A MODEL-BASED DEEP LEARNING APPROACH TO SPECTRUM MANAGEMENT IN DISTRIBUTED COGNITIVE RADIO NETWORKS

by

Mduduzi Comfort Hlophe

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

in the

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

UNIVERSITY OF PRETORIA

July 2020

SUMMARY

A MODEL-BASED DEEP LEARNING APPROACH TO SPECTRUM MANAGEMENT IN DISTRIBUTED COGNITIVE RADIO NETWORKS

by

Mduduzi Comfort Hlophe

Promoter(s):	Prof. B. T. Maharaj
Department:	Electrical, Electronic and Computer Engineering
University:	University of Pretoria
Degree:	Philosophiae Doctor (Electronic Engineering)
Keywords:	Cognitive radio, Cognitive radio networks, Deep learning, Deep neural networks, Deep reinforcement learning, Docitive networks, Dynamic spectrum access, Energy efficiency, Internet of things, Long short-term memory, Mean slowdown, Opportunistic spectrum access, Primary user, Quality of experience, Quality of service, Resource allocation, Secondary user, Spectrum management, Spectrum occupancy reconstruction, Spectrum sensing, Spectrum shortage, Spectrum under-utilization, Stacked auto-encoder, Stochastic gradient descent, Wireless Big Data.

The acceleration towards the fifth generation (5G) and beyond will see the internet of things (IoT) being the primary strategy of deployment, and wireless networks will become more distributed and autonomous. Furthermore, network users will demand delivery of multimedia content to various network devices in dissimilar contexts. Thus, the cognitive radio (CR) paradigm requires some improvements for it to rigorously resolve quality of service (QoS) and quality of experience (QoE) in an energy-efficient manner before the 5G network is commissioned. Therefore, solving the distributed RA problem through thorough and in-depth investigations into the essentials and intricacies of energy-efficient RA by integrating artificial intelligence (AI) and signal processing (SP) strategies is a requisite. Having identified this knowledge gap and several limiting factors, this thesis focuses on two

fronts to maximize the distributed opportunistic usage of the wireless spectrum with enhanced energy efficiency.

The first contribution of this study provides a solution for missing spectrum sensing information to improve spectrum occupancy measurements in distributed CRNs. This is a problem commonly encountered in distributed cooperative spectrum sensing scenarios, where secondary users (SUs) are faced with the missing spectrum sensing data (SSD) problem owing to several impairments such as (i) the use of specific collaborative spectrum sensing schemes and (ii) imperfect reporting channel conditions. This results in the SSD contributed by SUs having gaps of missing entries. This degrades the performance of spectrum sensing algorithms, especially when the amount of missing SSD is quite large. Therefore, spectrum occupancy reconstruction is proposed as a solution to deal with missing values through missing value imputation. This is a deep learning (DL)-based strategy that uses deep belief networks (DBNs) composed of restricted Boltzmann machines (RBMs) to capture the feature of the input space of the spectrum occupancy data from a Markov random field (MRF). Link energy functions from the Ising models and the Metropolis-Hastings algorithm are used to pre-train the RBM to obtain a spectrum occupancy data matrix. The size of training samples and learning rates are decided using Gibbs sampling during the training process and missing spectrum values are learned using a scaled stochastic gradient descent (SGD). The simulation results obtained indicate that spectrum occupancy reconstruction problems can be solved better using the SGD algorithm because it takes advantage of correlations in multiple dimensions better than singular value decomposition (SVD) in matrix factorization.

The second contribution provides a solution for energy saving and QoS provisioning for SUs with heterogeneous traffic, which is a problem exacerbated by the increased demand for multimedia services. This necessitates for the establishment of newer power control strategies for multimedia sources, where energy saving and QoS provisioning are viewed from the job arrival rate instead of the packet arrival rate perspective. Here, the model dynamics are formulated as a continuous-time non-linear input affine system which combines opportunistic transmission and opportunistic computing to obtain resource consumption efficiency. By treating the base station (BS) as a hybrid switching system, a weighted cost function is obtained and solved using model-based reinforcement learning (RL), which initiates a single look-ahead for optimum operating states. Then, using the resource consumption efficiency, a DL-based predictive control scheme was realized with control actions that drive a stacked auto-encoder (SAE) that plays dynamic games on queues and performs effective trade-offs between QoS provisioning and energy saving. The simulation results obtained indicate that the processor sharing (PS) scheduling scheme achieves better energy saving than first-come-first-served (FCFS) at higher job arrival rates.

The last contribution deals with the problem of distributed RA in energy-constrained CRN environments, with the objective of ensuring user satisfaction in terms of QoE and QoS in an energy-efficient manner. QoE evaluation is performed using docitive techniques and the results obtained indicate that transfer-learning through docitive approaches achieves better convergence rates and superior spectral efficiency compared to the traditional cognitive approaches. Then, a computationally efficient optimization technique that handles the energy efficiency learning model is achieved using factored Markov decision processes (FMDPs), which provides a solvable framework for energy minimization. This completes the hierarchical deep RL (DRL) with a deep Q-network (DQN) formulation that learns energy consumption subject to latency constraints. The results obtained show that the DQN approach with experience replay achieves better QoS performance compared to the traditional RL in terms of minimizing buffer delays and power consumption.

LIST OF ABBREVIATIONS

5G	Fifth-Generation Mobile Telecommunications Technology
AI	Artificial Intelligence
ANN	Artificial Neural Network
ARIMA	Auto-regressive Integrated Moving Average
BS	Base Station
CSI	Channel State Information
CR	Cognitive Radio
CRN	Cognitive Radio Network
DBN	Deep Belief Network
DL	Deep Learning
DNN	Deep Neural Network
DQN	Deep Q-network
DRL	Deep Reinforcement Learning
DDQN	Double Deep Q-network
DTMC	Discrete Time Markov Chain
DTMDP	Discrete Time Markov Decision Process
DSA	Dynamic Spectrum Access
ENAAM	ENergy-Aware and Adaptive Management
FMDP	Factored Markov Decision Process
FCFS	First Come First Served
GAN	Generative Adversarial Network
IMT	International Mobile Telecommunications
ITU-T	International Telecommunications Union-Technical
IoT	Internet of Things
IP	Internet Protocol
LSTM	Long Short-term Memory
ML	Machine Learning
MRF	Markov Random Field
MOS	Mean Opinion Score
MSE	Mean Square Error

MMSE	Minimum Mean Square Error
MEC	Mobile Edge Controller
NFV	Network Function Virtualization
OSA	Opportunistic Spectrum Access
POMDP	Partially Observable Markov Decision Process
PHY	Physical Layer
PRB	Physical Resource Block
PU	Primary User
PCA	Principal Component Analysis
PS	Processor Sharing
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RNN	Recurrent Neural Network
RL	Reinforcement Learning
RA	Resource Allocation
RBM	Restricted Boltzmann Machine
SU	Secondary User
SP	Signal Processing
SINR	Signal-to-Interference-plus-Noise Ratio
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
SDN	Software Defined Networking
SSD	Spectrum Sensing Data
SAE	Stacked Auto-Encoder
SGD	Stochastic Gradient Descent
VM	Virtual Machine
VNF	Virtualized Network Function

