The Dissertation Committee for Faris Bassam Mismar
certifies that this is the approved version of the following dissertation:

Improving Next-Generation Wireless Network Performance and Reliability with Deep Learning

Committee:

Brian L. Evans, Supervisor

Naofal Al-Dhahir

Jeffery G. Andrews

Joydeep Ghosh

Haris Vikalo

Improving Next-Generation Wireless Network Performance and Reliability with Deep Learning

by

Faris Bassam Mismar

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2019

Dedicated to my parents, my wife, and my children.

Acknowledgments

Foremost, I would like to express my sincerest appreciation to my supervisor Prof. Brian L. Evans for his guidance, kindness, and seemingly infinite support during and before my Ph.D. journey. In addition, I extend my deepest gratitude to the rest of my dissertation committee: Prof. Naofal Al-Dhahir, Prof. Jeffrey G. Andrews, Prof. Joydeep Ghosh, and Prof. Haris Vikalo for their encouragement and insightful comments on this dissertation and my research plan.

I wish to thank Prof. Constantine Caramanis for introducing me to machine learning with Python and believing in my "iron" determination. Also, I wish to thank Ahmed Alkhateeb, Ahmad AlAmmouri, and Jinseok Choi for the motivation, the numerous discussions, and fruitful research efforts. Special thanks to the rest of the members of the ESPL lab: Hugo Andrade, Yunseong Cho, Scott Johnston, and Junmo Sung for their preparedness to discuss my research results and presentations often at short notice. Further, I want to thank my friends at UT ECE: Rebal Jurdi, Rahi Kalantari, Anum Ali, Abolfazl Hashemi, Isfar Tariq, and Travis Cuvelier. I also want to extend my gratitude to the staff at WNCG: Melanie Gulick, Apipol Piman, and Jaymie Udan, who made my navigation of the logistics at UT ECE less turbulent. Melanie is an amazing person whose advice and assistance in handling my countless inquiries

since the day of the orientation in Fall 2015 has benefited me tremendously.

Pursuing a Ph.D. on part-time basis while working full-time and living more than 200 miles away from the campus requires unique personal traits and a multitude of enabling resources. I wish to express my sincere gratitude towards The Wallenberg Foundations in Sweden for their generous scholarship, which covered the entire cost of pursuing my degree. This four-year pursuit has left me with a myriad of indelible memories: my twice a week flights back and forth between Dallas Love Field and Austin Bergstrom airports, all the timetables I iteratively created in a serious attempt to make my trips time efficient, the moments on the Forty Acres where it rained without me carrying an umbrella, Cyril the CapMetro bus-driver of Route 100 who on my first day at UT paid the fare out of his pocket and dropped me at the nearest bus stop on campus, and the TSA agent Officer Murkledove whom I saw every single week at the security checkpoint and who told me on my first day that UT was the best school in the world!

I also wish to thank Prof. Hlaing Minn, Prof. Joseph C. Picken, Dr. P. K. Rajasekaran, and Prof. Lakshman Tamil from The University of Texas at Dallas, who supported me to pursue my doctorate. I leave this sentence as a placeholder to thank the mysterious supporters who may have offered me helping hands while I was too occupied in this journey to have noticed them.

Finally, I thank my great family: my parents, my wife Iman, and my children Bassam and Siwar for their boundless sacrifice and love. No words can describe my gratitude to you all, especially to my faithful wife Iman.

Traveling for eight hours twice a week for three long semesters to pursue a Ph.D. at a top-tier school almost sounds surreal. However, I believe that resilience, determination, and persistence are key to pushing through obstacles and achieving ambitions. My motivation to realize the seemingly impossible was eloquently stated by *al-Maʿarrī* the famous Abbasid poet (973-1057 CE): despite me belonging to a later era, I am set to achieve what the pioneers of earlier eras never could. And, as my children taught me from their school:

"I can and I will... watch me!"

Faris B. Mismar

The University of Texas at Austin
December 2019

Improving Next-Generation Wireless Network Performance and Reliability with Deep Learning

Publication No. _____

Faris Bassam Mismar, Ph.D.
The University of Texas at Austin, 2019

Supervisor: Brian L. Evans

A rudimentary question whether machine learning in general, or deep learning in particular, could add to the well-established field of wireless communications, which has been evolving for close to a century, is often raised. While the use of deep learning based methods is likely to help build intelligent wireless solutions, this use becomes particularly challenging for the lower layers in the wireless communication stack. The introduction of the fifth generation of wireless communications (5G) has triggered the demand for "network intelligence" to support its promises for very high data rates and extremely low latency. Consequently, 5G wireless operators are faced with the challenges of network complexity, diversification of services, and personalized user experience. Industry standards have created enablers (such as the network data analytics function), but these enablers focus on post-mortem analysis at higher stack layers and have a periodicity in the time scale of *seconds* (or larger).

The goal of this dissertation is to show a solution for these challenges and how a data-driven approach using deep learning could add to the field of wireless communications. In particular, I propose intelligent predictive and prescriptive abilities to boost reliability and eliminate performance bottlenecks in 5G cellular networks and beyond, show contributions that justify the value of deep learning in wireless communications across several different layers, and offer in-depth analysis and comparisons with baselines and industry standards.

First, to improve multi-antenna network *reliability* against wireless impairments with power control and interference coordination for both packetized voice and beamformed data bearers, I propose the use of a joint beamforming, power control, and interference coordination algorithm based on deep reinforcement learning. This algorithm uses a string of bits and logic operations to enable simultaneous actions to be performed by the reinforcement learning agent. Consequently, a joint reward function is also proposed. I compare the performance of my proposed algorithm with the brute force approach and show that similar performance is achievable but with faster run-time as the number of transmit antennas increases.

Second, in enhancing the *performance* of coordinated multipoint, I propose the use of deep learning binary classification to learn a surrogate function to trigger a second transmission stream instead of depending on the popular signal to interference plus noise measurement quantity. This surrogate function improves the users' sum-rate through focusing on pre-logarithmic terms in the sum-rate formula, which have larger impact on this rate.

Third, *performance* of band switching can be improved without the need for a full channel estimation. My proposal of using deep learning to classify the quality of two frequency bands prior to granting the band switching leads to a significant improvement in users' throughput. This is due to the elimination of the industry standard measurement gap requirement—a period of silence where no data is sent to the users so they could measure the frequency bands before switching.

In this dissertation, a group of algorithms for wireless network performance and reliability for *downlink* are proposed. My results show that the introduction of user coordinates enhance the accuracy of the predictions made with deep learning. Also, the choice of signal to interference plus noise ratio as the optimization objective may not always be the best choice to improve user throughput rates. Further, exploiting the spatial correlation of channels in different frequency bands can improve certain network procedures without the need for perfect knowledge of the per-band channel state information. Hence, an understanding of these results help develop novel solutions to enhancing these wireless networks at a much smaller time scale compared to the industry standards today.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The creation of models that can accurately predict the future is an arduous task, if not an impossible one. Deep learning is a class of machine learning models composed of multiple processing layers to learn representations of data with multiple levels of abstractions [1]. These models have brought about breakthroughs in image processing and in game playing [2]. As wireless communication networks continue to evolve in terms of peak throughputs, lower latency, and ubiquity in coverage, improving their performance and reliability becomes imperative. Consequently, the use of deep learning in wireless communication networks to improve their performance and reliability may become possible if improvement actions are based on data-driven approaches. Combining simulations of the data-driven approach with the knowledge of the subject matter, these learning models can offer insights and deliver results avoiding simplified propositions. Combining these learning models with the ability of the networks to independently identify bottlenecks and improve performance is the essence of autonomous networks. In this introduction, I start with discussing machine learning and then pivot into deep learning. Next, I discuss the relevant aspects of next-generation wireless networks. Then, I talk about the use cases of deep learning in communications and the pertinent challenges.

I conclude this chapter with a thesis statement, proposed contributions, the notations and abbreviations I use, and the organization of the rest of this dissertation.

1.1 Machine Learning

Machine learning (ML) is an application of statistics whereby computers estimate complicated functions [2]. This application plays an important role in many fields such as data mining and artificial intelligence (AI). The term "learning" simply means the ability to predict outcomes of new observations. This is performed by the "learner." The data used to enable learning is based on a set of "features." These learning features can be either 1) categorical or 2) continuous. Data is often structured in a data frame known as the "design matrix," where each row represents a single object and the columns of this matrix are the features that correspond to specific values for that object. These features then undergo a phase of feature engineering, where more features are created and others are dropped based on the understanding of the subject matter and the problem formulation.

Several learning algorithms fall under ML: supervised learning, unsupervised learning, and reinforcement learning. In order for the various algorithms to output results, they need to be trained first. A trained algorithm is called a "model." I will briefly talk about each one of algorithm types next.

1.1.1 Supervised Learning

Supervised learning provides access to a set of learning features measured over a number of observations and an outcome variable (also known as the label or the target) measured over these observations. Given its nature, supervised learning is intuitively well-understood, especially when the supervisory opinion is related to human characterization or judgement. There are two types of supervised learning: classification and regression. In classification, the learner deals with a problem where the outcome variable is categorical. A classification model tries to predict the value of one or more outcomes. In regression, the learner deals with a problem where this variable is continuous. Classification problems often occur more frequently in practice than regression [3].

The objective of a supervised learning model is to optimize a loss function. An example of a loss function is the binary cross-entropy in binary classification, and the mean square error (or the quadratic loss) in regression. Not all loss functions have to be convex.

An important intuition in optimizing these loss functions is that they should also lead to solutions that also optimize the objective of the problem formulation. Hence, a mathematical effort is often needed to ensure that the loss function and the problem objective do not result in contradictory solutions.

1.1.2 Unsupervised Learning

Unsupervised learning has no information about the outcome variable. The learner in this type categorizes inputs according to their similarities and differences. Hence, this type of learning enables us to learn relationships and structures from the data. Unsupervised learning can be broadly classified into two different types of problems: clustering and association. In clustering, the learner discovers the inherent groupings in the data. In association, the learner uncovers rules that describe large portions of the data.

1.1.3 Reinforcement Learning

Reinforcement learning is primarily concerned with the problems that require finding suitable actions to take in given states in order to maximize a reward related to an objective. When these elements are combined in a tuple, they are called "experience" [4]. Reinforcement learning agents learn through interaction with an environment to build this experience. In this type of algorithms, there is no dataset to learn from.

1.2 Deep Learning

Deep learning is an approach to ML that exploits its resemblance to the human brain [2]. In this section, I discuss the deep learning algorithms covered in my dissertation: deep neural networks (DNN) and deep reinforcement learning (DRL).

1.2.1 Deep Neural Networks

Deep neural networks in essence are multilayer perceptrons (MLPs) with multiple layers between the input and output nodes. MLPs are the feed-forward deep network that maps a set of input values to output values. Feed-forward networks approximation properties are very general and as a result, MLPs are said to be "universal approximators" [5].

Besides being a universal approximator, DNNs have recently gained lots of publicity with the increase in training dataset sizes, often referred to as "big data." With increased model sizes, aided by the availability of faster computational power and distributed computing, DNNs enjoyed more depth (number of connected layers) and width (the number of neurons per layer). This increase in the total number of neurons in the DNN has unleashed the ability of DNNs to perform complex classification tasks with high accuracy. The total number of connections per neurons in DNN designs today have approached the number of of connections per neurons as the brain of a cat [2]. The total number of neurons in DNNs today double in size every 2.4 years. At this rate, a DNN is likely to have a similar number of neurons as a human brain in the 2050s [2]. This growth makes DNN a very attractive solution to potentially complex AI problems.

1.2.2 Deep Reinforcement Learning

The implementation of reinforcement learning can be either 1) deep or 2) vanilla (i.e., not assisted by any deep neural networks). Vanilla RL is

often also called a "tabular" solution [4, 6] since it uses a table to store its experience. On the other hand, deep RL uses DNN to estimate its experience and predict expected discounted rewards in order to select suitable actions [7]. Deep reinforcement learning therefore resembles supervised learning in some sense: the DNN learns from a dataset constructed from the actions made by agent and the environment state and reward to find out the action that leads to the best expected future reward. The introduction of DNN to reinforcement learning as an estimator can cause convergence problems in deep RL—a problem that tabular RL can avoid. The deep reinforcement learning problem is the AI problem [6].

Now that I have introduced deep learning, I discuss next-generation wireless networks hereafter. Then, I cover the deep learning use cases and the challenges in employing it in communications.

1.3 Next-Generation Wireless Networks

The vision of next-generation wireless networks promises very high data rates, ultra-low latency, and ubiquitous coverage [8–11]. Proliferation of connected devices and demand for applications higher up in the value chain (e.g., industrial automation, healthcare wearables, extended reality, and mission critical services) are already burdening existing cellular networks today. In order for Wi-Fi 6, for example, to cater to this substantial demand in indoor environments, Wi-Fi 6 offers larger bandwidth, interference mitigation, longer symbol duration, and higher order modulation and code rates. Similarly, new

radio (NR), which is the air interface of the fifth generation of wireless communications (5G), introduces technologies such as scalable air interface, massive multiple-input multiple-output (MIMO), larger bandwidths, enhanced coordinated multipoint or cell-free MIMO, advanced coding techniques, and AI-assisted radio access networks [9,12–15]. However, these applications also elicit demands for sophisticated radio resource management (RRM) algorithms to ensure that the wireless impairments are predicted and circumvented before they impact the end-user perceived quality of experience.

Several standardization entities have been working to evolve wireless networks to the next generation. I next cover these entities and the most recent work they have done in this space.

1.3.1 International Telecommunication Union

The International Telecommunication Union (ITU) designed the standards for prior generations of wireless communications. Under its International Mobile Telecommunication (IMT-2020) program, members of ITU develop the standards required to achieve 5G performance with a forward outlook to performance beyond 5G [16]. ITU has three main areas of activity: radio communications, standardization, and development to bridge the digital divide. Therefore, ITU sets the vision for next generation research activities.

1.3.2 The 3rd Generation Partnership Project

The 3rd Generation Partnership Project (3GPP) is formed by a union of several telecommunications standard development organizations to provide technical reports and specifications. The 3GPP project provides a complete system description for telecommunications [17].

3GPP Release 15 standards [10] bring forth the complete set of 5G radio standards. This release is working towards aligning 5G NR with the IMT-2020 performance requirement. In readiness for the introduction of AI in 5G, 3GPP have introduced a set of AI-enablers for automation of next generation wireless networks such as the "Network Data Analytics Function" [18], which helps with analyzing 5G network problems in retrospect (i.e., post-mortem).

1.3.3 Institute of Electrical and Electronics Engineers

The Institute of Electrical and Electronics Engineers (IEEE) defines the standards dealing with local area networks and metropolitan area networks under the IEEE Project 802 [19]. In fact, the 802.11ay standard defines the broadband wireless access operating at the 60 GHz mmWave frequency band [20, 21]. It is the next-generation of wireless access using Wi-Fi that promises up to 100 Gbps through MIMO, channel bonding, improved channel access, and enhanced beamforming training. There is no indication (as of the time this dissertation was written) whether IEEE will submit any of its 802.11 specifications to be recognized as an IMT-2020 technology.

1.3.4 Open Radio Access Network Alliance

The Open Radio Access Network (O-RAN) alliance is led by mobile network operators (such as AT&T, NTT DoCoMo, and Verizon)—in contrast with 3GPP which is led by network infrastructure vendors. This alliance is an effort to build the next generation of radio access networks using virtualization and machine intelligence. O-RAN specifications have introduced an AI-enabled RAN intelligent controller for both non-real time and near-real time operations [22]. The primary goal of this controller is to support tasks both in non-real-time (i.e., intelligent radio resource management, higher layer procedure optimization, policy optimization) and near real-time (i.e., providing ML models). In fact, both 3GPP and O-RAN are independently working to introduce machine learning enablers to RAN with a vision of openness, functional efficiency, and reduction in operating expenses [18, 22].

1.4 Motivating Deep Learning in Communications

The efficient realization of deep learning based algorithms in communications faces several challenges. However, there are many reasons why the use of deep learning in communications can prove helpful in numerous ways that outweigh the challenges. In this section, I discuss some of these reasons and the challenges that I address in my dissertation. Further, I keep my argument in addressing these challenges simple, often abiding by Occam's Razor that states: "among competing hypotheses, the simplest is the best."

1.4.1 Mathematical Intractability

Problems related to optimizing the sum-rate capacity are NP-hard [23]. When multiple users are being studied, non-convexity can also make the problem formulation more challenging. Therefore, the choice of convex optimization, which is typical in such a problem (or others such as network resources maximization), may not deliver the desired outcome. Problems of this nature, and other mathematically intractable problems, make the choice of deep learning quite interesting.

1.4.2 Legacy Radio Resource Management

RRM is a group of algorithms for controlling radio parameters (e.g., transmit power, load balancing, beamforming, band switching, handovers) in the wireless network. The objective of RRM is to utilize the limited radio-frequency spectrum resources and radio network infrastructure as efficiently as possible [24]. Despite their increasing importance, many of the RRM procedures have only seen incremental changes over the evolution of multiple successive industry standards.

The RRM algorithms in the 3GPP industry standards today [10, 24] are inherited from the "legacy" RRM algorithms, where the networks were voice-centric and as a result were simplistic and reactive in nature [25, 26]. In spite of the successive evolutions of these industry standards, the incremental changes in the RRM algorithms form a dilemma: on the one hand, the explosive expectations of what the next generation networks will offer; and on the

other hand, the performance offered by these legacy RRM algorithms often poses a bottleneck. Examples of this dichotomy are:

1. A requirement of a period of data transmission silence (known as the "measurement gap" [27]) for the user equipment to estimate the channel prior to the change to a different frequency band.

2. Use of physical layer measurements (e.g., signal strength, signal quality, channel rank) instead of the coordinates of the user equipment to accurately estimate the channel.

3. At least two separate quantities fed by the user to the serving base station (known as the channel state information) in order for the base station (BS) to trigger spatial multiplexing transmission to the user: the rank indicator and the quality indicator. These quantities are constrained such that the predicted codeword retransmission rate of the next transmission is minimized.

Both examples may have worked well in low throughput or voice-centric networks. However, high data rates may certainly suffer as a result of these silence periods. Furthermore, the consideration of independent transmission rank indicator and quality indicator per transmission—instead of a surrogate composite trigger that also handles non-linear interactions of both—can also cause lower achievable throughputs. At high data rates, even a small percentage of retransmission over any codeword and any antenna can cause big loss of

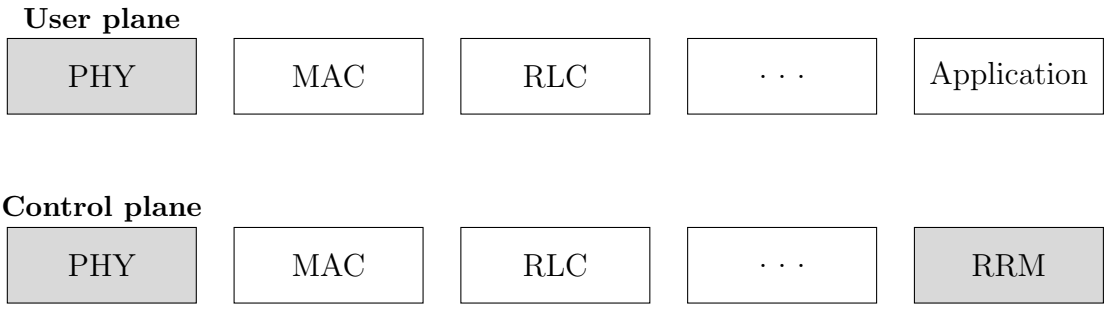


Figure 1.1: Air interface protocol stack showing the protocols covered in this dissertation (shaded). PHY is the physical layer, MAC is the medium access control layer, RLC is the radio link control layer, and RRM is the radio resource management.

throughput—a critical feature in next-generation wireless networks. The mitigation of such disconnect can be resolved through introducing novel algorithms with deep learning.

The NR air interface protocol stack is shown in Fig. 1.1. RRM falls under what is referred to as the "control plane" in the air interface protocol stack [10, 27]. This is contrast to the "user plane" where the user traffic (e.g., voice or data) falls under. However, these RRM algorithms can operate across different layers in the air interface protocol.

1.4.3 Autonomous Networks

Self-organizing network (SON) is a conglomerate of three solutions: self-configuration, self-healing, and self-optimization [28]. The SON architecture can be a centralized, distributed, or hybrid. It can be data-assisted (e.g., machine learning) or procedural. In any case, the trial and error accompa-

nying the introduction of SON autonomous actions to a real network poses a serious challenge: changing parameters and training algorithms can cause severe degradation in performance or network-wide outages in the worst case. However, it is obvious that the continuous and autonomous optimization that SON offers introduces several advantages: 1) minimizing fault impact duration and 2) reducing operational expenditure by reducing human efforts for manual tasks. SON combined with AI to self-configure, self-monitor, and self-maintain is often referred to as an "autonomous network."

Challenges: Classification requires a phase of fitting the hyperparameters to a dataset. Hyperparameters are the settings of the model performing the classification. They also control the capacity of the model—its ability to generalize its predictive results besides the training dataset. Achieving optimal capacity of models is not an easy task. In fact, finding the optimal set of hyperparameters in a deep learner is NP-hard [29]. In the case of deep RL, episodes are run for a duration of time until convergence is achieved, which is not guaranteed [7]. Further, finding the convergence episode with the optimal reward is also an NP-hard problem. It is obvious that learning does not happen in no time. During the time a classifier learns the desired algorithm, or the RL episode achieved convergence, many changes in the wireless channel may have taken place. This would render the models invalid to the task they were trained for.

Several approaches are taken to mitigate the time required for learning to happen. First, the computational architecture as I explained earlier. Com-

putational power implies the use of graphics processing units among multi-core central processing units with ample memory size. Second, proper initialization of models or the use of learning efficient algorithms such as deep RL to avoid the task of finding non-trivial initialization values. Third, the invalidation of the learned model after a certain time has passed. Fourth, in cases where a relationship between statistics in a wireless network is exploited (e.g., spatial correlation between channels in different frequency bands), the use of a small proportion of the measurements to train a deep learning model can overcome the time constraint keeping decisions relevant and near real-time, as I show in Chapter 4. Finally, the distributed training in models where one model exists per network entity can reduce the training complexity through a divide and conquer approach, which directly improves the training time.

1.5 Deep Learning in Communications Use Cases

The basis of communication in wireless networks is built on traditional probabilistic models (e.g., channel, noise, interference) [30]. Machine learning, and by inheritance deep learning, is changing this model-driven approach to a data-driven approach. In this data-driven approach, simulations lead to learning the models from extensive datasets available from field measurements from real networks or ray-tracing. This can help form a holistic understanding of the true wireless system instead of resorting to modeling some aspects of it. While data from field measurements is clearly preferred, generating such dataset is highly non-trivial.

Furthermore, the use of deep learning in communication can be broadly classified by their response time as near real-time and non-real time. In frequently changing wireless conditions, constraints about the data collection time and the invalidation of prior learned models need to be imposed. In addition, constraints about where the computational platforms reside with regards to the network (e.g., edge vs center) become important. Often, a hierarchical computational architecture is introduced in communication networks, where the training of the decision making model takes place in a central location. This central location then pushes a copy of this trained model to the network edge to be applied to the users.

Deep learning in communications can also be classified by the type of analysis offered following [31] as

1. Descriptive: offers insights about the use case by looking into the past.
2. Predictive: understands the future and allows decisions to be made in anticipation.
3. Prescriptive: provides advice about possible outcomes.

While descriptive analysis has its applications in communications such as root cause analysis, the focus in my dissertation is on the latter two. Furthermore, I do not use unsupervised learning techniques in my dissertation since I use labeled datasets and environments with a defined objective to improve. Thus, I use *deep learning classifiers* and *deep reinforcement learning* in

the solution of my proposed problems. In my dissertation, I focus on the 3GPP NR and LTE air interfaces in the wireless communication networks. The air interface is the interface between the user equipment and the base station.

1.6 Dissertation Summary

To summarize, I have contributed to the *problem formulation* of the radio resource management and physical layer algorithms to improve the performance of the next-generation wireless communications using deep learning.

Also, because the contributions in my dissertation use various deep learning algorithms to enhance the wireless networks performance, my focus is on offering depth in the motivation, the comparison with other algorithms, and the analysis of the results. Explaining the results of deep learning algorithms—and ML in general—is notoriously challenging because 1) trained deep learning models are often inexplicable and hence treated as black boxes and 2) data features are further complicated through the complex approach when extracted and combined.

1.6.1 Thesis Statement

In this dissertation, I defend the following thesis statement:

Next-generation wireless networks will require intelligent predictive and prescriptive abilities to disrupt the reactive legacy standards in order to boost reliability and eliminate performance bottlenecks.

1.6.2 Summary of Contributions

For the use of predictive abilities in next-generation wireless networks, I propose the following contributions:

- *Gap-free band switching*: using deep learning and exploiting the correlation between channels based on the coordinates of served UEs, I develop a novel algorithm to eliminate the measurement gap during the band switching procedure. I predict whether a band switching request between sub-6 GHz or mmWave frequency bands would lead to better end-user throughput, and grant the switch request only if so.

- *Surrogate MIMO trigger function*: using deep learning classification with the codeword retransmission rates, the received signal power, the received signal to interference plus noise ratio (SINR), and the MIMO transmission rank (i.e., the degree of freedom gain), I build a surrogate function of these quantities to trigger coordinated multipoint—a form of network distributed MIMO. This trigger function can improve the throughput of users served by the distributed MIMO network through exploiting the impact of pre-logarithmic terms instead of the SINR.

For the prescriptive abilities to eliminate performance bottlenecks, I propose the following contribution:

- *Downlink power control and interference coordination*: I introduce power control and interference coordination using deep RL. This work is motivated by the exploitation of semi-persistent scheduling in voice and user-specific beamforming in data, both of which provide a virtual sense of dedicated channels.

To make my contributions easy to refer to, I provide an overview which captures the contribution title, the problem, the goal, the parameters, and the approach. The contributions are numbered in the order they appear in this dissertation.

Contribution 1: Joint Beamforming, Power Control, and Interference Coordination

- Problem:
  - User served by a BS receives interference from neighboring base station.
  - BS serving the user causes interference to other users.
- Goal: Improve the SINR from serving BS to user.
- Parameters:
  - Beamforming (BF) to create a virtual sense of a user-specific channel for data.
  - Power control (PC) to control the transmit power of the serving BS towards a user.
  - Interference coordination (IC) to control the transmit power of the neighboring BS.
  - User spatial coordinates.
- Approach:
  - Perform binary encoding of BF, PC, and IC actions in a string of bits to enable joint actions.
  - If SINR of all users improve, then reward actions. This resolves the race between base stations attempting to control power and interference levels simultaneously.
  - Compare proposed solution with brute force solution.

Contribution 2: Improving Downlink Coordinated Multipoint Performance with Deep Learning

- Problem:

  - Industry implementations trigger coordinated multipoint based on user SINR, which leads to low throughput.

- Goal: Develop triggering function to improve the user throughput.

- Parameters:

  - Block error rate (BLER) target for codeword reception error.
  - Channel state information to help derive transmission rank.

- Approach:

  - Train a classifier to learn the relationship between the reported measurements and the BLER.
  - If a user is predicted to have a BLER lower than the target, configure rank-2 transmission.
  - Compare with SINR-based trigger.

Contribution 3: Deep Learning Predictive Band Switching in Wireless Networks

- Problem:

  - Users want to switch to a different frequency band if they expect to get higher throughput.

  - Switching between frequency bands requires a "measurement gap" which reduces user throughput.

- Goal: Improve user throughput by exploiting the spatial correlation to eliminate the measurement gap.

- Parameters:

  - Band switch request threshold which defines the rate below which the user requests a band switch.

  - Band switch grant threshold which defines the rate above which the user request is granted.

  - Percentage of users in sub-6 GHz or millimeter wave vs total users.

  - User spatial coordinates.

- Approach:

  - Employ a data-driven approach using a ray-tracing dataset.

  - Use deep learning to rank the downlink channel quality based on the users' coordinates.

  - Grant a band switch if predicted to improve the user throughput (no need for the gap).

1.7 Notation and Abbreviations

This dissertation uses the following notation: boldface lower case $\mathbf{a}$ and upper case $\mathbf{A}$ symbols represent column vectors and matrices, respectively. The i-th column vector of the matrix $\mathbf{A}$ is denoted by $\mathbf{a}_i$. Calligraphic letters $\mathcal{A}$ are for sets. The cardinality of a set is $|\cdot|$. The transpose and Hermitian transpose operators are $(\cdot)^\top$ and $(\cdot)^*$. $\mathbf{A}\otimes\mathbf{B}$ is the Kronecker product of $\mathbf{A}$ and $\mathbf{B}$. The trace of a matrix is $\mathrm{Tr}(\cdot)$. The expectation operator is $\mathbb{E}[\cdot]$. Probability is denoted by $\mathbb{P}[\cdot]$. The $:=$ symbol means equal by definition. The indicator function $\mathbb{1}[\cdot]$ is equal to one if the condition in the brackets is true and zero if false. $[\cdot]_{i,j}$ is the element in row i and column j of a matrix. $\#(\mathbf{y}=c)$ denotes the number of elements in $\mathbf{y}$ the values of which is equal to c. The ℓ_p-norm of a vector $\mathbf{x}$ is given by $\|\mathbf{x}\|_p$ with an implied subscript for the Euclidean norm. Finally, an M-by-N matrix whose elements are non-negative integers, real, or complex numbers is $\mathbb{Z}_+^{M\times N}$, $\mathbb{R}^{M\times N}$, or $\mathbb{C}^{M\times N}$, respectively. The abbreviations used in this dissertation are summarized in Table 1.1.

