDCM and the GMBL algorithms, where it showed better performance over the two baselines. The near optimal delay performance of the system is achieved before the optimal power consumption since the power consumption can only be learned after satisfying the packet holding cost constraint.

In routing problems it is beneficial to incorporate the concept of prospect theory concept that describes how decision makers choose between different prospects and how they estimate the perceived likelihood of each of these options. This is RL with foresight. In this work, the optimization objective was defined and the valuation function that was used was induced by an acceptance level through the RDCM for value functions that were specified in the prospect of route choices. These route choices have associated costs which are aimed at assisting in conflating observations in terms of alternatives leading to different

choices, as well as rewards resulting from different choices of value function parameters from the characteristics of the CMDP. However, computational complexity is the main obstacle observed in the application of the proposed algorithm. It should be noted that the complexity of the family of RL strategies comes with the repeated learning updates towards reaching the reward. The proposed DRL-RDCM differs from the conventional DRL in terms of task complexity measures.

CHAPTER 5 CONCLUSION

5.1 CONCLUDING REMARKS

This research work aimed to solve some of the challenges in IAB networks by providing flexible solutions with relatively lower complexity and that adhere to the 3GPP specifications. Consequently, it was proposed to look into providing solutions that combine traditional optimization techniques with machine learning strategies.

The literature review in chapter 2 highlighted the need for including machine learning strategies, in particular RL, in IAB network models. It was concluded that the dynamic resource management and backhaul routing problems in IAB networks had not been thoroughly investigated in literature. The research work then aimed to provide an end-to-end solution by first developing a context-aware user association and access resource management scheme, and then developing a DRL-based context-switching backhaul route establishment technique for multi-hop IAB networks.

5.2 RESULTS ACHIEVED

Chapter 3 presented a DRL-based model for context-aware resource management in mm-wave IAB networks. A DRL-based solution for the formulated optimization problem was developed and presented. The solution combined aspects of the RL strategy and the DL technique, where an SBS is taken as the DRL agent, to provide a reward maximization algorithm for determining the best possible association whilst satisfying the QoS requirements of associated UEs. The DRL algorithm was proposed for both individual learning and nearest neighbour cooperative learning. The proposed algorithms were shown to have less computational complexity for action-selection compared to the baseline algorithm. The individual learning scheme also had less complexity for the learning update, whereas the cooperative learning scheme had the same complexity as the baseline algorithm.

Simulation results showed that the congestion performance of both the individual learning and cooperative learning schemes was the same as that of the baseline algorithm, albeit the proposed schemes having reduced complexity. For the throughput performance and the QoE performance, the proposed learning schemes provided significantly better performance compared to the baseline for all considered cases of varying learning rate and number of associated UEs. However, it was seen that the cooperative algorithm is more sensitive to an increase in the learning rate and an increasing number of UEs compared to the individual learning scheme.

In chapter 4, another DRL-based model for mm-wave IAB networks was presented, where investigation of optimal backhaul routes was carried out. The optimization problem was formulated as a CMDP for which the DRL-RDCM solution method was proposed. In this proposed scheme, the mechanism for adjusting the reward value is flexible and it is thus well-suited for next generation wireless network routing applications. In addition to the DRL-RDCM scheme, cost analysis of the backhaul routing is performed by applying a post-decision state-based dynamic programming technique to compute the cost function.

For performance evaluation, the DRL-RDCM scheme was compared with conventional DRL and the GMBL technique. The simulation results showed that the proposed DRL-RDCM outperforms the conventional DRL and the GMBL approaches in terms of cumulative throughput performance, for all considered cases of various learning rates. The results also showed that for the proposed DRL-RDCM scheme, the oscillations caused by a large learning rate are reduced when the value of the learning rate also reduces. However, this was achieved at the cost of an increased time complexity. The performance evaluation was extended to evaluate how the average end-to-end throughput varies with increasing number of IAB nodes for different source rates. For the case of lower source rate, the proposed DRL-RDCM strategy was observed to be outperformed by the conventional DRL strategy when there are very few IAB nodes, but as more nodes became available, the proposed strategy outperformed both baselines. As the source rate was increased, the performance of the proposed strategy was shown to match that of DRL for few number of IAB nodes, and when the nodes increased, the proposed strategy again outperformed both baselines.

The cost function was evaluated to provide an insight into the power-delay trade-off depending on the number of packets arriving at a node/link. From the time-delay analysis of the learning process, it was seen that populating the known attributes has larger time complexities compared to populating the

unknown ones. It was also seen that both the mean delay and the mean buffer overflow cost increase linearly at the same rate as the packet arrival rate increases. Evaluation of packet holding costs and power consumption as a function of increasing packet arrival rates showed that the packet holding cost points decrease, while the power consumption points increase, as the packet arrival rate increases. This was as expected since increasing the delay forces the system to penalize any packet held in the buffer beyond a single time slot. Thus, the system responds by increasing the transmission power in order to increase the transmission rate and reduce the delay.

5.3 RECOMMENDATIONS FOR FUTURE WORK

Based on the work presented in chapter 3, future research work can look into improving (i) the reliability of the access network, (ii) the computation capability at the wireless network edge, and (iii) traffic prioritization. This is based on the assumption that with the proliferation of critical IoT applications, latency-sensitive traffic would be processed at the network edge. As such, computation offloading strategies for IAB-enabled MEC networks should be investigated in order to understand the interaction between MEC and mm-wave IAB networks. This MEC-IAB paradigm should be well equipped with edge computing capabilities in order to deal with the envisaged tantalizing amounts of real-time traffic in future networks.

Considering the work of chapter 4, whether the system learns better by reward or by punishment, as well as to what extent does the reward and/or the punishment influence the learning rate of the system have not been considered in RL-based IAB research solutions. However, it can be postulated, as an assumption, that the influence of the reward and punishment on the learning rate is subject to various complex mechanisms of the actions. For instance, the reward and punishment appear to be processed in different ways and the risk/loss aversion could also have an influence on the reward and punishment as well as algorithmic sensitivity to both the reward and punishment. It has been noted that few studies have investigated the influence of reward and punishment on learning rates, although this question has been addressed since the beginning of psychological research and is still unresolved in many aspects. Further research is not only required in the context of long-term effects (retention) of reward and punishment, but also whether reward or punishment lead to a higher learning rates and if at all and under what conditions reward and punishment lead to higher learning rates.

Overally, there is need for development of a scalable wireless backhaul strategy that is capable of dynamically re-configuring itself in the event of an outage. The scalable IAB network should not only

be limited to 5G mm-wave operation but it should also allow a straightforward and incremental upgrade pathway to improving capacity over time. This is would enable integration with future-generation wireless networks.

REFERENCES

[1] M. Belhabib, “Over-The-Air (OTA) Testing 5G Wirelesses Communications Systems,” *International Journal of Future Computer and Communication*, vol. 9, no. 1, pp. 1–4, Mar. 2020.

[2] Cisco, “Cisco Annual Internet Report (2018-2023),” San Jose, CA, USA, Tech. Rep., 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

[3] Ericsson, “Ericsson Mobility Report,” Stockholm, Sweden, Tech. Rep., 2019. [Online]. Available: https://www.ericsson.com/assets/local/mobility-report/documents/2019/ericsson-mobility-report-june-2019.pdf

[4] Keysight Technologies. "The ABC’s of 5G New Radio Standards". [Online]. Available: https://connectlp.keysight.com/AMO_5GNR-ABCs-AppBR#: (accessed: 01 Sep. 2020.)