LIST OF SYMBOL NOTATIONS

	Set Notation
$\mathbb{R}$	Denotes a set of real numbers
$\|X\|$	Denotes the cardinality of set X
$x \in X$	Denotes that x is an element of X
$\emptyset$	Denotes an empty set
$\mathcal{G}$	Denotes a graph
$\mathcal{V}$	Denotes the set of vertices $v \in \mathcal{V}$ of graph, $\mathcal{G}$
$\mathcal{E}$	Denotes the set of edges $e \in \mathcal{E}$ connecting vertices of graph, $\mathcal{G}$
	Vector and Matrix Notation
$\mathbf{X}$	Denotes a matrix
X_{ij}	Denotes the $(ij)^{th}$ element of matrix $\mathbf{X}$
$\mathbf{x}$	Denotes a vector
$\mathbf{x}^T$	Denotes the vector transpose, with T being the transpose operator
x_i	Denotes that x is the i^{th} element of vector $\mathbf{x}$
$\|\|\mathbf{x}\|\|$	Denotes the Euclidean norm of vector $\mathbf{x}$
$\mathbf{x} \cdot \mathbf{y}$	Denotes the Euclidean inner product of vectors $\mathbf{x}$ and $\mathbf{y}$
$\mathbb{E}(\cdot)$	Denotes a mathematical expectation operator
$\mathbb{E}_{x \sim p(x)}[f(x)]$	Denotes the expected value of function $f(x)$ where the random variable x is drawn from the distribution $p(x)$
	Signal Processing Notation
α	Denotes the path-loss exponent
K; k	Total number of SUs; where k identifies a particular SU
J; j	Total number of physical resource blocks (PRBs), j refers to a specific PRB
$\gamma_{j,k}$	Denotes the signal-to-noise-plus-interference ratio (SINR)
$g_{j,k}$	Denotes the channel gain between SU and BS,
$\mathcal{I}_k$; $\mathcal{I}^{th}$	Denotes the aggregate interference; interference threshold
$\mathcal{J}$	Denotes the set of available PRBs
$P_{j,k}$; $R_{j,k}$	Denotes the optimal transmission power; and spectral efficiency, respectively
P_{max}	Denotes the maximum allowed transmission power at secondary BS
ξ	Denotes the system bandwidth

ϕ	Denotes the SU admission control probability
ς^{th}	Denotes the resource percentage threshold (RPT)
δ^{th}	Denotes the threshold PU collision probability
$q(t)$	Denotes the queue length
λ_t; μ_t	Denotes the packet arrival rate; packet processing rate
$v(t)$	Denotes the base station switching mode
$\rho(t)$	Denotes the system utilization/resource consumption efficiency
$P(v,\rho,t)$	Denotes the load-dependent power consumption
$J(v,\rho,t)$	Denotes a weighted cost function
$x(t)$; $u(t)$	Denotes the system state; system control input
	Machine Learning Notation
$p_{data}(x)$	Denotes a data generating distribution
$(x,y) \in D$	Denotes training examples existing in data set D
$p(y\|x)$	Denotes the conditional distribution of a random variable y conditioned on the random variable x
θ	Denotes the distribution parameterization parameter of a conditional distribution
$p_{model}(D,\theta)$	Denotes the function space of probability distributions over the parameters θ
$p(x;\theta)$	Denotes the probability distribution of a random variable x is conditioned on a set of parameters, θ
$E(\cdot)$	Denotes an energy function in restricted Boltzmann machines
Ω	Denotes a sample space, i.e., a collection of all possible realizations of random variables
Φ	Denotes an activation function
$\alpha(\cdot,\cdot)$	Denotes the likelihood ratio between consecutive samples
α_t; γ^t	Denotes the learning rate; denotes the discount factor
δ	Denotes the delay constraint
ς^*	Denotes the packet dropping penalty function
$\mathcal{S}$; $\mathcal{A}$	Denotes the set of all states, s; set of all actions, a
$Q(s,a)$	Denotes the state-value function approximator
$\pi(s,a)$	Denotes a stationary policy mapping states to actions
Ψ; Λ	Denotes the set of all stationary policies; Lagrange multiplier
θ^-, θ	Denotes target network parameters (i.e., in DQNs)
D_i	Denotes the replay memory buffer in DQNs

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER 1 INTRODUCTION

Mobile and wireless networking technologies have advanced profoundly over the past decade towards achieving ubiquitous communication [1]. Tremendous advancements have also been made in the affordability and portability of mobile and wireless communication devices, which in conjunction with improved wireless coverage have resulted in worldwide acceptance of wireless communications [2]. As a consequence, the mobile and wireless network penetration has increased and the demand for spectrum resources to accommodate this positive explosion of wireless communication has also increased. Because of the advanced capabilities of mobile devices, network users have made extensive use of multimedia services, consequently requiring application content delivery at greater speeds. However, because of the existing absolute spectrum allocation scheme, the radio spectrum required to accommodate and support such demands is not available. On one hand, there are no new spectrum bands to be further allocated to support new technological advancements and their requirements [3]. On the other hand, the radio spectrum currently allocated using the absolute allocation scheme is under-utilized by the incumbent users [4], [5].

This turns the apparent spectrum scarcity problem into a spectrum shortage and spectrum under-utilization duality. A concise, yet clear explanation of the spectrum shortage and the spectrum under-utilization duality can be given by contrasting the spectrum allocation point of view with the spectrum usage point of view. In terms of spectrum allocation, spectrum bands can be allocated by regulators for a number of years to a single entity without any fine-grained temporal access and, ensuring the maintenance of acceptable interference with wireless systems, guard bands and spatial buffers are kept [6]. Spectrum usage is concerned with the overall utilization of allocated spectrum bands. Because of the wireless spectrum quagmire, measurements of spectrum usage have been conducted in different locations in the world and results have indicated extremely low spectrum utilization levels by the incumbents [7]. As a result, dynamic spectrum sharing using cognitive radio

(CR) technology has been advocated as the de facto solution to bridge the gap between spectrum allocation and utilization.

The CR technology enables spectrum sharing between the primary users (PUs) of the spectrum and secondary users (SUs) who want to use the spectrum opportunistically through dynamic spectrum access (DSA) [8]. A CR is defined as a wireless technology whose radio transceiver has the ability of intelligently sensing and detecting the occupancy and usage of the licensed spectrum [9]. As a result, in the context of CRNs, depending on the knowledge discovery and SU coexistence strategy with the PUs, spectrum sharing approaches were divided into three classes; (i) underlay, (ii) overlay and (iii) interweave [10]. In the underlay approach, SUs are allowed to concurrently utilize the spectrum with PUs as long as their transmissions do not disturb the integrity of the PUs. In the Overlay approach, SUs can opportunistically utilize holes in the spectrum only in the absense of the PU and vacate them immediately they sense the return of the PU, while the interweave approach exploits the advantages of both the underlay and overlay approaches [11]. This means that in the interweave approach SUs can switch between the two modes according to the state of the PUs. Upon detecting an unoccupied spectrum band, it can resume opportunistic transmission as long as the PU remains inactive. However, if no free spectrum band is detected, it continues sensing until it finds one. This intelligent approach to the utilization of the idle radio spectrum improves the efficiency of spectrum usage while simultaneously optimizing harmful interference with PUs and prioritized SUs [12]. Through this technique, opportunistic spectrum access (OSA) is realized, where SUs can reuse the licensed spectrum either temporally or spatially. In the temporal reuse case, SUs can access the spectrum when the PU is currently not using it; while in spatial reuse case, SUs can access the spectrum when they are far enough from the PU to ensure that it will not interfere with the PU's activities [13]. Thus, CR technology is currently the only mechanism that offers a purely technical solution to the apparent spectrum scarcity problem without the need to redesign the existing primary systems.