Table 1.1: Summary of abbreviations and acronyms

3GPP	3rd Generation Partnership Project	**O-RAN**	Open Radio Access Network
BLER	Block Error Rate	**OFDM**	Orthogonal Frequency Division Multiplexing
BS	Base Station		
CDF	Cumulative Distribution Function	**PHY**	Physical Layer
CoMP	Coordinated Multipoint	**PRB**	Physical Resource Block
CQI	Channel Quality Indicator	**QoE**	Quality of Experience
CSI	Channel State Information	**RAN**	Radio Access Network
DFT	Discrete Fourier Transform	**RL**	Reinforcement Learning
DNN	Deep Neural Network	**ROC**	Receiver Operating Characteristic
DQN	Deep Q-Network	**RRM**	Radio Resource Management
DRL	Deep Reinforcement Learning	**RSRP**	Reference Symbol Received Power
FDD	Frequency Division Duplex	**SGD**	Stochastic Gradient Descent
FPA	Fixed Power Allocation	**SINR**	Signal to Interference plus Noise Ratio
JBPCIC	Joint Beamforming Power Control and Interference Coordination	**SNR**	Signal to Noise Ratio
		SON	Self-Organizing Networks
LOS	Line of Sight	**SVM**	Support Vector Machine
LTE(-A)	Long Term Evolution (-Advanced)	**TTI**	Transmit Time Interval
MAC	Medium Access Control	**UE**	User Equipment
MIMO	Multiple Input Multiple Output	**ULA**	Uniform Linear Array
ML	Machine Learning	**UPA**	Uniform Planar Array
NLOS	Non-Line of Sight	**VoLTE**	Voice over Long Term Evolution
NR	New Radio	**ZF**	Zero-Forcing

1.8 Organization

The remainder of this dissertation is as follows: In Chapter 2, I use reinforcement learning to perform joint beamforming and power control for data bearers and power control and interference coordination for voice users, all of which are PHY layer procedures. Then, in Chapter 3, I show the ability of deep neural networks to learn when to trigger coordinated multipoint and exceed the performance of the algorithmic industry standards. I do so through a surrogate trigger function which I create using a data-driven simulation approach. Next, I move up the air interface protocol stack and extend the band switching procedure across frequency bands to a eliminate measurement gaps required for band switching in cellular networks in Chapter 4. Finally, concluding remarks and suggestions for future work are in Chapter 5.

Chapter 2

Joint Beamforming, Power Control, and Interference Coordination

In Chapter 1, I have introduced deep learning, next-generation wireless networks, the use cases of deep learning in wireless communications, and the problems realizing these use cases in practice. In this chapter[†], I show how disrupting the conventional way power control and interference coordination is done for voice bearers can bring forth enhanced user performance. Further, I show that deep learning can achieve significant execution speedup compared to brute force algorithms without compromising the user performance.

Beamforming focuses a wireless signal in a specific direction. As a result, beamforming can offer better downlink coverage to the users served by a base station supporting it. When the beamforming is applied to a certain user, it is called user-specific beamforming. User-specific beamforming for data bearers, analogous to semi-persistent scheduling for packetized voice bearers, brings about an important benefit: a virtual sense of a *dedicated* channel,

[†]This chapter is based on the work in the following submitted journal paper: F. B. Mismar, B. L. Evans, A. Alkhateeb "Deep Reinforcement Learning for 5G Networks: Joint Beamforming, Power Control, and Interference Coordination," *IEEE Transactions on Communications*, 2019, available through IEEE Early Access. This work was supervised by Prof. Brian L. Evans. Prof. Ahmed Alkhateeb (Arizona State University) provided important ideas about the application of beamforming that greatly improved the work.

where the base station can control the power it is transmitting to that user, by controlling the power of the physical resource blocks that are involved in the beamforming. When the power transmitted to the same resource blocks at the same time belong to a different user, this power becomes interference and needs to be coordinated.

2.1 Overview

In this chapter, I formulate the joint design of beamforming, power control, and interference coordination as a non-convex optimization problem to maximize the signal to interference plus noise ratio (SINR) and solve this problem using deep reinforcement learning. By using the greedy nature of deep Q-learning (defined in Section 1.2.2) to estimate future rewards of actions and using the reported coordinates of the users served by the network, I propose an algorithm for voice bearers and data bearers in sub-6 GHz and millimeter wave (mmWave) frequency bands, respectively. The algorithm improves the performance measured by SINR and sum-rate capacity. In realistic cellular environments, the simulation results show that my algorithm outperforms the link adaptation industry standards for sub-6 GHz voice bearers. For data bearers in the mmWave frequency band, my algorithm approaches the maximum sum rate capacity, but with less than 4% of the required run time.

2.2 Introduction

The massive growth in traffic volume and data rate continues to evolve with the introduction of the fifth generation of wireless communications (5G). Also evolving is enhanced voice call quality with better reliability and improved codecs. Future wireless networks are therefore expected to meet this massive demand for both the data rates and the enhanced voice quality. In an attempt to learn the characteristics of inter-cellular interference and inter-beam interference, I propose an online learning based algorithm based on a reinforcement learning (RL) framework. I use this framework to derive a policy to maximize the end-user SINR and sum-rate capacity. The importance of reinforcement learning in power control has been demonstrated in [32–34]. Power control in voice bearers makes them more robust against wireless impairments, such as fading. It also enhances the usability of the network and increases the cellular capacity. For data bearers, beamforming, power control, and interference coordination, can improve the robustness of these data bearers, improve the data rates received by the end-users, and avoid retransmissions.

A major question here is whether there exists a method that (1) can jointly solve for the power control, interference coordination, and beamforming, (2) achieve the upper limit on SINR, and (3) avoids the exhaustive search in the action space for both bearer types. The aim of this chapter is to propose an algorithm for this joint solution by utilizing the ability of reinforcement learning to explore the solution space by learning from interactions. This algorithm applies to both voice and data bearers alike. Furthermore, I study the

overhead introduced as a result of passing information to a central location, which computes the solution through online learning.

2.2.1 Related Work

Performing power control and beamforming in both uplink and downlink was studied in [35–38]. A jointly optimal transmit power and beamforming vector was solved for in [38] to maximize the SINR using optimization, but without regards for scattering or shadowing, which are critical phenomena in millimeter wave (mmWave) propagation.

The industry standards adopted the method of almost blank subframe (ABS) to resolve the co-channel inter-cell interference problem in LTE where two base stations (BSs) interfere with one another [27]. While ABS works well in fixed beam antenna patterns, the dynamic nature of beamforming reduces the usefulness of ABS [39].

An online learning algorithm for link adaptation in multiple-input multiple-output (MIMO) bearers was studied in [32]. The algorithm computational complexity was comparable to existing online learning approaches, but with minimal spatial overhead. Interference avoidance in a heterogeneous network was studied in [34]. A Q-learning framework for the coexistence of both macro and femto BSs was proposed. The feasibility of decentralized self-organization of these BSs was established where the femtocells interference towards the macro BSs was mitigated. The use of a Q-learning framework was also proposed in [33]. The framework focused on packetized voice power control in a

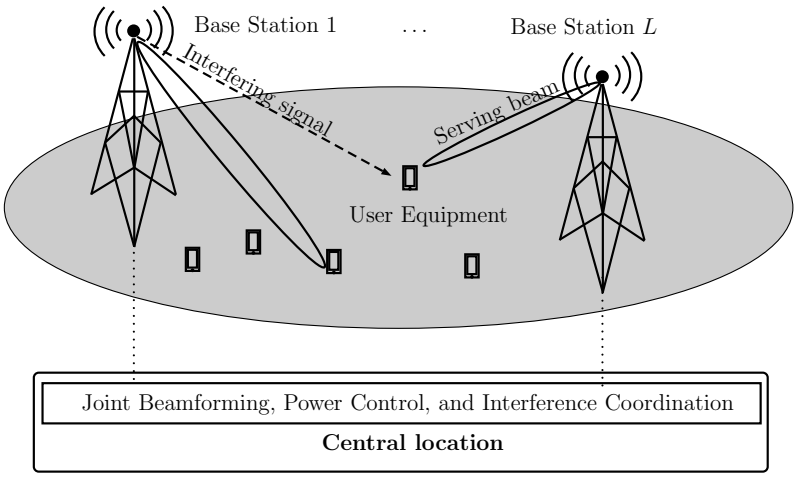


Figure 2.1: Performing joint beamforming and power control on the signal from the serving base station while coordinating interference from the other BS. The decisions are computed at a central location, which can be one of the L BSs. The measurements from the UEs are relayed to the central location over the backhaul.

multi-cell indoors environment. It exploits the use of semi-persistent scheduling [40], which establishes a virtual sense of a dedicated channel. This channel enabled the power control of the downlink to ensure enhanced voice clarity compared to industry standards, which are based on fixed power allocation.

Joint power control in massive MIMO was introduced in [35]. This approach led to a reduced overhead due to a limited exchange of channel state information between the BSs participating in the joint power control. The joint power control scheme led to enhanced performance measured by the SINR. In the uplink direction, power control with beamforming was studied in [36]. An optimization problem was formulated to maximize the achievable sum rate of the two users while ensuring a minimal rate constraint for each user. Using reinforcement learning to solve the problem for the uplink is computationally

expensive and can cause a faster depletion of the user equipment (UE) battery. I on the other hand focus on the downlink and on interference cancellation alongside power control and beamforming.

Over the last two years, the use of deep learning in wireless communications was studied in [37,41–47]. The specific use of deep reinforcement learning to perform power control for mmWave was studied in [37]. This approach was proposed as an alternative to beamforming in improving the non-line of sight (NLOS) transmission performance. The power allocation problem to maximize the sum-rate of UEs under the constraints of transmission power and quality targets was solved using deep reinforcement learning. In this solution, a convolutional neural network was used to estimate the Q-function of the deep reinforcement learning problem. In [41], a policy that maximizes the successful transmissions in a dynamic correlated multichannel access environment was obtained using deep Q-learning. The use of deep convolutional neural networks was proposed in [42] to enhance the automatic recognition of modulation in cognitive radios at low SINRs.

In [46], deep neural networks were leveraged to predict mmWave beams with low training overhead using the omni-directional received signals collected from neighboring base stations. In [47], the authors generalized [46] by mapping the channel knowledge at a small number of antennas to an SINR-optimal beamforming vector for a larger array, even if this array was at a different frequency at a neighboring BS. The use of adversarial reinforcement learning in beamforming for data bearers was proposed in [48], where an algorithm to

Table 2.1: Literature comparison

Reference	Bearer	Band	Objective	Procedure*	Algorithm†
[35]	data	unspecified	downlink SINR	PC	convex optimization
[36]	data	mmWave	uplink sum-rate	BF, PC	convex optimization
[37]	data	mmWave	dowlink SINR, sum-rate	PC	DRL
[43]	data	unspecified	uplink power, sum-rate	PC	DNN
[44]	data	unspecified	downlink throughput	PC	DNN
[45]	data	unspecified	SINR, spectral efficiency	PC	CNN
[46]	data	mmWave	downlink achievable rate	BF	DNN
[47]	data	mmWave and sub-6	downlink spectral efficiency	BF	DNN
[48]	data	unspecified	downlink sum-rate	BF	adversarial DRL
[49]	voice	sub-6	downlink SINR	PC	tabular RL
[50]	data	mmWave	downlink sum-rate	BF, IC	DNN
[51]	data	unspecified	downlink SINR	BF	DNN
Proposed	voice and data	mmWave and sub-6	downlink SINR	BF, PC, IC	DRL

* PC is power control, IC is interference coordination, and BF is beamforming.
† DRL is deep reinforcement learning, CNN is convolutional neural networks, DNN is deep neural networks.

derive antenna diagrams with near-optimal SINR performance was devised. There was no reference to power control or interference coordination. Voice bearers in the sub-6 GHz frequency band was studied in [49] but only in a single co-located BS environment, in contrast with this chapter where I study voice in a multi-access network with multiple BSs. Joint beamforming and interference coordination at mmWave was performed in [50] using deep neural networks, which require knowledge of the channel to make decisions. The performance of deep neural networks on beamforming was studied in [51] but without the use of reinforcement learning. Table 2.1 shows how my work compares with earlier work.

2.2.2 Motivation

In this chapter, I provide an answer to the question whether a method exists that can perform the joint beamforming, power control, and interference coordination by introducing a different approach to power control in wireless networks. In such a setting, it is not only the transmit power of the serving BS that is controlled as in standard implementations, but also the transmit powers of the interfering base stations from a central location as shown in Fig. 2.1. As a result of this apparent conflict, a *race condition* emerges, where the serving BS of a given user is an interfering BS of another user. The reason why I choose deep reinforcement learning (DRL) is as follows:

1. The proposed solution does not require the knowledge of the channels in order to find the SINR-optimal beamforming vector. This is in contrast with the upper limit SINR performance, which finds the optimal beamforming vector by searching across all the beams in a codebook that maximizes the SINR (and this requires perfect knowledge of the channel).

2. The proposed solution minimizes the involvement of the UE in sending feedback to the BS. In particular, the UE sends back its received SINR along with its coordinates, while the agent handles the power control and interference coordination commands to the involved BSs. Industry specifications [27] require that the UE reports its channel state information which is either a vector of length equal to the number of antenna

elements or a matrix of dimension equal to the number of antenna elements in each direction. In my case, I achieve a reduction in the reporting overhead by using the UE coordinates instead.

3. The implementation complexity of brute force SINR performance message passing for joint beamforming, power control, and interference coordination commands when multiple BSs are involved is prohibitive.

4. Having explicit power control and interference coordination (PCIC) commands sent by the UE to the serving and interfering BSs requires a modification to the current industry standards [40]. These standards today only require the serving BS to send power control commands to the UE for the uplink direction.

2.2.3 Contributions

In finding a different approach to power control in wireless networks, this chapter makes the following specific contributions:

- Formulate the joint beamforming, power control, and interference coordination problem in the downlink direction as an optimization problem that optimizes the users' received SINR.

- Resolve the race condition between the involved base stations in sub-exponential times in the number of antennas. The race condition is handled by a central location (similar to coordinated multipoint [52]) based on the user reported downlink SINR and coordinates.

- Show how to create a deep reinforcement learning based solution where multiple actions can be taken at once using a binary encoding of the relevant actions performed by the BS, which I define in Section 2.9.1.

2.3 Network, System, and Channel Models

In this section, I describe the adopted network, system, and channel models.

2.3.1 Network Model

I consider an orthogonal frequency division multiplexing (OFDM) multi-access downlink cellular network of L BSs. This network is comprised of a serving BS and at least one interfering BS. I adopt a downlink scenario, where a BS is transmitting to one UE. The BSs have an inter-site distance of R and the UEs are randomly scattered in their service area. The association between the users and their serving BS is based on the distance between them. A user is served by one BS maximum. The cell radius is $r > R/2$ to allow overlapping of coverage. Voice bearers run on sub-6 GHz frequency bands while the data bearers use mmWave frequency band. I employ analog beamforming for the data bearers to compensate for the high propagation loss due to the higher center frequency.

2.3.2 System Model

Considering the network model in Section 2.3.1, and adopting a multi-antenna setup where each BS employs a uniform linear array (ULA) of M antennas and the UEs have single antennas, the received signal at the UE from the ℓ-th BS can be written as

$$y_\ell = \mathbf{h}_{\ell,\ell}^* \mathbf{f}_\ell x_\ell + \sum_{b \neq \ell} \mathbf{h}_{\ell,b}^* \mathbf{f}_b x_b + n_\ell \tag{2.1}$$

where $x_\ell, x_b \in \mathbb{C}$ are the transmitted signals from the ℓ-th and b-th BSs, and they satisfy the power constraint $\mathbb{E}[|x_\ell|^2] = P_{\text{TX},\ell}$ (similarly for b). The $M \times 1$ vectors $\mathbf{f}_\ell, \mathbf{f}_b \in \mathbb{C}^{M \times 1}$ denote the adopted downlink beamforming vectors at the ℓ-th and b-th BSs, while the $M \times 1$ vectors $\mathbf{h}_{\ell,\ell}, \mathbf{h}_{\ell,b} \in \mathbb{C}^{M \times 1}$ are the channel vectors connecting the user at the ℓ-th BS with the ℓ-th and b-th BSs, respectively. Finally, $n_\ell \sim \text{Normal}(0, \sigma_n^2)$ is the received noise at the user sampled from a complex Normal distribution with zero-mean and variance σ_n^2. The first term in (2.1) represents the desired received signal, while the second term represents the interference received at the user due to the transmission from the other BSs.

Beamforming vectors: Given the hardware constraints on the mmWave transceivers, I assume that the BSs use analog-only beamforming vectors, where the beamforming weights of every beamforming vector $\mathbf{f}_\ell, \ell = 1, 2, ..., L$ are implemented using constant-modulus phase shifters, i.e., $[\mathbf{f}_\ell]_m = e^{j\theta_m}$. Further, I assume that every beamforming vector is selected from a beamsteering-based beamforming codebook $\mathcal{F}$ of cardinality $|\mathcal{F}| := N_{\text{CB}}$, with the n-th

element in this codebook defined as

$$\begin{aligned}\mathbf{f}_n &:= \mathbf{a}(\theta_n) \\ &= \frac{1}{\sqrt{M}} \left[1, e^{jkd\cos(\theta_n)}, ..., e^{jkd(M-1)\cos(\theta_n)}\right]^\top,\end{aligned} \tag{2.2}$$

where d and k denote the antenna spacing and the wave-number, while θ_n represents the steering angle. Finally, $\mathbf{a}(\theta_n)$ is the array steering vector in the direction of θ_n. The value of θ_n is obtained by dividing the antenna angular space between 0 and π radians by the number of antennas M.

Power control and interference coordination: Every BS ℓ is assumed to have a transmit power $P_{\text{TX},\ell} \in \mathcal{P}$, where $\mathcal{P}$ is the set of candidate transmit powers. I define the set of the transmit powers as the power offset above (or below) the BS transmit power. My choice of the transmit power set $\mathcal{P}$ is provided in Section 2.9.1. This choice of $\mathcal{P}$ follows [40].

Power control and interference coordination take place over a semi-dedicated channel. For voice, this is facilitated through the semi-persistent scheduling, which creates a virtual sense of a dedicated channel as I have mentioned in Section 2.2. For data bearers, the use of beamforming provides a dedicated beam for a given UE, through which power control and interference coordination takes place.

2.3.3 Channel Model

In this chapter, I adopt a narrow-band geometric channel model, which is widely used for analyzing and designing mmWave systems [53–55]. With this geometric model, the downlink channel from a BS b to the user in BS ℓ

can be written as

$$\mathbf{h}_{\ell,b} = \frac{\sqrt{M}}{\rho_{\ell,b}} \sum_{p=1}^{N^p_{\ell,b}} \alpha^p_{\ell,b} \mathbf{a}^* \left(\theta^p_{\ell,b}\right), \tag{2.3}$$

where $\alpha^p_{\ell,b}$ and $\theta^p_{\ell,b}$ are the complex path gain and angle of departure (AoD) of the p-th path, and $\mathbf{a}(\theta^p_{\ell,b})$ is the array response vector associated with the AoD, $\theta^p_{\ell,b}$. Note that $N^p_{\ell,b}$ which denotes the number of channel paths is normally a small number in mmWave channels compared to sub-6 GHz channels [56, 57], which captures the sparsity of the channels in the angular domain. Finally, $\rho_{\ell,b}$, represents the pathloss between BS b and the user served in the area of BS ℓ. Note that the channel model in (2.3) accounts of both the LOS and NLOS cases. For the LOS case, I assume that $N^p_{\ell,b} = 1$.

I define $P_{\text{UE}}[t]$ as the received downlink power as measured by the UE over a set of physical resource blocks (PRBs) at a given time t as

$$P^{\ell,b}_{\text{UE}}[t] = P_{\text{TX},b}[t] \left|\mathbf{h}^*_{\ell,b}[t]\mathbf{f}_b[t]\right|^2 \tag{2.4}$$

where $P_{\text{TX},b}$ is the PRB transmit power from BS b. Next, I compute the received SINR for the UE served in BS ℓ at time step t as follows:

$$\gamma_\ell[t] = \frac{P_{\text{TX},\ell}[t]|\mathbf{h}^*_{\ell,\ell}[t]\mathbf{f}_\ell[t]|^2}{\sigma_n^2 + \sum_{b\neq\ell} P_{\text{TX},b}[t]|\mathbf{h}^*_{\ell,b}[t]\mathbf{f}_b[t]|^2}. \tag{2.5}$$

This is the received SINR that I will optimize in Sections 2.6 and 2.7.

2.4 Problem Formulation

My objective is to jointly optimize the beamforming vectors and the transmit power at the L BSs to maximize the achievable sum rate of the users. I

formulate the joint beamforming, power control, and interference coordination optimization problem as

$$\begin{aligned} \underset{\substack{P_{\mathrm{TX},j}[t],\ \forall j \\ \mathbf{f}_j[t],\ \forall j}}{\text{maximize}} \quad & \prod_{j\in\{1,2,\ldots,L\}} \gamma_j[t] \\ \text{subject to} \quad & P_{\mathrm{TX},j}[t] \in \mathcal{P}, \qquad \forall j, \\ & \mathbf{f}_j[t] \in \mathcal{F}, \qquad \forall j, \\ & \gamma_j[t] \geq \gamma_{\mathrm{target}}. \end{aligned} \tag{2.6}$$

where γ_{target} denotes the target SINR of the downlink transmission. $\mathcal{P}$ and $\mathcal{F}$ are the sets of candidate transmit powers and beamforming codebook, respectively as stated earlier. This problem is a non-convex optimization problem due to the non-convexity of the first two constraints. The ℓ-th BS attempts to solve this problem to find optimal $P_{\mathrm{TX},\ell}$ and $\mathbf{f}_\ell$ for the UE served by it at time t. I solve this optimization problem at a central location by searching over the space of the Cartesian product of $\mathcal{P} \times \mathcal{F}$. The optimal solution to this problem is found through an exhaustive search over this space (i.e., by brute force). The complexity of this search is known to be exponential in the number of BSs. I discuss this and the overhead of the communication to a central location in Section 2.7.

Next, I provide a brief overview on deep reinforcement learning in Section 2.5 before delving into the proposed algorithm in Sections 2.6 and 2.7.

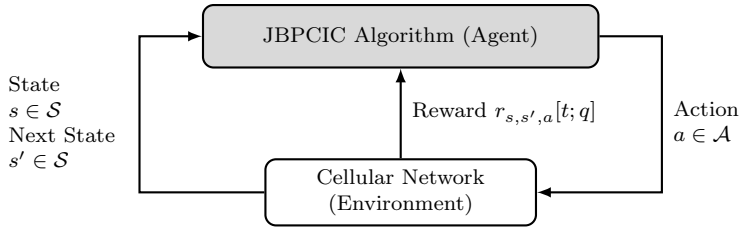


Figure 2.2: The agent-environment interaction in reinforcement learning.

2.5 A Primer on Deep Reinforcement Learning

In this section, I describe deep reinforcement learning (DRL), which is a special type of reinforcement learning [7]. Reinforcement learning (defined in Section 1.1.3) is a machine learning technique that enables an *agent* to discover what action it should take to maximize its expected future *reward* in an interactive *environment*. The interaction between the agent and the environment is shown in Fig. 2.2. In particular, DRL exploits the ability of deep neural networks to learn better representations than handcrafted features and act as a universal approximator of functions. I have described DRL in Section 1.2.2.

Reinforcement learning elements: Reinforcement learning has several elements [4]. These elements interact together, and are as follows:

- *Observations*: Observations are continuous measures of the properties of the environment and are written as a p-ary vector $\mathbf{O} \in \mathbb{R}^p$, where p is the number of properties observed.

- *States*: The state $s_t \in \mathcal{S}$ is the discretization of the observations at time step t. Often, states are also used to mean observations.

- *Actions*: An action $a_t \in \mathcal{A}$ is one of the valid choices that the agent can make at time step t. The action changes the state of the environment from the current state s to the target state s'.

- *Policy*: A policy $\pi(\cdot)$ is a mapping between the state of the environment and the action to be taken by the agent. I define my stochastic policy $\pi(a \mid s) : \mathcal{S} \times \mathcal{A} \to [0, 1]$.

- *Rewards*: The reward signal $r_{s,s',a}[t; q]$ is obtained after the agent takes an action a when it is in state s at time step t and moves to the next state s'. The parameter $q \in \{0, 1\}$ is the bearer selector, which is a binary parameter to differentiate voice bearers from data bearers.

- *State-action value function*: The state-action value function under a given policy π is denoted $Q_\pi(s, a)$. It is the expected discounted reward when starting in state s and selecting an action a under the policy π.

These elements work together and their relationship is governed by the objective to maximize the future discounted reward for every action chosen by the agent, which causes the environment to transition to a new state. The policy dictates the relationship between the agent and the state. The value of the expected discounted reward is learned through the training phase.

If $Q_\pi(s, a)$ is updated every time step, then it is expected to converge to the optimal state-action value function $Q^\star_\pi(s, a)$ as $t \to +\infty$ [4]. However, this may not be easily achieved. Therefore, I use a function approximator instead

aligned with [7]. I define a neural network with its weights at time step t as $\boldsymbol{\Theta}_t \in \mathbb{R}^{u \times v}$ as in Fig. 2.3. Also, if I define $\boldsymbol{\theta}_t := \mathrm{vec}\,(\boldsymbol{\Theta}_t) \in \mathbb{R}^{uv}$, I thus build a function approximator $Q_\pi(s, a; \boldsymbol{\theta}_t) \approx Q^\star_\pi(s, a)$. This function approximator is neural network based and is known as the Deep Q-Network (DQN) [7]. Activation functions, which are non-linear functions that compute the hidden layer values, are an important component of neural networks. A common choice of the activation function is the sigmoid function $\sigma\colon x \mapsto 1/(1+e^{-x})$ [2]. This DQN is trained through adjusting $\boldsymbol{\theta}$ at every time step t to reduce the mean-squared error loss $L_t(\boldsymbol{\theta}_t)$:

$$\underset{\boldsymbol{\theta}_t}{\text{minimize}} \qquad L_t(\boldsymbol{\theta}_t) := \mathbb{E}_{s,a}\left[(y_t - Q_\pi(s, a; \boldsymbol{\theta}_t))^2\right] \tag{2.7}$$

where $y_t := \mathbb{E}_{s'}[r_{s,s',a} + \gamma \max_{a'} Q_\pi(s', a'; \boldsymbol{\theta}_{t-1}) \mid s_t, a_t]$ is the estimated function value at time step t when the current state and action are s and a respectively. The process of interacting with the environment and the DQN to obtain a prediction and compare it with the true answer and suffer a loss $L_t(\cdot)$ is often referred to as "online learning." In online learning, the UEs feedback their data to the serving BS, which in turns relays it to the central location for DQN training. This data represent the state of my network environment $\mathcal{S}$, as I explain further in Section 2.9.

DQN dimension: I set the dimension of the input layer in the DQN to be equal to the number of states $|\mathcal{S}|$. The dimension of the output layer is equal to the number of actions $|\mathcal{A}|$. For the hidden layer dimension, I choose a small depth since the depth has the greatest impact on the computational

complexity. The dimension of the width follows [58] as I show further in Section 2.9.1.

Deep reinforcement training phase: In the training phase of the DQN, the weights $\boldsymbol{\theta}_t$ in the DQN are updated after every iteration in time t using the stochastic gradient descent (SGD) algorithm on a minibatch of data. SGD starts with a random initial value of $\boldsymbol{\theta}$ and performs an iterative process to update $\boldsymbol{\theta}$ using a step size $\eta > 0$ as follows:

$$\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \eta \nabla L_t(\boldsymbol{\theta}_t). \tag{2.8}$$

The training of the DQN is facilitated by "experience replay" [59]. The experience replay buffer $\mathcal{D}$ stores the experiences at each time step t. An experience e_t is defined as $e_t := (s_t, a_t, r_{s,s',a}[t;q], s'_t)$. I draw samples of experience at random from this buffer and perform minibatch training on the DQN. This approach offers advantages of stability and avoidance of local minimum convergence [7]. The use of experience replay also justifies the use of off-policy learning algorithms, since the current parameters of the DQN are different from those used to generate the sample from $\mathcal{D}$.

I define the state-action value function estimated by the DQN $Q^\star_\pi(s,a)$ as

$$Q^\star_\pi(s_t, a_t) := \mathbb{E}_{s'}\left[r_{s,s',a} + \gamma \max_{a'} Q^\star_\pi(s', a') \,\middle|\, s_t, a_t \right], \tag{2.9}$$

which is known as the Bellman equation. Here, $\gamma\colon 0 < \gamma < 1$ is the *discount factor* and determines the importance of the predicted future rewards. The

next state is s' and the next action is a'. My goal using DQN is to find a solution to maximize the state-action function $Q^{\star}_{\pi}(s_t, a_t)$.

Often compared with deep Q-learning is the tabular version of Q-learning [4]. Despite the finite size of the states and action space, tabular Q-learning is slow to converge is because its convergence requires the state-action pairs to be sampled infinitely often [4,60]. Further, tabular RL requires a non-trivial initialization of the $\mathbf{Q} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$ table to avoid longer convergence times [49]. However, deep Q-learning convergence is not guaranteed when using a non-linear approximator such as the DQN [7]. I discuss tabular Q-learning in Section 2.6.

Policy selection: In general, Q-learning is an off-policy reinforcement learning algorithm. An off-policy algorithm means that a policy can be found even when actions are selected according to an arbitrary exploratory policy [4]. Due to this, I choose a near-greedy action selection policy. This policy has two modes:

1. *exploration*: the agent tries different actions at random at every time step t to discover an effective action a_t.