[5] Ericsson. "Carrier Aggregation in 5G: A must to deploy better 5G". [Online]. Available: https://www.ericsson.com/en/networks/offerings/5g/carrier-aggregation (accessed: 11 Nov. 2020).

[6] O. Teyeb, A. Muhammad, G. Mildh, and E. D. F. B. B. Makki, “Integrated Access Backhauled Networks,” in *IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–5.

[7] 3GPP, “Study on Integrated Access and Backhaul (Release 15),” Sophia Antipolis, France, Tech. Rep. TR 38.874, 2017.

[8] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated Access and Backhaul in 5G mmWave Networks: Potential and Challenges," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 62–68, Mar. 2020.

[9] 3GPP, "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G," Sophia Antipolis, France, Tech. Rep. TR 121 916, 2021.

[10] I. F. Akyildiz, D. M.Gutierrez-Estevez, and E. ChavarriaReyes, "The Evolution to 4G Cellular Systems: LTE-Advanced," *Physical Communication*, vol. 3, no. 4, pp. 217–244, Dec. 2010.

[11] M. N. Islam, S. Subramanian, and A. Sampath, "Integrated access backhaul in millimeter wave networks," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, 2017, pp. 1–6.

[12] S. Singh, M. N. Kulkarni, A. Ghosh, and J. G. Andrews, "Tractable Model for Rate in Self-Backhauled Millimeter Wave Cellular Networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 10, pp. 2196–2211, Oct. 2015.

[13] 3GPP, "Technical Specification Group Services and System Aspects (Release 15)," Sophia Antipolis, France, Tech. Rep. TR 21.915, 2019.

[14] C. Madapatha, B. Makki, C. Fang, O. Teyeb, E. Dahlman, M.-S. Alouini, and T. Svensson, "On Integrated Access and Backhaul Networks: Current Status and Potentials," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 1374–1389, Sept. 2020.

[15] E. Dahlman, S. Parkvall, and J. Sköld, *4G LTE/LTE-Advanced for Mobile Broadband*. Academic Press, 2011.

[16] M. N. Kulkarni, J. G. Andrews, and A. Ghosh, "Performance of Dynamic and Static TDD in Self-Backhauled Millimeter Wave Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6460–6478, Oct. 2017.

[17] J. G. Andrews, S. Singh, Q. Ye, X. Lin, and H. S. Dhillon, "An Overview of Load Balancing in HetNets: Old Myths and Open Problems," *IEEE Wireless Communications*, vol. 21, no. 2, pp. 18–25, Apr. 2014.

[18] B. Bangerter, S. Talwar, R. Arefi, and K. Stewart, "Networks and Devices for the 5G Era," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 90–96, Feb. 2014.

[19] D. Cohen, "What You Need to Know About 5G Wireless Backhaul," Ceragon, Tech. Rep., Feb. 2016. [Online]. Available: https://www.ceragon.com/hubfs/Content_for_campaigns_/Ceragon_-_What_you_need_to_know_about_5G_Wireless_Backhaul.pdf

[20] R. Cole, Y. Dodis, and T. Roughgarden, "Bottleneck Links, Variable Demand, and the Tragedy of the Commons," *Networks*, vol. 60, no. 3, pp. 194–203, Oct. 2012.

[21] D. Li, G. Zhang, Y. Xu, H. Zhao, and F. Tian, "Integrating Distributed Grids With Green Cellular Backhaul: From Competition to Cooperation," *IEEE Access*, vol. 6, pp. 75 798–75 812, Nov. 2018.

[22] H. Tullberg, M. Fallgren, K. Kusume, and A. Höglund, *5G Use Cases and System Concept*. Cambridge University Press, 2016, ch. 2.

[23] A. Alexiou, "The road to 5g - visions and challenges," in *5G Wireless Technologies*, A. Othman, Ed. London, UK: IET, 2018, ch. 1, pp. 1–15.

[24] V. Sridhar, T. Casey, and H. Hämmäinen, "Flexible Spectrum Management for Mobile Broadband Services: How Does it Vary Across Advanced and Emerging Markets?" *Telecommunications Policy*, vol. 37, no. 2–3, pp. 178–191, Mar.-Apr. 2013.

[25] K. I. Pedersen, F. Frederiksen, C. Rosa, H. Nguyen, L. G. U. Garcia, and Y. Wang, "Carrier Aggregation for LTE-Advanced: Functionality and Performance Aspects," *IEEE Communications Magazine*, vol. 49, no. 6, pp. 89–95, Jun. 2011.

[26] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5G on the Horizon: Key Challenges for the Radio-Access Network," *IEEE Vehicular Technology Magazine*, vol. 8, no. 3, pp. 47–53, Sept. 2013.

[27] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell Networks: A Survey," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 59–67, Sept. 2008.

[28] AALTO, "Study on Small Cells and Dense Cellular Networks Regulatory Issues," Global 5G, Tech. Rep., 2018. Accessed: 03 Nov. 2019. [Online]. Available: https://global5g.eu/news/global5gorg-study-small-cells-and-dense-cellular-networks-regulatory-issues

[29] Small Cell Forum, "Small Cells: What's the Big Idea?" Tech. Rep., 2012. Accessed: 13 Oct. 2020. [Online]. Available: http://www.scf.io/en/documents/030_-_Small_cells_big_ideas.php

[30] GSMA, "5G, the Internet of Things (IoT) and Wearable Devices: What do the new uses of wireless technologies mean for radio frequency exposure?" London, UK, Tech. Rep., 2017. Accessed: 03 Oct. 2020. [Online]. Available: https://www.gsma.com/latinamerica/resources/5g-the-iot-and-wearable-devices-what-do-the-new-uses-of-wireless-technologies-mean-for-radio-frequency-exposure/

[31] ICNIRP, "ICNIRP Guidelines for Limiting Exposure to Time-Varying Electric, Magnetic and Electromagnetic Fields (up to 300 GHz)," *Health Physics*, vol. 118, no. 5, pp. 483–524, Mar. 2020.

[32] Z. Li, F. S. Moya, G. Fodor, J. M. B. D. S. Jr, and K. Koufos, "Device-to-device (D2D) communications," in *5G Mobile and Wireless Communications Technology*, A. Osseiran, J. F. Monserrat, and P. Marsch, Eds. UK: Cambridge University Press, 2016, ch. 5, pp. 107–136.

[33] N. K. Pratas and P. Popovski, "Wireless device-to-device (D2D) links for machine-to-machine (M2M) communication," in *5G Wireless Technologies*, A. Othman, Ed. London, UK: IET, 2018, ch. 6, pp. 193–221.

[34] M. Mustonen, M. Matinmikko, D. Roberson, and S. Yrjölä, "Evaluation of Recent Spectrum Sharing Models from the Regulatory Point of View," in *Proceedings of the 1st International Conference on 5G for Ubiquitous Connectivity*, 2014, pp. 11–16.

[35] C.-L. I, J. Huang, C. Bai, R. Duan, and R. Ren, "Wireless networks virtualization," in *5G Wireless Technologies*, A. Othman, Ed. London, UK: IET, 2018, ch. 10, pp. 341–361.