There is industrious research activity regarding resource allocation (RA) in CR networks (CRNs), using several optimization techniques such as mathematical programming (i.e., game theory, queuing theory), bio-inspired programming (i.e., particle swarm, genetic programming), artificial intelligence (AI)-inspired learning architectures (i.e., supervised learning, unsupervised learning, reinforcement learning (RL), deep reinforcement learning (DRL), and deep learning (DL)) and interesting results have been reported. However, with the acceleration into 5G driven by the internet of things (IoT), where billions of wireless devices will be connected to the internet to create a global network of dimensions never seen before [14], [15] and energy efficiency will remain a pervasive challenge. As IoT continues to steer operations in the wireless world, different spectrum use cases with dissimilar characteristics and dissenting resource demands will take over wireless networks. In all these use cases, energy-efficient communication among different network devices is crucial to accomplish the objective of ubiquity in future mobile and wireless networking. For this objective to be feasible, a high level of coordination among mobile and wireless devices to optimize both spectrum usage and energy efficiency is a requisite. Since wireless devices will have stringent performance requirements such as low latency, wireless networks need to be equipped with intelligence to achieve superior energy efficiency [16].

Energy efficiency has been the most pervasive problem in wireless networks mostly because most mobile and wireless devices are usually battery-powered, thus they are unable to have power provided all the time they are connected to the network. In addition, over the past few years data transmission throughput has increased drastically in response to the increase in multimedia services; the higher transmission throughput has led to more energy consumed. Contrary to traditional wireless communications, energy efficiency in CRNs includes several aspects such as spectrum sensing, transmission and channel switching, and different constraints need to be considered in dealing with energy efficiency problems. As a consequence, energy efficiency is one of the critical problems hindering the deployment of CRNs and this problem has been studied extensively, using power control techniques in the context of voice and data communications. In view of the increasing demand for multimedia services, it is necessary to establish new algorithms for energy efficiency in the context of information sources other than voice and data.

If the energy problem can be solved for multimedia information sources, CR technology will enable fine-grained reuse of scarce spectrum resources. However, because of the growing popularity of multimedia applications, there is a conflict between achieving high data rates and simultaneously reducing the energy consumption costs. This is exacerbated by the vexing problem of spectrum shortage and the fact that wireless networks are deployed in a distributed manner to ensure improved scalability. A better solution to the energy efficiency problem in terms of green networks is also an attractive option to cut down the operational expenditures but also to alleviate the rising concerns about the global environmental consequences due to operating the communication infrastructures using fossil fuel. Thus, it is a critical requirement for future mobile and wireless communication network design to reduce energy consumption while still providing high-speed communication. Therefore, new energy-efficient and energy-saving strategies should focus not only on single objective optimization, but mainly on how better trade-offs can be achieved in multi-objective optimization problems.

1.1 PROBLEM STATEMENT

The energy efficiency problem means that future mobile and wireless networks will not only be faced with the problem of creating sufficient transmission capacity, but also energy-efficient operations intertwined with low latency requirements. A credible solution to these problems will constitute a vital step towards guaranteeing network sustainability and minimizing the cost of network operation and expenditure. The relevance of energy efficiency in CRNs will mostly depend on the complexity of the scenarios in which the CR devices have to operate in, and also on the performance levels they are required to guarantee. Depending on the complexity of the operating scenarios, accurate multi-objective mathematical models might not be currently available. This is because every current model is inherently an approximation such that trade-offs exist between model accuracy and model complexity. Regarding the complexity of the scenario, accurate models have proven to be too complex to handle mathematically, while the simplified models are usually not or may not be accurate enough.

The current state-of-the-art optimization solutions have however shown that these issues can be ignored if the operating scenarios do allow for the derivation of simpler mathematical models that are both accurate and tractable enough and conform to the desired performance requirements. However, because of the forecast heterogeneity of characteristics of future wireless networks - heterogeneity of services, user demands and user technologies, service requirements might no longer form renewal processes. As a result, accurate energy-efficient solutions that are flexible enough to adapt to the heterogeneity of service requirements and the operating environments need to be found.

1.1.1 Context of the Problem

To improve the energy efficiency and transmission throughput of CRNs significantly, closed-form solutions need to consider the energy consumption, throughput and interference levels of PUs and the transmission duration of SUs simultaneously. Maximizing the throughput of SUs by finding the optimal spectrum sensing duration and scheduling a transmission action is subject to the battery capacity constraint. However, most research investigations have focused primarily on throughput in spectrum sensing and on spectrum access and have left the vast picture of the energy uncertainty non-perceptible. For example, the drawbacks associated with SUs employing energy-efficient spectrum sensing have not been attended to. Furthermore, the energy saving based on application/service differentiation oriented energy efficiency, which is supposed to give fine-grained details of network energy consumption have not yet been looked into. Energy efficiency and energy-saving strategies for future mobile and wireless networking should exploit the traffic load variations, the diversity of the QoS and the trade-off between energy consumption and transmission delay in order to support non-perceptible migration from traditional voice and data to real-time communications.

The integration of these functionalities into the same network architecture requires a paradigm shift in the way in which mobile and wireless networks are designed and managed, which will lead to an exceedingly flexible and more adaptive network architecture. However, this move seems to be in sharp contrast to the approach adopted in traditional wireless networks, where the control and data planes are bundled inside the network nodes. In light of this, it is clear that the deployment of more performing technologies may not ensure the required flexibility to accommodate the diverse user classes with extremely heterogeneous service requirements. New architecture and management solutions have already been proposed to tackle this issue; however, most of them leverage host-based and network-based protocols that use traditional signal processing (SP) techniques. As a consequence, the handover delays and packet losses are still too high for real-time services, and a detailed analysis of traditional approaches and proposed machine learning (ML)-based solutions are as follows:

- SP strategies have their solid foundations firmly rooted in statistics and information theory and are often optimal for tractable mathematical models, which are generally linear and stationary and possess Gaussian statistics [17]. In the context of wireless networks, statistics and information theory and the assumptions made about those seem valid for some situations as long as they make the problems mathematically convenient. However, practical wireless communication situations often involve time-varying, correlated and non-Gaussian quantities - with many imperfections and non-linearities [18]. For example, the time-varying nature of a wireless communication channel results in such large differences in the instantaneous received signal strength, that even with the advent of efficient SP techniques, the bottleneck remains the complexity diversity of the wireless environment.

However, ML techniques such as neural networks (NNs) have proven to be universal function approximators in such complex wireless environments. Recent works have shown remarkable algorithmic learning capabilities through the use of recurrent NNs (RNNs) [19], [20] that are known to be Turing complete owing to their highly parallelized computational architectures [21]. There is plethora of evidence in literature that algorithms of this kind could be executed faster and at lower energy cost than the current SP strategies [22]. Therefore, for this reason, since ML-based systems do not require mathematically tractable models and can be optimized for different hardware configurations and channel environments, might be able to be better optimized for such imperfections.