2. *exploitation*: the agent chooses an action at time step t that maximizes the state-action value function $Q_{\pi}(s, a; \boldsymbol{\theta}_t)$ based on the previous experience.

In this policy, the agent performs exploration with a probability ϵ and exploitation with probability of $1 - \epsilon$, where ϵ: $0 < \epsilon < 1$ is a hyperparameter

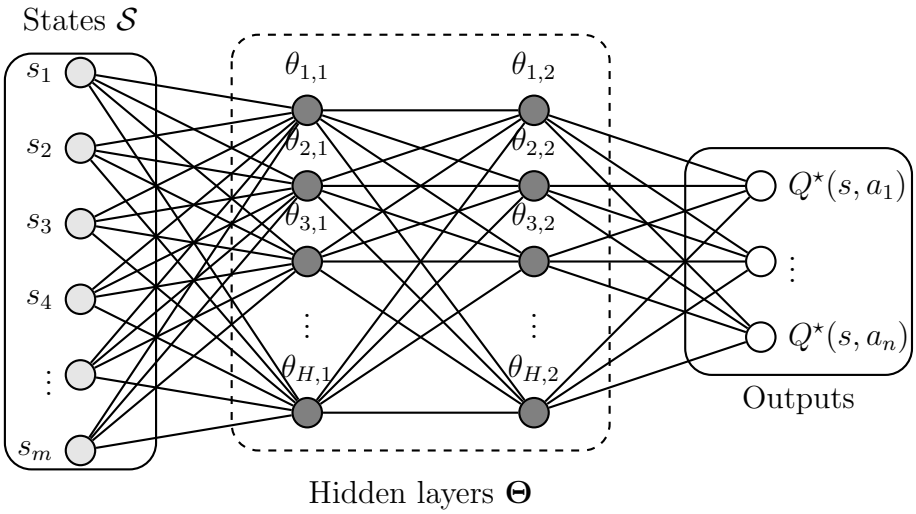


Figure 2.3: Structure of the deep Q-network used for the implementation of the algorithms with two hidden layers each of dimension H. Here, $(u, v) = (H, 2)$, $|\mathcal{S}| = m$, and $|\mathcal{A}| = n$.

that adjusts the trade-off between exploration and exploitation. This trade-off is why this policy is also called the ϵ-greedy action selection policy. This policy is known to have a linear regret in t (regret is the opportunity loss of one time step) [61].

At each time step t, the UEs move at speed v and the agent performs a certain action a_t from its current state s_t. The agent receives a reward $r_{s,s',a}[t; q]$ and moves to a target state $s' := s_{t+1}$. I call the period of time in which an interaction between the agent and the environment takes place an *episode*. One episode has a duration of T time steps. An episode is said to have *converged* if within T time steps the target objective was fulfilled.

In my DQN implementation, I particularly keep track of the UE co-

ordinates. When UE coordinates are reported back to the network and used to make informed decisions, the performance of the network improves [62]. Therefore, UE coordinates need to be part of the DRL state space $\mathcal{S}$.

2.6 Deep Reinforcement Learning in Voice Power Control and Interference Coordination

In this section, I describe my proposed voice power control and interference coordination reinforcement learning algorithm as well as the baseline solutions which I compare my solution against. First, I describe the fixed power allocation algorithm, which is the industry standard algorithm today, and then the implementation of the proposed algorithm using tabular and deep implementations of Q-learning. Finally, I explain the brute force algorithm.

2.6.1 Fixed Power Allocation

I introduce the fixed power allocation (FPA) power control as a baseline algorithm that sets the transmit signal power at a specific value. No interference coordination is implemented in FPA. Total transmit power is simply divided equally among all the PRBs and is therefore constant:

$$P_{\mathrm{TX},b}[t] := P_{\mathrm{BS}}^{\max} - 10\log N_{\mathrm{PRB}} + 10\log N_{\mathrm{PRB},b}[t] \qquad (\mathrm{dBm}). \tag{2.10}$$

Here, N_{PRB} is the total number of physical resource blocks in the BS and $N_{\mathrm{PRB},b}$ is the number of available PRBs to the UE in the b-th BS at the time step t.

FPA with adaptive modulation and coding is the industry standard algorithm [40]. In this standard algorithm, the BS fixes its transmit power and only changes the modulation and code schemes of the transmission. This change is known as the "link adaptation." Link adaptation takes place based on the reports sent by the UE back to the BS (i.e., the SINR and received power). Since the BS transmit power is fixed, the link adaptation takes place based on either periodic or aperiodic measurement feedback from the voice UE to the serving BS. This results in an improved effective SINR and a reduction in the voice packet error rate. There is no measurement sent to the interfering BS based on FPA.

2.6.2 Tabular Reinforcement Learning

I use a tabular setting of Q-learning (or "vanilla" Q-learning) to implement the algorithm for voice communication. In a tabular setting, the state-action value function $Q_\pi(s_t, a_t)$ is represented by a table $\mathbf{Q} \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{A}|}$. There is no neural network involvement and the Q-learning update analog of (2.9) is defined as:

$$Q_\pi(s_t, a_t) := (1-\alpha)Q_\pi(s_t, a_t) + \alpha\left(r_{s,s',a} + \gamma \max_{a'} Q_\pi(s', a')\right) \tag{2.11}$$

where $Q_\pi(s_t, a_t) := [\mathbf{Q}]_{s_t,a_t}$. Here, $\alpha > 0$ is the learning rate of the Q-learning update and defines how aggressive the experience update is with respect to the prior experience. Computationally, the tabular setting suits problems with small state spaces, and maintaining a table $\mathbf{Q}$ is possible.

2.6.3 Proposed Algorithm

I propose Algorithm 1 which is a DRL-based approach. This algorithm performs both power control and interference coordination without the UE sending explicit power control or interference coordination commands. This use of the DQN may provide a lower computational overhead compared to the tabular Q-learning depending on the number of states and the depth of the DQN [49]. The main steps of Algorithm 1 are as follows:

- Select an optimization action at a time step t.
- Select a joint beamforming, power control, and interference coordination action.
- Assess the impact on the effective SINR $\gamma_{\ell,\text{eff}}[t]$.
- Reward the action taken based on the impact on $\gamma_{\ell,\text{eff}}[t]$ and its distance from γ_{target} or γ_{min}.
- Train the DQN based on the outcomes.

Power control for the serving BS b is described as

$$P_{\text{TX},b}[t] = \min(P_{\text{BS}}^{\max}, P_{\text{TX},b}[t-1] + \text{PC}_b[t]). \tag{2.12}$$

I add one more condition for the interference coordination on the interfering BS ℓ as

$$P_{\text{TX},\ell}[t] = \min(P_{\text{BS}}^{\max}, P_{\text{TX},\ell}[t-1] + \text{IC}_\ell[t]) \tag{2.13}$$

where the role of the BS (serving vs interfering) can change based on the UE being served. IC and PC commands are actually the same, but the role of the BS makes one an interferer (which needs coordination) and the other a server (which needs power control). I model the PCIC algorithm using deep Q-learning as shown in Algorithm 1. My proposed algorithm solves (2.6).

Different from [33], I use the effective SINR $\gamma_{\ell,\text{eff}}[t]$ (i.e., the SINR including coding gain) for all three voice algorithms where the adaptive code rate β is chosen based on the SINR $\gamma_\ell[t]$. I use an adaptive multirate (AMR) codec and quadrature phase shift keying modulation for voice. I choose to fix the modulation since voice bearers do not typically require high data rates [33]. This effective SINR $\gamma_{\ell,\text{eff}}[t]$ is the quantity I optimize in Algorithm 1.

For FPA, the run-time complexity is $\mathcal{O}(1)$. For tabular Q-learning PCIC, the run-time complexity is $\mathcal{O}(|\mathcal{S}^{\text{voice}}||\mathcal{A}^{\text{voice}}|)$ [49], where $\mathcal{S}^{\text{voice}}, \mathcal{A}^{\text{voice}}$ are the state and action sets for voice bearers. Deep Q-learning can have a run time orders of magnitude slower than the tabular version.

Since one of the L BSs also serves as a central location to the surrounding BSs in my proposed algorithm, the overhead due to transmission over the backhaul to this central location for a total of N_{UE} UEs in the service area is in $\mathcal{O}(gLN_{\text{UE}})$, where the periodicity g is the number of measurements sent by any given UE during time step t [11].

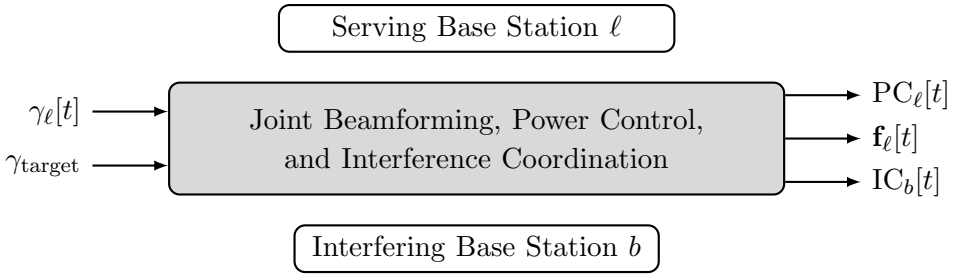


Figure 2.4: Downlink joint beamforming, power control, and interference coordination module.

2.6.4 Brute Force

The brute force PCIC algorithm uses an exhaustive search in the Euclidean space $\mathcal{P}$ per BS to optimize the SINR. This algorithm solves (2.6) and is the upper limit of the performance for jointly optimizing the SINR for the voice bearers in my problem.

2.7 Deep Reinforcement Learning in mmWave Beamforming Power Control and Interference Coordination

In this section, I present my proposed algorithm and quantify the changes in the SINR as a result of the movement of the UEs and optimization actions of the RL-based algorithm.

2.7.1 Proposed Algorithm

I propose a DRL-based algorithm where the beamforming vectors and transmit powers at the base stations are jointly controlled to maximize the

Algorithm 1: Deep Reinforcement Learning in Joint Beamforming and PCIC (JBPCIC)

Input: The downlink received SINR measured by the UEs.
Output: Sequence of beamforming, power control, and interference coordination commands to solve (2.6).

Initialize time, states, actions, and replay buffer $\mathcal{D}$.
repeat
  repeat
    $t := t + 1$
    Observe current state s_t.
    $\epsilon := \max(\epsilon \cdot d, \epsilon_{\min})$
    Sample $r \sim \text{Uniform}(0, 1)$
    if $r \leq \epsilon$ **then**
      Select an action $a_t \in \mathcal{A}$ at random.
    else
      Select an action $a_t = \arg\max_{a'} Q_\pi(s_t, a'; \boldsymbol{\theta}_t)$.
    end
    Compute $\gamma_{\ell,\text{eff}}[t]$ and $r_{s,s',a}[t; q]$ from (2.17).
    if $\gamma_{\ell,\text{eff}}[t] < \gamma_{\min}$ **then**
      $r_{s,s',a}[t; q] := r_{\min}$
      Abort episode.
    end
    Observe next state s'.
    Store experience $e[t] \triangleq (s_t, a_t, r_{s,s',a}, s')$ in $\mathcal{D}$.
    Minibatch sample from $\mathcal{D}$ for experience $e_j \triangleq (s_j, a_j, r_j, s_{j+1})$.
    Set $y_j := r_j + \gamma \max_{a'} Q_\pi(s_{j+1}, a'; \boldsymbol{\theta}_t)$
    Perform SGD on $(y_j - Q_\pi(s_j, a_j; \boldsymbol{\theta}_t))^2$ to find $\boldsymbol{\theta}^\star$
    Update $\boldsymbol{\theta}_t := \boldsymbol{\theta}^\star$ in the DQN and record loss L_t
    $s_t := s'$
  until $t \geq T$
until convergence or aborted
if $\gamma_{\ell,\text{eff}}[t] \geq \gamma_{\text{target}}$ **then** $r_{s,s',a}[t; q] := r_{s,s',a}[t; q] + r_{\max}$

objective function in (2.6). The use of a string of bits as an action register enables us to jointly perform several actions concurrently.

First, selecting the beamforming vector is performed as follows. The

agent steps up or down the beamforming codebook using circular increments $(n+1)$ or decrements $(n-1)$

$$n \mapsto \mathbf{f}_n[t] \colon n := (n \pm 1) \bmod M \tag{2.14}$$

for BSs b and ℓ independently. I monitor the change in γ_ℓ as a result of the change in the beamforming vector. I use a code gain of unity in computing $\gamma_{\ell,\text{eff}}$ for the data bearers (i.e., $\gamma_{\ell,\text{eff}} = \gamma_\ell$).

When the beamforming vectors are selected for a given UE, the agent also performs power control of that beam by changing the transmit power of the BS to this UE (or the interference coordination of other BSs). The selection of the transmit power is governed by (2.12) and (2.13), both of which define the set $\mathcal{P}$.

For proposed algorithm, the run time of the deep reinforcement learning is significantly faster than the brute force algorithm for all antenna sizes M as I show in Section 2.9.3. Also, the reporting of the UE coordinates (i.e., longitude and latitude) to the BS instead of the channel state information reduces the reporting overhead from M complex-valued elements to the two real-valued coordinates and its received SINR only. If I assume that the reporting overhead for M complex-valued elements is $2M$, then for reporting the UE coordinates, I achieve an overhead reduction gain of $1 - 1/M$.

I call my algorithm the joint beamforming, power control, and interference coordination (JBPCIC) algorithm.

Table 2.2: Reinforcement learning hyperparameters

Parameter	Value	Parameter	Value
Discount factor γ	0.995	Exploration rate decay d	0.9995
Initial exploration rate ϵ	1.000	Min. exploration rate $(\epsilon_{\text{min}}^{\text{voice}}, \epsilon_{\text{min}}^{\text{bf}})$	(0.15,0.10)
Number of states $\lvert\mathcal{S}\rvert$	8	Number of actions $\lvert\mathcal{A}\rvert$	16
Deep Q-Network width H	24	Deep Q-Network depth	2

2.7.2 Brute Force

The brute force beamforming and PCIC algorithm uses an exhaustive search in the Euclidean space $\mathcal{P} \times \mathcal{F}$ per BS to optimize the SINR. As in the voice bearers brute force algorithm, this is also the upper limit in the performance for jointly optimizing the SINR in my problem. While the size of $\mathcal{P}$ can be selected independently of the number of the antennas in the ULA M, the size of $\mathcal{F}$ is directly related to M. Similar to the brute force algorithm for voice bearers, this algorithm solves (2.6) and may perform well for small M and small number of BSs L for data bearers. However, I observe that with large M the search time becomes prohibitive. This is because the run time for this algorithm in $\mathcal{O}((|\mathcal{P}||\mathcal{F}|)^L) = \mathcal{O}(M^L)$, which is much larger than the run time for the proposed algorithm, as I show in Section 2.9.3.

2.8 Performance Measures

In this section I introduce the performance measures I use to benchmark my algorithms.

Table 2.3: Joint beamforming power control algorithm – radio environment parameters

Parameter	Value
Base station (BS) maximum transmit power $P_{\text{BS}}^{\max}$	46 dBm
Cellular geometry	circular
Propagation model (voice, bf)	(COST231, [63])
Antenna gain $(G_{\text{TX}}^{\text{voice}}, G_{\text{TX}}^{\text{bf}})$	(11, 3) dBi
Max. number of UEs per BS N	10
Probability of LOS $(p_{\text{LOS}}^{\text{voice}}, p_{\text{LOS}}^{\text{bf}})$	(0.9, 0.8)
Number of transmit antennas $(M^{\text{voice}}, M^{\text{bf}})$	$(1,\{4, 8, 16, 32, 64\})$
Downlink frequency band	(2100 MHz, 28 GHz)
Cell radius r	(350, 150) m
User equipment (UE) antenna gain	0 dBi
Inter-site distance R	(525, 225) m
Number of multipaths N_p	(15, 4)
UE average movement speed v	(5, 2) km/h
Radio frame duration $(T^{\text{voice}}, T^{\text{bf}})$	(20, 10) ms

2.8.1 Convergence

I define convergence ζ in terms of the episode at which the target SINR is fulfilled over the entire duration of T for all UEs in the network. I expect that as the number of antennas in the ULA M increase, the convergence time ζ will also increase. In voice, convergence as a function of M is not applicable, since I only use single antennas. For several random seeds, I take the aggregated percentile convergence episode.

2.8.2 Run time

While calculating the upper bound of the brute force algorithm runtime complexity is possible, obtaining a similar expression for the proposed deep Q-learning algorithm may be challenging due to lack of convergence and

stability guarantees [7]. Therefore, I obtain the run time from simulation per antenna size M.

2.8.3 Coverage

I build a complement cumulative distribution function (CCDF) of $\gamma_{\ell,\text{eff}}$ following [64] by running the simulation many times and changing the random seed, effectively changing the way the users are dropped in the network.

2.8.4 Sum-Rate Capacity

Using the effective SINRs, I compute the average sum-rate capacity as

$$C = \frac{1}{T} \sum_{t=1}^{T} \sum_{j \in \{\ell, b\}} \log_2(1 + \gamma_{j,\text{eff}}[t]) \tag{2.15}$$

which is an indication of the data rate served by the network. I then obtain the maximum sum-rate capacity resulting from computing (2.15) over many episodes.

2.9 Simulation Results

In this section, I evaluate the performance of my RL-based proposed solutions in terms of the performance measures in Section 2.8. First, I describe the adopted setup in Section 2.9.1 before delving into the simulation results in Sections 2.9.2 and 2.9.3.

2.9.1 Setup

I adopt the network, signal, and channel models in Section 2.3. The users in the urban cellular environment are uniformly distributed in its cov-

erage area. The users are moving at a speed v with both log-normal shadow fading and small-scale fading. The cell radius is r and the inter-site distance $R = 1.5r$. For the voice bearer, I set the adaptive code rate β between 1:3 to 1:1 based on reported SINR and use an AMR voice codec bitrate of 23.85 kbps and a voice activity factor $\nu = 0.8$. The users experience a probability of line of sight of p_{LOS}. The bandwidth of both the voice and data bearers is equal to one resource block. The rest of the parameters are shown in Table 2.3. I set the target effective SINRs as:

$$\begin{aligned} \gamma_{\text{target}}^{\text{voice}} &:= 3\,\text{dB}, \\ \gamma_{\text{target}}^{\text{bf}} &:= \gamma_0^{\text{bf}} + 10\log M\,\text{dB} \end{aligned} \tag{2.16}$$

where γ_0^{bf} is a constant threshold (i.e., not dependent on the antenna size). I set the minimum SINR at -3 dB below which the episode is declared aborted and the session is unable to continue (i.e., dropped).

The hyperparameters required to tune the RL-based model are shown in Table 2.2. I refer to my source code [65] for further implementation details. Further, I run Algorithm 1 on the cellular network with its parameters in Table 2.3. The simulated states $\mathcal{S}$ are setup as:

$$\begin{aligned} (s_t^0, s_t^1) &:= \text{UE}_\ell(x[t], y[t]), \qquad (s_t^2, s_t^3) := \text{UE}_b(x[t], y[t]), \\ s_t^4 &:= P_{\text{TX},\ell}[t], \qquad s_t^5 := P_{\text{TX},b}[t], \\ s_t^6 &:= \mathbf{f}_\ell[t], \qquad s_t^7 := \mathbf{f}_b[t], \end{aligned}$$

where (x, y) are the Cartesian coordinates (i.e., longitude and latitude) of the given UE.

To derive the actions $\mathcal{A}$, I exploit the fact that $\mathcal{F}$ and $\mathcal{P}$ each has a cardinality that is a power of two. This enables us to construct the binary encoding of the actions using a string of bits in a register $\mathbf{a}$ as shown in Fig. 2.5. With bitwise-AND, masks, and shifting, the joint beamforming, power control, and interference coordination commands can be derived. I choose the following code:

1. When $q = 0$:
    - $\mathbf{a}_{[0,1]} = 00$: decrease the transmit power of BS b by 3 dB.
    - $\mathbf{a}_{[0,1]} = 01$: decrease the transmit power of BS b by 1 dB.
    - $\mathbf{a}_{[0,1]} = 10$: increase the transmit power of BS b by 1 dB.
    - $\mathbf{a}_{[0,1]} = 11$: increase the transmit power of BS b by 3 dB.
    - $\mathbf{a}_{[2,3]} = 00$: decrease the transmit power of BS ℓ by 3 dB.
    - $\mathbf{a}_{[2,3]} = 01$: decrease the transmit power of BS ℓ by 1 dB.
    - $\mathbf{a}_{[2,3]} = 10$: increase the transmit power of BS ℓ by 1 dB.
    - $\mathbf{a}_{[2,3]} = 11$: increase the transmit power of BS ℓ by 3 dB.
2. When $q = 1$:
    - $\mathbf{a}_{[0]} = 0$: decrease the transmit power of BS b by 1 dB.
    - $\mathbf{a}_{[0]} = 1$: increase the transmit power of BS b by 1 dB.
    - $\mathbf{a}_{[1]} = 0$: decrease the transmit power of BS ℓ by 1 dB.
    - $\mathbf{a}_{[1]} = 1$: increase the transmit power of BS ℓ by 1 dB.

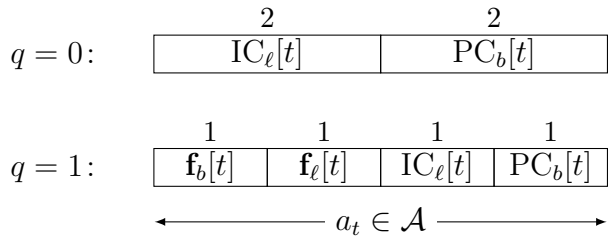


Figure 2.5: Binary encoding of joint beamforming, power control, and interference coordination actions using a string of bits in a register $\mathbf{a}$ for different bearer types ($q = 0$ for voice bearers and $q = 1$ for data bearers).

- $\mathbf{a}_{[2]} = 0$: step down the beamforming codebook index of BS ℓ.
- $\mathbf{a}_{[2]} = 1$: step up the beamforming codebook index of BS ℓ.
- $\mathbf{a}_{[3]} = 0$: step down the beamforming codebook index of BS b.
- $\mathbf{a}_{[3]} = 1$: step up the beamforming codebook index of BS b.

Here, I can infer that $\mathcal{P} = \{\pm 1, \pm 3\}$ dB offset from the transmit power. The choice of these values is motivated by 1) aligning with industry standards [40] which choose integers for power increments and 2) maintaining the non-convexity of the problem formulation (2.6) by keeping the constraints discrete. The actions to increase and decrease BS transmit powers are implemented as in (2.12) and (2.13). I introduce 3-dB power steps for voice only to compensate for not using beamforming, which is aligned with the industry standards of not having beamforming for packetized voice bearers [40].

The reward I use in my proposed algorithms is divided into two tiers: 1) based on the relevance of the action taken and 2) based on whether the target SINR has been met or the SINR falls below the minimum. I start with

defining a function $\mathrm{p}(\cdot)$ which returns one of the elements of $\mathcal{P}$ based on the chosen code. Here, $\mathrm{p}(00) = -3, \mathrm{p}(01) = -1, \mathrm{p}(10) = 1, \mathrm{p}(11) = 3$. Next, I write the received SINR due to the aforementioned encoded actions in $\mathbf{a}$ as $\gamma_b^{\mathbf{a}_{[0]},\mathbf{a}_{[3]}}$ and $\gamma_\ell^{\mathbf{a}_{[1]},\mathbf{a}_{[2]}}$ for BSs b and ℓ, respectively.

I further write the joint reward for both voice and data bearers as follows:

$$r_{s,s',a}[t;q] := \Big(\mathrm{p}(\mathbf{a}_{[0,1]}[t]) - \mathrm{p}(\mathbf{a}_{[2,3]}[t])\Big)(1-q) + \Big(\gamma_b^{\mathbf{a}_{[0]}[t],\mathbf{a}_{[3]}[t]} + \gamma_\ell^{\mathbf{a}_{[1]}[t],\mathbf{a}_{[2]}[t]}\Big)q \tag{2.17}$$

where $q = 0$ for voice bearers and $q = 1$ for data bearers. I reward the agent the most per time step when a joint power control and beamforming action is taken for data bearers and when a joint power control and interference coordination takes place for a voice bearer. I abort the episode if any of the constraints in (2.6) becomes inactive. At this stage, the RL agent receives a reward $r_{s,s',a}[t;q] := r_{\min}$. Either a penalty $r_{\min}$ or a maximum reward $r_{\max}$ is added based on whether the minimum $\gamma_{\min}$ has been violated or γ_{target} has been achieved as shown in Algorithm 1. Here, it is also clear that for data bearers the agent is rewarded more for searching in the beamforming codebook than attempting to power up or down. However, for voice bearers, I reward the agent more if it chooses to power control the serving BS b than if it chooses to control the interference from the other BS ℓ.

In my simulations, I use a minibatch sample size of $N_{\text{mb}} = 32$ training examples. With $|\mathcal{A}| = 16$, the width of the DQN can be found using [58] to be $H = \sqrt{(|\mathcal{A}| + 2)N_{\text{mb}}} = 24$. I refer to my code [65] for details.

2.9.2 Outcomes

1. Convergence: I study the normalized convergence under (2.16) where $\gamma_0^{\text{bf}} = 5$ dB. Every time step in an episode is equal to one radio subframe, the duration of which is 1 ms [11]. During this time the UE is likely to be using a sub-optimal selection of beam obtained from a prior iteration. This would cause the UE throughput to degrade by a factor as I show in Section 2.9.3. As the size of the ULA M increases, the number of episodes required converge increases with minimal effect of the constant threshold γ_0^{bf} since $M \gg \gamma_0^{\text{bf}}$. This is justified since the number of attempts to traverse the beamforming codebook increases almost linearly with the increase of M.

2. Run time: I study the normalized run time and observe that as the number of antennas M increase, so does the run-time complexity for the proposed algorithm. This is justified due to the increase in the number of beams required for the algorithm to search through to increase the joint SINR.

3. Coverage: for voice bearers I observe that the coverage as defined by the SINR CCDF improves everywhere. For data bearers, the coverage improves where the SINR monotonically increases with the increase in M which is expected because the beamforming array gain increases with an increase in M.

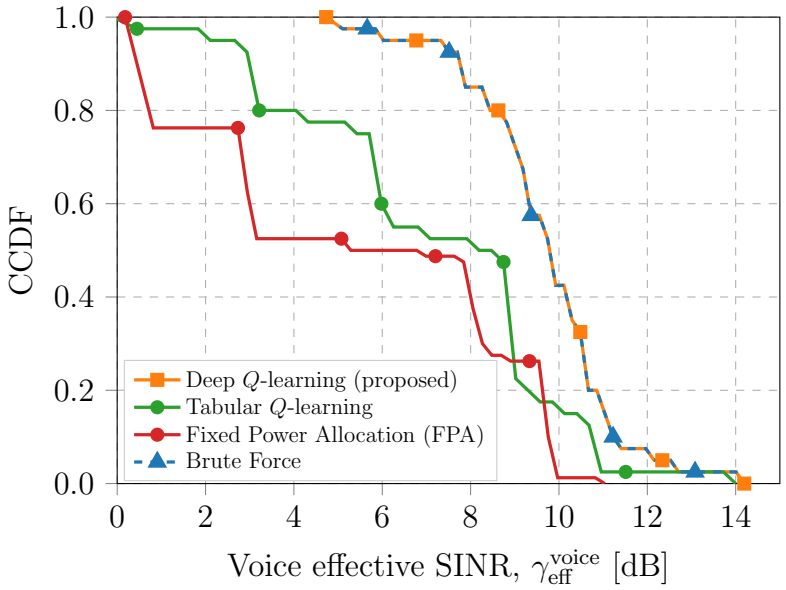


Figure 2.6: Coverage CCDF plot of $\gamma_{\text{eff}}^{\text{voice}}$ for different voice power control and interference coordination algorithms.

4. Sum-rate: the sum-rate capacity increases logarithmically as a result of the increase of M, which is justified using (2.5) and (2.15).

2.9.3 Figures

Fig. 2.6 shows the CCDF of the effective SINR γ_{eff} for the voice PCIC algorithms all for the same episode. This episode generates the highest reward. Here I see that the FPA algorithm has the worst performance especially at the cell edge (i.e., low effective SINR regime), which is expected since FPA has no power control or interference coordination. The tabular implementation of my proposed algorithm has better performance compared with the FPA. This is since power control and interference coordination are introduced to the base

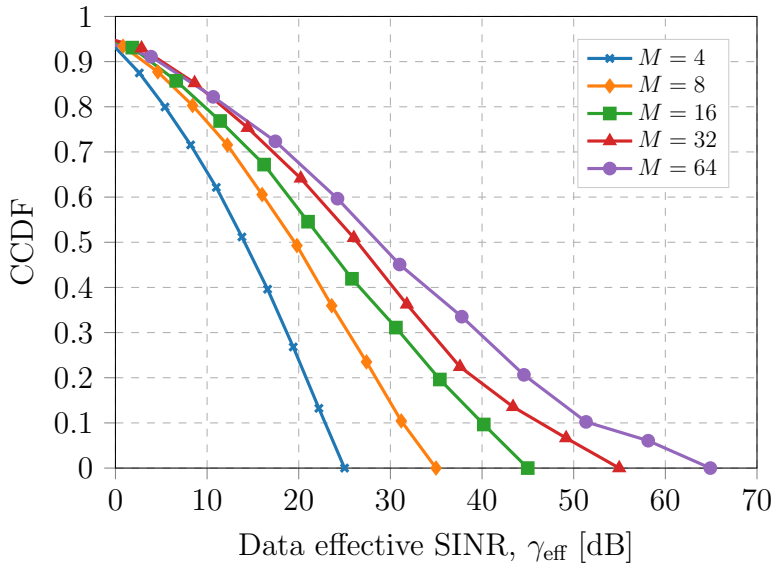


Figure 2.7: Coverage CCDF plot of the effective SINR γ_{eff} for the proposed deep Q-learning algorithm vs. the number of antennas M.

stations, though not as effectively, which explains why close to $\gamma_{\text{eff}} = 9$ dB tabular Q-learning PCIC underperforms FPA. Further, I observe that deep Q-learning outperforms the tabular Q-learning implementation of the PCIC algorithm, since deep Q-learning has resulted in a higher reward compared to tabular Q-learning. This is because deep Q-learning has converged at a better solution (identical to the solution obtained through brute force), unlike the tabular Q-learning the convergence of which may have been impeded by the choice of a initialization of the state-action value function. However, as the effective SINR γ_{eff} approaches 13 dB, the users are close to the BS center and therefore all power control algorithms perform almost similarly thereafter.