[36] J. Heide, F. H. P. Fitzek, M. V. Pedersen, and M. Katz, "Green mobile clouds: Network coding and user cooperation for improved energy efficiency," in *IEEE 1st International Conference on Cloud Networking (CLOUDNET)*, 2012, pp. 111–118.

[37] Open Networking Foundation, "Software-defined networking: The new norm for networks," Palo Alto, CA, USA, Tech. Rep., 2012. Accessed: 01 Oct. 2020. [Online]. Available: www.opennetworking.org/images/stories/downloads/whitepapers/ wp-sdn-newnorm.pdf

[38] H. Droste, I. L. D. Silva, P. Rost, and M. Boldi, "The 5G architecture," in *5G Mobile and Wireless Communications Technology*, A. Osseiran, J. F. Monserrat, and P. Marsch, Eds. UK: Cambridge University Press, 2016, ch. 3, pp. 50–76.

[39] AT&T et al., "Network Function Virtualization: An Introduction, Benefits, Enablers, Challenges & Call for Action," ETSI, Tech. Rep., 2012. Accessed: 03 Oct. 2020. [Online]. Available: http://portal.etsi.org/NFV/NFV_White_Paper.pdf

[40] 5G Americas, "Advanced Antenna Systems for 5G," Tech. Rep., 2019. Accessed: 03 Oct. 2020. [Online]. Available: https://www.5gamericas.org/wp-content/uploads/2019/08/5G-Americas_Advanced-Antenna-Systems-for-5G-White-Paper.pdf

[41] 5G PPP, "View on 5G Architecture," Tech. Rep., 2019. Accessed: 13 Aug. 2020. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2020/02/5G-PPP-5G-Architecture-White-Paper_final.pdf

[42] 5G Americas, "Network slicing for 5G networks and services," Tech. Rep., 2019. Accessed: 14 Aug. 2020. [Online]. Available: https://www.5gamericas.org/network-slicing-for-5g-networks-

services/

[43] Rysavy, "Global 5G: Implications of a Transformational Technology," 5G Americas, Bellevue, WA, USA, Tech. Rep., 2019. [Online]. Available: https://www.5gamericas.org/global-5g-implications-of-a-transformational-technology

[44] GSMA, "Mobile backhaul options: Spectrum analysis and recommendations," Tech. Rep., 2018. Accessed: 03 Jun. 2020. [Online]. Available: https://www.gsma.com/spectrum/resources/mobile-backhaul-options/

[45] R. Ravindran, P. Suthar, A. Chakraborti, S. O. Amin, A. Azgin, and G. Wang, "Deploying ICN in 3GPP's 5G NextGen Core Architecture," in *IEEE 5G World Forum (5GWF)*, 2018, pp. 1–6.

[46] Qualcomm. Just in: 3GPP completes 5G NR Release 17. [Online]. Available: https://www.qualcomm.com/news/onq/2022/03/just-3gpp-completes-5g-nr-release-17 (accessed: 18 Feb. 2023).

[47] Small Cell Forum, "Small cell siting challenges and recommendations," 5G Americas, Tech. Rep., Aug. 2018.

[48] M. Coldrey, J.-E. Berg, L. Manholm, C. Larsson, and J. Hansryd, "Non-Line-of-Sight Small Cell Backhauling Using Microwave Technology," *IEEE Communications Magazine*, vol. 51, no. 9, pp. 78–84, Sept. 2013.

[49] S. Jin, J. Liu, X. Leng, and G. Shen, "Self-Backhaul Method and Apparatus in Wireless Communication Networks," U.S. Patent 0 110 005 A1, May, 2007.

[50] R. Taori and A. Sridharan, "Point-to-multipoint in-band mmwave backhaul for 5G networks," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 195–201, Jan. 2015.

[51] W. Feng, Y. Li, D. Jin, L. Su, and S. Chen, "Millimetre-Wave Backhaul for 5G Networks: Challenges and Solutions," *Sensors*, vol. 16, no. 892, Jun. 2016.

[52] S. Nie, G. R. MacCartney, S. Sun, and T. S. Rappaport, "28 GHz and 73 GHz signal outage study for millimetre wave cellular and backhaul communications," in *Proceedings of 2014 IEEE International Conference on Communications (ICC)*, 2014, pp. 4856–4861.

[53] Y. Zhu, Z. Zhang, Z. Marzi, C. Nelson, U. Madhow, B. Y. Zhao, and H. Zheng, "Demystifying 60 GHz outdoor picocells," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, 2014, pp. 5–16.

[54] T. S. Rappaport, J. N. Murdock, and F. Gutierrez, "State of the Art in 60-GHz Integrated Circuits and Systems for Wireless Communications," *Proceedings of the IEEE*, vol. 99, no. 8, pp. 1390–1436, Aug. 2011.

[55] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, and T. S. Rappaport, "Millimeter Wave Channel Modeling and Cellular Capacity Evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, June 2014.

[56] M. N. Islam, N. Abedini, G. Hampel, S. Subramanian, and J. Li, "Investigation of performance in integrated access and backhaul networks," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 597–603.

[57] T. Tian, Y. Dou, G. Ren, L. Gu, J. Chen, Y. Cui, T. Takada, M. Iwabuchi, J. Tsuboi, and Y. Kishiyama, "Field Trial on Millimeter Wave Integrated Access and Backhaul," in *IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, 2019, pp. 1–5.

[58] Y. Zhang, M. A. Kishk, and M.-S. Alouini, "A Survey on Integrated Access and Backhaul Networks," *Frontiers in Communications and Network*, vol. 2, pp. 1–24, Jun. 2021.

[59] A. Betzler, D. Camps-Mur, E. Garcia-Villegas, I. Demirkol, and J. J. Aleixendri, "SODALITE: SDN Wireless Backhauling for Dense 4G/5G Small Cell Networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1709–1723, Dec. 2019.

[60] X. Ge, H. Cheng, M. Guizani, and T. Han, "5G wireless backhaul networks: Challenges and research advances," *IEEE Network*, vol. 28, no. 6, pp. 6–11, Nov.-Dec. 2014.

[61] C. Saha and H. S. Dhillon, "Millimeter Wave Integrated Access and Backhaul in 5G: Performance Analysis and Design Insights," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 12, pp. 2669–2684, Dec. 2019.

[62] M. Haenggi, *Stochastic Geometry for Wireless Networks*. Cambridge University Press, 2012.

[63] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic Geometry and Random Graphs for the Analysis and Design of Wireless Networks," *IEEE Journal on Selected on Areas in Communications*, vol. 27, no. 7, pp. 1029–1046, Sept. 2009.

[64] F. Baccelli and B. Błaszczyszyn, "Stochastic Geometry and Wireless Networks: Volume II Applications," *Foundations and Trends in Networking*, vol. 4, no. 1–2, pp. 1–312, Jan. 2010.

[65] V. Chandrasekhar and J. G. Andrews, "Spectrum allocation in tiered cellular networks," *IEEE Transactions on Communications*, vol. 57, no. 10, pp. 3059–3068, Oct. 2009.

[66] B. Zhuang, D. Guo, E. Wei, and M. L. Honig, "Large-Scale Spectrum Allocation for Cellular Networks via Sparse Optimization," *IEEE Transactions on Signal Processing*, vol. 66, no. 20, pp. 5470–5483, Oct. 2018.