- The basic principle in wireless communication systems design is splitting the SP into a chain of multiple independent blocks, each block optimized to execute a well-defined and isolated function such as source or channel coding, modulation, channel estimation, etc [23]. This principled approach has resulted in the efficient, versatile and controllable wireless communication systems that are currently available. It has also demonstrated the possibility of learning full transmitter and receiver implementations for any given channel model, optimized for any chosen loss function (e.g., minimizing the block error rate). However, considering future generations of mobile and wireless communications, this seems to be a rigid approach. This is because it is not clear whether the individually optimized processing blocks do achieve the best possible end-to-end performance. Some examples include: (i) the separation of source and channel coding for many practical channels and short block lengths [24]; as well as (ii) the separation of coding and modulation [25]. Jointly optimizing these individual components using factor graphs does provide significant performance gains, but leads to unwiedly and computationally complex systems and rigid modular structures [26].

  A fully learned end-to-end communication system is however unlikely to have such a rigid modular structure, as it will be optimized for end-to-end performance using parallel processing. Massive parallel processing architectures with distributed memory architectures and increasingly specialized chips for NN inference have also proven to be very energy-efficient and capable of impressive computational throughput when fully utilized by concurrent algorithms [27], [28]. However, the performance of such architectures has been largely limited by the ability of algorithms and higher programming languages to make efficient use of them. Nevertheless, the inherently concurrent nature of computation and memory access across wide and deep neural networks (DNNs) has demonstrated a surprising ability to achieve high resource utilization on these architectures with minimal application-specific tuning or optimization required.

Apart from all the intellectual beauty of SP techniques that have given rise to fully learned wireless communication systems, the above reasons prove why ML enhancements and extensions could massively improve the existing wireless communication physical layer solutions. This entails resorting to model-driven paradigms for network design, in which the best policies to be applied at particular instances come not only from an analysis of mathematical models, but are also derived from the study and processing of previously acquired communication network data. Then, the performance obtained by given policies from previous communication episodes, the system should have the ability to decide on the best policy to apply at the present communication episode. This is the fundamental operational principle that underlies the operation of model-based RL strategies.

1.1.2 Research Gap

From the above analysis, it is evident that the best way to overcome the current problems in CRNs is applying a good extension of the traditional SP strategies using ML techniques, deep architectures in particular. Deep architectures such as DL and DRL have become the most rapidly advancing components of ML and exhibit far superior performance compared to the other ML strategies. As a result, they are expected soon to become, by far, the largest drivers of future mobile and wireless networking [31]. Surprisingly enough, the application of deep architectures in wireless communication systems has only been realized recently and has since been explored using data-driven approaches. Data-driven approaches focus on building systems that can identify the best solution based on having seen a large number of solution examples of question-answer pairs. Most of the existing research works in DL focus on data-driven approaches, which treat wireless communication systems as "black boxes" and train them by using a huge volume of data [170].

Training a wireless network in this way requires sufficient computing resources and extensive time, both of which are seldom found in communication devices. As a result, this fundamental postulate of data-driven DL approaches will be significantly weakened in the near future, which necessitates model-driven DL in communication system design. Thus, in contrast to the current data-driven DL approaches, this thesis proposes the use of model-driven DL strategies that combine the communication systems' domain knowledge together with DL strategies with the objective of reducing over-reliance on huge amounts of training data. Although the main reason for using model-based DL approaches is to reduce reliance of network design and operation on huge amounts of training data, the vision of this work is not to completely replace data-driven approaches. However, in order to overcome the complexity crunch, it is believed that a hybrid of model-based and data-driven approaches, which is a vision that will be supported throughout this thesis, is necessary.

1.2 RESEARCH QUESTION(S) AND OBJECTIVES

1.2.1 Research Question(s)

The main question is however not whether DL or model-based DL strategies will be integrated in future mobile and wireless networking solutions, but instead how and when will this integration take place? This question is backed by the positions recently taken by several leading telecommunication companies, [33], [34], and also supported by the fact that initial steps towards the standardization of intelligent wireless communication systems have already been taken. In February of 2017, the European Telecommunications Standards Institute (ETSI) initiated an Industry Specification Group (ISG) - the Experiential Network Intelligence (ENI) whose task was to define a cognitive network management architecture that is capable of utilizing AI techniques [29]. Their mandate included designing context-aware policies that will adjust the offered services based on the changes in user requirements, the environmental conditions, and up-to-date business goals. This would lead to a new paradigm referred to as the observe-orient-decide-act control, which would represent the first standardization step towards the definition of an experiential system, which is a kind of system that learns from previous experiences to improve its knowledge of how to act in future situations. With the IoT concept having revolutionized the usage of sensors that produce relevant data, how can CRs exploit this paradigm together with the flexibility of the already available data in order to simplify the implementation of spectrum sensing and spectrum access?

1.2.2 Research Objectives

The general research objectives are to identify, through literature investigation, the limiting problems associated with spectrum sensing and spectrum RA in distributed CRNs. With the identified problems identified in both spectrum sensing and spectrum RA, which are indeed the factors limiting the realization of optimality in spectrum management, then this research is split into three specific objectives as follows:

- In distributed spectrum sensing scenarios, the received spectrum sensing data (SSD) may be incomplete owing to malicious SUs, imperfect reporting channels or specific collaborative spectrum sensing schemes (CSSSs). In another similar case, malicious SUs might launch Byzantine attacks and disrupt collaborative spectrum sensing by generating and sending falsified spectrum sensing data [30]. Regarding imperfect reporting channels, the spectrum sensing results contributed by other SUs might not reach their destinations owing to poor channel conditions, while in the case of specific CSSSs, SUs using energy-saving spectrum sensing schemes might decide not to contribute their observations when they have no data to send. For these reasons, the SSD that was contributed will have gaps of missing entries. As a consequence, the performance of spectrum detection may degrade, especially when the amount of missing SSD entries is large. Therefore, the first objective is to develop a system model that incorporates the limiting factors and makes it possible to investigate a viable solution to address the missing SSD problem.
- One other major challenge that would be encountered in distributed spectrum access (DSA) in future mobile and wireless communications is the issue of energy-efficient operation for heterogeneous traffic. Energy-efficient operation in mobile and wireless networking has been extensively studied and has achieved outstanding results for voice and data applications. One widely adopted energy efficiency strategy is in the context of power control where base stations (BSs) alternate their operation modes based on the amount of traffic offered. However, with the increasing demand for multimedia services, it is necessary to establish other power control algorithms for multimedia information sources. Thus, the second objective is to develop an efficient spectrum access strategy for multimedia information sources that uses resource consumption efficiency and traffic prediction to effect a trade-off between QoS and energy saving.
- Another pervasive problem hindering the deployment of CRNs for the next generation of mobile and wireless communications is the limited and strict energy constraints of network end devices. This problem is exacerbated by the fact that IoT forms a key component of next generation deployment strategy where networks will become more distributed and autonomous. As the network becomes increasingly distributed with autonomous operation, energy consumption in network devices increases, mostly owing to the overheads caused by the resulting transmissions of control information. Thus, the issue of energy efficiency needs to be addressed, since in

distributed resource allocation, network entities have to make decisions to find the optimal sub-band and power level for transmission without having to wait for global information. Making distributed decisions locally may incur limited transmission overheads, but under the uncertainty of the network environment they are prone to resource and energy wastage owing to unused time-slots and packet collisions, respectively.