I show the coverage CCDF in Fig. 2.7. As M increases, so does the

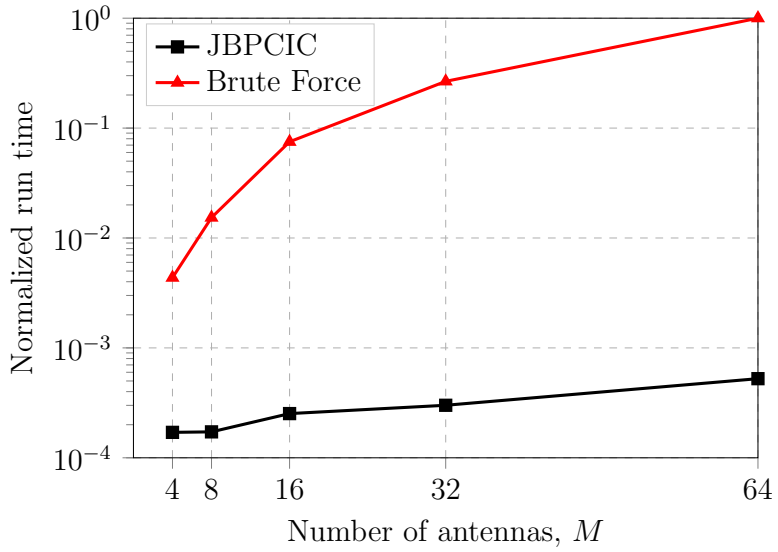


Figure 2.8: The normalized run times for the proposed deep Q-learning algorithm as a function of the number of antennas M.

probability of achieving a given effective SINR, since the effective SINR depends on the beamforming array gain which is a function of M as stated earlier. The improvement in the run time is shown in Fig. 2.8. The brute force algorithm has a significantly larger run time compared to the proposed algorithm. The run time increases as the number of antennas M increase, though much steeper in the brute force algorithm, due to the exponential nature of the run-time complexity. At $M = 4$, only 4% of the run time of the brute force algorithm was needed for my proposed algorithm. In Fig. 2.9, at smaller ULA sizes M, the impact of the constant threshold γ_0^{bf} becomes dominant and it takes almost similar times to converge for values of M. This is likely to be due to the wider beams in the grid of beams, which are able to cover the UEs moving at speeds v. However, for the large antenna size regime, as the size

of the ULA M increases, the number of episodes required converge increases with minimal effect of γ_0^{bf} as I explained earlier. This is due to the longer time required for the agent to search through a grid of beams of size $|\mathcal{F}|$, which are typically narrower at large M. This causes the agent to spend longer time to meet the target SINR. This time or delay can have a negative impact on the throughput and voice frames of the data and voice bearers respectively. If I assume the data bearer transmits b bits over a total duration of T^{bf} for beamformed data bearers, then the impact of the convergence time would cause these b bits to be transmitted over a duration of $T^{\text{bf}}\zeta$. The throughput due to convergence then becomes $b/T^{\text{bf}}\zeta$. For voice, the number of lost voice frames due to this convergence time is $\lceil \nu\zeta \rceil$.

The achieved SINR is proportional to the ULA antenna size M as shown in Fig. 2.10. This is expected as the beamforming array gain is $\|\mathbf{f}_b\|^2 \leq M$. The transmit power is almost equal to the maximum. Fig. 2.10 also shows the relative performance of JBPCIC compared with the upper limit of performance outlined in Section 2.7.2. I observe that the performance gap of both the transmit power of the base stations and the SINR is almost diminished all across M. This is because of the DQN ability to estimate the function that leads to the upper limit of the performance. Further, I observe that the solution for the race condition is for both BSs to transmit at maximum power.

Finally, Fig. 2.11 shows the sum-rate capacity of both the JBPCIC algorithm and the upper limit of performance. Similarly, the performance gap diminishes across all M for the same reason discussed earlier.

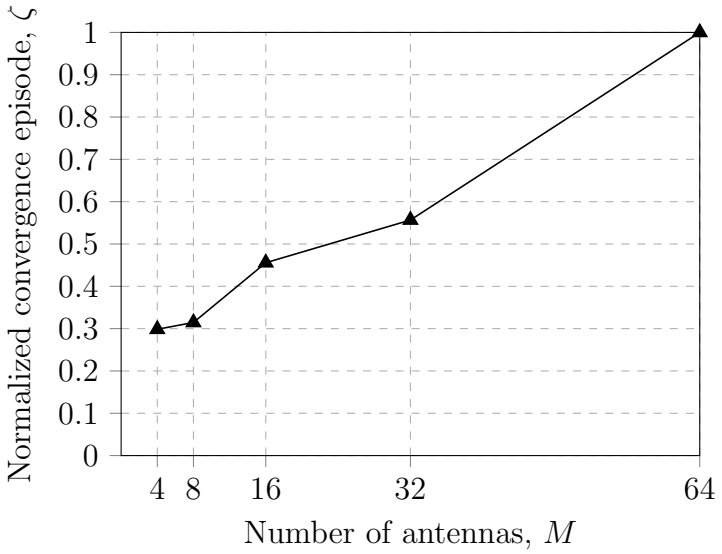


Figure 2.9: The normalized convergence time for the proposed deep Q-learning algorithm as a function of the number of antennas M.

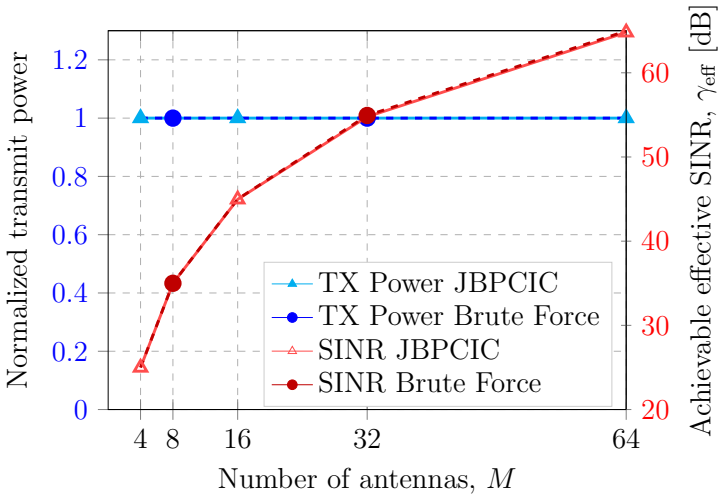


Figure 2.10: Achievable SINR and normalized transmit power for both the brute force and proposed JBPCIC algorithms as a function of the number of antennas M.

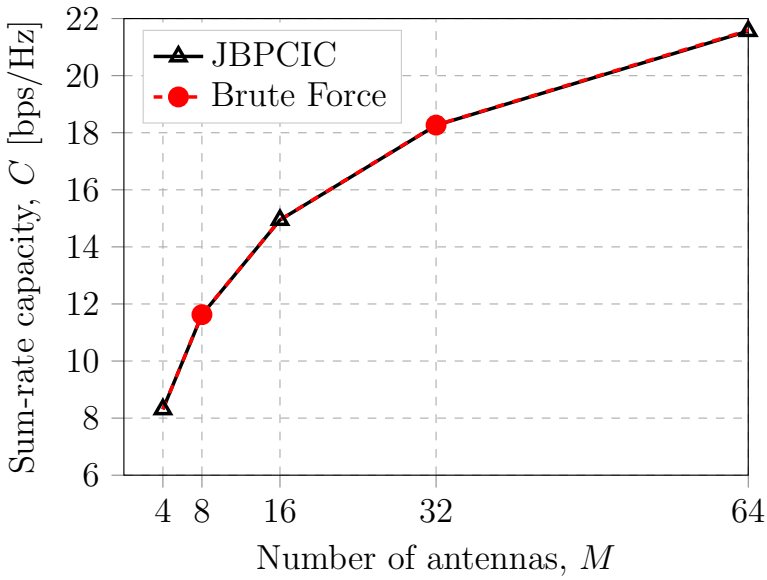


Figure 2.11: Sum-rate capacity as a function of the number of antennas M.

2.10 Conclusion

In this chapter, I sought to maximize the downlink SINR in a multi-access OFDM cellular network from a multi-antenna base station to single-antenna user equipment. The user equipment experienced interference from other multi-antenna base stations. My system used sub-6 GHz frequencies for voice and mmWave frequencies for data. I assumed that each base station could select a beamforming vector from a finite set. The power control commands were also from a finite set. I showed that a closed-form solution did not exist, and that finding the optimum answer required an exhaustive search. An exhaustive search had a run time exponential in the number of base stations.

To avoid an exhaustive search, I developed a joint beamforming, power control, and interference coordination algorithm (JBPCIC) using deep rein-

forcement learning. This algorithm resides at a central location and receives UE measurements over the backhaul. For voice bearers, my proposed algorithm outperformed both the tabular Q-learning algorithm and the industry standard fixed power allocation algorithm.

My proposed algorithm for joint beamforming, power control and interference coordinations requires that the UE sends its coordinates and its received SINR every millisecond to the base station. The proposed algorithm, however, does not require the knowledge of the channel state information, which removes the need for channel estimation and the associated training sequences. Moreover, the overall amount of feedback from the UE is reduced because the UE sends its coordinates and would not need to send explicit commands for beamforming vector changes, power control, or interference coordination.

In the next chapter, I talk about enhancing the performance of the coordinated multipoint through using a deep learning based triggering function, which finds an improved relationship between the retransmission overhead and the MIMO transmission rank.

Chapter 3

Improving Downlink Coordinated Multipoint Performance with Deep Learning

In Chapter 2, I have introduced deep reinforcement learning to perform joint beamforming, power control, and interference coordination for data bearers; and power control and interference coordination for voice bearers. In this chapter[†], I enhance the performance of coordinated multipoint. Coordinated multipoint improves the user performance by utilizing the capacity of more than one base station close to the edge to enable communications. In the downlink direction, coordinated multipoint has three schemes: joint transmission, dynamic point selection, and coordinated scheduling/coordinated beamforming. I seek to optimize the communication performance of a joint transmission scheme. My focus is on using a surrogate function to trigger a higher simultaneous network-based transmission rank towards the user. This surrogate trigger function again disrupts the conventional way coordinated multipoint is triggered and offers enhanced user performance.

[†]This chapter is based on the work published in the journal paper: F. B. Mismar and B. L. Evans, "Deep Learning in Downlink Coordinated Multipoint in New Radio Heterogeneous Networks," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1040-1043, Aug. 2019. This work was supervised by Prof. Brian L. Evans.

3.1 Overview

In this chapter, I propose a method to improve the performance of the downlink coordinated multipoint (CoMP) in heterogeneous networks. This method is compliant with industry standards and is based on the construction of a surrogate CoMP trigger function using deep learning. This surrogate function enhances the downlink user throughput distribution through online learning of non-linear interactions of user equipment reported measurements, which I use as learning features to build a classifier to change the transmission rank. The cooperating set is a single-tier of sub-6 GHz heterogeneous base stations operating in the frequency division duplex (FDD) mode (i.e., no channel reciprocity). In simulation, I show that the average user throughput obtained by my proposed method outperforms industry standards by 13.5% in a realistic and scalable heterogeneous cellular environment.

3.2 Introduction

The aggregate demand for data traffic over fifth generation of wireless communications (5G) cellular networks is expected to increase a thousand times compared to the previous generation [8]. Heterogeneous networks, in which small cells are deployed along with macro base stations, are one of the most important solutions to increase the network capacity. Downlink coordinated multipoint (CoMP) will play an important role in improving data rates and cellular capacity in 5G by using a centralized unit to coordinate the operation of multiple New Radio (NR) base station units [66]. The principle of

downlink CoMP extends to other 5G techniques such as the "cell-free massive MIMO," which achieves coherent processing across geographically distributed base stations [15].

Downlink CoMP has various implementations. My focus is on the *joint transmission* scheme, where the spatially multiplexed data streams of the user equipment (UE) are available at more than one point participating in the data transmission in a time-frequency resource. These points (or base stations) form the CoMP cooperating set. This effectively forms a distributed multiple input multiple output (MIMO) channel with spatially decorrelated streams from each base station (BS) in the CoMP cooperating set [67], which increases the average user throughput. This seems intuitive as spatial decorrelation increases the number of independent channels. The base stations in the CoMP cooperating set communicate over low-latency backhaul to maintain synchronization. A common approach in CoMP today is to use a static absolute triggering threshold based on the UE reported measurements.

3.2.1 Prior Work

A traffic analysis model based on Markov chains was introduced in [68] to approximate the computation of the received power sum of multiple log-normal random components in multi-BS environments in a joint transmission downlink CoMP scheme. Call admission control was introduced to improve the resource utilization based on transmission from multiple cells. I, on the other hand, focus on user perceived throughput and spectral efficiency without

regards to the call blocking. Joint transmission downlink CoMP was studied in [69] to derive an expression for network coverage probability of a UE located equidistant from cooperating BSs in a heterogeneous network using stochastic geometry. On the other hand, I do not have the equidistant constraint. The performance of the zero-forcing beamforming for CoMP coordinated beamforming was evaluated in [70], with base stations randomly dropped on a two-dimensional plane for multiple times. A near-optimal number of antennas was found to be equal to two in zero-forcing beamforming, which is the maximum number of transmit antennas I use.

Interference mitigation and handover management in CoMP-formed clusters were discussed in [71, 72]. The objective was to reduce the backhaul utilization in star clusters of base stations without harming end-user throughput. In contrast, my algorithm aims to improve end-user throughput. Furthermore, I do not require the formation of star clusters as heterogeneous networks in CoMP are likely to serve a user capable of two receive streams in pairs composed of one high power macro and one low power node.

3.2.2 Contributions

In this chapter, I further improve the CoMP joint transmission average user throughput performance from my previous work [73]. In my previous work, I used support vector machine (SVM) binary classification for my CoMP trigger function in frequency division duplex mode of operation. I propose an online deep learning algorithm which acquires physical layer measurement re-

ports from the connected UEs within the channel coherence time in a radio frame to trigger CoMP instead of conventional algorithms. My proposed algorithm is compliant with the industry standard for CoMP for Long Term Evolution Advanced (LTE-A) [67]. The proposed algorithm formulates a surrogate CoMP trigger function to enhance the downlink performance. This surrogate function factors in the received power, the received signal to interference plus noise ratio, the MIMO rank indication, and the codeword retransmission rate. The algorithm computation can take place in a centralized location as part of a self-organizing network (SON) as shown in Fig. 3.1. My choice of deep neural networks (DNNs) allows the creation of more learning features than shallow architectures such as SVM. This is due to the combinatorial and non-linear nature of the hidden layers of a DNN. Furthermore, DNNs perform particularly well when channels are complicated [74,75]. Also, SVMs tend to underperform when the classification problem is imbalanced [76].

I choose a heterogeneous network due to the relatively shorter distances of small cells from the macro, making backhaul more suitable for CoMP [67]. However, using macro BSs only may be possible with certain backhaul constraints [67].

3.3 Network and Signal Model

In this section, I describe the network environment, the signal model, and the deep learning related to it.

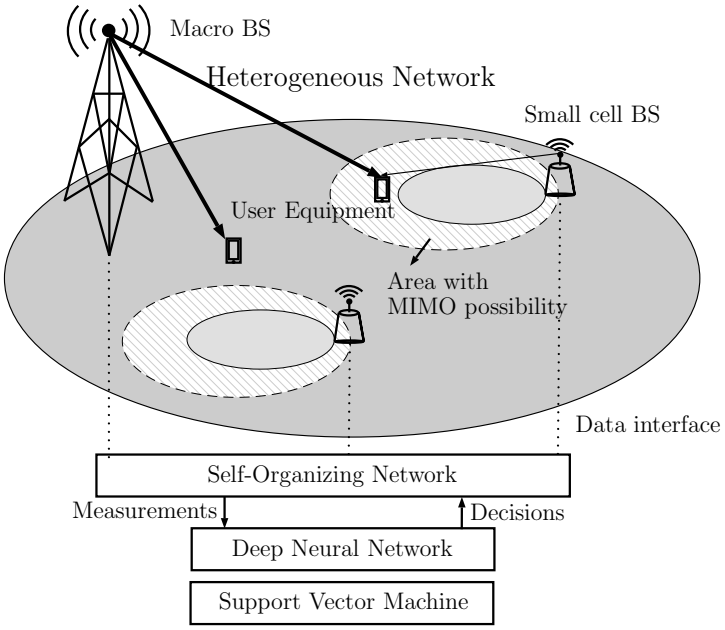


Figure 3.1: Joint transmission in a coordinated multipoint New Radio heterogeneous network with interfaces to the self-organizing network.

3.3.1 Network Environment

My heterogeneous network is based on orthogonal frequency division multiplexing (OFDM). I uniformly scatter small cells for densification of the macro coverage in an urban environment. Non-stationary UEs with multiple antennas are randomly placed and uniformly distributed in the service area. The network is comprised of a macro BS with one tier of surrounding macro BSs. All macro BSs have three sectors with directional antennas. I also add uniformly scattered small cells in the service area with omni-directional antennas. The BSs are transmitters and the UEs are the receivers. I use LTE-A as a multi-access wireless network in the sub-6 GHz range.

3.3.2 Signal Model

I adopt a narrow-band network MIMO channel model where the received signal in the discrete time domain for the i-th UE is written as

$$\mathbf{r}_i = \sqrt{\frac{E_{s,i}}{n_t}}\mathbf{H}_i\mathbf{s}_i + \mathbf{v}_i \tag{3.1}$$

where $\mathbf{H}_i \in \mathbb{C}^{n_r \times n_t}$ is the channel formed by the transmitting BSs with both large- and small-scale gain for the i-th UE, $E_{s,i}$ is the energy transmitted from the BSs to the i-th UE, $\mathbf{s}_i \in \mathbb{C}^{n_t}$ is the transmitted signal from the BSs in the cooperating set such that $\mathbb{E}[\|\mathbf{s}_i\|^2] = 1$, and $\mathbf{v}_i \in \mathbb{C}^{n_r}$ is the noise vector at the receiver. The latter two quantities are also assumed to be circularly symmetric Gaussian with zero mean and variance $\sigma_v^2\mathbf{I}$. Also, n_t and n_r are the number of transmit and receive streams respectively such that the maximum number of streams at the i-th UE $n_s := \min(n_r, n_t)$.

As I sample the signal model at discrete time steps t with unity increments, $E_{s,i}$ is also equal to the power of the transmitting BS in the CoMP cooperating set to the i-th UE $P_{\mathrm{BS}}^{(i)}$. Therefore, I can write that $E_{s,i} := P_{\mathrm{BS}}^{(i)}$.

I use zero-forcing (ZF) reception at the UE. Hence, the received signal to interference plus noise (SINR) for the i-th UE per receive stream j is derived as:

$$\gamma_j^{(i)} = \frac{P_{\mathrm{BS}}^{(i)}}{n_t\sigma_{\tilde{v}}^2}/[\mathbf{H}_i^*\mathbf{H}_i]_{j,j}^{-1}, \qquad j = 1, \ldots, n_s \tag{3.2}$$

where $\sigma_{\tilde{v}}^2$ is the variance of the receiver enhanced noise, measured at the i-th UE. This noise is defined as

$$\tilde{\mathbf{v}}_i := \sqrt{n_t}\mathbf{W}_{\mathrm{ZF},i}\mathbf{v}_i \tag{3.3}$$

where $\mathbf{W}_{\mathrm{ZF},i} \in \mathbb{C}^{n_r \times n_t}$ is the ZF matrix and is equal to the pseudo-inverse of the channel $\mathbf{H}_i$ (i.e., $\mathbf{W}_{\mathrm{ZF},i} = \mathbf{H}_i^{\dagger}$).

Similarly, the i-th UE received power per receive stream j is:

$$P_{\mathrm{UE},j}^{(i)} := \sigma_{\bar{v}}^2 \gamma_j^{(i)} = \frac{P_{\mathrm{BS}}^{(i)}}{n_t} / [\mathbf{H}_i^* \mathbf{H}_i]_{j,j}^{-1}, \qquad j = 1, \ldots, n_s \tag{3.4}$$

all for the i-th UE. The parameters of the radio environment are listed in Table 3.3.

I also define the reference symbol received power for the i-th UE (measured on the first receive branch as)

$$P_{\mathrm{RS}}^{(i)} = P_{\mathrm{UE},j=1}^{(i)} / (N_{\mathrm{SC}} N_{\mathrm{PRB}}) \tag{3.5}$$

where N_{SC} is the number of subcarriers per physical resource block (PRB) in the OFDM radio frame, and N_{PRB} is the number of PRBs allocated over the transmission bandwidth.

I define $\beta_{j,i}$, which is the respective codeword reception error, based on the block error rate (BLER) for the j-th stream of the i-th UE. BLER has a direct relationship with the number of streams and their chosen modulation and code scheme. I introduce two physical measurements which I use as learning features each of size M: a) the CSI reference symbol received power (CSI-RSRP) which is given in (3.5) and b) a transformation of the signal to interference plus noise ratio of the data channel (CSI-SINR) as measured at the j-th receive antenna. This resembles the channel quality indicator (CQI) [40] from LTE-A and is the name I adopt here. The CQI plot as a function of the

CSI-SINR is shown in Fig. 3.2. The CQI is reported for up to n_s antennas, which is the also known as the transmission rank. To obtain the machine learning features $\mathbf{X} := [\mathbf{x}_i]_{i=1}^{N}, \mathbf{x}_i \in \mathbb{R}^M$, I choose CQI and CSI-RSRP because they are physical channel measurement quantities that have weak statistical correlation: despite sharing the same coordinates where the measurement is made, CSI-RSRP ($\mathbf{x_1}$) is the received power of the *narrowband* NR reference symbols on the first receive antenna, while CQI ($\mathbf{x_2}$) is a quantized indication of the received *wideband* SINR per antenna [77]. The space-time decorrelation in the channel $\mathbf{H}_i$, where MIMO is possible in the cooperating set service area, can be indicated by per-antenna CQI values [67].

Surrogate CoMP trigger function: In building my surrogate CoMP trigger function, I create the supervisory signal labels vector $\mathbf{y} \in \{0, 1\}^M$. To do so, I use the aggregate BLER for the UE i, β_i, and write

$$y_i := \mathbb{1}[\beta_i \leq \beta_{\text{target}}] \tag{3.6}$$

where β_{target} is the retransmission target. The vector $\mathbf{y}$ is likely to be imbalanced in the two classes as a result. When $y_i = 1$, CoMP triggers a second stream thereby causing the cooperating set to behave as a distributed MIMO environment. The MIMO possibility is therefore not only due to the reported CSI measurements by the UE, but also by the error observed in received codewords.

With space-time independent receive streams j, I exploit statistical independence and find that the aggregate received BLER per user i, β_i, can

be calculated from the number of streams n_s and the BLER per stream j, $\beta_{j,i}$, as

$$\beta_i := 1 - \prod_{j=1}^{n_s} (1 - \beta_{j,i}), \tag{3.7}$$

which shows that the aggregate BLER for the i-th user increases as the number of streams n_s increases. The components of the surrogate CoMP trigger function for the i-th user are: BLER, CSI-RSRP, CQI, and the number of streams n_s, all of the i-th UE.

Finally, the relationship between the effective UE throughput and the elements of the surrogate CoMP trigger function for a given UE i is given by $R_i^{\text{eff}} = R_i(1 - \beta_i)$ where R_i is the instantaneous achievable rate given by $R_i := \sum_j B \log_2(1 + \gamma_j^{(i)})$ [78], all for the i-th UE [79]. This means that the relationship between the throughput and the BLER is inversely linear. An intuition here is that the SINR has a smaller impact on the effective UE throughput compared to the BLER since the latter is a pre-logarithmic factor.

3.4 Machine Learning

In this section, I describe two machine learning algorithms: support vector machines and deep neural networks. I also go over the various steps required in training a classifier.

3.4.1 Support Vector Machines

SVM classifiers maximize the margin around the separating hyperplane of two classes. In essence, the SVM classifier is the solution of the hinge-loss

kernel-modified Lagrangian dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize:}} \quad & \sum_i \lambda_i - \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{M}\lambda_m\lambda_n y_m y_n K(\mathbf{x}_m, \mathbf{x}_n) \\ \text{subject to:} \quad & \sum_{m=1}^{M}\lambda_m y_m = 0, \\ & 0 \le \lambda_m \le C_{\text{Box}}, \qquad m = 1, \ldots, M \end{aligned} \tag{3.8}$$

where $\mathbf{x}_m, \mathbf{x}_n$ (with non-bold typeface subscript) are the support vectors to the m-th and n-th data point respectively, λ_m, λ_i are elements in the Lagrangian multiplier vector $\boldsymbol{\lambda} \in \mathbb{R}^M$, and C_{Box} is a hyperparameter to control overfitting, also known as the Box constraint. Lastly, $K(\cdot, \cdot)$ is the SVM kernel and is defined as

$$K(\mathbf{x}, \mathbf{x}') \triangleq \phi(\mathbf{x})^\top \phi(\mathbf{x}') \tag{3.9}$$

where $\phi(\cdot)$ is a function that maps $\mathbf{x}$ to a higher dimension. A scale factor γ can also be introduced to the SVM kernel [80]. Since the dual problem is a maximization problem of a quadratic function subject to linear constraints, it

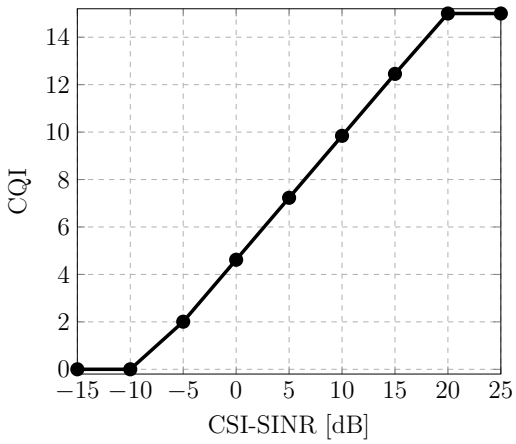


Figure 3.2: Downlink channel quality indicator (CQI) to CSI-SINR mapping.

Table 3.1: Machine learning features for CoMP improvement

	Parameter	Type	Description
$\mathbf{x_0}$	Bias term	Integer	This is equal to unity.
$\mathbf{x_1}$	CSI-RSRP	Float	Narrowband received power measurement.
$\mathbf{x_2}$	CQI	Integer	Wideband received SINR on first antenna linearly mapped.
$\mathbf{x_3}$	Rank	Integer	The number of streams j of the UE.

can be solved efficiently by quadratic programming. Also, the primal-dual gap is zero for SVM due to Slater's condition (strong duality) since the optimization objective in the primal problem is convex [81].

The SVM model is trained and tune the hyperparameters in Table 3.2 using grid search and K-fold cross-validation. I perform this grid search over the space defined by the Cartesian product of a few values of each hyperparameter.

3.4.2 Deep Learning

I use a fully connected DNN classifier with the sigmoid activation function in the implementation of this algorithm as shown in Fig. 3.3. I define the learning features in a design matrix $\mathbf{X}$ based on the physical measurements in the previous subsection. These features (listed in Table 3.1) are scaled such that their values lie in the closed interval $[0, 1]$. If the features were correlated, an inflation in the training error variance would have turned the learning results useless. Further, the block matrix of the weights of the fully connected DNN hidden layers is $\mathbf{\Theta} := [\boldsymbol{\theta}_\ell]_{\ell=1}^{d+1}, \boldsymbol{\theta}_\ell \in \mathbb{R}^{w \times M}$.

This introduction of a supervisory signal with deep learning enables the classifier to capture more effects more implicitly as I explain later in Sec-

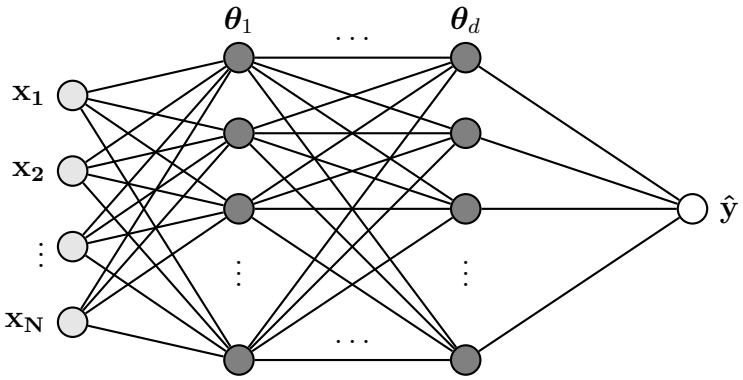


Figure 3.3: Structure of the fully connected deep neural network used in the implementation of my modified algorithm.

tion 3.5.