[67] C. Saha, M. Afshang, and H. S. Dhillon, "Bandwidth Partitioning and Downlink Analysis in Millimeter Wave Integrated Access and Backhaul for 5G," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8195–8210, Dec. 2018.

[68] Y. Li, J. Luo, R. A. Stirling-Gallacher, and G. Caire, "Integrated Access and Backhaul Optimization for Millimeter Wave Heterogeneous Networks," *ArXiv:1901.04959*, pp. 1–30, Jan. 2019.

[69] J. Y. Lai, W.-H. Wu, and Y. T. Su, "Resource Allocation and Node Placement in Multi-Hop Heterogeneous Integrated-Access-and-Backhaul Networks," *IEEE Access*, vol. 8, pp. 122 937–122 958, Jul. 2020.

[70] Y. Liu, A. Tang, and X. Wang, "Joint Incentive and Resource Allocation Design for User Provided Network Under 5G Integrated Access and Backhaul Networks," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 673–685, Apr.–Jun. 2019.

[71] R. Sofia and P. Mendes, "User-provided networks: Consumer as provider," *IEEE Communications Magazine*, vol. 46, no. 12, pp. 86–91, Dec. 2008.

[72] Y. Niu, C. Gao, Y. Li, L. Su, and A. V. Vasilakos, "Exploiting Device-to-Device Communications in Joint Scheduling of Access and Backhaul for mmWave Small Cells," *IEEE Journal in Selected Areas in Communication*, vol. 33, no. 10, pp. 2052–2070, Oct. 2015.

[73] M. Polese, M. Giordani, A. Roy, D. Castor, and M. Zorzi, "Distributed Path Selection Strategies for Integrated Access and Backhaul at mmWaves," in *IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–7.

[74] H. Elsawy, H. Dahrouj, T. Y. Alnaffouri, and M. S. Alouini, "Virtualized Cognitive Network Architecture for 5G Cellular Networks," *IEEE Communications Magazine*, vol. 53, no. 7, pp. 78–85, Jul. 2015.

[75] B. N. Schilit and M. Theimer, "Disseminating active map information to mobile hosts," *IEEE Network*, vol. 8, no. 5, pp. 22–32, Sept.–Oct. 1994.

[76] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a Better Understanding of Context and Context-Awareness," in *1st International Symposium on Handheld and Ubiquitous Computing (HUC)*, 1999, pp. 304–307.

[77] G. W. Musumba and H. O. Nyongesa, "Context Awareness in Mobile Computing: A Review," *IEEE International Journal of Machine Learning and Applications*, vol. 2, no. 1, pp. 1–10, May 2013.

[78] J. S. Perez, S. K. Jayaweera, and S. Lane, "Machine Learning Aided Cognitive RAT Selection for 5G Heterogeneous Networks," in *Black Sea Conference on Communications and Networking (BlackSeaCom)*, 2017.

[79] J. Mitola and G. Q. Maguire, "Cognitive Radio: Making Software Radios More Personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, Aug. 1999.

[80] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Selected Areas in Communication*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[81] I. F. Akyildiz, W.-Y. Lee, M. C.Vuran, and S. Mohanty, "NeXt Generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey," *Computer Networks*, vol. 50, no. 3, pp. 2127–2159, Sept. 2006.

[82] M. Zorzi, A. Zanella, A. Testolin, M. D. F. D. Grazia, and M. Zorzi, "Cognition-Based Networks: A new perspective on Network Optimization using Learning and Distributed Intelligence," *IEEE Access*, vol. 3, pp. 1512–1530, Sept. 2015.

[83] M. Bkassiny, Y. Li, and S. K. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1136–1160, Third Quarter 2013.

[84] B. T. Maharaj and B. S. Awoyemi, *Developments in cognitive radio networks: future directions for beyond 5G*. Springer Nature, 2021.

[85] R. S. Michalski, "Learning and cognition," in *World Conference on the Fundamentals of Artificial Intelligence (WOCFAI '95)*, 1995, pp. 507–510.

[86] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. DaSilva, "Learning and Reasoning in Cognitive Radio Networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1761–1777, Fourth Quarter 2013.

[87] E. Alpaydin, *Introduction to Machine Learning*, 3rd ed. MIT Press, 2014.

[88] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine Learning Paradigms for Next-Generation Wireless Networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 98–105, Apr. 2017.

[89] E. M. Joo and Y. Zhou, Eds., *Theory and Novel Applications of Machine Learning*. InTechOpen, 2009.

[90] X. Zhou, M. Sun, G. Y. Li, and B.-H. Juang, "Intelligent Wireless Communications Enabled by Cognitive Radio and Machine Learning," *China Communications*, vol. 15, no. 12, pp. 16–48, Dec. 2018.

[91] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. Springer, 2009.

[92] O. Simeone, *A Brief Introduction to Machine Learning for Engineers*. Now, 2018.

[93] M. Buhmann, "Radial basis function," *Scholarpedia*, vol. 5, no. 5, p. 9837, 2010.

[94] P. Garg, D. Neider, P. Madhusudan, and D. Roth, "Learning Invariants using Decision Trees and Implication Counterexamples," *ACM SIGPLAN Notices*, vol. 15, no. 1, pp. 499–512, Jan. 2016.

[95] B. K. Donohoo, C. Ohlsen, S. Pasricha, Y. Xiang, and C. Anderson, "Context-Aware Energy Enhancements for Smart Mobile Devices," *IEEE Transactions on Mobile Computing*, vol. 13, no. 8, pp. 1720–1732, Aug. 2014.

[96] V. Feng and S. Y. Chang, "Determination of Wireless Networks Parameters through Parallel Hierarchical Support Vector Machines," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 3, pp. 505–512, Mar. 2012.

[97] C.-K. Wen, S. Jin, K.-K. Wong, J.-C. Chen, and P. Ting, "Channel Estimation for Massive MIMO Using Gaussian-Mixture Bayesian Learning," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1356–1368, Mar. 2015.

[98] C. Ghosh, C. Cordeiro, D. P. Agrawal, and M. B. Rao, "Markov chain existence and hidden Markov models in spectrum sensing," in *IEEE International Conference on Pervasive Computing and Communications*, 2009, pp. 1–6.

[99] T. Rondeau, C. Rieser, T. Gallagher, and C. Bostian, "Online Modeling of Wireless Channels with Hidden Markov Models and Channel Impulse Responses for Cognitive Radios," in *IEEE MTT-S International Microwave Symposium Digest*, 2004, pp. 739–742.

[100] Y. Yao, Z. Feng, W. Li, and Y. Qian, "Dynamic Spectrum Access with QoS Guarantee for Wireless Networks: A Markov Approach," in *IEEE Global Telecommunications Conference (GLOBECOM)*, 2010, pp. 1–5.

[101] Z.-J. Zhao, Z. Shilian, C.-Y. Xu, and X.-Z. Kong, "Discrete channel modelling based on genetic algorithm and simulated annealing for training hidden Markov model," *Chinese Physics*, vol. 16, no. 6, pp. 1619–1623, June 2007.

[102] K. W. Choi and E. Hossain, "Estimation of Primary User Parameters in Cognitive Radio Systems via Hidden Markov Model," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 782–795, Feb. 2013.