1.3 RESEARCH HYPOTHESIS AND APPROACH

Following from the two research questions stated above, this research work has followed the well-established pattern of carrying out technical research in the field of wireless communications engineering - literature review, system modeling, simulation and analysis.

- **Systematic review**: The first part of this research work is dedicated to an in-depth exploration of ML concepts and their application in mobile and wireless networking, spectrum management problems in particular. A systematic review of the application of ML and DL strategies in spectrum management research categorized into different domains and techniques is provided. The first research question on how to tailor DL to mobile and wireless environments is answered in this section, which culminates by pointing out current research challenges that need to be pursued in the direction of wireless Big Data.
- **System modeling**: The second research question is addressed by developing various models that are tailored to specific problems in spectrum resource management. Each system model seeks to solve a specific problem by incorporating several identified factors that limit the realization of optimal resource management solutions. The solution models developed employ concepts of DL and DRL in solving the identified limiting problems associated with resource allocation in CRNs. These models, capturing and addressing each of these limiting challenges, are thoroughly analyzed and results are obtained and presented in the various chapters of this thesis.
- **Simulation and numerical analysis**: The solutions developed are simulated in two stages using MATLAB^{TM} software. First, the RA is simulated using the disciplined optimization toolbox called CVX used for solving optimization problems. Then, secondly, a DL toolbox (formerly called a neural network toolbox), which provides a framework for designing and implementing DNNs with algorithms is used to train the CR systems and monitor the training-learning progress.

1.4 RESEARCH SCOPE AND CONTRIBUTIONS

1.4.1 Research Scope

This research focuses on enhancing energy-efficient operations for spectrum management using DL techniques to uphold the decision-making processes in distributed CRNs. This thesis contains six chapters, Chapter 2 is a systematic review, Chapters 3, 4, and 5 form the main contributions of the thesis. Chapter 6 gives the conclusions and makes recommendations on future research. The main chapters:

- **Chapter 2** gives a systematic literature review which is an in-depth exploration of the state-of-the-art approaches in spectrum management. It reviews ML techniques used in spectrum management, makes suggestions for future research and then advocates the use of DL techniques and wireless Big Data for intelligent spectrum management.
- **Chapter 3** deals with problems encountered in distributed spectrum sensing scenarios, where the received SSD may be incomplete owing to imperfect reporting channels or specific CSSSs. Because of incomplete SSD, the performance of distributed spectrum detection approaches may degrade, especially when the percentage of missing SSD entries is large. Thus, spectrum occupancy reconstruction is performed based on incomplete spectrum sensing results and a computational DL approach is proposed as a solution. This approach generally assumes that during spectrum message collaboration, PUs remain inactive, thus interference concerns are generally ignored at this point.
- **Chapter 4** focuses on the major challenges in spectrum management in future mobile and wireless communications, among which is energy efficiency. One widely adopted strategy, which has been studied extensively in the context of power control, is the ON/OFF BS operation. Here, BSs alternate their operation modes based on the amount of traffic and have achieved good results for voice and data applications. With the increasing demand for multimedia services, it is necessary to establish other power control algorithms for multimedia information sources.
- **Chapter 5** deals with the constrained energy management problem in CRNs by proposing a hierarchical DRL technique, which is a CRN environment using a DL approach. This enables for the use of dis-aggregate instantaneous user-specific information to inform and improve CR operations on a network scale under the assumption that CRNs have limited and strict energy

constraints. Because of the current lack of computationally efficient optimization techniques to enable for the use of integrated models to design sustainable energy-efficient RA strategies, a Markov decision process (MDP) is used to embed energy consumption and traffic behavior models into a deep Q-network (DQN) architecture to achieve energy-efficient CRN operation.

1.4.2 Research Contributions

The overall contribution of this thesis is the utilization of a framework that combines the use of both model-based and data-driven approaches to address the research problems identified earlier. The specific contributions include the following;

Chapter 2: Deep Architectures for CR Spectrum Management in Future Generations of Mobile and Wireless Networking: A Systematic Review

- This chapter offers clear identification of DL approaches that are employed in spectrum management and energy efficiency solutions to achieve better network sustainability. A careful block structured analysis of traditional communication systems is presented and the need for a DL technique for realizing end-to-end communications is clearly spelt out. Opportunities and challenges in realizing a DL end-to-end system are discussed in parallel with future directions. The scope of this chapter is to seek to answer the following questions: (i) As software defined radio (SDR) and CR become feasible, how will the decrease in sensitivity influence overall network performance and client power consumption in streamlined radio deployments negatively in future mobile and wireless networking? (ii) Why is there a need for improved spectrum management approaches as the drive towards the IoT intensifies? (iii) Why is DL seen as a promising method for solving mobile networking problems on a broader scale? (iv) How can researchers and developers tailor signal processing and DL methods to achieve spectrum management schemes specifically for future mobile and wireless networking problems? (v) What other cutting-edge DL methods that are relevant to future mobile and wireless networking? (vi) What are the currently successful DL applications in the CRN domain? (vii) Which are the most noteworthy research directions for future research pursuit?

Chapter 3: Spectrum Occupancy Reconstruction in Distributed Cognitive Radio Networks Using Deep Learning

In this part, a spectrum occupancy reconstruction technique for distributed CRNs is proposed, where the problem of missing spectrum occupancy data is treated as a missing value imputation problem and solved through a stochastic gradient decent (SGD) algorithm.

- The spectrum occupancy problem was modeled as a magnetic excitation state recovery problem, in the context of ferro-magnetism, where the CRN is represented by a plenary grid on a Markov random field (MRF). This is an idea borrowed from probabilistic DL where a certain area can be factorized into a Bayesian network. Thus, the CRN topology is illustrated using the plenary grid where the distribution of SUs can be factorized into a cluster within a 1500 m $\times$ 1500 m geographical area. This area is further divided into 50×50 sub-areas where each grid is a 30 m $\times$ 30 m square area.
- The binary logic of SU spectrum observations and interactions including only one-hop SUs as the nearest neighborhood forms a two-point clique and the problem is summarized as a matrix factorization problem. The researcher then introduces and motivates a new training principle for DL representation based on the idea of making the learned representations robust to partial corruption of the input data pattern. Here, a time-series spectrum occupancy prediction using deep belief networks (DBNs), which are a kind of probabilistic generative NNs composed of a restricted Boltzmann machine (RBM), is used.
- An RBM captures the feature of input space of spectrum occupancy data from an MRF using link energy functions from the Ising model. After pre-training of the RBM using their energy functions, a spectrum occupancy data matrix is obtained using the Metropolis-Hastings algorithm. To determine the size of the samples and learning rates, Gibbs sampling is adopted during the training process and SGD is used to process, learn and impute the missing entries. More precisely, the computational DL technique is trained to map between observed and unobserved spectrum sensing results from collaborating SUs to predict the values of the missing values accurately.