Feature engineering and missing data: The UEs are not scheduled all the time. Furthermore, not all the antennas are always receiving data either. Therefore, I dropped the CQI from the second antenna, and created a learning feature which captures n_s. When a UE is not receiving data, the CQI for that user in that time step is missing, even though the RSRP is still reported. I drop this data point from my dataset.

Training, cross-validation, and testing: The gathered data $\mathbf{X}$ and $\mathbf{y}$ is periodically split to a training and a test dataset. I then train the model and tune the hyperparameters in Table 3.2. I use grid search over the hyperparameters search space to find the optimal settings and K-fold cross-validation to prevent under- or over-fitting.

Classification performance: The vector containing the estimated supervisory signal labels from the classifier is $\hat{\mathbf{y}}$. Since the classifiers are trained

with the training dataset, the anticipated generalization performance of the classifiers is represented by the test data misclassification error μ

$$\mu := \frac{1}{M_{\text{test}}} \sum_{m} \mathbb{1}[\hat{y}_m \neq \hat{y}_{\text{test},m}] \tag{3.10}$$

where M_{test} is the test data size. High misclassification error can be attributed to classifier poor bias-variance characteristic or rapidly changed radio conditions. When the data has highly imbalanced classes (i.e., $\#(\mathbf{y} = c) \gg \#(\mathbf{y} = c'),\ c, c' \in \{0, 1\}, c \neq c'$), as is the case in this problem, the receiver operating characteristic (ROC) area under the curve ξ has advantage over the misclassification error. This is because ROC tries all the thresholds in its attempt to compute the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative example.

3.5 Problem Formulation

A common industry approach to enable downlink CoMP or disable it in the cooperating set is to use absolute thresholds of the downlink SINR reported by the UEs. These thresholds are subjective and have diminished effects on the user rate since SINR is a logarithmic term in its formula. Therefore, they are unlikely to yield an improved downlink CoMP performance. To improve the CoMP performance, I formulate the problem of triggering CoMP as a trade-off between BLER β and transmission rank n_s. Therefore, unlike the subjective SINR-based threshold, I now have a dynamic data-driven pre-logarithmic threshold that triggers CoMP. The data is collected from the

radio measurements from all the UEs served by the cooperating set during the time duration of T_{CoMP}. This duration cannot exceed the channel coherence time $T_{\text{coherence}}$ or the radio frame duration T_{RF}. I use the approximation $T_{\text{coherence}} \approx c/(vf_c)$ where c is the speed of light, v is the speed of the receiver UE, and f_c is the center frequency of the OFDM carrier. In other words, $T_{\text{CoMP}} \leq \min(T_{\text{coherence}}, T_{\text{RF}})$. Given this, the matrix $\mathbf{X}$ has a number of rows M upper bounded by $n_s N_{\text{UE}} g T_{\text{CoMP}}$, given the CSI reporting periodicity of g reports per transmission time interval (TTI) as in [11].

The collected data is then used to train a deep learning classifier and if its performance is acceptable, it can override the common approach for the next TTI. Otherwise, the common trigger is always the fallback. The DNN classifier performance is measured through the decision threshold ε, which can also control misclassifications due to training outside the channel coherence time or poor model fitting in general.

The DNN classifier must be periodically invalidated (i.e., purged and retrained with new measurements) at a periodicity of T_{CoMP}. Invalidation ensures that the model is updated with measurements within the channel coherence time. Otherwise, the channel state information (CSI) may have changed but may not have the proper reflection onto the classifier. I can therefore build a deep learning binary classifier where y_i is found as in (3.6). This enables us to reformulate the problem as a machine learning problem

that minimizes the binary cross-entropy loss function $L(\cdot,\cdot)$:

$$\underset{\boldsymbol{\Theta}}{\text{minimize: }} L(\mathbf{y},\hat{\mathbf{y}};\boldsymbol{\Theta}) := -\sum_{k} y_k \log \hat{y}_k + (1-y_k)\log(1-\hat{y}_k). \tag{3.11}$$

Here, $\hat{\mathbf{y}}$ is estimated from the DNN classifier. The value of $\hat{\mathbf{y}}$ instructs the CoMP cooperating set to form or teardown a dynamic MIMO channel through changing n_s. This is done per user for all users i during a given TTI. Minimization of the loss function is done through stochastic gradient descent (SGD) optimizer. SGD starts with a random initial value of $\boldsymbol{\Theta}$ and performs an iterative process to update $\boldsymbol{\Theta}$ as follows

$$\boldsymbol{\Theta} := \boldsymbol{\Theta} - \eta \nabla L(\mathbf{y},\hat{\mathbf{y}};\boldsymbol{\Theta}) \tag{3.12}$$

where $\eta\colon 0 < \eta \leq 1$ is the learning rate of SGD and $\nabla L_(\mathbf{y},\hat{\mathbf{y}};\boldsymbol{\Theta})$ is the gradient of the loss function (3.11) with respect to $\boldsymbol{\Theta}$. The weights $\boldsymbol{\Theta}$ are updated after every time step t. I also use the sigmoid function $\sigma : x \mapsto 1/(1+e^{-x})$ as the activation function of all the nodes in the DNN. The sigmoid function is a continuous non-linear differentiable function for all x making it a preferred choice, especially when the data is scaled in the interval $[0,1]$ as mentioned in Section 3.4.2.

Deep learning transforms (3.11) to higher dimensions through combinatorial and non-linear nature of the hidden layers. Using the fully connected DNN in Fig. 3.3, I write my DNN-based *surrogate function* $\hat{\mathbf{y}}$ in terms of the inputs, the trainable weights $\boldsymbol{\Theta}$, and non-linear activation functions $\sigma_\ell(\cdot), \ell \in \{1,2,\ldots,d+1\}$ as

$$\hat{\mathbf{y}} = \sigma_{d+1}(\boldsymbol{\theta}_{d+1}\sigma_d(\ldots\sigma_1(\boldsymbol{\theta}_1\tilde{\mathbf{X}}))) \tag{3.13}$$

where $d+1$ is the output layer, $\tilde{\mathbf{X}}$ is the normalized matrix $\mathbf{X}$, and the non-linear activation functions are applied element-wise on vectors. It can be inferred from the surrogate function formula that the number of learning features generated from these inputs is in $\mathcal{O}(w^d)$. This surrogate function is a CoMP-triggering function used for the next TTI.

3.6 Algorithm

In this section, I present my proposed algorithm in comparison to other algorithms.

3.6.1 Static SINR-based Algorithm

The decision to enable or disable CoMP in the cooperating set for users is based on an absolute threshold of the downlink SINR on the first antenna reported by the distribution of users. It is triggered based on the majority of the UEs reporting CQI above the trigger SINR threshold. In other words, I can write the SINR-based algorithm as

$$y_i[t] = \mathbb{1}[\gamma_{(i)}[t] \geq \gamma_{\text{CoMP}}] \tag{3.14}$$

where $\gamma_{i)}[t]$ is the received SINR as measured on the first antenna and γ_{CoMP} is the CoMP SINR trigger.

3.6.2 Dynamic Algorithm

The dynamic algorithm to trigger CoMP comes from [73]. The asymptotic time complexity of SVM training is in $\mathcal{O}(M^3)$ where M is the number of

rows in the matrix $\mathbf{X}$ as computed in Section 3.3. However, performing a grid search cross-validation has a multiplicative impact on this bound [2, 81].

3.6.3 Deep Learning Algorithm

The improved proposed dynamic algorithm to trigger CoMP is shown in Algorithm 2. The decision to trigger CoMP is based on the class with the maximum frequency: $\hat{y}^{\star}[t] := \arg\max_{\hat{y}} f(\hat{y}[t-1]), f(\hat{y}_i) := \#(\hat{\mathbf{y}} = \hat{y}_i)/M$, where the vector $\hat{\mathbf{y}}$ is obtained using the surrogate CoMP function from all the served UEs (3.13). The lower bound time complexity of training a DNN with d hidden layers and w neurons per hidden layer is in $\mathcal{O}(Mw^d)$ [82]. Otherwise, with equal hyperparameter search space size and cross-validation folds, DNN run-time complexity outperforms SVM if $d \log w < 2 \log M$.

The main steps of Algorithm 2 are as follows:

- Construct the dataset for the UEs in the cooperating set, which contains the measurements and the MIMO rank.
- Train the DNN classifier using this dataset.
- Use this classifier to make a decision about the enabling or disabling CoMP (i.e., the distributed MIMO channel rank) towards the UEs in this cooperating set.

Algorithm 2: Deep Learning CoMP Algorithm

Input: Decision threshold ε, measurements collection period T_{CoMP}, current triggering downlink SINR. Table 3.4 has example values.

Output: Triggering decision for downlink CoMP for all N_{UE} UEs in T_{sim} TTIs.

```
for T := 1 to T_sim do
    if T mod T_CoMP ≠ 0 then
        Acquire the learning features X in Table 3.1 from all UE
         measurements during time t = T, ..., (T + T_CoMP − 1).
        Compute the classification label y.
        Use the static algorithm for CoMP trigger
    else
        Split the measurement data [X | y] to training and test data.
        Scale the features in X to interval [0, 1].
        Train the DNN model using the training data and use grid
         search on K-fold cross-validation to tune the hyperparameters
         (in Table 3.2) and find optimum weights Θ.
        Compute the misclassification error μ.
        if μ > ε then fallback to static algorithm.
        Compute ŷ using the trained classifier for all UEs.
        Use arg max_ŷ f(ŷ) to decide to enable CoMP in the next TTI.
    end
    Invalidate the DNN model.
    Purge collected measurement data.
end
```

3.7 Performance Measures

The user downlink throughput empirical cumulative distribution function (CDF) [83] and the average user throughput are used as performance measures of the respective algorithms as presented in Section 3.6.

Table 3.2: Classifier hyperparameters

DNN Hyperparameter	Search range	SVM Hyperparameter	Search range
DNN depth d	{1,3,5}	Kernel	{`gaussian`, `polynomial`*}
DNN width w	{1,3,10}	Box constraint C_{Box}	{0.01,1,10}
Optimizer	Stochastic Gradient Descent	Kernel scale γ	`auto` [73]

* Degrees $p \in \{1, 2, 3, 4\}$.

Table 3.3: CoMP algorithm – radio environment parameters

Parameter	Value
Bandwidth B	10 MHz
Downlink center frequency f_c	2100 MHz
Downlink user scheduler	Proportional Fair
Macro BS maximum power	46 dBm
Small cell BS maximum power	37 dBm
Maximum number of streams n_s	2
Number of PRBs N_{PRB}	50

3.8 Simulation Results

In this section, I evaluate the performance of my proposed solution in terms of the performance measurements. I describe the adopted setup and then delve into the simulation outcomes.

3.8.1 Setup

I use a MATLAB-based simulator to implement my algorithm [83, 84]. The UEs move at an average speed of $v = 5\,\text{km/h}$. I use a $K = 3$ K-fold cross-validation with a training-test data split of 70-30 and SGD optimizer learning rate $\eta = 0.05$. I set the retransmission target β_{target} to 10%. The number of subcarriers per PRB $N_{\text{SC}} = 12$, and the radio frame duration $T_{\text{RF}} = 10$ ms. The channel coherence time $T_{\text{coherence}} \approx 103$ ms. The network is shown in

Table 3.4: CoMP simulation parameters

Parameter	Value
Static CoMP SINR trigger γ_{CoMP}	-3.5 dB
Total number of connected UEs in the cluster N_{UE}	184
Number of small cells	17
Number of macro BSs	21
Measurements collection period T_{CoMP}	3 TTIs
Simulation time T_{sim}	30 TTIs
Misclassification error threshold ε	15%

Fig. 3.5. The small cells are scattered according to a homogeneous density of 80 per km^2. The important simulation parameters are in Table 3.4.

3.8.2 Outcomes

The performance improvement empirical CDF of the user throughput over the simulation period is shown in Fig. 3.4. This improvement is due to the

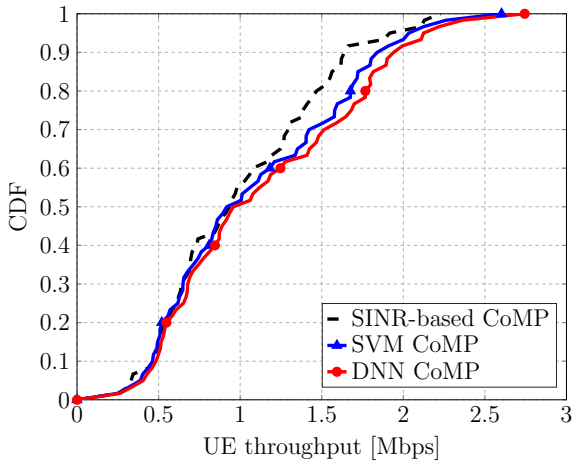


Figure 3.4: Empirical cumulative distribution function (CDF) of the UE downlink throughput for all three algorithms.

learning of an improved surrogate CoMP triggering function as shown in Fig. 3.7. The supervisory signal labels are imbalanced with $\#(\mathbf{y} = 0) = 1{,}522$ out of 9,180. Therefore, the class $y = 1$ is the majority class. This means that the BLER was maintained below the 10% target. I observe that the SINR-based algorithm makes decisions to enable or disable CoMP in the cooperating set

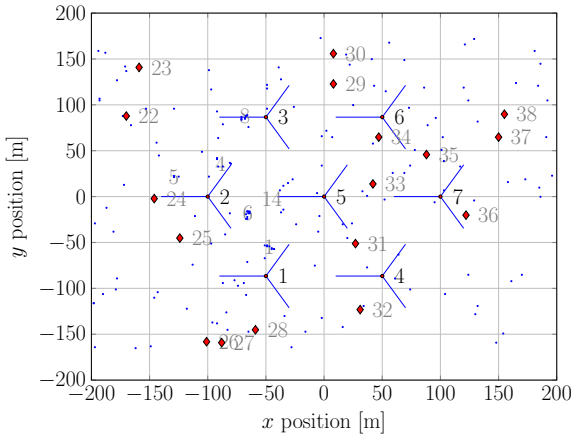


Figure 3.5: CoMP network with UEs in blue and small cells in red diamonds.

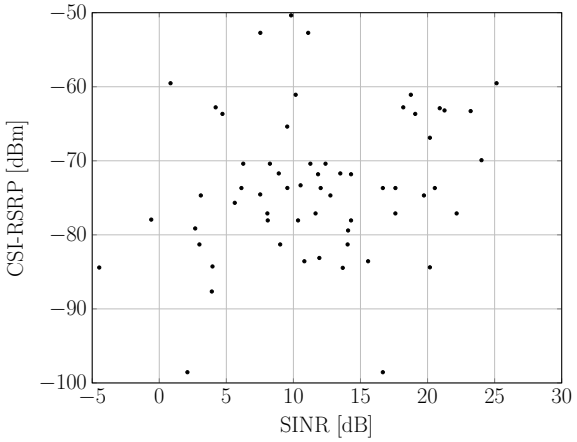


Figure 3.6: Scatter plot showing the CSI-RSRP and the SINR as measured on antenna $j = 1$. There is weak correlation between the two quantities.

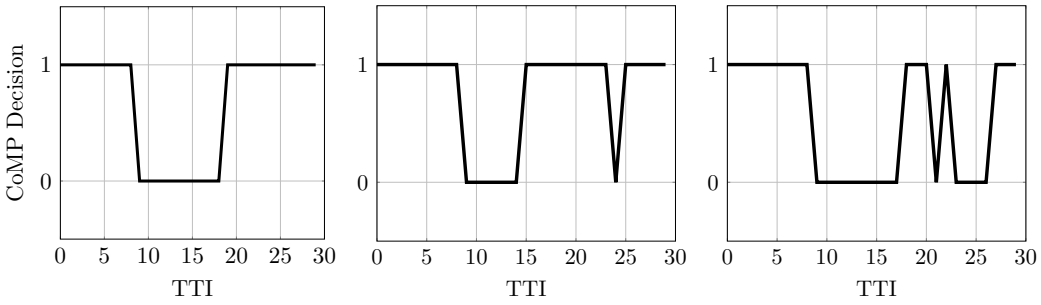
Table 3.5: CoMP downlink link-level performance

	Average				
Algorithm	User Throughput [Mbps]	BLER β_i	Streams n_s	CQI	CSI-RSRP [dBm]
SINR-based CoMP[‡]	1.02	-	-	-	-
SVM CoMP	1.10	7.15%	1.59	3	−58.17
DNN CoMP	**1.16**	3.76%	1.55	3	−58.17

[‡] Quantities not reported in the published version.

when the improved dynamic algorithms do the opposite. The reason for DNN outperforming the SVM-based CoMP algorithm is two-fold. First, the depth of the DNN allows the creation of more interaction features. As I show in Table 3.6, the number of features in DNN is $\mathcal{O}(w^d)$, compared to the most feature-generating polynomial SVM kernel of degree p with the number of features being $\mathcal{O}(p)$. Second, SVM tends to suffer bias towards the majority class ($y = 1$) when the training supervisory signal labels are imbalanced [76]. Due to the decisions made by the computed surrogate function (3.13), the CoMP cooperating set prevents UEs from receiving less codewords with lower BLER penalty on average (3.7). Furthermore, the cooperating set activates CoMP at times the SVM decision is biased towards disabling CoMP. The optimization of triggering CoMP with an increased BLER and a larger number of streams n_s brings about the observed downlink throughput gain on average. Neither the average CQI nor the average RSRP was impacted as shown in Table 3.5.

Fig. 3.6 shows that there is no strong correlation between the RSRP and the SINR as measured on the first antenna (when measured in the linear



(a) SINR-based **(b)** SVM **(c)** DNN

Figure 3.7: Downlink coordinated multipoint being enabled (state = 1) and disabled (state = 0) for the SINR-based (left), the SVM-based (middle), and the DNN proposed algorithm (right).

Table 3.6: Comparison of complexities

Algorithm	Asymptotic run-time	Number of features
SINR-based CoMP	$\mathcal{O}(1)$	$\mathcal{O}(1)$
SVM CoMP	$\mathcal{O}(M^3)$	$\mathcal{O}(p)$
DNN CoMP	$\mathcal{O}(Mw^d)$	$\mathcal{O}(w^d)$

scale). The coefficient of determination of these quantities, which indicates the proportion of the variance of one quantity that can be predicted from the other, is in the order of 10^{-3}.

3.8.3 Future Work

Note that I have focused on the joint transmission type of coordinated multipoint, where different codewords are transmitted from different BSs to the UE. In this case, the UE benefits from the increase in the throughput when MIMO is possible. However, an interesting improvement is to extend the idea to cell-free massive MIMO [15] or network-coordinated beamforming,

where the DNN learns the best assignment of joint beamforming vectors to the scheduled UEs, enhancing the overall performance.

3.9 Conclusion

In this chapter, I motivated the use of a surrogate trigger function for CoMP. My surrogate trigger function captures the received signal power, the SINR, the codeword retransmissions, and the transmission rank in an attempt to enhance the means of which the transmission rank is triggered in the CoMP cooperating set. I obtained this function through applying online machine learning to physical layer measurements in a realistic LTE-A FDD heterogeneous network using both SVM and DNN. My standards-compliant method using DNN improved the downlink rates compared to SVM with virtually no impact on the reported CQI or the narrowband received power. This improvement is due to introducing the retransmissions awareness to the CoMP trigger function, the increase in the number of relevant learning features, and the lower bias in the DNN classification model compared to SVM.

In the next chapter, I introduce predictive band switching in dual-band cellular networks. The idea of proactively granting band switching without ceasing the transmission can improve the downlink UE throughput.

Chapter 4

Deep Learning Predictive Band Switching in Wireless Networks

In Chapter 3, I have introduced deep learning to enhance the downlink coordinated multi-point performance. In this chapter[†], I use deep learning to perform predictive band switching and eliminate the measurement gap bottleneck. Band switching is the procedure during which the user requests changing their serving frequency band. It is often referred to as the *vertical* handover. This procedure takes place at the radio resource management layer in the air interface.

[†]This chapter is based on the work submitted in the journal paper: F. B. Mismar, A. AlAmmouri, A. Alkhateeb, J. G. Andrews, and B. L. Evans, "Deep Learning Predictive Band Switching in Wireless Networks," *IEEE Transactions on Wireless Communications*, submitted, Oct. 2, 2019. Part of this work also appeared in the conference paper: F. B. Mismar and B. L. Evans, "Partially Blind Handovers for mmWave New Radio Aided by Sub-6 GHz LTE Signaling," in *Proceedings of IEEE International Conference on Communications Workshops*, May 2018. This work was supervised by Prof. Brian L. Evans. Mr. Ahmad AlAmmouri, Prof. Jeffrey G. Andrews, and Prof. Ahmed Alkhateeb (Arizona State University) provided important ideas about the use of the ray-tracing dataset, the problem formulation, and the depth of analysis of the results which greatly improved the work. Prof. Jeffrey G. Andrews wrote the last two sentences in the Abstract (Section 4.1 here) and the first paragraph in the Introduction (Section 4.2 here).

4.1 Overview

In this chapter, I deal with the band switching problem in wireless networks. In cellular systems, the user equipment (UE) can request a change in the frequency band when its rate drops below a threshold on the current band. The UE is then instructed by the base station (BS) to measure the quality of candidate bands, which requires a *measurement gap* in the data transmission, thus lowering the data rate. I propose a band switching approach based on machine learning that does not require any measurement gap. My proposed classifier-based band switching policy instead exploits spatial and spectral correlation between radio frequency signals in different bands based on knowledge of the UE location. I focus on switching between a lower (e.g., 3.5 GHz) band and a millimeter wave band (e.g., 28 GHz), and design and evaluate two classification models that are trained on a ray-tracing dataset. A key insight is that measurement gaps are overkill, in that only the relative order of the bands is necessary for band selection, rather than a full channel estimate. My proposed machine learning-based policies achieve roughly 30% improvement in mean effective rates over those of the industry standard policy, while achieving misclassification errors well below 0.5%.

4.2 Introduction

With each successive cellular standard using a rapidly increasing number of different frequency bands in different parts of the spectrum, the band selection problem has become ever more complicated. In particular, UEs wish

to use the band or bands that maximize their quality of experience (QoE), which is highly correlated to their achieved data rate. The choice of the optimal frequency band can be challenging. On the one hand, lower frequency bands generally have more benign propagation and thus produce higher signal to noise ratio (SNR), but higher frequency bands such as millimeter wave (mmWave) offer much more bandwidth as well as beamforming gains and will typically be more lightly loaded. So, if the SNR on a mmWave band is acceptable, it is likely to provide a much higher data rate than a lower band and a UE would usually benefit from being efficiently switched over to the mmWave band. Similarly, if coverage is lost on the mmWave band, the UE should be quickly switched back to the lower band.

Despite its increasing importance, the procedure for band switching has seen only incremental changes over the evolution of multiple successive 3GPP standards [26,85]. This procedure is shown in Fig. 4.1 and described as follows. If the received power at the user drops below a certain threshold on its current frequency band, call it f_j, it requests a band switch from its serving base station (BS). This request is followed by a *measurement gap*, where the data flow is stopped to allow the user to tune its reception circuitry to the frequency of the target band, call it $f_{j'}, j' \neq j$, to measure the channel. After obtaining the measurements, the user reports them back to the BS. The BS estimates, based on the measurements, whether the user would benefit from switching to $f_{j'}$ or not, and hence, grants or denies the request. A key issue with the aforementioned procedure is its dependence on the measurement gap which

causes interruption in the data flow and reduces the user overall throughput. The 3GPP standards also introduced mobile load balancing (MLB) as a means of transferring traffic served by a congested BS to nearby BSs that have spare resources [28]. However, MLB requires periodic communication between BSs about their resources, thereby introducing a significant overhead. MLB is also triggered by the BS desire to relieve its congestion, while the band switch is triggered when the UE desires to maximize its QoE.

It would be desirable to introduce a reliable method that can support the band switch procedure without interrupting the user data flow by measurement gaps. The aim of this chapter is to propose a novel gap-free algorithm for band switching that utilizes the spatial and spectral correlations over different frequency bands along with the previous band switching requests and decisions for nearby users. More precisely, I propose predictive algorithms, based on the extreme gradient boosting (XGBoost) trees [86] and deep neural network (DNN) classifiers, which allow the BS to decide whether to grant or deny the band switching requests without the need for measurement gaps.

4.2.1 Related Work

Predicting the success of a band switch from one frequency band to another without explicitly measuring the channel at the target frequency band falls under the genre of problems commonly referred to as "channel estimation using out-of-band information" [87]. In the simplest form of this problem, there are forward and backward (downlink and uplink) links occupying the

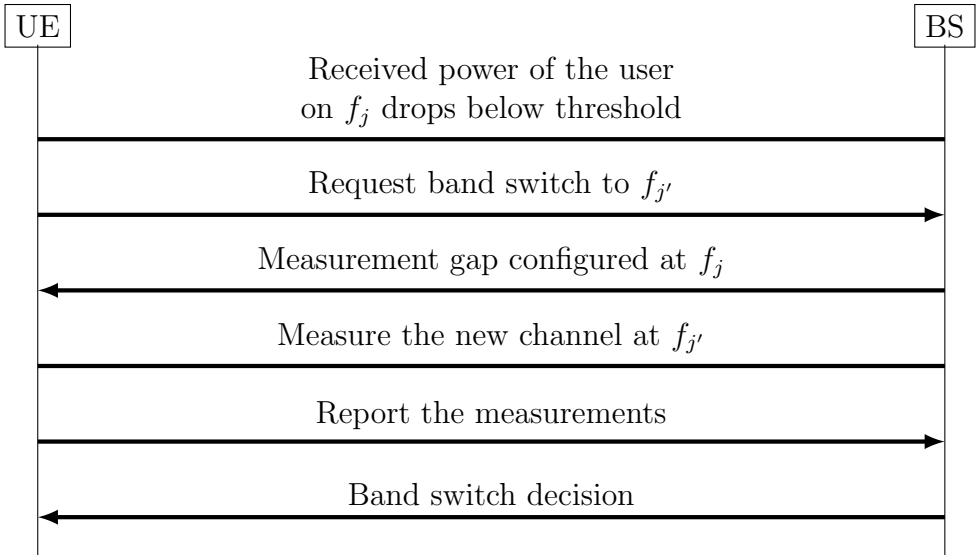


Figure 4.1: The band switch procedure between frequency bands in one base station (BS) in the downlink direction.

same frequency bands at different time slots. In this case, I can use channel reciprocity [88] to estimate the channel of the backward link using the measurements on the forward link, or vice versa. Even with a separation of frequency duplex bands on the order of ten megahertz, a spatial correlation between the signals on the two frequency bands still exists due to the common paths, blockages, and reflectors [47, 88]. Interestingly, the spatial correlation between two frequency bands that are separated by tens of gigahertz still exits [89]. However, it cannot be directly used to accurately estimate the channel on one frequency band by only using the measurements from the other, but it can be used to aid the channel estimation and reduce its complexity. For example, this correlation was exploited in [90–93] for cell discovery, channel covariance estimation, and beam selection in mmWave bands using sub-6 GHz

measurements. In the case of band switching, I am not interested in accurate channel estimation since the objective is not to use the estimate in decoding the messages, beam selection, or precoder/combiner design used in multiple-input multiple-output (MIMO) communications. Instead, my goal is much simpler: *ranking* the downlink channel quality of the two frequency bands or technologies.

The major challenge in exploiting the spatial correlations between frequency bands is the lack of accurate mathematical models that describe how the channel changes across these frequencies (or technologies). This challenge makes a data-driven or a machine learning (ML) approach more attractive to follow and implement. With more publicly available datasets that are based on field-measurements or sophisticated ray-tracing simulations [94, 95], I expect the interest in this approach to dramatically increase. Nevertheless, the applications involving dual-band ray-tracing datasets with ML classification to study channels is a nascent research area.

Although relevant to dual-band resource management, the work in [96] did not address the impact of measurement gaps on UE data rates. It focused on granting resources to users at mmWave first, while I allow granting resources to both mmWave and sub-6 GHz simultaneously without any specific preference. Furthermore, statistical path loss models were used for both mmWave and sub-6 GHz bands which may be privy to the spatial and spectral correlation of channels that I otherwise capture using ray-tracing datasets.

The work in [97] studied only one type of 3GPP dual-band handovers,

which I call the "legacy" policy later in this chapter. However, similar to [96], the use of statistical path loss models voids the opportunity to exploit the correlation across bands; therefore insights about the performance of the various algorithms, including the second type of 3GPP dual-band handover algorithms—the "blind" policy—could not be derived. Furthermore, the objective was to improve energy efficiency through handover avoidance, unlike my proposed algorithm the objective of which is to improve the UE data rates by eliminating measurement gaps.

In [98] dual connectivity was studied. Dual connectivity requires a local coordinator to manage the traffic between the cells, unlike band switch procedures. As a result, a backhaul latency constraint between the BSs was imposed. Furthermore, empirical pathloss models were used. Multiple BSs with a single UE were simulated while my focus is on a single BS with dual band and multiple UEs. The use of a single UE may prevent the employment of ML techniques due to the limited number of learning observations—a problem I avoid altogether through the use of a dataset. Moreover, a band switch time-to-trigger mechanism was introduced, whereby the band switch is only granted after the band switch criterion is fulfilled for a period of time. This, unlike my proposed approach, introduces further latency to the band switch procedure thereby making it unsuitable for the high-rate and ultra-low latency targets promised in the fifth generation of wireless communications (5G) [8].

4.2.2 Contributions

In this chapter, I provide an answer to the question whether a reliable band switch method exists to maximize the users' achievable data rates. Specifically, this chapter makes the following contributions:

1. Motivate the use of deep learning in ranking the downlink channel quality of the two frequency bands—a mathematically intractable problem and a requirement for the band switching procedure.