[103] A. Assra, J. Yang, and B. Champagne, "An EM Approach for Cooperative Spectrum Sensing in Multiantenna CR Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 3, pp. 1229–1243, Mar. 2016.

[104] C.-K. Yu, K.-C. Chen, and S.-M. Cheng, "Cognitive Radio Network Tomography," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1980–1997, May 2010.

[105] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. The MIT Press, 1998.

[106] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An Introduction to Deep Reinforcement Learning," *Foundations and Trends in Machine Learning*, vol. 11, no. 3–4, pp. 219–354, 2018.

[107] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018.

[108] R. McFarlane, "A Survey of Exploration Strategies in Reinforcement Learning," 2003, pp. 1–10.

[109] S. B. Thrun, *The role of exploration in learning control*. Van Nostrand Reinhold, 1992.

[110] B. H. Abed-Alguni, "Action-Selection Method for Reinforcement Learning Based on Cuckoo Search Algorithm," *Arab Journal of Science and Engineering*, vol. 43, no. 1, pp. 6771–6785, Oct 2018.

[111] R. W. Dearden, N. Friedman, and S. J. Russell, "Bayesian Q-learning," in *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence (IAAI)*, 1998, pp. 761–768.

[112] J. Wyatt, "Exploration and Inference in Learning from Reinforcement," Ph.D. thesis, Department of Artificial Intelligence, University of Edinburgh, 1997.

[113] M. Sugiyama, *Statistical Reinforcement Learning: Modern Machine Learning Approaches*, 1st ed. CRC Press, 2015.

[114] M. C. Hlophe and B. T. Maharaj, "AI Meets CRNs: A Prospective Review on the Application of Deep Architectures in Spectrum Management," *IEEE Access*, vol. 9, pp. 2169–3536, Aug. 2021.

[115] M. van Otterlo and M. A. Wiering, *Reinforcement Learning and Markov Decision Processes*. Springer, 2012, ch. 1, pp. 3–42.

[116] Z. Wei, J. Xu, Y. Lan, J. Guo, and X. Cheng, "Reinforcement Learning to Rank with Markov Decision Process," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 945–948.

[117] C. Boutilier, "Planning, learning and coordination in multiagent decision processes," in *Proceedings of the 6th Conference on Theoretical aspects rationality and knowledge (TARK)*, 1996, pp. 195–210.

[118] E. Altman, "Constrained Markov decision processes with total cost criteria: Occupation measures and primal LP," *Mathematical Methods of Operations Research*, vol. 43, pp. 45–72, Feb. 1996.

[119] O. Onireti, A. Zoha, J. Moysen, A. Imran, L. Giupponi, M. A. Imran, and A. Abu-Dayya, "Cell Outage Management Framework for Dense Heterogeneous Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 4, pp. 2097–2114, Apr. 2016.

[120] T. Jiang, P. D. Mitchell, and D. Grace, "Efficient Exploration in Reinforcement Learning-based Cognitive Radio Spectrum Sharing," *IET Communications*, vol. 5, no. 10, pp. 1309–1317, Jul. 2011.

[121] S. B. Thrun, "Efficient Exploration in Reinforcement Learning," Carnegie Mellon University, Pittsburgh, PA, USA, Tech. Rep., 1992. Accessed 03 Oct. 2020. [Online]. Available: https://www.ri.cmu.edu/pub_files/pub1/thrun_sebastian_1992_1/thrun_sebastian_1992_1.pdf

[122] A. Aprem, C. R. Murthy, and N. B. Mehta, "Transmit Power Control Policies for Energy Harvesting Sensors with Retransmissions," *IEEE Journal on Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 895–906, Oct. 2013.

[123] B. J. Chang and W. T. Chang, "Cost-Reward-based Carrier Aggregation with Differentiating Network Slicing for Optimizing Radio RB Allocation in 5G New Radio Network," in *Proceedings of the IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2019, pp. 813–820.

[124] B. F. Lo and I. F. Akyildiz, "Reinforcement Learning-based Cooperative Sensing in Cognitive Radio Ad hoc Networks," in *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2010, pp. 2244–2249.

[125] S. Maghsudi and S. Stańczak, "Channel Selection for Network-Assisted D2D Communication via No-Regret Bandit Learning with Calibrated Forecasting," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1309–1322, Mar. 2015.

[126] S. S. Mwanje, L. C. Schmelz, and A. Mitschele-Thiel, "Cognitive CellularNetworks: A Q-Learning Framework for Self-Organizing Networks," *IEEE Transactions on Network and Service Management*, vol. 13, no. 1, Mar. 2016.

[127] P. Semov, P. Koleva, K. Tonchev, V. Poulkov, and A. Mihovska, "Autonomous Learning Model for Achieving Multi Cell Load Balancing Capabilities in HetNet," in *IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, 2016, pp. 1–5.

[128] C. Wu, K. R. Chowdhury, M. D. Felice, and W. Meleis, "Spectrum Management of Cognitive Radio using Multi-agent Reinforcement Learning," in *9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2010, pp. 1705–1712.

[129] H. Jiang, H. He, L. Liu, and Y. Yi, "Q-Learning for Non-Cooperative Channel Access Game of Cognitive Radio Networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, 2018, pp. 1–7.

[130] X. Yang, P. Yu, L. Feng, F. Zhou, W. Li, and X. Qiu, "A Deep Reinforcement Learning based Mechanism for Cell Outage Compensation in 5G UDN," in *IFIP/IEEE Symposium on Integrated Network and Service Management*, 2019, pp. 1–6.

[131] G. Alnwaimi, S. Vahid, and K. Moessner, "Dynamic Heterogeneous Learning Games for Opportunistic Access in LTE-Based Macro/Femtocell Deployments," *IEEE Transactions on Wireless Communications*, vol. 14, no. 4, pp. 2294–2309, Apr. 2015.

[132] C. Wu, C. Wang, J. Sheng, and Y. Wang, "Cooperative Learning for Spectrum Management in Railway Cognitive Radio Network," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5809–5819, June 2019.

[133] C. Wang, Q. Wu, Z. Tang, J. Sheng, C. Wu, and Y. Wang, "Spectrum Management in High-Speed Railway Cooperative Cognitive Radio Network Based on Multi-agent Reinforcement Learning," in *International Wireless Communications and Mobile Computing (IWCMC)*, 2020, pp. 702–507.

[134] A. Das, S. C. Ghosh, N. Das, and A. D. Barman, "Q-Learning Based Co-Operative Spectrum Mobility in Cognitive Radio Networks," in *IEEE 42nd Conference on Local Computer Networks (LCN)*, 2017, pp. 502–505.

[135] A. Galindo-Serrano and L. Giupponi, "Distributed Q-learning for Aggregated Interference Control in Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1823–1834, May 2010.

[136] J. He, J. Peng, F. Jiang, G. Qin, and W. Liu, "A distributed Q learning Spectrum Decision Scheme for Cognitive Radio Sensor Network," *International Journal of Distributed Sensor Networks*, vol. 11, no. 5, pp. 1–10, May 2015.

[137] P. Sun, Z. Guo, G. Wang, J. Lan, and Y. Hu, "MARVEL: Enabling controller load balancing in software-defined networks with multi-agent reinforcement learning," *Computer Networks*, vol. 177, pp. 1–10, Jan. 2020.