Chapter 4: QoS provisioning and Energy Saving Scheme for Distributed Cognitive Radio Networks Using Deep Learning

In this part, a learning scheme for upholding the QoS requirements in an energy-efficient manner evaluated in a single BS scenario is proposed. This is a spectrum management scheme that uses resource consumption efficiency and traffic prediction via DL to make effective trade-offs between QoS provision and energy saving.

- Firstly, a distributed dynamic RA based on uplink (UL) power allocation and SU resource reservation protocol is proposed. Resource reservation is a transport layer protocol designed to reserve resources across a network for QoS using the integrated services model and is adopted in this paper to provide seamless SU hand-offs and also handle the transmission of mission critical applications under the main constraint of bandwidth limitation. Resource reservation and resource allocation are respectively solved using geometric programming (GP) and weighted bipartite matching to obtain a resource consumption efficiency. It should however be noted that the weighted bipartite graph is only used as a means to bridge from traditional optimization to the DL cost function.
- Secondly, the resource consumption efficiency is utilized as an optimization weight between power consumption and traffic load. A corresponding weighted cost function is then defined, in which power consumption are added together with different weights reflecting their contribution to the overall system power consumption. Deep learning is used to solve the cost function and derive system states and control actions associated with the new systems' state-space.
- Finally, a traffic flow matrix is fed into a SAE DNN and control actions are applied and using this formulation, relevant parameters of the operating environment such as workload arrival patterns are estimated and used by the model to predict the future behavior over a finite horizon. The output of the SAE is a regression between the previous and current states and an appropriate packet processing scheme is chosen between mean slowdown MS FCFS and and mean slowdown MS PS. The belief that MS is important in packet-by-packet processing as a measure of the systems energy efficiency is proven by the results that different traffic flows consume significantly different amounts of resources and the choice of a processing scheme determines the overall energy efficiency of the system.

Chapter 5: QoE-Driven Resource Management in Energy Constrained CRNs Using Deep Hierarchical Reinforcement Learning

In this part, the QoS and energy-saving scheme studied in the previous section is modified to a QoE/QoS scheme for constrained energy management in CRNs in order to handle more realistic network dynamics, such as power consumption and transmission latencies. It is assumed that coupling QoE and QoS in an instantaneous power consumption model can yield detailed network-wide power consumption estimates for subsequent RA decisions. Due to algorithm deficit in this pervasive problem, an FMDP solution was derived and then modified using a DQN with experience replay algorithm.

- Firstly, to improve QoE evaluation using the perfect docition to demonstrate the possibility of handling dynamic RA problems using an innovative user-centric and context-aware technique. SUs experiencing heterogeneous traffic use the mean opinion score (MOS) to measure QoE in a distributed manner and the RA problem is solved using DRL.
- Secondly, in improving the CRN-level QoS, an energy-efficient dynamic RA scheme that minimizes power consumption, subject to transmission delays using MDPs was proposed. A constrained power minimization technique that chooses optimal and energy-efficient decisions for each application-allocated resource and penalizes packet losses was used. Using MDPs through the analytic bounds of queuing theory and the scaling laws of energy consumption, a better energy consumption model is formulated as a constrained MDP and solved using FMDPs and a Lagrange approximation technique.
- Lastly, the solution obtained through FMDPs was transported into a DQN framework, which using experience replay to estimate the state-action value solutions without supervision, offered a quicker way of minimizing buffer overflows and power consumption.

1.4.3 List of Publications

1.4.3.1 Publications in Peer-reviewed Journals

- **M. C. Hlophe** and B. T. Maharaj, "Spectrum Occupancy Reconstruction in Distributed Cognitive Radio Networks Using Deep Learning," *IEEE Access*, vol. 7, no. 2, pp. 14294 - 14307, January 2019.
- **M. C. Hlophe** and B. T. Maharaj, "QoS Provisioning and Energy Saving Scheme for Distributed Cognitive Radio Networks Using Deep Learning," *Journal of Communications and Networks*, vol. 22, no. 3, pp. 185 - 204, June 2020.
- **M. C. Hlophe** and B. T. Maharaj, "QoE-Driven Resource Management in Energy Constrained CRNs Using Deep Hierarchical Reinforcement Learning," *IEEE Transactions on Vehicular Technology*, **Under Review**.

1.4.3.2 Publications in Peer-reviewed Conferences

- **M. C. Hlophe**, B. T. Maharaj and S. Hamouda, "Distributed Spectrum Sensing in Distributed Cognitive Radio Systems Using Graph Theory," in *Proceedings of IEEE AFRICON*, Cape Town, South Africa, 18 - 20 September 2017.
- M. M. Sande, **M. C. Hlophe**, and B. T. Maharaj, "Machine Learning-based Base Station Association for Resource Allocation in 5G Heterogeneous Cognitive Radio Networks," *40th Meeting of the Wireless World Research Forum (WWRF'40)*, Durban, South Africa, 31 May - 01 June 2018.
- **M. C. Hlophe** and B. T. Maharaj, "Optimization and Learning in Energy Efficient Resource Allocation for Cognitive Radio Networks," in *Proceedings of IEEE Vehicular Technology Conference*, Kuala Lumpur, Malaysia, 28 April - 01 May 2019.
- **M. C. Hlophe** and B. T. Maharaj, "QoE-Driven Resource Allocation for SUs with Heterogeneous Traffic using Deep Reinforcement Learning," in *Proceedings of IEEE Wireless Africa Conference*, Pretoria, South Africa, 18 - 20 August 2019.
- **M. C. Hlophe** and B. T. Maharaj, "Secondary User Experience-oriented Resource Allocation in AI-empowered Cognitive Radio Networks Using Deep Neuroevolution," in *Proceedings of the 91st IEEE Vehicular Technology Conference (VTC2020-Spring)*, Antwerp, Belgium, 25 - 28 May 2020.

CHAPTER 2 A SYSTEMATIC REVIEW OF DEEP ARCHITECTURES FOR SPECTRUM MANAGEMENT IN DISTRIBUTED COGNITIVE RADIO NETWORKS

2.1 CHAPTER OVERVIEW

Proper utilization of wireless network resources is an essential requirement operating for CRNs in the next generation of mobile and wireless networking. The CR technology aims to fulfil the requirements of next-generation networks by exploiting AI strategies in order to realize a cognitive engine that exploits awareness about the surrounding radio environment, optimize the use of radio resources and adapt relevant transmission parameters. This chapter systematically explores different ML approaches to realize the decision-making function of the cognitive engine that aims at configuring the radio transmission parameters according to the environmental conditions. This requires a definition of a set of parameters that brings about dynamic automation in order to usher predictive RA to the forefront of CRNs. However, among the existing research contributions and literature surveys about the integration of ML strategies in wireless communication networks, a comprehensive survey is still lacking.

Therefore, this chapter aims at filling the gap between ML and mobile and wireless communication networks by presenting a systematic review that addresses the hybridization of ML techniques with SP strategies into CRNs. As a result, the concept of intelligent spectrum management is first introduced and several reasons why ML-based spectrum management is required in future CRN deployments are given. Then an in-depth review of deep architectures and their advocated solutions for future mobile and wireless networking is provided, with emphasis on CR spectrum management.