2. Offer several insights about the different band switch policies and their respective impact on performance. Furthermore, I show how the choice of the band switch threshold can have adverse impacts on the performance.

3. Motivate a data-driven approach to band switching, where I use a ray-tracing dataset in deep learning.

4. Create a unified framework to describe the band switch policies in a single equation and use this equation to explain the various band switching policies and their relevant performance.

4.3 System Model

In this section, I describe the adopted network and channel models.

4.3.1 Network Model

I consider a radio network comprised of one BS serving single-antenna UEs in an arbitrary association area. The BS has two frequency bands; one in the sub-6 range and one at mmWave. Moreover, the BS utilizes a different number of antennas for each frequency band. Let $j \in \{\text{sub-6}, \text{mmWave}\}$ denote the frequency band and let $N^{(j)}$ denote the number of antennas on the j-th band, then the received signal at the i-th UE from the BS at the j-th frequency band is

$$r_{(i,j)} = P_{\text{TX}}^{(j)} \mathbf{h}_{(i,j)}^{*} \mathbf{f}_{(i,j)} s_{(i,j)} + n_{(i,j)}, \tag{4.1}$$

where $P_{\text{TX}}^{(j)}$ is the transmit power of BS on the j-th frequency band, $\mathbf{h}_{(i,j)} \in \mathbb{C}^{N^{(j)} \times 1}$ is the channel vector, $\mathbf{f}_{(i,j)}$ is the beamforming vector, $s_{(i,j)}$ is the transmitted signal, and $n_{(i,j)} \sim \text{Normal}(0, \sigma_n^2)$ is the thermal noise. I focus on codebook-based analog beamforming, where the beamforming vector is chosen from a pre-defined codebook $\mathcal{F}^{(j)}$ [99]. In this case, the BS chooses the optimal beamforming vector $\mathbf{f}^{\star}$ that maximizes the receive SNR at the user from the codebook $\mathcal{F}^{(j)}$

$$\mathbf{f}_{(i,j)}^{\star} := \underset{\mathbf{f}_{(i,j)} \in \mathcal{F}^{(j)}}{\arg\max} \, |\mathbf{h}_{(i,j)}^{*} \mathbf{f}_{(i,j)}|^2. \tag{4.2}$$

Let the codebook size be denoted by $N_{\text{CB}}^{(j)}$ and assume that all codewords are normalized, i.e., $\|\mathbf{f}_{(i,j)}\|^2 = 1$. Based on this, the received SNR at time step t at the i-th UE on the j-th frequency band is

$$\gamma^{(i,j)}[t] = \frac{P_{\text{TX}}^{(j)}[t]}{\sigma_n^2} |\mathbf{h}_{(i,j)}^{*}[t] \mathbf{f}_{(i,j)}^{\star}[t]|^2, \tag{4.3}$$

and the instantaneous achievable rate is

$$R^{(i,j)}[t] = B^{(j)} \log_2(1 + \gamma^{(i,j)}[t]), \tag{4.4}$$

where $B^{(j)}$ is the available bandwidth at the j-th frequency. Note that the rate in (4.4) does not include the overhead of switching to a different frequency band nor the beam training overhead. These overheads cause a loss in throughput, which is typically related to the coherence time of the channel and the frame length.

4.3.2 Channel Model

Here I discuss the channel coherence time, the beam training time, the band switching overhead, and the effective throughput.

Channel coherence time: Let the coherence time for sub-6 GHz and mmWave frequency bands be denoted as $T_C^{\text{sub-6}}$ and T_C^{mmWave}, respectively. The exact values depend on the environment, the antenna configuration, and the user movement. Hence, to maintain the generality of the framework, I do not assume specific values for the channel coherence times and I discuss my choice of the coherence times in Section 4.7, which is only needed to numerically evaluate the performance of the different algorithms.

Beam training time: For the beam training overhead, I define the training penalty per beam as T_{beam}. Thus, the total beam training time, T_B, is related to the number of all possible beams, which is the size of the codebook $\mathcal{F}^{(j)}$ in my case (i.e., $T_B = T_{\text{beam}} N_{\text{CB}}^{(j)}$).

Band switching overhead: At the beginning of each frame, the UE can request a band switch operation from its serving BS if it is not satisfied by its current signal quality. The BS uses a certain policy to determine whether the change to a different frequency band should be granted or denied. However, there is a time penalty for the band switch request, which is used by the BS to take a decision regarding the user request. I denote this overhead by T_H, which is determined by the algorithm or the policy used in the BS to respond to the band switch request and the existence or absence of a measurement gap. The exact values of T_H are given in Section 4.4, where I present different band switch policies.

Effective throughput: Using the previous definitions for the channel coherence time T_C, the beam training time T_B, and the band switch overhead T_H, I can compute the effective throughput for the i-th UE that is connected to the j-th frequency band at the time step t as follows

$$R_E^{(i,j,k)}[t] = \left(1 - \frac{T_B^{(j)} + T_H^{(k)}}{T_C^{(j)}}\right) R^{(i,j)}[t], \tag{4.5}$$

where the j-th band is the band after the band switch decision is made, which is a new frequency if the band switch was granted and the old frequency if the band switch was denied.

After discussing the system model and providing the necessary definitions, I present the current polices discussed in the industry standards [26] for the BS to make band switch decisions in the next section.

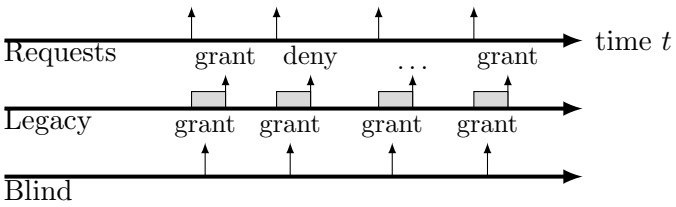


Figure 4.2: Legacy band switch time diagram. The shaded gray rectangles represent the measurement gaps.

4.4 Band Switch Policies

A band switch policy has to answer the following two questions: (i) when should the UE request this band switch? (ii) what is the information needed by the BS to make a decision for the band switch request and how? The first is typically solved by a pre-defined rate threshold $r_{\text{threshold}}$, such that if the UE rate is below this threshold, it requests a band switch. For the second question, the standards specify two policies today [26]. These policies are the measurement-based legacy approach and the blind approach. I also discuss the optimal policy as a benchmark. To provide a unified framework for the different polices, I define the following decision variables: $x_{\text{br}}, y \in \{0, 1\}$, where $x_{\text{br}} = 1$ if the UE requests a band switch, and $x_{\text{br}} = 0$ otherwise, and $y = 1$ if the BS grants the band switch and $y = 0$ otherwise. It is understood that y is only defined if $x_{\text{br}} = 1$. Further, the threshold $r_{\text{threshold}}$ is defined for all policies except the optimal policy. It is set based on how soon the UE should request the band switch from the BS.

4.4.1 Legacy Policy

The legacy policy, also known as the measurement-based policy, is shown in Fig. 4.2. When the user throughput is below the threshold $r_{\text{threshold}}$, it requests a band switch from the BS and it stops its transmission to measure the channel at the desired frequency band. After measuring the downlink channel, the user reports the measurements back to the BS, which decides whether to grant or deny the band switch request based on the measurements provided by the user. The measurement gap duration, denoted by T_G, is set to be a fraction of the coherence time [26]

$$T_G := \rho T_C, \qquad \rho > 0. \tag{4.6}$$

Further, if I denote the overhead due to a band switch signaling request and its decision response as $b > 0$, then the band switch time overhead $T_H^{(\text{legacy})}$ is equal to $T_G + b$. By using this policy, the BS can make an informed decision regarding the band switch using the rates from both bands, which guarantees a certain QoE for the user. However, this comes at the expense of the measurement gap,where the BS stops its transmission to the UE so it measures the target channel, which causes an interruption in the data flow and reduces the UE throughput.

By employing this policy, one of three scenarios are possible at the beginning of each frame: (i) the UE does not request a band switch, which happens if its current rate is higher than the threshold, (ii) the UE requests a band switch and it is granted by the BS, which happens if the user's current

rate is lower than the threshold, and the rate at the target band is higher than its current rate, or (*iii*) the UE requests a band switch, but it is denied, which happens if the UE current rate is lower than the threshold, and the rate at the target band is lower than its current rate. In terms of the decision variables I mentioned, they are defined as follows

$$x_{\text{br}}^{(i)}[t] = \mathbb{1}[(R^{(i,j)}[t] < r_{\text{threshold}})], \qquad \forall i, \tag{4.7}$$

$$y^{(i)}[t] = \mathbb{1}[(\hat{R}^{(i,j')}[t] > R^{(i,j)}[t])], \qquad \forall i, \tag{4.8}$$

where j is the current serving BS, j' is the target BS, and $\hat{R}^{(i,j')}$ is the estimated rate the UE would get if the band switch were granted.

4.4.2 Blind Policy

Similar to the legacy policy, when the UE throughput is below the threshold, it requests a band switch from the BS. However, in this policy, the BS instructs the UE to band switch to a different band without any need for a measurement gap. Given the nature of this band switch approach, if the SNR is worse at the target frequency, the throughput drops significantly. Hence, although the measurement gap is eliminated, the BS cannot guarantee the user a higher throughput after the band switch, which causes a low QoE for the user. The decision variables in this case are as follows

$$x_{\text{br}}^{(i)}[t] = \mathbb{1}[(R^{(i,j)}[t] < r_{\text{threshold}})], \qquad \forall i, \tag{4.9}$$

$$y^{(i)}[t] = 1, \qquad \forall i, \tag{4.10}$$

since the band switch requests are always granted by the BS. Here, $T_H^{(\text{blind})} = b$, which is only for the signaling overhead since there is no measurement gap requirement.

4.4.3 Optimal

To define an upper bound for the various band switch policies, I define the optimal policy to be the one where the BS knows the instantaneous quality of the channels of the different bands perfectly, so there is no need for a measurement gap. Hence, it asks the user to switch to a different band if the target rate is higher than its current rate. It also eliminates the need for a pre-defined rate threshold, since the band switch request and decision are combined and executed at the beginning of each frame by the BS. This also costs a time overhead of b. Based on this, the optimal effective throughput in this case is given by

$$R_E^{\star(i)}[t] = \max_{j \in \{\text{sub-6,mmWave}\}} \left(1 - \frac{T_B^{(j)}}{T_C^{(j)}}\right) R^{(i,j)}[t]. \tag{4.11}$$

Finally, the decision variables can be written as

$$x_{\text{br}}^{(i)}[t] = 1, \qquad \forall i, \tag{4.12}$$

$$y^{(i)}[t] = \mathbb{1}[(\hat{R}^{(i,j')}[t] > R^{(i,j)}[t])], \qquad \forall i. \tag{4.13}$$

4.4.4 Overhead of Band Switching

Based on the previous discussion, besides the standards-imposed signaling overhead requirement of b, which is common across all policies, only

the legacy policy causes a band switch overhead. This overhead is equal to the measurement gap. Hence, $T_H^{(i,j,k)} = T_G^{(i,j,k)} + b$ for $k \in \{\text{legacy}\}$ and $T_H^{(i,j,k)} = b, k \in \{\text{blind}, \text{optimal}\}$. Note that deterioration in user throughput due to band switch overhead in the legacy policy drives the setting of the predefined threshold to lower values to avoid spending long times in measurement gaps. When this threshold is set low, the signal quality has to be bad for the user to request the band switch. This prevents the user from utilizing possibly better channels on other frequency bands or technologies. Moreover, with the introduction of mmWave frequency bands [100] in the 5G standard, the design of the band switch procedure becomes yet more critical since radio frequency signals at mmWave bands are more sensitive to blockages by various objects. For example, it was shown in [101] that the antenna gains on the mmWave bands can suffer from up to 25 dB attenuation due to the user hand grip on the mobile device and it varies with the different hand grips. Hence, under large blockage losses, the user would benefit from a fast transition to other frequency bands, and relying on the measurement gaps does not help.

The objective of this work is to propose a new band switch policy that eliminates the measurement gap, as in the blind policy, but ensures a certain QoE as in the legacy policy. My policy relies on deep learning classification, which I introduce in the next section, before discussing the details of my algorithm.

Table 4.1: Deep neural network classifier learning features

	Parameter	Type	Description
$\mathbf{x_0}$	Bias term	Integer	This is equal to unity.
$\mathbf{x_1}$	$R_E^{(\text{sub-6})}$	Float	Effective achievable rate in the sub-6 GHz band.
$\mathbf{x_2}$	$R_E^{(\text{mmWave})}$	Float	Effective achievable rate in the mmWave band.
$\mathbf{x_3}$	Source technology	Binary	$(= 1$ for sub-6 and $= 0$ for mmWave).
$(\mathbf{x_4}, \mathbf{x_5}, \mathbf{x_6})$	Coordinates	Float	The coordinates of the UEs distance from the base station based on the coordinates of i-th UE.
$\mathbf{x_7}$	Band switch requested	Binary	UE requested band switch $(x_{\text{br}}^{(i)} = 1)$?
$\mathbf{y}$	Band switch decision	Binary	UE band switch request is granted $(y^{(i)} = 1)$?

4.5 Proposed Policy

For my proposed policy, I use the locations and measurements of a set of users $\mathcal{U}$ to improve the band switch performance for a set of geographically nearby users $\mathcal{N} \setminus \mathcal{U}$ from the total set of users $\mathcal{N}$, where $|\mathcal{N}| = N_{\text{UE}}$ such that $\mathcal{U} \subset \mathcal{N}$, without the need of a measurement gap. This is achieved by exploiting the spatial and spectral correlation of the channels over different frequency bands and different locations. I keep the minimum threshold criteria used in the legacy and blind polices, where the UE requests a band switch if its rate drops below a pre-defined threshold, $r_{\text{threshold}}$. Then the BS grants the band switch if the estimated rate at the target frequency is higher than the UE current rate. The difference is that the BS does not ask the UE to interrupt its transmission to measure the channel, but instead uses a machine learning approach to estimate the rate at the target frequency, using the user current rate and the previous measurements from other UEs. Hence, the decision variables x_{br}, y are defined in the same way as in the legacy approach given in

(4.7) and (4.8).

The major challenge in this approach, which relies on exploiting the spatial and spectral correlations between the channels, is the lack of accurate mathematical models that capture these correlations. Hence, I propose the use of DNN or XGBoost classifiers in the solution of my problem. My proposed algorithm trains the classifier based on a fraction of the locations of the UEs and whether a band switch was granted or denied in these locations. Then, I use this classifier to predict the band switch of other locations in the proximity of these learned locations. My algorithm is illustrated in Fig. 4.3 and is specified in Algorithm 3. The main steps of this algorithm are as follows:

- Construct the dataset for the UEs in the region, which contains the rates from the spatially correlated wireless channels and band switch decisions.
- Train the classifier using this dataset.
- Use this classifier to predict the band switch for the UEs outside this region.

For the classifier I consider two options: DNN and XGBoost. DNN is a feed-forward architecture that uses layers of neurons of a given depth d and width w [2]. An activation function $\sigma(\cdot)$ defines the output of a neuron with respect to its inputs. A DNN optimizes a convex loss function through a learning rate $\eta > 0$. XGBoost optimizes an objective function containing a convex loss function (e.g., binary logistic loss) and a regularization term

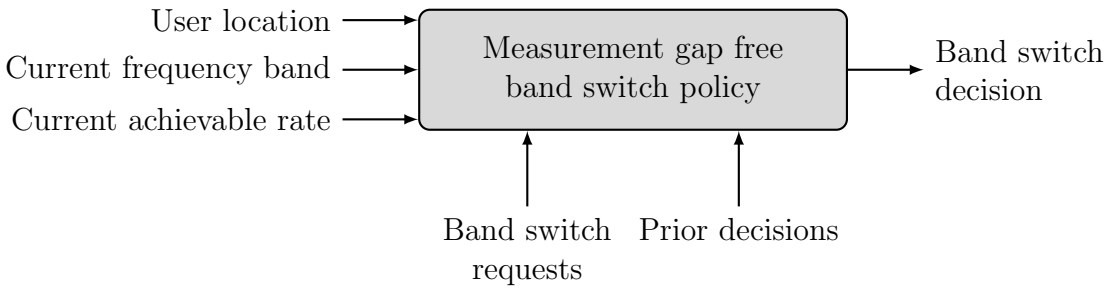


Figure 4.3: Illustration of my proposed algorithm. The list of learning features is shown in Table 4.1.

$\alpha\|\mathbf{w}\|_1 + \frac{1}{2}\lambda\|\mathbf{w}\|_2^2 + \gamma T$, where $\mathbf{w}$ is the vector containing the leaf weights in the boosted tree, α and λ are the regularization terms for their respective norms, γ is the complexity control, and T is the number of leaves.

Classifier training: I train the hyperparameters of the classifiers using grid search and K-fold cross-validation. The list of learning features is shown in Table 4.1. Let the feature matrix be denoted by $\mathbf{X} \in \mathbb{R}^{N_{\text{UE}} \times p}, p > 2$. The industry standards require two features for the band switch decision, as I showed earlier. The supervisory label vector is a column vector $\mathbf{y} \in \{0, 1\}^{N_{\text{UE}}}$, where 0 means the band switch was denied and 1 means granted based on (4.7).

My proposed approach is shown in comparison to the legacy approach in Fig. 4.4 and it operates in two phases: a) learning phase and b) exploitation phase.

Learning phase: In the learning phase, the UE follows the legacy approach discussed earlier while the proposed algorithm stores the learning features $\mathbf{X}$ and $\mathbf{y}$. Machine learning is then applied on this data to build

Algorithm 3: Measurement Gap Free Band Switch Policy

Input: Parameters listed in Table 4.2 and Table 4.3. Instantaneous rates by UE location per frequency band (or technology) of a given region. Band switch threshold rate. Set of all UEs $\mathcal{N}$ and target UEs $\mathcal{U} \subset \mathcal{N}$.

Output: A vector $\hat{\mathbf{y}}$ containing a prediction whether the measurement gap-free band switch should be granted or denied for the set $\mathcal{N} \setminus \mathcal{U}$ in the proximity of that region, a confusion matrix $\mathbf{C}$, and the area under the receiver operating characteristic (ROC) curve for the set.

1 $N_{\text{learning}} := \lceil (1 - r_{\text{exploitation}}) \cdot N_{\text{UE}} \rceil$

2 At random, sample with no replacement a set users from $\mathcal{N}$ which has N_{UE} users.

3 Build the dataset $[\mathbf{X} \,|\, \mathbf{y}]$ for UEs $\{1, 2, \ldots, N_{\text{learning}}\}$, where $\mathbf{X}$ is in Table 4.1 and $\mathbf{y}$ is based on (4.8), both for all N_{learning} UEs.

4 Randomly split the data $[\mathbf{X} \,|\, \mathbf{y}]$ into a training and a test data (using r_{training} split ratio).

5 Train the DNN classifier using the training data and use grid search on K-fold cross-validation to tune the hyperparameters based on the binary cross-entropy loss function [2].

6 **forall** $u \in \mathcal{N} \setminus \mathcal{U}$ **do**

7 | Use the DNN classifier to predict $\hat{y}^{(u)}$ based on $\mathbf{X}^{(u)}$.

8 **end**

9 Obtain ξ the area under the ROC curve for this model using $\hat{\mathbf{y}} := [\hat{y}^{(u)}]$.

10 Build the confusion matrix $\mathbf{C}$ by observing $\mathbf{y}$ and $\hat{\mathbf{y}}$.

a classifier that estimates band switch decisions but without the need for a measurement gap. During this phase, I let all UEs request band switches by setting $\mathbf{x}_7$ to unity (or $r_{\text{threshold}}$ to $+\infty$). This is in order for the classifier to learn the relationship between channels regardless of band switch requests. I use both DNN and XGBoost classification models in the implementation of this phase.

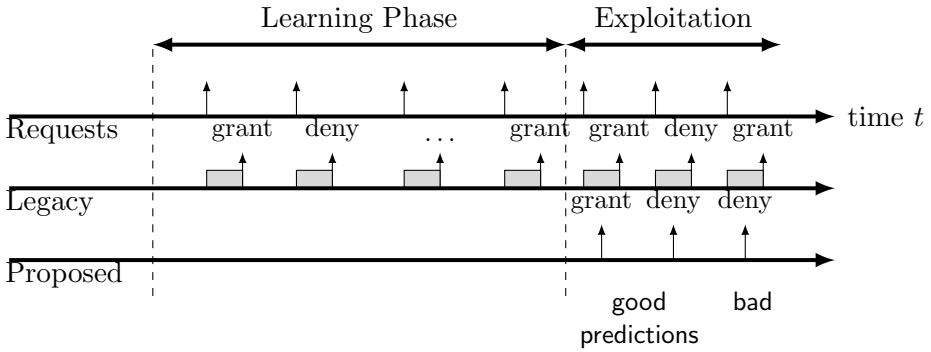


Figure 4.4: Proposed band switch time diagram. The shaded gray rectangles represent the transmission gaps.

Exploitation phase: In the exploitation phase, the classifier uses prediction to eliminate the measurement gap but for the set of UEs $\mathcal{U}$ which were not used in the learning phase, as stated earlier. This set $\mathcal{U}$ is a group of UEs that are geographically close to the ones in the learning phase which can benefit from the spatial correlation of the channels. The predicted decision either grants or denies the band switch from the j-th band. The exploitation phase essentially represents the generalization capacity of the classifier.

4.6 Performance Measures

In this section I describe my choice of the performance measures to benchmark my algorithm. These measures describe the performance of both the QoE of the users and the performance of the classifier.

4.6.1 Effective Achievable Rate

I evaluate the effective rate of all the users in the network using (4.5) employing the different polices I discussed. I am interested in the statistics of the effective achievable rates. Namely, the cumulative distribution function (CDF) and the mean.

4.6.2 Confusion Matrix

I define the misclassification count $E\colon 0 \leq E \leq n$ as the number of incorrectly predicted band switches during the exploitation phase. I build a confusion matrix $\mathbf{C} \in \mathbb{Z}_{+}^{2\times 2}$ having the true and predicted band switch decision counts and write

$$E := \mathrm{Tr}(\mathbf{J}\mathbf{C}^{\top}), \tag{4.14}$$

where $\mathbf{J}$ is a 2×2 anti-diagonal identity matrix. The misclassification error μ can be derived by dividing E by $n := \lfloor r_{\text{exploitation}} N_{\text{UE}} \rfloor$.

4.6.3 Receiver Operating Characteristic Area Under the Curve

The receiver operating characteristic (ROC) curve is a two-dimensional curve used to visualize classifiers based on the tradeoff between hit rates and false alarm rates. To compare the performance of classifiers, I reduce the ROC performance to a single scalar quantity known as the ROC area under the curve [102]. This area $\xi\colon 0.5 \leq \xi \leq 1.0$ where 0.5 means the classifier is as good as a random guess and 1.0 means it produces perfect prediction.

So far, I have discussed the different band switch policies, including my

proposed policy. I have highlighted the desired performance measures I am interested in. In the next section, I discuss the data I use and how I construct in detail.

4.7 Dataset Construction

To test the performance of the proposed algorithm, I rely on the DeepMIMO dataset [94]. The choice of this dataset is based on its use of accurate ray-tracing tools to generate spatially and spectrally correlated channels for specific scenarios. Hence, I avoid using oversimplified mathematical models that could lead to misleading results. A better choice would be to use a dataset that is based on actual field-measurements. However, such dataset is not available yet to the best of my knowledge, and is highly non-trivial to generate.

In the O1 outdoor scenario of the DeepMIMO dataset, the UEs are placed on a uniform grid on a main street for both the sub-6 GHz and mmWave frequency bands, where the BS uses OFDM and uniform planar array (UPA) antennas. The adopted O1 scenario is shown in Fig. 4.5.

4.7.1 Channel Coherence Time

The channel coherence time over which the channel remains highly correlated, is known to be given by [103]

$$T_C(\alpha) \approx \frac{c}{f_c v_s \sin \alpha}, \tag{4.15}$$

where c is the speed of light, v_s is the speed of the UE, α is the angle between the direction of travel and the direction of the BS, $v_s \sin \alpha$ is the relative speed of the user with regards to the BS, and f_c is the center frequency. This equation has been widely used to measure the channel coherence time for the sub-6 GHz range, where omni-directional antennas are used. However, at mmWave, where directional antennas along with beamforming are employed to combat the high isotropic path loss, (4.15) does not accurately measure coherence time [104]. This is because by combining directional antenna arrays with beamforming, the signal power is focused on a beamwidth-defined angular space directed towards the UE location. Hence, only the variations in the channel within this angular space are relevant, which increases the channel coherence compared to (4.15). The coherence time of the beamformed channel, referred to as the beam coherence time, is given by [104]

$$T_C(\alpha) \approx \frac{D}{v_s \sin \alpha} \frac{\Theta}{2}, \tag{4.16}$$

where D is the Euclidean distance from the serving BS and Θ is the beamwidth of the beams used by serving BS (in radians). Since UEs are located at different locations with different distances to the BS, they have different coherence times. However, to maintain a fixed frame length for all users connected at the same band, I assume the cell-center beam coherence time (i.e., the 1^{st} percentile). This conservative assumption is also motivated by the practical case where the BS may not have full knowledge of the UE parameters, such as their distance and accurate location. To sum up, I assume that the coherence

time for sub-6 GHz and mmWave bands is given by

$$T_C^{\text{sub-6}} := \left(\frac{c}{f_c v_s \sin \alpha} \right)_{0.01}, \tag{4.17}$$

$$T_C^{\text{mmWave}} := \left(\frac{D}{v_s \sin \alpha} \frac{\Theta}{2} \right)_{0.01}, \tag{4.18}$$

respectively, where $(\mathcal{X})_{0.01}$ is the 1^{st} percentile of the set $\mathcal{X}$. For the frame time, I set the frame duration to be equal to the channel coherence time for simplicity. Hence, the overheads for beam training and band switch, mentioned earlier, are related to a single parameter, which is the coherence time.

4.7.2 Band-Selective Blockage

Further, I choose to have occasional blockage in the mmWave frequency band. To generate this blockage using DeepMIMO, I generate two channels: one with blockage and the other without blockage. I further combine the mmWave channels into one by introducing a Bernoulli random variable for the i-th UE:

$$b_i \sim \text{Bernoulli}(p), \qquad i = 1, 2, \ldots, N_{\text{UE}} \tag{4.19}$$

$$\mathbf{h}^{(i)}[t] = b_i \mathbf{h}_b^{(i)}[t] + (1 - b_i)\mathbf{h}_{nb}^{(i)}[t], \tag{4.20}$$

where p is the blockage probability, $\mathbf{h}_b^{(i)}$ is the channel with blockage on the first mmWave path, and $\mathbf{h}_{nb}^{(i)}$ is the channel with no blockage, all for the i-th UE. Hence, some locations along the street are assumed to be blocked from the BS (non-line of sight), while others have a line of sight.

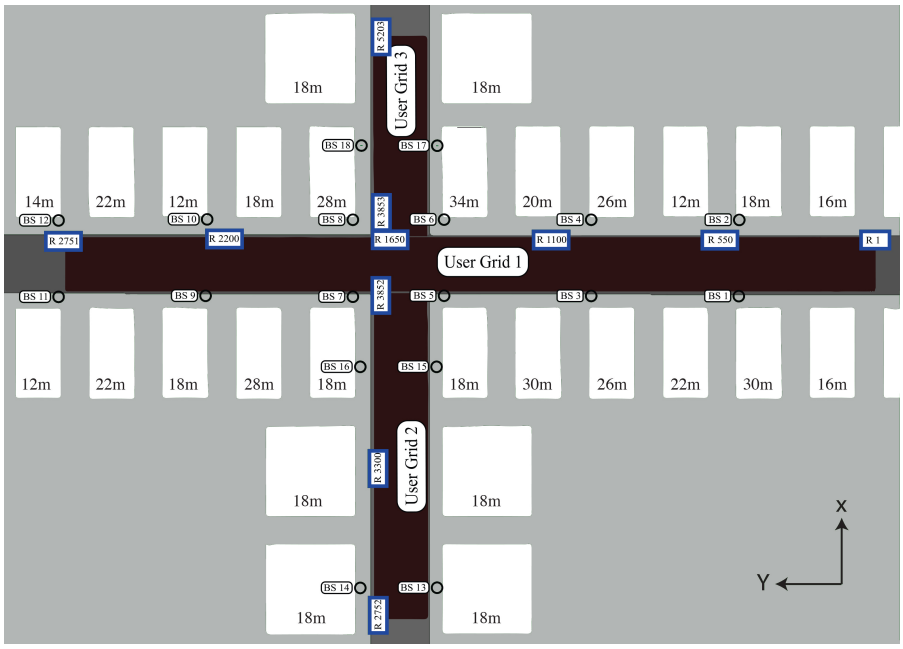


Figure 4.5: Scenario O1 of the DeepMIMO dataset. I use base station (BS) 3 and users on User Grid 1.