[138] A. Valadarsky, M. Schapira, and D. S. A. Tamar, "Learning to Route," in *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, 2017, pp. 185–191.

[139] D. R. Militani, H. P. de Moraes, R. L. Rosa, L. Wuttisittikulkij, M. A. Ramírez, and D. Z. Rodríguez, "Enhanced Routing Algorithm Based on Reinforcement Machine Learning—A Case of VoIP Service," *Sensors*, vol. 21, no. 2, pp. 1–32, Jan. 2021.

[140] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep Reinforcement Learning: A Brief Survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, Nov. 2018.

[141] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[142] Y. Bengio, "Learning Deep Architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, Jan. 2009.

[143] J. Ma, M. K. Yu, S. Fong, K. Ono, E. Sage, B. Demchak, R. Sharan, and T. Ideker, "Using Deep Learning to Model the Hierarchical Structure and Function of a Cell," *Nature Methods*, vol. 15, no. 4, pp. 290–298, Mar. 2018.

[144] E. Guresena and G. Kayakutlu, "Definition of artificial neural networks with comparison to other networks," *Procedia Computer Science*, vol. 3, pp. 426–433, 2011.

[145] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of Trends in Practice and Research for Deep Learning," in *Proceedings of the 2nd International Conference on Computational Sciences and Technologies (INCCST 20)*, 2020, pp. 1–10.

[146] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. The MIT Press, 2016.

[147] P. Saravanan. Understanding Loss Functions in Machine Learning. [Online]. Available: https://www.section.io/engineering-education/understanding-loss-functions-in-machine-learning/ (accessed: 12 Oct. 2021).

[148] C. Zhang, P. Patras, and H. Haddadi, "Deep Learning in Mobile and Wireless Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2224–2287, 3rd quater 2019.

[149] E. Bjornson and P. Giselsson, "Two Applications of Deep Learning in the Physical Layer of Communication Systems," *IEEE Signal Processing Magazine*, vol. 37, no. 5, pp. 134–140, Sep. 2020.

[150] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to Optimize: Training Deep Neural Networks for Interference Management," *IEEE Transactions on Signal Processing*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018.

[151] A. Ly and Y.-D. Yao, "A Review of Deep Learning in 5G Research: Channel Coding, Massive MIMO, Multiple Access, Resource Allocation, and Network Security," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 396–408, Feb. 2021.

[152] P. Yu, F. Zhou, X. Zhang, X. Qiu, M. Kadoch, and M. Cheriet, "Deep Learning-Based Resource Allocation for 5G Broadband TV Service," *IEEE Transactions on Broadcasting*, vol. 66, no. 4, pp. 800–813, Dec. 2020.

[153] D. Huang, Y. Gao, Y. Li, M. Hou, W. Tang, S. Cheng, X. Li, and Y. Sun, "Deep Learning Based Cooperative Resource Allocation in 5G Wireless Networks," *Mobile Networks and Applications*, pp. 1–8, Dec. 2018.

[154] Y. Zhou, Z. M. Fadlullah, B. Mao, and N. Kato, "A Deep-Learning-Based Radio Resource Assignment Technique for 5G Ultra Dense Networks," *IEEE Network*, vol. 32, no. 6, pp. 28–34, Nov. 2018.

[155] Y. M. Lee, "Classification of node degree based on deep learning and routing method applied for virtual route assignment," *Ad Hoc Networks*, vol. 58, pp. 70–85, Apr. 2017.

[156] T. V. Chien, E. Bjornson, and E. G. Larsson, "Sum Spectral Efficiency Maximization in Massive MIMO Systems: Benefits from Deep Learning," in *IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

[157] Y. Jin, J. Zhang, S. Jin, and B. Ai, "Channel Estimation for Cell-Free mmWave Massive MIMO Through Deep Learning," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 325–10 329, Oct. 2019.

[158] Y. Liao, H. Yao, Y. Hua, and C. Li, "CSI Feedback Based on Deep Learning for Massive MIMO Systems," *IEEE Access*, vol. 7, pp. 86 810–86 820, Jun. 2019.

[159] M. Bashar, A. Akbari, K. Cumanan, H. Q. Ngo, A. G. Burr, P. Xiao, M. Debbah, and J. Kittler, "Exploiting Deep Learning in Limited-Fronthaul Cell-Free Massive MIMO Uplink," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 8, pp. 1678–1697, Aug. 2020.

[160] H.-Y. Kim and J.-M. Kim, "A load balancing scheme based on deep-learning in IoT," *Cluster Computing*, vol. 20, pp. 873–878, Nov. 2016.

[161] G. Gui, H. Huang, Y. Song, and H. Sari, "Deep Learning for an Effective Nonorthogonal Multiple Access Scheme," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8440–8450, Sep. 2018.

[162] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3175, 4th Quarter 2019.

[163] M. Riedmiller, "Neural Fitted Q Iteration – First Experiences with a Data Efficient Neural Reinforcement Learning Method," in *Proceedings of the 16th European conference on Machine Learning*, 2005, pp. 317–328.

[164] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[165] H. V. Hasselt, "Double Q-learning," in *Proceedings of Advances in Neural Information Processing Systems 23*, 2010, pp. 1–9.

[166] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-Learning," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 2094–2100.

[167] T. T. Anh, N. C. Luong, D. Niyato, Y.-C. Liang, and D. I. Kim, "Deep Reinforcement Learning for Time Scheduling in RF-Powered Backscatter Cognitive Radio Networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2019, pp. 1–7.

[168] M. S. Shokry, D. Ebrahimi, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Leveraging UAVs for Coverage in Cell-Free Vehicular Networks: A Deep Reinforcement Learning Approach," *IEEE Transactions on Mobile Computing*, vol. 20, no. 9, pp. 2835–2847, Sept. 2020.

[169] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep Reinforcement Learning Based Resource Allocation for V2V Communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp.

3163–3173, Apr. 2019.

[170] Y. Al-Eryani, M. Akrout, and E. Hossain, "Multiple Access in Cell-Free Networks: Outage Performance, Dynamic Clustering, and Deep Reinforcement Learning-Based Design," *IEEE Journal on Selected Areas in Communications (Early Access)*, 2020.

[171] N. Zhao, Y.-C. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5141–5152, Nov. 2019.

[172] J. Zhu, Y. Song, D. Jiang, and H. Song, "A New Deep-Q-Learning-Based Transmission Scheduling Mechanism for the Cognitive Internet of Things," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2375–2385, Aug. 2018.

[173] M. Chu, H. Li, X. Liao, and S. Cui, "Reinforcement Learning-Based Multiaccess Control and Battery Prediction With Energy Harvesting in IoT Systems," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2009–2020, Apr. 2019.

[174] S. Chinchali, P. Hu, T. Chu, M. Sharma, M. Bansal, R. Misra, M. Pavone, and S. Katti, "Cellular Network Traffic Scheduling With Deep Reinforcement Learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, pp. 1–9.

[175] M. Gadaleta, F. Chiariotti, M. Rossi, and A. Zanella, "D-DASH: A Deep Q-Learning Framework for DASH Video Streaming," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 703–718, Dec. 2017.

[176] H. Mao, R. Netravali, and M. Alizadeh, "Neural Adaptive Video Streaming with Pensieve," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, 2017, pp. 197–210.