In addition to reviewing the most recent literature, DL model strategies are outlined regarding future spectrum management problems. Also, a further investigation of how DL strategies could be tailored into future mobile and wireless networking problems to achieve the best performance in complex environments is conducted.

2.2 INTRODUCTION AND BACKGROUND

With the migration to the 5G era, in the broader context of future mobile and wireless networks, wireless devices are expected to demand more effective and efficient communications than ever before. However, the vexing dual problem of spectrum scarcity and spectrum under-utilization is one major bottleneck in the development CRNs that has highlighted the need for improved approaches to spectrum management. CR-driven DSA spectrum sharing has been advocated as the de facto solution to spectrum management and has consequently become topical among researchers in enabling efficient migration to 5G and beyond [33]. However, despite CR technology having been studied for close to two decades now, there are still challenges that have been overlooked and need to be reconsidered and addressed. Examples of remaining challenges are establishing the interference threshold for PUs and determining how SUs can autonomously become aware of the interference they are causing.

Furthermore, the effect of SU activities in the primary network is still being characterized as a dynamic system, while the throughput of PUs is still considered as a time series with temporal dependencies. In this case, the accuracy of spectrum sensing has to be high, which entails that SU systems need to be able to robustly and efficiently sense spectrum holes if they want to utilize the unlicensed spectrum opportunistically [36]. However, in a dynamically changing wireless network environment spectrum detection techniques based on a single PU parameter cannot guarantee the accuracy of harmless spectrum access. Thus, spectrum shortage and under-utilization, coupled with the growing diversity of services of future networks require that all CR devices be equipped with a technology that will enable more autonomous operation. This necessitates the devising of distributed spectrum management mechanisms. Therefore, in order to improve their ability to utilize wireless resources to meet the unprecedented user demands, the integration of AI strategies is a requisite [37].

Future mobile and wireless networks will become more distributed with an increase in heterogeneity of services and vertically integrated networks, and the integration of AI strategies into

CR operation will enhance the coordination of all network devices and improve the effectiveness of spectrum management. This integration will enhance distributed CR deployments since mainstream research in CRs has always focused on infrastructure-based (i.e., centralized and decentralized) network architectures; as a result devising distributed spectrum management mechanisms has remained substantially unaddressed [38]. The integration of AI strategies into a distributed wireless networks will not only have a positive effect the transmission technologies, but will also have a significant impact on the way in which the wireless network should be controlled. Due to their autonomous nature, future mobile and wireless networks will be mostly controlled through feedback signals to avoid instability and malfunctioning. This is because the futuristic use cases such as autonomous driving and the tactile internet, and many others will force a fundamental change in the way in which networks are currently managed.

The main challenge with distributed spectrum management is the simultaneous handling of wireless parameters such as frequency band, symbol modulation, coding rate, etc., to ensure an optimal network operating point that is both energy-efficient and does not cause harm to PUs [39]. With the 5G era being an era of ubiquitous connected devices, requirements of high-speed connectivity, high network efficiency and high security, which have been put under three categories - enhanced mobile broadband), large-scale IoT, and ultra-reliable and low-latency connections [40] - have been put forward. With the IoT driving the 5G era, a staggering number of mobile devices will have to access the wireless network simultaneously. This will create a heavy burden of authentication computing in the wireless network owing to the low computational capabilities of network devices [41]. This necessitates lightweight access methods for intensive application scenarios. The necessary condition for IoT operation is enabling the interconnection and communication between a large number of network devices and equipment. This brings about an increase in data transmission rate requirements, the level of heterogeneity and the expected rate of change of network traffic and presents new challenges related not only to the management of resource allocation, but also to network energy consumption [42].

The consequences of having a more distributed network with mobile and heavy traffic users (e.g., high-definition video streaming), where every user demands more immediate bandwidth, are the limitations associated with energy efficiency in CRNs. Solving energy-efficient resource allocation problems has to be done in parallel with transmission delay requiring cross-layer treatment, which, if not elusive, is computationally complex. This challenge is exacerbated by the lack of low-complexity joint-layer strategies and coherent research contributions on this aspect are scattered across all layers of the protocol stack. This problem can be effectively addressed using the exploitation-exploration technique often modeled as multi-armed bandit (MAB) problems in reinforcement learning (RL) [43].

The MAB technique intuitively describes the problem that a gambler repeatedly pulls on one of the arms of a gambling machine (i.e., the bandit) for a certain amount of reward (i.e., dispensing a certain number of coins). The MAB seeks to maximize the average reward after pulling arms several times which can help in coming up with effective cross-layer strategies. Thus, the entry into the IoT era not only suggests and motivates the implementation of DL strategies, but also that they should be implemented in a distributed manner. As a consequence, this condition brings about other serious challenges that need urgent attention. For example, massive IoT deployments are usually constrained in terms of computation, storage and energy capabilities which implies that design efforts should be directed at the optimization of power consumption and operational costs rather that optimizing performance. Therefore, this chapter systematically reviews AI strategies applied to wireless networks and their integration into future mobile and wireless networking.

2.2.1 Scope of Review

The prime objective of this review is to provide a systematic review of ML techniques in the broader context of mobile and wireless networking domain with specific context towards spectrum sensing and spectrum sharing in CRNs. While the scope of this chapter is on DL strategies, it also covers a wide range of supervised learning and RL techniques that are increasingly important and have not yet been covered in previous reviews. The first part provides an overview of CR spectrum management and then provides an introduction to ML communication systems. The first part does this while giving a concrete overview of their applications and fundamental descriptions of the algorithms presented as solutions to wireless communication problems. Since the material in this paper is presented from a probabilistic perspective, this part introduces ML concepts by reviewing the methods from the mathematical background of supervised learning. This is because in as much as the ML techniques are rooted in probability and information theory, they are actually based on supervised learning. The second part provides a concrete review of RL and many of its concepts, which leads to the realization of its associated deep architecture (i.e., DRL) and the variety of applications across wireless networking.

The third part of this review deals with the core concepts of DL and their applications in mobile and wireless networking. These include AEs and generative adversarial networks (GANs), which have their roots in supervised learning but have found applications in DL problems. In summary, in order to apply DL methods in mobile and wireless networking, one must realize that it is paramount to identify the intimate connections between RL and DL, which become evident when dealing with issues of network traffic and user analytics. The last part of this review is aimed at providing a rigorous presentation of deep architectures together with the advantages and disadvantages of their application in the wireless networking domain. Consequently, this might not be an all-encompassing and comprehensive review, but it has been made as systematic as it could possibly be. Thus, in each of the sub-fields of ML, the researcher first introduces the mathematical background and applications, followed by discussions of challenges encountered in the training of the architectures. Even though more emphasis is put on deep architectures (i.e., DRL and DL), the presentation is intended to be as rigorous as necessary to allow the reader to understand the relationship between the different ML methods and how they are applied to wireless communication problems.