4.7.3 Analog Beamforming

I adopt a multi-antenna setup, where the BS employs a UPA of $M_y^{(j)}$ and $M_z^{(j)}$ antennas in the elevation and azimuth directions respectively at the j-th band. Therefore, I write the channel in (4.1) as $\mathbf{h} \in \mathbb{C}^{M_y^{(j)} M_z^{(j)} \times 1}$ in a vectorized form. In my implementation of analog beamforming, I focus on discrete Fourier transform (DFT) codebooks. I focus on DFT codebooks because they are simple to implement. Let the $M \times N_{\text{CB}}^{(j)}$ matrix $\mathbf{F}^{(j)}$ be the concatenation of M beamforming vectors in the codebook $\mathcal{F}^{(j)}$, then the matrix $\mathbf{F}^{(j)}$ is constructed as

$$\mathbf{F}^{(j)} = \mathbf{F}_z^{(j)} \otimes \mathbf{F}_y^{(j)} \tag{4.21}$$

Table 4.2: Predictive band switching algorithm – radio environment parameters

Parameter	Value
Subcarrier bandwidth (sub-6, mmWave)	$(180, 1800)$ kHz
Center frequency	(3.5, 28) GHz
UE noise figure	7 dB
DeepMIMO Scenario O1 Base Station	3
DeepMIMO Scenario O1 number of antennas (M_x, M_y, M_z)	$(1, 64, 4)$
DeepMIMO Scenario O1 OFDM limit	64
Band switch threshold for sub-6 GHz $r_{\text{threshold}}^{\text{sub-6}}$	1.72 Mbps
Band switch threshold for mmWave $r_{\text{threshold}}^{\text{mmWave}}$	7.00 Mbps
Measurement gap fraction of coherence time ρ	0.6

where $\mathbf{F}_y^{(j)} \in \mathbb{C}^{M_y^{(j)} \times M_y^{(j)}}$ and $\mathbf{F}_z^{(j)} \in \mathbb{C}^{M_z^{(j)} \times M_z^{(j)}}$ concatenate the DFT codebook vectors in the y and z directions for the j-th frequency band. In the next section, I use this dataset to evaluate the performance of the proposed algorithm and compare it with the other algorithms discussed earlier.

4.8 Simulation Results

In this section, I evaluate the performance of my proposed algorithm in simulations using several performance measures outlined in Section 4.6.

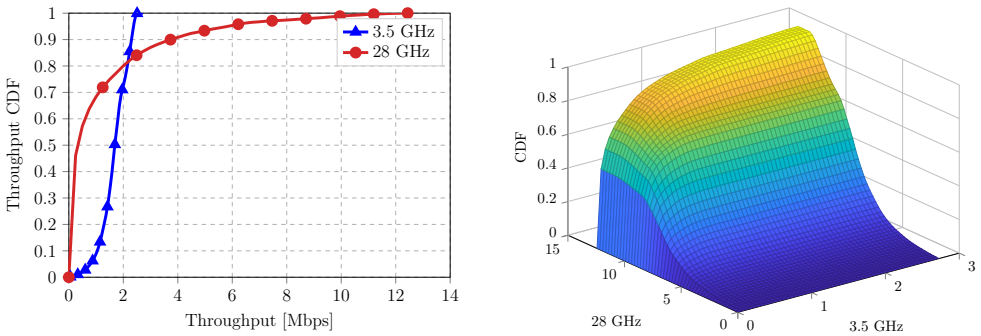
4.8.1 Setup

The DNN and XGBoost classifier hyperparameters are both shown in Table 4.3. As mentioned earlier, the users are placed on a uniform grid on a main street in the service area of this co-sited BS such that the i-th UE has the Cartesian coordinate (x_i, y_i, z_i). The height of all UEs $z_i = 2$ m is constant throughout the simulation. The users move at a vehicular speed

Table 4.3: Hyperparameters of classifiers used in the implementation of my algorithm

DNN		XGBoost	
Parameter	Value	Parameter	Value
Exploitation split $r_{\text{exploitation}}$	0.8	Exploitation split $r_{\text{exploitation}}$	0.8
K-fold cross-validation K	2	K-fold cross-validation K	2
Optimizer	[105]	ℓ_1 regularization term α	{0,1}
Learning rate η	0.05	ℓ_2 regularization term λ	{0,1}
Activation function $\sigma(\cdot)$	sigmoid	Complexity control term γ	{0,0.02,0.04}
Depth of neural network d	{1,3,5}	Sample weights	{0.5,0.7}
Width of the hidden layer w	{3,5,10}	Child weights	{0,10}

$v_s = 50$ km/h within the BS service area every discrete time step t. In an attempt to find the training ratio that best maximizies the ROC area, I choose from $r_{\text{training}} \in \{1 \times 10^{-3}, 5 \times 10^{-3}, 7 \times 10^{-3}, 1 \times 10^{-2}, 3 \times 10^{-2}, 5 \times 10^{-2}, 7 \times 10^{-2}, 0.1, 0.3, 0.4, 0.5, 0.7\}$. I set the beam training time $T_{\text{beam}} := 1$ μs [104]. With $N_{\text{SC}} = 12$ OFDM subcarriers per PRB and a subcarrier spacing $\Delta f = 15$ kHz, I have the bandwidth $B = 180$ kHz per PRB. I use one PRB for the sub-6 GHz frequency band and ten PRBs for the mmWave frequency band. In other words, $B^{\text{sub-6}} = 180$ kHz and $B^{\text{mmWave}} = 1800$ kHz. The coherence times for sub-6 GHz and mmWave based on (4.17) and (4.18) are 6.17 and 19.16 ms respectively. I choose the transmit energy of 0.1W/Hz at mmWave and set the transmit energy at sub-6 GHz to 1W/Hz. Further, I arbitrarily set the band switch thresholds as 1.72 Mbps and 7.00 Mbps for sub-6 GHz and mmWave. The exploitation ratio $r_{\text{exploitation}}$ of 0.8 means that the data available for training and testing is 20% of the entire dataset. Using the training ratios r_{training} above, the proportion of data used towards learning



(a) Marginal distributions **(b)** Joint distribution

Figure 4.6: Instantaneous throughput distributions for both sub-6 and mmWave frequency bands.

does not exceed $20\% \times 0.7 = 14\%$. In the learning phase, I set the band switch request threshold to $+\infty$. This allows us to capture all the available spatial correlation information between the channels without any omission.

I simulate the radio environment (given in Table 4.2) using three different scenarios:

- *Scenario A:* All users start in sub-6 and attempt to change band to mmWave.
- *Scenario B:* All users start in mmWave and attempt to change band to sub-6 GHz.
- *Scenario C:* 70% are in sub-6 and 30% are in mmWave.

I refer to the source code for the details of the implementation of this simulation [106]. Before presenting the results for these scenarios, I start with

analyzing the raw data from the dataset. In Fig. 4.6, I show the distribution of the effective throughput over all users for the 3.5 GHz and the 28 GHz bands. From the marginal distributions, I can see that the effective throughput for the mmWave bands goes up to 12 Mbps, while it only goes up to 3 Mbps for the 3.5 GHz band. This is due to the large bandwidth that is available in the mmWave band. However, due to blockage, these high rates only occur with a small probability, since the two CDF curves cross at 0.8. Overall, the figure shows that the effective rate at mmWave can be very high, due to the large bandwidth, or very low, due to the blockage. This wide range of rates motivates the optimized design of the band switch policy, since ineffective design can cause a big deterioration in the UE throughput, or can prevent the user from harvesting a high rate from mmWave bands.

From the joint distribution, the general trend is that a higher throughput on 3.5 GHz means a higher throughput on 28 GHz. This is due to the correlation between the channels caused by common paths, reflectors, and obstructions. By simple computations using the joint distributions, one can see that the 25% of the users can get higher throughput on the 28 GHz band. Intuitively, these are the users who do not suffer from blockage and are at a short distance from the BS. These users would benefit from operating at mmWave. To quantify this gain, I plot Fig. 4.7, which shows the distribution of the absolute difference between the throughput at 3.5 GHz and 28 GHz, $|\Delta R_E| := |R_E^{\text{sub-6}} - R_E^{\text{mmWave}}|$. As the figure shows, the difference between throughput can go up to 10 Mbps, which can have a detrimental impact on

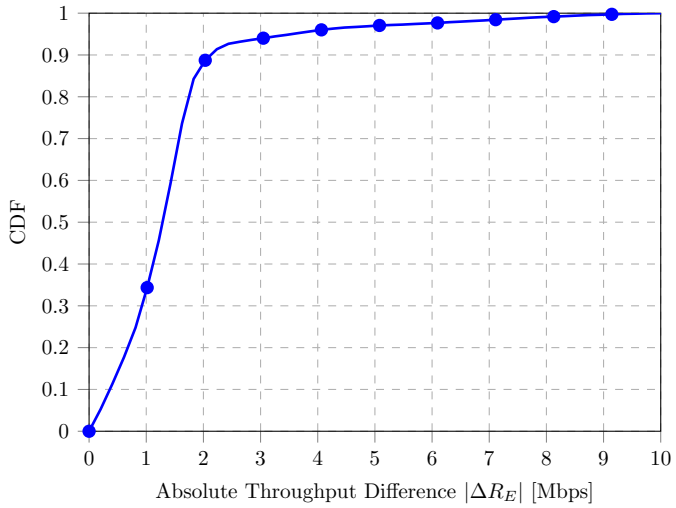


Figure 4.7: The distribution of the absolute difference between the rate on 3.5 GHz and 28 GHz.

the users throughput and motivates a careful design for the band switch policy. Next, I analyze the performance of the different band switch policies discussed earlier.

4.8.2 Band Switch Polices

4.8.2.1 Scenario A

I start with Scenario A, where all the users start in 3.5 GHz band and can request a band switch to the 28 GHz band. The results are shown in Fig. 4.8, where I show the distribution of the effective throughput under different band switch policies. Starting by the legacy approach, the effect of the measurement gap is clear in the figure and results in a performance gap compared to the optimal policy, especially in the low rate regime, where the negative impact of the measurement gap is detrimental. To be precise, the

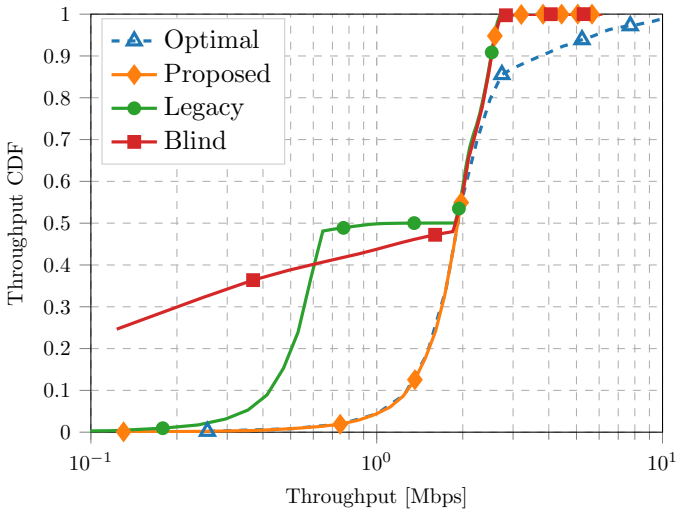


Figure 4.8: The distribution of the effective throughput for Scenario A under different band switch polices.

effect is more dominant for users suffering from a low throughput at sub-6 GHz and requested a band switch from the BS, but their request got denied because the throughput at mmWave is also low, possibly due to blockage. Hence, having a measurement gap makes their throughput even worse. This can be observed from the curve for the small throughput regime. Moving to Fig. 4.8, I observe that there are more users in the small throughput range (i.e., less than 0.5 Mbps) in the blind policy than the legacy policy. This can be justified as follows: for the same class of users suffering from extremely low throughput at sub-6, but would not benefit from switching to mmWave, the legacy approach instructs them to stay in the sub-6 GHz band at the expense of a measurement gap, while the blind policy switches these users to the mmWave band, which deteriorates their throughput even more. However,

there is a point where the throughput at mmWave is around the same as the sub-6 GHz. At this point, the blind policy is more efficient since it has the advantage of not requiring a measurement gap.

For the proposed algorithm, Fig. 4.8 also shows that it has the best performance compared to the previous two; it is identical to the optimal in the low rate regime and identical to the other policies in the high rate regime. This is due to: (i) the elimination of the measurement gap, hence users with low throughput do not suffer more if their band switch request got denied as in the legacy approach, and (ii) the accurate band switch decisions, which prevent switching users to a band with low throughput as in the blind policy. Note that there is a performance gap between the three policies and the optimal in the high rate regime. This performance gap is due to the band switch threshold introduced in these policies, but missing from the optimal. Hence, users with high throughput on sub-6 GHz do not benefit from the higher throughput on mmWave bands following these policies. But this is not the case for the optimal algorithm, since the BS picks the band with the maximum throughput each frame without a threshold. However, as I will show later, my proposed algorithm can overcome this issue by increasing the band switch threshold, without losing its accuracy. Finally, to quantify the gains provided by the different band switch policies, I list the mean effective throughput in Table 4.4. Based on the values for Scenario A, the proposed algorithm provides a gain of 39% and 37% over the blind and the legacy policies, respectively, and just 20% behind the optimal algorithm. Overall, the results for this scenario

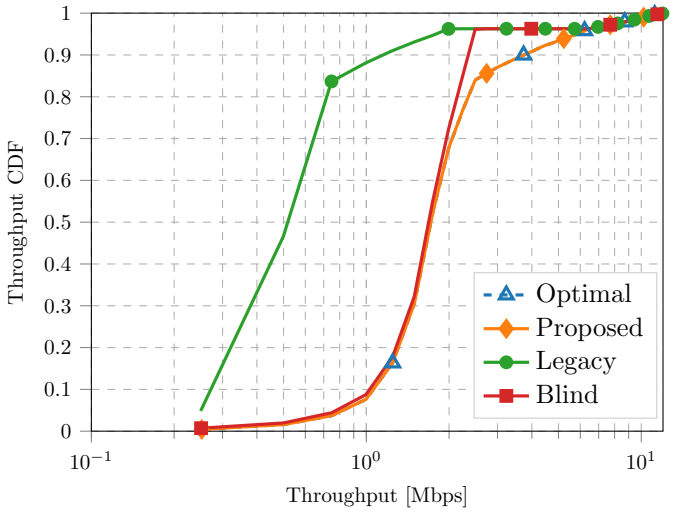


Figure 4.9: The distribution of the effective throughput for Scenario B under different band switch polices.

Table 4.4: Normalized mean effective throughput for different scenarios

	Normalized mean effective throughput R_E [Mbps]			
Scenario	Legacy	Blind	Proposed	Optimal
Scenario A	0.55	0.54	0.75	1.00
Scenario B	0.43	0.88	1.00	1.00
Scenario C	0.35	0.76	1.00	1.00

are promising and show the effectiveness of the proposed algorithm.

4.8.2.2 Scenario B

For the second scenario, the results are shown in Fig. 4.9. There are a few differences between this scenario and Scenario A. Firstly, all policies achieve the same performance in the high rate regime, which is due to the assumption that all users start in the mmWave band. To be precise, the high throughput regime (above 4 Mbps) can only be achieved on mmWave

bands as shown in Fig. 4.6. Hence, given that the users start in the mmWave band and this range is above the band switch threshold, these users remain in mmWave regardless of the policy, which justifies the identical performance of the different policies in the high throughput regime. Secondly, the blind achieves an identical performance to the optimal for for the low throughput region. To justify this, I need the data in Table 4.5, which show the number of band switch requests and the number of the granted ones for each policy and each scenario. From this table, I can see that around 70% of the band switch requests are granted in this scenario assuming the optimal policy. Hence, the blind algorithm is identical to the optimal 70% of the time. Among this 70% of the users, are the users who suffer from extremely low throughput at mmWave, mostly due to blockage. Hence, the blind algorithm results in the optimal decision for these users which justifies the identical performance for the low throughput regime. However, the blind also makes the wrong decision 30% of the time, which results in a gap between this policy and the optimal in the medium throughput regime. Finally, the legacy is always the worst in this scenario, which is due the measurement gap and the fact that the blind is accurate 70% of the time without having a measurement gap.

The means of the effective throughput are also shown in Table 4.4. The proposed policy achieves 130% of throughput compared to the legacy policy and 13% compared to the blind policy. Also, the effective throughput for the proposed is almost identical to the optimal.

Table 4.5: Band switch grants and requests based on the band switch policy and the scenario

Scenario	Policy	Band switches requested	Band switches granted
Scenario A	Legacy	22,458	2,751
	Blind	22,458	22,458
	Proposed	22,458	2,724
Scenario B	Legacy	41,609	32,558
	Blind	41,609	41,609
	Proposed	41,609	32,569
Scenario C	Legacy	43,033	21,514
	Blind	43,033	43,033
	Proposed	43,033	21,619

4.8.2.3 Scenario C

The results for this scenario are presented in Fig. 4.10. As expected, the results lie between the previous two, and all the curves can be justified using the same arguments I above. The reason I include this scenario is to have an idea on the gains I might observe in practice, since part of the users will be using mmWave and the others using sub-6 GHz. The mean gains are also presented in Table 4.5.

Overall, the results for the different scenarios show the superiority of the proposed policy compared to the legacy and the blind policies; up to 130% improvement in the effective throughput depending on the considered scenario. It also justifies the use of a machine learning approach to solve this problem. Next, I provide more technical discussions on the accuracy of the proposed algorithm, more insights, and possible extensions to this work.

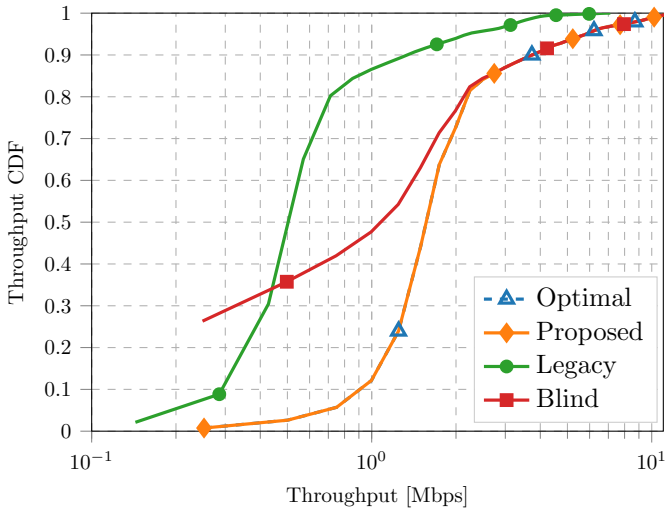
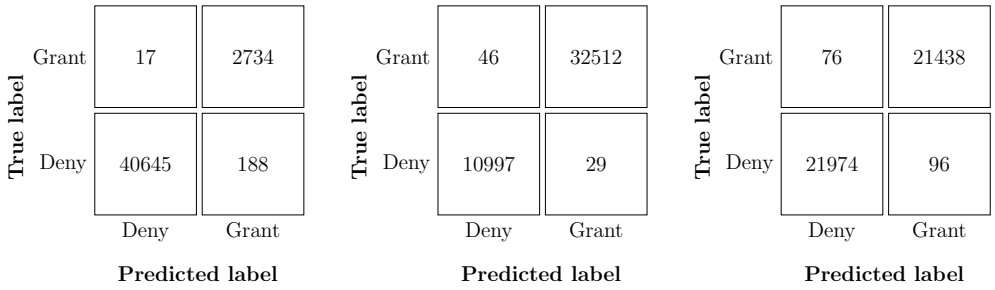


Figure 4.10: The distribution of the effective throughput for Scenario C under different band switch polices.

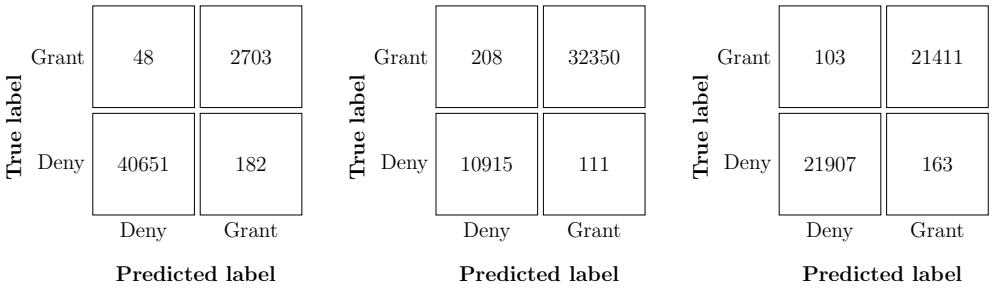


(a) Scenario A **(b)** Scenario B **(c)** Scenario C

Figure 4.11: The confusion matrix **C** for the three scenarios using DNN.

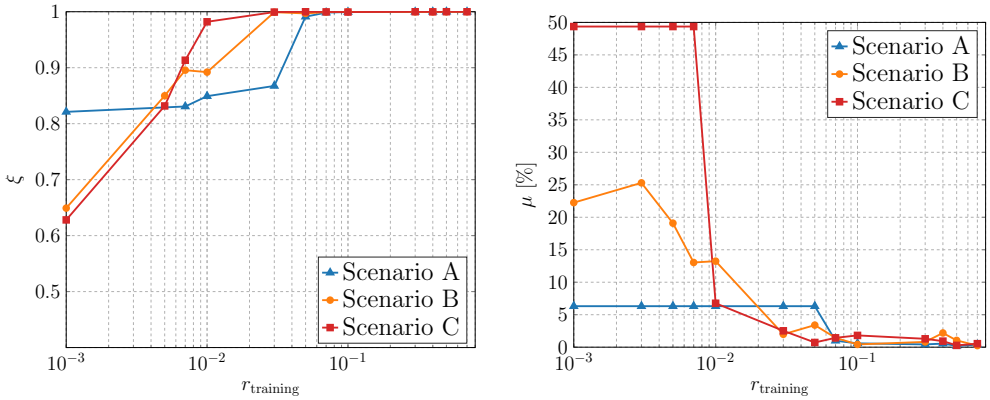
4.8.3 Discussion

I start with discussing the predictive accuracy of the proposed algorithm. Fig. 4.11 shows that my proposed algorithm usually made the right decisions; only a very few times did it deny the band switch when it was supposed to grant it (and vice versa).



(a) Scenario A **(b)** Scenario B **(c)** Scenario C

Figure 4.12: The confusion matrix **C** for the three scenarios using XGBoost.



(a) ROC Area ξ **(b)** Misclassification error μ

Figure 4.13: The classification performance of the proposed algorithm for different training data sizes and different scenarios.

To compare with other ML classifiers, I show the performance of XGBoost alongside DNN. In Fig. 4.11 and Fig. 4.12, I show the confusion matrix for the three considered scenarios using DNN and XGBoost respectively. Precisely, the misclassification error (μ) using DNN (XGBoost) is 0.47% (0.53%), 0.17% (0.73%), and 0.39% (0.61%) for Scenarios A, B, and C, respectively. The run-time complexity of XGBoost using the hyperparameters in Table 4.3

has an upper bound in $\mathcal{O}(n + n \log n) = \mathcal{O}(n \log n)$ [86], where $n := N_{\text{UE}}$. However, for DNN this complexity is in $\mathcal{O}(nw^d)$ [82], where d and w are the depth and width of the DNN, respectively. Hence, the classifier choice between DNN and XGBoost is a trade-off between decision speed and accuracy: if accuracy is desired, then choose DNN, but if less run-time complexity is desired for decision speed, XGBoost is a more attractive choice.

The second point I highlight here is the amount of training data that I require to have an accurate prediction. In Fig. 4.13, I show the ROC area (ξ) and the misclassification error (μ) for different training data sizes. In particular, the figure shows that training using only 1/40 of the data is enough to have an excellent performance—less than 2% misclassification error. In other words, having knowledge about the previous band switch decisions for 1/40 of the locations at random is enough to predict the band switch decisions for the rest of the locations. This ratio depends on the spatial correlation between the channels on different locations, as well as the hyperparameters and the choice of the machine learning algorithm.

Note that my presented results so far are for a single band switch threshold value. However, I claim that the performance gap between the optimal algorithm and the proposed one can be reduced by increasing the threshold. To verify this claim, I show the mean effective throughput for different band switch thresholds in Table 4.6. In Scenario A, I observe that as I increase the band switch threshold $r_{\text{threshold}}$, the performance gap between the mean effective throughputs of the proposed and the optimal rates shrink considerably.

Table 4.6: Impact of different band switch thresholds for Scenarios A and B

		Normalized mean effective throughput R_E [Mbps]			
	$r_{\text{threshold}}$	Legacy	Blind	Proposed	Optimal
Scenario A	1.72	0.55	0.54	0.75	1.00
	2.00	0.45	0.46	0.77	1.00
	2.60	0.34	0.60	1.00	1.00
Scenario B	2.00	0.43	0.88	1.00	1.00
	9.00	0.39	0.84	1.00	1.00
	12.50	0.33	0.76	1.00	1.00

While both the legacy and blind rates also get better, their performance is not close to the optimal: the legacy because of the measurement gap and the blind because of the undesired band switch. However, in Scenario B, both the blind and legacy rates deteriorate as I increase the band switch thresholds. In the legacy policy, it is also due to the measurement gaps, and for the blind policy, it is because users who were getting up to 10 Mbps on mmWave are now getting 3 Mbps at best as shown in Fig. 4.6. As expected, I do not see much of a change in the proposed rate as I increase the band switch threshold, aligned with the CDFs in Fig. 4.7 and Fig. 4.9.

4.9 Conclusions

In this chapter, I used both deep neural networks and XGBoost classifiers to rank the downlink channel quality of the frequency bands prior to the band switch, which is a mathematically intractable problem. The use of classifiers eliminate the dependence on measurement gaps during a band switch in a dual-band base station. I exploited the spatial and spectral relationships

in both the sub-6 GHz and mmWave bands through the use of a ray-tracing dataset. I revealed insights as to why the deep learning classification method was needed and why it worked. I simulated one dual-band base station with many UEs in its association area. In this simulation, my method improved downlink throughput by up to 1.3x compared with the legacy policy over different scenarios with a misclassification error less than 0.3%. The observed improvement is due to the classifier ability to exploit the spatial correlation of channels across the different frequency bands and thus accurately predict the effective achievable rate on the target frequency without the dependency on a measurement gap. This band selection method is better suited for 5G and beyond where maintaining high data rates is desired without interrupting the data flow. I focused on the case where the BS has only two bands: one centered at 3.5 GHz and the other at 28 GHz, since the dataset I use supports these two bands. An interesting extension is for multiple bands, or when a handoff between multiple BSs is required.

In the next chapter, I close with concluding remarks about my dissertation. I summarize my contributions and offer a few possible directions for future research.

Chapter 5

Concluding Remarks

5.1 Summary

This dissertation focused on improving the performance in next-generation wireless communication networks using deep learning. These choice of deep learning in next-generation wireless networks was motivated by 1) absence of accurate mathematical formulations of channels, 2) incremental changes in radio resource management (RRM) algorithms over the evolution of successive network standards, and 3) operators' desire to build autonomous networks. The summary of my contributions are shown in Table 5.1. First, I proposed a joint beamforming, power control, and interference coordination algorithm using deep reinforcement learning for both voice and data bearers. Then, I developed a method to improve the performance of the coordinated multipoint (CoMP) using a data-driven approach, different physical layer quantities, and deep neural networks. Finally, I developed the employment of this approach by introducing ray-tracing datasets to perform predictive band switching.

For the beamforming and power control problem, I developed a formulation of the joint design of beamforming (BF), power control (PC), and interference coordination (IC) as a non-convex optimization problem to max-

Table 5.1: Summary of contributions

	Dissertation Contributions		
	1. Joint BF, PC, IC	2. Improved CoMP	3. Band Switching
Chapter	2	3	4
Reference	[107]	[52]	[108]
Frequency band	mmWave and sub-6 GHz	sub-6 GHz	mmWave and sub-6 GHz
Stack layer*	PHY	PHY	RRM
Algorithm†	DRL	SVM and DNN	DNN and XGBoost
Direction	downlink		
User	multi-user		

* PHY is the physical layer and RRM is the radio resource management layer.
† DRL is deep reinforcement learning, SVM is support vector machines, DNN is deep neural networks, and XGBoost is extreme gradient boosting.

imize the signal to interference plus noise (SINR) per user and solved this problem using deep reinforcement learning. The algorithm used the reported SINR from connected users, the transmit powers of the base stations (BS), and the coordinates of the connected users to improve the performance measured by coverage and sum-rate capacity. My proposed algorithm did not require the channel state information and relaxed the need for channel estimation—a requirement for optimal beamforming. Simulation results showed that my algorithm outperformed the link adaptation industry standards for sub-6 GHz voice bearers and approached the optimal limits for mmWave data bearers but without an exhaustive search in the action space. This led to a reduction of the computational run time to 4% of the optimal solution run time depending on the antenna size.

For the improved downlink coordinated multipoint problem, I devel-

oped a surrogate CoMP triggering function that, unlike industry standards, configures the rank of transmission to the user equipment (UE) not based on the reported SINR but factors in the retransmission rate of the codewords on these streams. I motivate the choice of using deep neural networks by comparing the network performance against using support vector machines. Deep neural networks converts the reported channel state information and the codewords retransmissions into a classification problem using the retransmission targets set by the industry standards. The surrogate CoMP triggering function then either triggers a rank-two transmission or maintains a rank-one transmission based on the outcome of the classifier. In simulation, I showed an improved throughput distribution, which was the result of the feature interaction created by the deep neural network. The improvement was approximately 13.5% of the SINR-triggered industry baseline.

Finally, I showed that the dual-band predictive band switch procedure between both sub-6 GHz and mmWave bands can be formulated as a ranking problem. This allowed me to introduce deep learning to band switching. The spatial and spectral correlation between radio frequency signals at different bands was exploited to propose a classifier-based band switching policy. This policy ranked the frequency bands by their channel quality. I built a unified framework to describe the impact of the band switching policies on the effective achievable rates. I exploit the fact that band switching can benefit from the ranking of channels instead of accurate channel estimations. The use of classifiers eliminated the dependency on measurement gaps for band switch-

ing, which improved these rates. I used two popular classification models and showed the band switching trade-off between both models. I showed a key insight about the employment of measurement gaps in band switching, since only the order of the bands is necessary for this procedure, rather than a full channel estimate. The results showed that my proposed band switch thresholds provided 30% gain in the mean effective rates compared with the industry standard policy rates, while achieving misclassification errors well below 0.5%.