[177] Y. He, Z. Zhang, F. R. Yu, N. Zhao, H. Yin, V. C. M. Leung, and Y. Zhang, "Deep-Reinforcement-Learning-Based Optimization for Cache-Enabled Opportunistic Interference

Alignment Wireless Networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10 433–10 445, Nov. 2017.

[178] S. Liu, X. Hu, and W. Wang, "Deep Reinforcement Learning Based Dynamic Channel Allocation Algorithm in Multibeam Satellite Systems," *IEEE Access*, vol. 6, pp. 15 733–15 742, Feb. 2018.

[179] P. V. R. Ferreira, R. Paffenroth, A. M. Wyglinski, T. M. Hackett, S. G. Bilén, R. C. Reinhart, and D. J. Mortensen, "Multiobjective Reinforcement Learning for Cognitive Satellite Communications Using Deep Neural Network Ensembles," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 5, pp. 1030–1041, May 2018.

[180] Y. He, F. R. Yu, N. Zhao, V. C. M. Leung, and H. Yin, "Software-Defined Networks with Mobile Edge Computing and Caching for Smart Cities: A Big Data Deep Reinforcement Learning Approach," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 31–37, Dec. 2017.

[181] Y. He, N. Zhao, and H. Yin, "Integrated Networking, Caching, and Computing for Connected Vehicles: A Deep Reinforcement Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 1, pp. 44–55, Jan. 2018.

[182] Y. He, Z. Zhang, and Y. Zhang, "A Big Data Deep Reinforcement Learning Approach to Next Generation Green Wireless Networks," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2017, pp. 1–6.

[183] Z. Tang, X. Zhou, F. Zhang, W. Jia, and W. Zhao, "Migration Modeling and Learning Algorithms for Containers in Fog Computing," *IEEE Transactions on Services Computing*, vol. 12, no. 5, pp. 712–725, Sept. 2019.

[184] N. Tafintsev, D. Moltchanov, M. Simsek, S.-P. Yeh, S. Andreev, Y. Koucheryavy, and M. Valkama, "Reinforcement Learning for Improved UAV-Based Integrated Access and Backhaul Operation," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–7.

[185] M. M. Sande, M. C. Hlophe, and S. Maharaj, "Instantaneous Load-Based User Association in

Multi-Hop IAB Networks using Reinforcement Learning," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, 2020, pp. 1–6.

[186] M. M. Sande, M. C. Hlophe, and B. T. Maharaj, "Access and Radio Resource Management for IAB Networks Using Deep Reinforcement Learning," *IEEE Access*, vol. 9, pp. 114 218–114 234, Aug. 2021.

[187] B. Zhang, F. Devoti, and I. Filippini, "RL-based Resource Allocation in mmWave 5G IAB Networks," in *Mediterranean Communication and Computer Networking Conference (MedComNet)*, 2020, pp. 1–8.

[188] M. Gupta, A. Rao, E. Visotsky, M. Cudak, A. Ghosh, and J. G. Andrews, "Learning-based Delay Optimization for Self-Backhauled Millimeter Wave Cellular Networks," in *53rd Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1–8.

[189] B. Zhang, F. Devotia, I. Filippini, and D. D. Donno, "Resource Allocation in mmWave 5G IAB Networks: A Reinforcement Learning Approach Based on Column Generation," *Computer Networks*, vol. 196, p. 108248, Sep. 2021.

[190] W. Lei, Y. Ye, and M. Xiao, "Deep Reinforcement Learning Based Spectrum Allocation in Integrated Access and Backhaul Networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 980–989, May 2020.

[191] Q. Cheng, Z. Wei, and J. Yuan, "Deep Reinforcement Learning-based Spectrum Allocation and Power Management for IAB Networks," in *IEEE International Conference on Communications Workshops (ICC Workshops)*, 2021, pp. 1–6.

[192] L. Yanjun, L. Xiaobo, and Y. Osamu, "Traffic engineering framework with machine learning based meta-layer in software-defined networks," in *Proceedings of the 4th IEEE International Conference on Network Infrastructure and Digital Content*, 2014, pp. 121–126.

[193] J. Reis, M. Rocha, T. K. Phan, D. Griffin, F. Le, and M. Rio, "Deep Neural Networks for Network Routing," in *International Joint Conference on Neural Networks (IJCNN)*, 2019, pp.

1–8.

[194] S. Yeo, Y. Naing, T. Kim, and S. Oh, "Achieving Balanced Load Distribution with Reinforcement Learning-Based Switch Migration in Distributed SDN Controllers," *Electronics*, vol. 10, no. 2, pp. 1–16, Jan. 2021.

[195] B. Mao, Z. M. Fadlullah, F. Tang, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "Routing or Computing? The Paradigm Shift Towards Intelligent Computer Network Packet Transmission Based on Deep Learning," *IEEE Transactions on Computers*, vol. 66, no. 11, pp. 1946–1960, Nov. 2017.

[196] Z. Li, X. Zhou, J. Gao, and Y. Qin, "SDN Controller Load Balancing Based on Reinforcement Learning," in *Proceedings of the IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, 2018, pp. 1–7.

[197] M. Jiang, D. Xenakis, S. Costanzo, N. Passas, and T. Mahmoodi, "Radio resource sharing as a service in 5G: A software-defined networking approach," *Computer Communications*, vol. 107, pp. 13–29, Jul. 2017.

[198] G. Sun, K. Xiong, G. O. Boateng, D. Ayepah-Mensah, G. Liu, and W. Jiang, "Autonomous Resource Provisioning and Resource Customization for Mixed Traffics in Virtualized Radio Access Network," *IEEE Systems*, vol. 13, no. 3, pp. 2454–2467, Sept. 2019.

[199] K. Xiong, S. S. R. Adolphe, G. O. Boateng, G. Liu, and G. Sun, "Dynamic Resource Provisioning and Resource Customization for Mixed Traffics in Virtualized Radio Access Network," *IEEE Access*, vol. 7, pp. 115 440–115 453, Aug. 2019.

[200] M. Simsek, D. Zhang, D. Öhmann, M. Matthé, and G. Fettweis, "On the Flexibility and Autonomy of 5G Wireless Networks," *IEEE Access*, vol. 5, pp. 22 823–22 835, Jun. 2017.

[201] M. B. A. Sadi and A. Nadia, "Call Admission Scheme for Multidimensional Traffic Assuming Finite Handoff User," *Journal of Computer Networks and Communications*, vol. 1, pp. 1–5, Jan. 2017.

[202] A. Chydzinski and B. Adamczyk, "Queues with the Dropping Function and General Service Time," *PLoS ONE*, vol. 14, no. 7, pp. 1–21, Jul. 2019.

[203] W. Fang, J. Chen, L. Shu, T. Chu, and D. Qian, "Congestion Avoidance, Detection and Alleviation in Wireless Sensor Networks," *Journal of Zhejiang University Science C*, vol. 11, no. 63, pp. 63–73, Dec. 2010.

[204] V. Pla, A. S. Alfa, J. Martinez-Bauset, and V. Casares-Giner, "Discrete-time Analysis of Cognitive Radio Networks with Nonsaturated Source of Secondary Users," *Wireless Communications and Mobile Computing*, vol. 2019, pp. 1–12, Dec. 2019.