5.2 Future Work

In my dissertation, I showed some applications of deep learning in communications. However, there are still a myriad of performance problems in wireless networks that result in mathematically intractable problems, and adopting a data-driven approach poses viable solution. A caveat to this data-driven approach to work is the aforementioned key challenges. I present several possible research directions related to this dissertation next.

Multi-user joint beamforming, power control, and interference coordination: My work in Chapter 2 discussed the use of reinforcement learning and a string of bits with binary logic operations to formulate a joint set of actions for both packetized voice and data bearers. I also proposed one user per base station. To assess the scalability of my proposed algorithm, increasing the number of users per base station is one way forward.

Improved cell-free massive MIMO: In Chapter 3, I discussed the use of joint transmission to formulate a network-based MIMO channel. I also

propose to improve the cell-free massive MIMO operation. Cell-free MIMO can benefit from joint user-specific beamforming using deep learning, where the beamforming vectors of the coordinated base stations can be derived using deep reinforcement learning without full knowledge of the downlink channel.

Generalized multi-band predictive handovers: My work in Chapter 4 eliminated the need for the measurement gaps for dual-band BS band change (i.e., vertical handover) procedures. However, another procedure that is relevant is the inter-BS handover procedure. Handovers are a result of the mobility of the UEs away from the service area of one BS to the service area of another. Exploiting the spatial correlation of channel measurements and the use of channel signatures with the aid of deep learning can help build predictors that generalize measurement gap free handovers. Further, instead of focusing only on a dual-band configuration, the use of deep learning to rank multiple frequencies belonging to the same band (e.g., mmWave on 39 GHz and 60 GHz) exploiting the spatial correlation between the channels can be very interesting. This work is likely to widen the gap with the legacy based algorithm, where longer measurement gaps are required to measure several target frequencies.

Optimal hybrid beamforming: In Chapter 2, I proposed the use of deep reinforcement learning to find the optimal analog beamforming vector from a codebook of DFT beamforming vectors, which are simple to implement. In hybrid digital/analog beamforming, the solution to derive the precoder and the combiner is by solving for each independently. One way forward is to com-

pute jointly solve for the precoder and the combiner using a deep reinforcement learning approach and eliminate the dependence on analog beamforming.

One relevant direction to consider further is an NR link-level simulator software release completely built in Python using ray-tracing propagation model data. The importance of reproducible research continues to grow. With the proliferation of deep learning and the various machine learning libraries, new software languages have emerged. Most cellular network system simulators nowadays are written in MATLAB. Porting Python to MATLAB works but with slow execution times. Further, some of these simulators use statistical propagation models which do not clearly capture implicit effects across frequency bands or do not benefit from the graphics processing unit acceleration.

In conclusion, the application of deep learning to next generation wireless networks does not end at NR. The doors are still wide-open for deep learning applications that benefit from multi-access edge computing, wireless sensors, video predictive caching and processing for wireless users, and radio resource management beyond the fifth generation of wireless communications.

Appendix

Publications

Publications related to the dissertation

1. F. B. **Mismar** and B. L. Evans, "Deep Learning in Downlink Coordinated Multipoint in Heterogeneous Networks," *IEEE Wireless Communication Letters,* vol. 8, no. 4, pp. 1040-1043, Aug. 2019.
   doi: 10.1109/LWC.2019.2904686.

2. F. B. **Mismar**, B. L. Evans, and A. Alkhateeb, "Deep Reinforcement Learning for 5G Networks: Joint Beamforming, Power Control, and Interference Coordination," *IEEE Transactions on Communications*, 2019, available through IEEE Early Access.

3. F. B. **Mismar**, A. AlAmmouri, A. Alkhateeb, J. G. Andrews, and B. L. Evans, "Deep Learning Predictive Band Switching in Wireless Networks," *IEEE Transactions on Wireless Communications*, submitted Oct. 2, 2019. [Online]. Available: https://arxiv.org/abs/1910.05305

Other publications

1. F. B. **Mismar**, J. Choi, and B. L. Evans, "A Framework for Automated Cellular Network Tuning with Reinforcement Learning," *IEEE Transactions on Communications*, vol. 67, no. 10, pp. 7152-7167, Oct. 2019. doi: 10.1109/TCOMM.2019.2926715.

2. F. B. **Mismar** and B. L. Evans, "Partially Blind Handovers for mmWave New Radio Aided by Sub-6 GHz LTE Signaling," *Proceedings of IEEE International Conference on Communications Workshops*, May 2018.

3. F. B. **Mismar** and B. L. Evans, "Deep Q-Learning for Self-Organizing Networks Fault Management and Radio Performance Improvement," *Proceedings of the 52nd Annual Asilomar Conference on Signals, Systems, and Computers*, Oct. 2018.

4. F. B. **Mismar** and B. L. Evans, "Q-Learning Algorithm for VoLTE Closed-Loop Power Control in Indoor Small Cells," *Proceedings of the 52nd Annual Asilomar Conference on Signals, Systems, and Computers*, Oct. 2018.

5. F. B. **Mismar** and B. L. Evans, "Machine Learning in Downlink Coordinated Multipoint in Heterogeneous Networks," Technical Report, Feb. 2019. [Online]. Available: https://arxiv.org/abs/1608.08306

Bibliography

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature International Journal of Science*, vol. 521, pp. 436–444, May 2015.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. Cambridge, MA, USA: The MIT Press, 2016.

[3] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*. New York: Springer, 2013.

[4] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 2nd ed. Cambridge, MA, USA: MIT Press, 2017.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, 1st ed. Springer, 2007.

[6] V. Mnih, "Deep Reinforcement Learning Part 1," Max Planck Institute for Intelligent Systems, Jun. 2017. [Online]. Available: https://youtu.be/WeT_a2zVPcI?t=24

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari With Deep Reinforcement Learning," in *NIPS Deep Learning Workshop*, Dec. 2013.

[8] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What Will 5G Be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[9] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 1617–1655, Feb. 2016.

[10] 3GPP, "NR; NR and NG-RAN Overall Description," 3rd Generation Partnership Project, TS 38.300, Jun. 2018.

[11] 3GPP, "NR; Physical channels and modulation," 3rd Generation Partnership Project, TS 38.211, Jun. 2018.

[12] Qualcomm, "Making 5G NR a Reality," Apr. 2017, accessed on January 10, 2019. [Online]. Available: https://www.qualcomm.com/system/files/document/files/powerpoint_presentation_-_making_5g_nr_a_reality_september_2018_web.pdf

[13] Intel, "5G NR – Driving Wireless Evolution into New Vertical Domains," Aug. 2018, accessed on October 4, 2019. [Online]. Available: https://www.intel.com/content/dam/www/public/us/en/documents/guides/5g-nr-technology-guide.pdf

[14] Huawei, "Huawei Launches the Evolution Strategy for 5G-oriented Wireless Target Network," Nov. 2018, accessed on November 28, 2018.

[Online]. Available: http://www.globenewswire.com/NewsRoom/AttachmentNg/735a89a1-62e7-4ece-9196-3941a4b37cdf/en

[15] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-Free Massive MIMO Versus Small Cells," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1834–1850, Mar. 2017.

[16] ITU-R, Recommendation M.2083-0, "IMT Vision – Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond," Sep. 2005.

[17] 3GPP, "About 3GPP," 2019, accessed on August 20, 2019. [Online]. Available: https://www.3gpp.org/about-3gpp

[18] 3GPP, "Technical Specification Group Services and System Aspects; Study of enablers for Network Automation for 5G," 3rd Generation Partnership Project, TR 23.791, Jun. 2019. [Online]. Available: http://www.3gpp.org/dynareport/23791.htm

[19] "Overview and Guide to the IEEE 802 LMSC," Oct. 2016. [Online]. Available: http://www.ieee802.org/IEEE-802-LMSC-OverviewGuide-06-Oct-2016-v2.pdf

[20] I. Selinis, K. Katsaros, M. Allayioti, S. Vahid, and R. Tafazolli, "The Race to 5G Era; LTE and Wi-Fi," *IEEE Access*, vol. 6, pp. 56 598–56 636, Oct. 2018.

[21] IEEE, "802.11ay IEEE Draft Standard for Information Technology," Institute of Electrical and Electronics Engineers, Tech. Rep., 2016. [Online]. Available: https://standards.ieee.org/project/802_11ay.html

[22] O-RAN Alliance. O-RAN: Towards an Open and Smart RAN. 2018. [Online]. Available: https://www.o-ran.org/resources

[23] P. C. Weeraddana, M. Codreanu, M. Latva-aho, A. Ephremides, and C. Fischione, *Weighted Sum-Rate Maximization in Wireless Networks: A Review*. Now Publisher, 2011.

[24] 3GPP, "NR; User Equipment (UE) conformance specification; Radio Resource Management (RRM)," 3rd Generation Partnership Project, TS 38.533, Jun. 2019. [Online]. Available: http://www.3gpp.org/dynareport/38533.htm

[25] 3GPP, "Technical Specification Group Radio Access Network; Radio Resource Control; Protocol Specification," 3rd Generation Partnership Project, TS 25.331, Dec. 2011. [Online]. Available: http://www.3gpp.org/dynareport/25331.htm

[26] 3GPP, "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification; Protocol specification; Release 15," 3rd Generation Partnership Project, TS 36.331, Dec. 2018. [Online]. Available: http://www.3gpp.org/dynareport/36331.htm

[27] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Overall description," 3rd Generation Partnership Project, TS 36.300, Jan. 2019. [Online]. Available: http://www.3gpp.org/dynareport/36300.htm

[28] 3GPP, "Technical Specification Group Services and System Aspects; Telecommunication management; Study on enhancements of OAM aspects of distributed Self-Organizing Networks (SON) functions (Release 14)," 3rd Generation Partnership Project, TR 32.860, Jun. 2016.

[29] R. Zadeh, "The hard thing about Deep Learning," KD Nuggets, Dec. 2016. [Online]. Available: https://www.kdnuggets.com/2016/12/hard-thing-about-deep-learning.html

[30] C. Saha and H. S. Dhillon, "Machine Learning Meets Stochastic Geometry: Determinantal Subset Selection for Wireless Networks," May 2019. [Online]. Available: https://arxiv.org/abs/1905.00504

[31] S. Song, D. J. Kim, M. Hwang, J. Kim, D. Jeong, S. Lee, H. Jung, and W. Sung, "Prescriptive Analytics System for Improving Research Power," in *Proceedings of IEEE International Conference on Computational Science and Engineering*, Dec. 2013.

[32] S. Yun and C. Caramanis, "Reinforcement Learning for Link Adaptation in MIMO-OFDM Wireless Systems," in *Proceedings of IEEE Global Telecommunications Conference*, Dec. 2010.

[33] F. B. Mismar and B. L. Evans, "Q-Learning Algorithm for VoLTE Closed Loop Power Control in Indoor Small Cells," in *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, Oct. 2018.

[34] M. Bennis and D. Niyato, "A Q-learning Based Approach to Interference Avoidance in Self-Organized Femtocell Networks," in *Proceedings of IEEE Globecom Workshops*, Dec. 2010.

[35] J. Choi, "Massive MIMO with Joint Power Control," *IEEE Wireless Communications Letters*, vol. 3, no. 4, pp. 329–332, Aug. 2014.

[36] L. Zhu, J. Zhang, Z. Xiao, X. Cao, D. O. Wu, and X. Xia, "Joint Power Control and Beamforming for Uplink Non-Orthogonal Multiple Access in 5G Millimeter-Wave Communications," *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 6177–6189, Sep. 2018.

[37] C. Luo, J. Ji, Q. Wang, L. Yu, and P. Li, "Online Power Control for 5G Wireless Communications: A Deep Q-Network Approach," in *Proceedings of IEEE International Conference on Communications*, May 2018.

[38] F. Rashid-Farrokhi, L. Tassiulas, and K. J. R. Liu, "Joint Optimal Power Control and Beamforming in Wireless Networks Using Antenna Arrays," *IEEE Transactions on Communications*, vol. 46, no. 10, pp. 1313–1324, Oct. 1998.

[39] R. Kim, Y. Kim, N. Y. Yu, S. Kim, and H. Lim, "Online Learning-based Downlink Transmission Coordination in Ultra-Dense Millimeter Wave

Heterogeneous Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2200–2214, Mar. 2019.

[40] 3GPP, "Evolved Universal Terrestrial Radio Access; Physical layer procedures," 3rd Generation Partnership Project, TS 36.213, Dec. 2008.

[41] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, "Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 2, pp. 257–265, Jun. 2018.

[42] Y. Wang, M. Liu, J. Yang, and G. Gui, "Data-Driven Deep Learning for Automatic Modulation Recognition in Cognitive Radios," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 4074–4077, Apr. 2019.

[43] H. S. Jang, H. Lee, and T. Q. S. Quek, "Deep Learning-Based Power Control for Non-Orthogonal Random Access," *IEEE Communications Letters*, 2019, available through IEEE Early Access.

[44] M. K. Sharma, A. Zappone, M. Debbah, and M. Assaad, "Deep Learning Based Online Power Control for Large Energy Harvesting Networks," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2019.

[45] W. Lee, M. Kim, and D. Cho, "Deep Power Control: Transmit Power

Control Scheme Based on Convolutional Neural Network," *IEEE Communications Letters*, vol. 22, no. 6, pp. 1276–1279, Jun. 2018.

[46] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep Learning Coordinated Beamforming for Highly-Mobile Millimeter Wave Systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, Jun. 2018.

[47] M. Alrabeiah and A. Alkhateeb, "Deep Learning for TDD and FDD Massive MIMO: Mapping Channels in Space and Frequency," May 2019. [Online]. Available: https://arxiv.org/abs/1905.03761

[48] T. Maksymyuk, J. Gazda, O. Yaremko, and D. Nevinskiy, "Deep Learning Based Massive MIMO Beamforming for 5G Mobile Network," in *Proceedings of IEEE International Symposium on Wireless Systems*, Sep. 2018.

[49] F. B. Mismar, J. Choi, and B. L. Evans, "A Framework for Automated Cellular Network Tuning with Reinforcement Learning," *IEEE Transactions on Communications*, vol. 67, no. 10, pp. 7152–7167, Oct. 2019.

[50] P. Zhou, X. Fang, X. Wang, Y. Long, R. He, and X. Han, "Deep Learning-Based Beam Management and Interference Coordination in Dense mmWave Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 592–603, Jan. 2019.

[51] W. Xia, G. Zheng, Y. Zhu, J. Zhang, J. Wang, and A. P. Petropulu, "A Deep Learning Framework for Optimization of MISO Downlink Beam-

forming," *IEEE Transactions on Communications*, Dec. 2019, available through IEEE Early Access.

[52] F. B. Mismar and B. L. Evans, "Deep Learning in Downlink Coordinated Multipoint in New Radio Heterogeneous Networks," *IEEE Wireless Communications Letters*, vol. 8, no. 4, pp. 1040–1043, Aug. 2019.

[53] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath Jr., "Channel Estimation and Hybrid Precoding for Millimeter Wave Cellular Systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[54] R. W. Heath Jr., N. González-Prelcic, S. Rangan, W. Roh, and A. Sayeed, "An Overview of Signal Processing Techniques for Millimeter Wave MIMO Systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 436–453, Apr. 2016.

[55] P. Schniter and A. Sayeed, "Channel Estimation and Precoder Design for Millimeter Wave Communications: The Sparse Way," in *Proceedings of Asilomar Conference on Signals, Systems and Computers*, Nov. 2014.

[56] T. Rappaport, F. Gutierrez, E. Ben-Dor, J. Murdock, Y. Qiao, and J. Tamir, "Broadband Millimeter-Wave Propagation Measurements and Models Using Adaptive-Beam Antennas for Outdoor Urban Cellular Communications," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 4, pp. 1850–1859, Apr. 2013.

[57] T. S. Rappaport, R. W. Heath Jr., R. C. Daniels, and J. N. Murdock, *Millimeter Wave Wireless Communications*. Westford, MA, USA: Pearson Education, 2014.

[58] G.-B. Huang, "Learning Capability and Storage Capacity of Two-hidden-layer Feedforward Networks," *IEEE Transactions on Neural Networks*, vol. 14, no. 2, pp. 274–281, Mar. 2003.

[59] L.-J. Lin, "Reinforcement Learning for Robots Using Neural Networks," Ph.D. dissertation, School of Computer Science, Carnegie-Mellon University, Pittsburg, PA, 1993.

[60] M. Simsek, A. Czylwik, A. Galindo-Serrano, and L. Giupponi, "Improved Decentralized Q-learning Algorithm for Interference Reduction in LTE-femtocells," in *Proceedings of IEEE Wireless Advanced Conference*, Jun. 2011.

[61] D. Silver, "Advanced Topics – Reinforcement Learning," 2015. [Online]. Available: http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html

[62] F. B. Mismar and B. L. Evans, "Partially Blind Handovers for mmWave New Radio Aided by Sub-6 GHz LTE Signaling," in *Proceedings of IEEE International Conference on Communications Workshops*, May 2018.

[63] A. I. Sulyman, A. Alwarafy, G. R. MacCartney, T. S. Rappaport, and A. Alsanie, "Directional Radio Propagation Path Loss Models for Millimeter-Wave Wireless Networks in the 28-, 60-, and 73-GHz Bands," *IEEE*

Transactions on Wireless Communications, vol. 15, no. 10, pp. 6939–6947, Oct. 2016.

[64] T. Bai and R. W. Heath Jr., "Coverage and Rate Analysis for Millimeter-Wave Cellular Networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 1100–1114, Feb. 2015.

[65] F. B. Mismar, "Source code," 2019. [Online]. Available: https://github.com/farismismar/Deep-Reinforcement-Learning-for-5G-Networks

[66] 3GPP, "5G; NG-RAN; Architecture description," 3rd Generation Partnership Project, TS 38.804, Jul. 2018.

[67] 3GPP, "Coordinated Multi-Point Operation for LTE," 3rd Generation Partnership Project, TR 36.819, Sep. 2013.

[68] S. Y. Kim and C. H. Cho, "Call Blocking Probability and Effective Throughput for Call Admission Control of CoMP Joint Transmission," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 622–634, Jan. 2017.

[69] G. Nigam, P. Minero, and M. Haenggi, "Coordinated Multipoint Joint Transmission in Heterogeneous Networks," *IEEE Transactions on Communications*, vol. 62, pp. 4134–4146, Oct. 2014.

[70] P. Xia, C. H. Liu, and J. G. Andrews, "Downlink Coordinated Multi-Point with Overhead Modeling in Heterogeneous Cellular Networks,"

IEEE Transactions on Wireless Communications, vol. 12, no. 8, pp. 4025–4037, Aug. 2013.

[71] H. Zhang, C. Jiang, J. Cheng, and V. C. M. Leung, "Cooperative Interference Mitigation and Handover Management for Heterogeneous Cloud Small Cell Networks," *IEEE Wireless Communications*, vol. 22, no. 3, pp. 92–99, Jun. 2015.

[72] S. Wesemann and G. Fettweis, "Decentralized Formation of Uplink CoMP Clusters Based on Affinity Propagation," in *Proceedings of International Symposium on Wireless Communication Systems*, Oct. 2012.

[73] F. B. Mismar and B. L. Evans, "Machine Learning in Downlink Coordinated Multipoint in Heterogeneous Networks," Technical Report, Feb. 2019. [Online]. Available: https://arxiv.org/abs/1608.08306

[74] H. Ye, G. Y. Li, and B. Juang, "Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems," *IEEE Wireless Communication Letters*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[75] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep Learning for Super-Resolution Channel Estimation and DOA Estimation Based Massive MIMO System," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8549–8560, Sep. 2018.

[76] K. Veropoulos, C. Campbell *et al.*, "Controlling the Sensitivity of Support Vector Machines," in *Proceedings of International Joint Conference*

on Artificial Intelligence, 1999.

[77] 3GPP, "NR; Physical layer measurements," 3rd Generation Partnership Project, TS 38.215, Sep. 2018.

[78] A. Paulraj, A. Rohit, R. Nabar, and D. Gore, *Introduction to Space-Time Wireless Communications*. New York, USA: Cambridge University Press, 2003.

[79] I. Ahmad, Z. Kaleem, and K. Chang, "Block Error Rate and UE Throughput Performance Evaluation using LLS and SLS in 3GPP LTE Downlink," Oct. 2018. [Online]. Available: http://arxiv.org/abs/1810.01162

[80] A. Karatzoglou, D. Meyer, and K. Hornik, "Support Vector Machines in R," *Journal of Statistical Software, Articles*, vol. 15, no. 9, pp. 1–28, Apr. 2006.

[81] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[82] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[83] M. Rupp, S. Schwarz, and M. Taranetz, *The Vienna LTE-Advanced Simulators: Up and Downlink, Link and System Level Simulation*. Singapore: Springer, 2016.

[84] F. B. Mismar, "Source Code," 2019. [Online]. Available: https://github.com/farismismar/DL-CoMP-Machine-Learning

[85] 3GPP, "Evolved Universal Terrestrial Radio Access; Radio Resource Control; Protocol specification; Release 8," 3rd Generation Partnership Project, TS 36.311, Nov. 2007.

[86] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Aug. 2016.

[87] N. Gonzalez-Prelcic, A. Ali, V. Va, and R. W. Heath Jr., "Millimeter-Wave Communication with Out-of-Band Information," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 140–146, Dec. 2017.

[88] K. Hugl, K. Kalliola, and J. Laurila, "Spatial Reciprocity of Uplink and Downlink Radio Channels in FDD Systems," in *Proceedings of European Cooperation in the Field of Scientific and Technical Research*, May 2002.

[89] T. Nitsche, A. B. Flores, E. W. Knightly, and J. Widmer, "Steering with Eyes Closed: mm-Wave Beam Steering without In-Band Measurement," in *Proceedings of IEEE International Conference on Computer Communications*, Apr. 2015.

[90] A. Ali, N. González-Prelcic, and R. W. Heath Jr., "Millimeter Wave Beam-Selection Using Out-of-Band Spatial Information," *IEEE Trans-*

actions on Wireless Communications, vol. 17, no. 2, pp. 1038–1052, Feb. 2018.

[91] Y. Xiu, W. Wang, and Z. Zhang, "A Message Passing Approach to Acquire mm-Wave Channel State Information Based on Out-of-Band Data," *IEEE Access*, vol. 6, pp. 45 665–45 680, Jul. 2018.

[92] Y. Xiu, J. Wu, C. Xiu, and Z. Zhang, "Millimeter Wave Cell Discovery Based on Out-of-Band Information and Design of Beamforming," *IEEE Access*, vol. 7, pp. 23 076–23 088, Feb. 2019.

[93] A. Ali, N. González-Prelcic, and R. W. Heath Jr., "Spatial Covariance Estimation for Millimeter Wave Hybrid Systems Using Out-of-Band Information," *IEEE Transactions on Wireless Communications*, 2019, available through IEEE Early Access.

[94] A. Alkhateeb, "DeepMIMO: A Generic Deep Learning Dataset for Millimeter Wave and Massive MIMO Applications," in *Proceedings of Information Theory and Applications Workshop*, Feb. 2019.

[95] A. Alkhateeb. DeepMIMO. [Online]. Available: https://github.com/DeepMIMO/DeepMIMO-codes

[96] O. Semiari, W. Saad, and M. Bennis, "Joint Millimeter Wave and Microwave Resources Allocation in Cellular Networks With Dual-Mode Base Stations," *IEEE Transactions on Wireless Communications*, vol. 16, no. 7, pp. 4802–4816, Jul. 2017.

[97] O. Semiari, W. Saad, M. Bennis, and B. Maham, "Caching Meets Millimeter Wave Communications for Enhanced Mobility Management in 5G Networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 2, pp. 779–793, Feb. 2018.

[98] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, "Improved Handover Through Dual Connectivity in 5G mmWave Mobile Networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2069–2084, Sep. 2017.

[99] J. Mo, B. L. Ng, P. Huang, M. Kulkarni, A. AlAmmouri, J. C. Zhang, S. Chang, J. Lee, and W.-J. Choi, "Beam Codebook Design for 5G mmWave Terminals," *IEEE Access*, vol. 7, pp. 98 387–98 404, Jul. 2019.

[100] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez Jr., "Millimeter Wave Mobile Communications for 5G Cellular: It Will Work!" *IEEE Access*, vol. 1, no. 1, pp. 335–349, May 2013.

[101] A. AlAmmouri, J. Mo, B. L. Ng, J. C. Zhang, and J. G. Andrews, "Hand Grip Impact on 5G mmWave Mobile Devices," *IEEE Access*, vol. 7, pp. 60 532–60 544, May 2019.

[102] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, Jun. 2006.

[103] A. Ghosh, J. Zhang, J. G. Andrews, and R. Muhamed, *Fundamentals of LTE*. Boston, MA, USA: Prentice-Hall, 2010.

[104] V. Va, J. Choi, and R. W. Heath Jr., "The Impact of Beamwidth on Temporal Channel Variation in Vehicular Channels and Its Implications," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 5014–5029, Jun. 2017.

[105] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proceedings of International Conference on Learning Representations*, May 2014.

[106] F. B. Mismar, "Source Code," 2019. [Online]. Available: https://github.com/farismismar/Bandswitch-DeepMIMO

[107] F. B. Mismar, B. L. Evans, and A. Alkhateeb, "Deep Reinforcement Learning for 5G Networks: Joint Beamforming, Power Control, and Interference Coordination," *IEEE Transactions on Communications*, 2019, available through IEEE Early Access.

[108] F. B. Mismar, A. AlAmmouri, A. Alkhateeb, J. G. Andrews, and B. L. Evans, "Deep Learning Predictive Band Switching in Wireless Networks," *IEEE Transactions on Wireless Communications*, 2019, submitted. [Online]. Available: https://arxiv.org/abs/1910.05305

[109] T. Cover and J. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ, USA: Wiley-Interscience, 2006.

[110] 3GPP, "NR; User Equipment (UE) radio transmission and reception," 3rd Generation Partnership Project, TS 38.101, Oct. 2018.

[111] F. B. Mismar, "Source Code," 2018. [Online]. Available: https://github.com/farismismar/Deep-Q-Learning-SON-Perf-Improvement

[112] D. Tse and P. Vishwanath, *Fundamentals of Wireless Communication*. New York, USA: Cambridge University Press, 2005.

[113] Vienna Simulators LTE-A Downlink System Simulator. [Online]. Available: https://www.nt.tuwien.ac.at/wp-content/uploads/2015/11/LTEsystemDoc_v1_9Q2_2016.pdf

[114] F. B. Mismar and B. L. Evans, "Deep Q-Learning for Self-Organizing Networks Fault Management and Radio Performance Improvement," in *Proceedings of Asilomar Conference on Signals, Systems, and Computers*, Oct. 2018.

[115] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Frequency (RF) system scenarios," 3rd Generation Partnership Project, TR 36.942, Mar. 2017. [Online]. Available: http://www.3gpp.org/dynareport/36942.htm

[116] A. Ali, N. González-Prelcic, and R. W. Heath Jr., "Spatial Covariance Estimation for Millimeter Wave Hybrid Systems using Out-of-Band Information," Apr. 2018. [Online]. Available: https://arxiv.org/abs/1804.11204

[117] A. Alkhateeb, I. Beltagy, and S. Alex, "Machine Learning for Reliable mmWave Systems: Blockage Prediction and Proactive Handoff," in *Proceedings of IEEE Global Conference on Signal and Information Processing*, Nov. 2018.

[118] Y. Wang, A. Klautau, M. Ribero, A. C. K. Soong, and R. W. Heath Jr., "MmWave Vehicular Beam Selection With Situational Awareness Using Machine Learning," *IEEE Access*, vol. 7, pp. 87 479–87 493, Jun. 2019.

[119] 3GPP, "Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specifications," 3rd Generation Partnership Project, TS 36.321, Dec. 2015. [Online]. Available: http://www.3gpp.org/dynareport/36321.htm

[120] T. Jaakkola, M. I. Jordan, and S. P. Singh, "Convergence of Stochastic Iterative Dynamic Programming Algorithms," in *Proceedings of the International Conference on Neural Information Processing Systems*, Dec. 1994.

[121] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep Learning Coordinated Beamforming for Highly-Mobile Millimeter Wave Systems," *IEEE Access*, vol. 6, pp. 37 328–37 348, Jun. 2018.

[122] T. Erpek, Y. E. Sagduyu, and Y. Shi, "Deep Learning for Launching and Mitigating Wireless Jamming Attacks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 1, pp. 2–14, Mar. 2019.

[123] 3GPP, "5G; NR; Radio Resource Control (RRC); Protocol specification," 3rd Generation Partnership Project, TS 38.331, Oct. 2018. [Online]. Available: http://www.3gpp.org/dynareport/38331.htm

Vita

Faris B. Mismar received his BS degree in electrical and computer engineering from The University of Jordan in 2004, the MS degree in electrical engineering from The University of Texas at Dallas in 2011, and the MBA degree from The University of Texas at Dallas in 2014. He has been working in the wireless communication industry since 2004. He has consulted with many leading wireless operators across the globe and has held various senior positions at Motorola, Ericsson, and Samsung. He was a recipient of the Marcus Wallenberg Scholarship for Scientific Research and Education in 2016 and 2017. His research interests include machine learning, artificial intelligence, and wireless communications. He is currently working towards the Ph.D. degree in electrical and computer engineering at The University of Texas at Austin and is a member of the Embedded Signal Processing Laboratory (ESPL) directed by Prof. Brian L. Evans. ESPL is affiliated with the Wireless Networking and Communications Group and the Center for Perceptual Systems.

Permanent address: `faris.mismar@gmail.com`

This dissertation was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.