[205] A. Slalmi, H. Chaibi, A. Chehri, R. Saadane, and G. Jeon, "Toward 6G: Understanding Network Requirements and Key Performance Indicators," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 3, pp. 1–18, Mar. 2021.

[206] R. Cimurs, J. H. Lee, and I. H. Suh, "Goal-oriented Obstacle Avoidance with Deep Reinforcement Learning in Continuous Action Space," *Electronics*, vol. 9, no. 3, pp. 1–16, Feb. 2020.

[207] Ankit Choudhary. A Hands-On Introduction to Deep Q-Learning using OpenAI Gym in Python. [Online]. Available: https://www.analyticsvidhya.com/blog/2019/04/introduction-deep-q-learning-python/ (accessed: 01 Jun. 2021).

[208] M. C. Hlophe and B. T. Maharaj, "Secondary User Experience-oriented Resource Allocation in AI-empowered Cognitive Radio Networks Using Deep Neuroevolution," in *Proceedings of the IEEE Vehicular Technology Conference (VTC-Spring)*, 2020, pp. 1–5.

[209] M. Carter and B. van Brunt, *The Lebesgue-Stieltjes Integral: A Practical Introduction*. Springer, 2002.

[210] M. C. Hlophe and B. T. Maharaj, "Spectrum Occupancy Reconstruction in Distributed Cognitive Radio Networks using Deep Learning," *IEEE Access*, vol. 7, pp. 14 294–14 307, Jan. 2019.

[211] H. Cuayáhuitl, S. Keizer, and O. J. Lemon, "Strategic Dialogue Management via Deep Reinforcement Learning," *ArXiv:1511.08099*, pp. 1–10, Nov. 2015.

[212] C. Ding and et al., "Circnn: Accelerating and Compressing Deep Neural Networks Using Block-circulant Weight Matrices," in *Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture*, 2017, pp. 395–408.

[213] S. Koenig and R. G. Simmons, "Complexity Analysis of Real-time Reinforcement Learning," in *AAAI*, 1993, pp. 99–107.

[214] S. M. Kakade, "On the Sample Complexity of Reinforcement Learning," Ph.D. thesis, Gatsby Computational Neuroscience Unit, Univ. College London, 2003.

[215] B. Price and C. Boutilier, "Accelerating Reinforcement Learning Through Implicit Imitation," *Journal of Artificial Intelligence Research*, vol. 19, no. 1, pp. 569–629, Aug. 2003.

[216] M. Botvinick, S. Ritter, J. X. Wang, Z. Kurth-Nelson, C. Blundell, and D. Hassabis, "Reinforcement Learning, Fast and Slow," *Trends in Cognitive Sciences*, vol. 23, no. 5, pp. 408–422, May 2019.

[217] J. Hyun, N. V. Tu, and J. W.-K. Hong, "Towards Knowledge-defined Networking Using in-band Network Telemetry," in *IEEE/IFIP Network Operations and Management Symposium (NOMS 2018)*, 2018, pp. 1–7.

[218] S. Jouet, C. Perkins, and D. Pezaros, "OTCP: SDN-managed congestion control for data center networks," in *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2016, pp. 171–179.

[219] M. D. Hill, "Three Other Models of Computer System Performance," *ArXiv:1901.02926*, pp. 1–5, Jan. 2019.

[220] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Distributed Federated Learning for Ultra-Reliable Low-Latency Vehicular Communications," *IEEE Transactions on Communications*,

vol. 68, no. 2, pp. 1–5, Feb. 2020.

[221] C. Xu, W. Zhuang, and H. Zhang, "A Deep-reinforcement Learning Approach for SDN Routing Optimization," in *Proceedings of the 4th International Conference on Computer Science and Application Engineering (CSAE)*, 2020, pp. 1–5.

[222] T. Ai, A. A. Wahid, and V. Wijeratne, "Impact of buffer sizing on energy efficiency and performance," *IET Networks*, vol. 4, no. 1, pp. 1–9, Jan. 2015.

[223] N. Mastronarde and M. van der Schaar, "Joint Physical-Layer and System-Level Power Management for Delay-Sensitive Wireless Communications," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 694–709, Apr. 2013.

[224] M. Zimmermann and E. Frejinger, "A Tutorial on Recursive Models for Analyzing and Predicting Path Choice Behavior," *EURO Journal on Transportation and Logistics*, vol. 9, no. 2, pp. 1–12, Jun. 2020.

[225] F. L. Lec and B. Tarroux, "On Attitudes to Choice: Some Experimental Evidence on Choice Aversion," *Journal of the European Economic Association*, vol. 18, no. 5, pp. 2108–2134, Oct. 2020.

[226] D. Fudenberg and T. Strzalecki, "Dynamic Logit with Choice Aversion," *Econometrica*, vol. 83, no. 2, pp. 651–691, Mar. 2015.

[227] A. Hansen, "The Three Extreme Value Distributions: An Introductory Review," *Frontiers in Physics*, vol. 8, pp. 1–8, Dec. 2020.

[228] Q. Liang and E. Modiano, "Optimal Network Control with Adversarial Uncontrollable Nodes," in *Proceedings of the IEEE Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing (INFOCOM)*, 2019, pp. 101–110.

[229] T. K. Vu, C.-F. Liu, M. Bennis, M. Debbah, M. Latva-aho, and C. S. Hong, "Ultra-reliable and Low Latency Communication in mmWave-enabled Massive MIMO Networks," *IEEE*

Communications Letters, vol. 21, no. 9, pp. 2041–2044, Sep. 2017.

[230] L. G. Afanaseva and S. Grishunina, "Stability Conditions for a Multiserver Queueing System with a Regenerative Input Flow and Simultaneous Service of a Customer by a Random Number of Servers," *Queueing Systems*, vol. 94, no. 3, pp. 213–241, Apr. 2020.

[231] R. Livni, S. Shalev-Shwartz, and O. Shamir, "On the Computational Efficiency of Training Neural Networks," *Advances in Neural Information Processing Systems*, vol. 1, pp. 1–15, Oct. 2014.

[232] L. Fredenslund. Computational Complexity Of Neural Networks. [Online]. Available: https://lunalux.io/series/introduction-to-neural-networks/computational-complexity-of-neural-networks (accessed: 22 Mar. 2022).

[233] T. Lattimore, M. Hutter, and P. Sunehag, "The Sample-complexity of General Reinforcement Learning," in *Proceedings of the 30th International Conference on Machine Learning*, 2013, pp. 28–36.

[234] Y. Song and S. Ermon, "Generative Modeling by Estimating Gradients of the Data Distribution," *Advances in Neural Information Processing Systems*, vol. 32, pp. 11 918–11 930, Dec. 2019.

[235] A. Mauleon, S. Schopohl, A. Taalaibekova, and V. Vannetelbosch, "Coordination on Networks with Farsighted and Myopic Agents," *International Journal of Game Theory*, pp. 1–28, Jan. 2022.

[236] A. Hagberg, P. Swart, and D. S. Chult, "Exploring Network Structure, Dynamics, and Function Using NetworkX," in *Proceedings of the 7th Python in Science Conference (SciPy2008)*, 2008, pp. 11–16.

[237] W. Su, L. Chen, M. Wu, M. Zhou, Z. Liu, and W. Cao, "Exploring Network Structure, Dynamics, and Function Using NetworkX," in *11th Asian Control Conference (ASCC)*, 2017, pp. 1063–1068.