2.2.2 Summary of Contributions

Beyond reviewing the most relevant literature, this review also discusses the key pros of various DL techniques and outlines DL model selection strategies applicable to future mobile and wireless networking problems. Methods of tailoring DL into individual mobile and wireless networking tasks aimed at achieving better performance in complex operating environments, with the ultimate goal of providing a guide for networking researchers who intend to employ DL strategies in solving wireless problems, are further investigated. Thus, this chapter seeks to provide answers to the following key questions: (i) As SDR and CR become feasible, what will be the negative influence on overall network performance and client power consumption in streamlined radio deployments in future mobile and wireless networking? (ii) Why is there a need for improved spectrum management approaches as the drive towards the IoT intensifies? (iii) Why is DL seen as a promising method for solving mobile and wireless networking problems on a broader scale? (iv) How can researchers and developers tailor SP techniques and DL methods to achieve spectrum management schemes specifically for future mobile and wireless networking problems? (v) What other cutting-edge DL methods that are relevant to future mobile and wireless networking? (vi) What are the currently successful DL applications in the CRN domain? and (vii) Which are the most noteworthy research directions for future research pursuit?

While the research contributions reviewed only provide partial answers to these questions, the contribution of this chapter goes beyond the current context of CRNs by attempting to provide comprehensive answers to the above questions. The first question addresses the challenges facing CRs in future mobile and wireless communications. The second question is addressed in the section "CRs in the IoT era", which also discusses previously unanswered questions related to model and algorithmic deficit problems. The third question is addressed by specifically focusing on the crossovers between DL and mobile and wireless networking and the use of DL in wireless Big Data. In response to the third and fourth questions, it is clarified that even though the ML vision is quite appealing, it is currently still unclear to what extent ML techniques can augment the expert knowledge that has been developed during the last century.

Regarding the fifth question, it has been found that it is still very difficult to obtain light-weight solutions that meet the resource-constrained application scenarios. The sixth question is addressed through the integration of DRL into future wireless communications networks will surely revolutionize the conventional model-based network optimization to model-free approaches and meet various network application demands. DRL methods, through their interaction with the environment, can provide autonomous decision-making mechanisms for the network entities to solve non-convex, complex model-free problems such as dynamic spectrum access, handover, scheduling, RA, and energy efficiency. Though DRL has shown great potential to address emerging issues in complex wireless networks, there are still domain-specific challenges that require further investigation. The aspects that address the second part of the sixth question include the design of proper deep neural network (DNN) architectures to capture the characteristics of 5G network optimization problems, the state explosion in dense networks, multi-agent learning in dynamic networks, limited training data and exploration space in practical networks, the inaccessibility and high cost of network information, as well as the balance between information quality and learning performance.

2.3 OVERVIEW OF COGNITIVE RADIO SPECTRUM MANAGEMENT

A definition of a CR that gives a clear understanding was offered by the ITU as a radio system that employs a technology that enables it to acquire knowlwdge of its operational environment, and through the use of established policies together with its internal state, dynamically adjusts its operational parameters [44]. In a nutshell, it is an intelligent communication system capable of obtaining awareness of its operating environment through learning, adapts its internal state to the statistical variations of the network in real-time [8]. The primary objective of CRs is to solve the spectrum shortage and spectrum under-utilization duality through DSA (i.e., the opportunistic transmissions of SUs), by monitoring the radio spectrum to locally determine regions that are not occupied by their incumbent users and transmits in those bands [45].

CRN architectures can be centralized, decentralized, or distributed. In centralized architecture, a central and robust unit called the fusion center (FC) incorporates advanced capabilities to sustain the other CR stations [46]. The centralized model is the most cost-effective, since backup operations are less complicated and can be replicated. They also require less control, since maintenance issues are primarily located at the FC. However, the major concern with this architecture is that in case of failure of the FC, the proper functioning of the whole network is compromised. The distributed CR architecture, also referred to as ad hoc, is an alternative to the centralized approach where CR nodes can communicate with another via ad hoc point-to-point connections over the licensed spectrum bands. The distributed architecture has strong advantages, since each node may act selfishly and can endure various network failures independently. In this way, the network contains no weak points, since each node is able to handle its backup and control operations locally [47]. As a result, the distributed approach requires additional hardware resources to be implemented at the node level, hence it is cost-effective in terms of deployment.

Since in this chapter the researcher is mostly interested in the decision-making, readers interested in detailed analysis of CRN architectures are referred to a comprehensive review in [11]. In CRs, decision-making can either be centralized (i.e., database access) or distributed (i.e., consensus-based). A high-level illustration of both these decision-making processes, defined in [48] and [49], is presented in Figure 2.1 below.

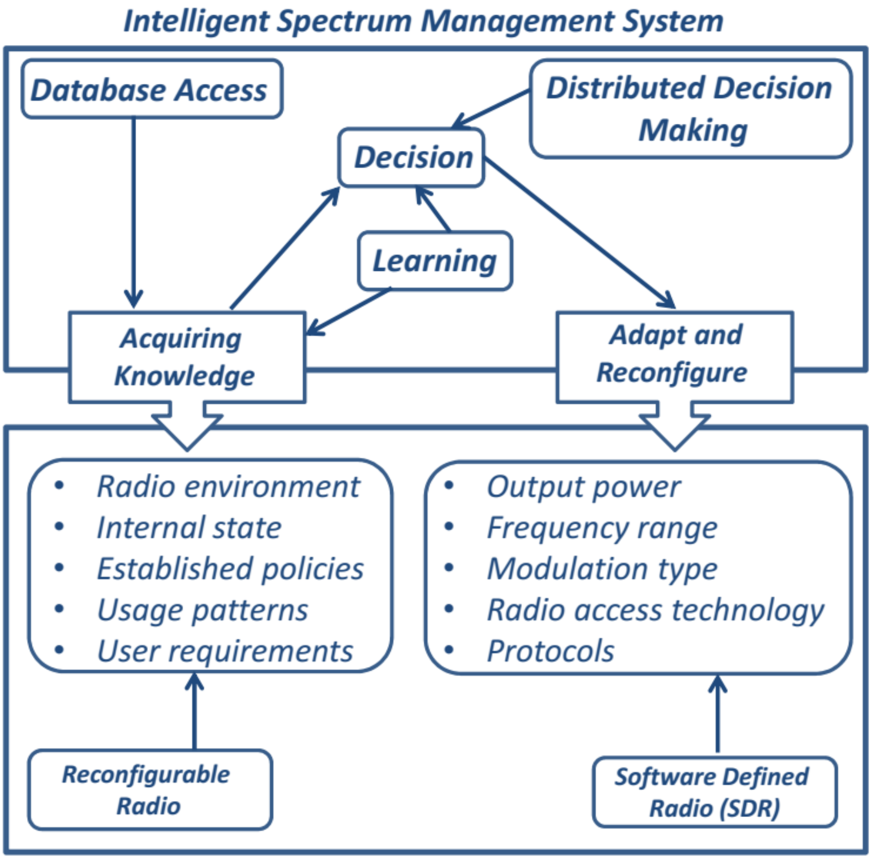

Figure 2.1. High-level Concept of Intelligent Spectrum Management in CRNs.

2.3.1 Reconfigurable Radio

The reconfigurable radio, left-hand side of Figure 2.1, was developed in the early years of TV white space research after regulators had determined that the use of spectrum sensing as a stand-alone method could not sufficiently protect incumbent TV users [50]. Thus, because of uncertainties and restrictions in spectrum sensing, spectrum databases are a favourable option in spectrum management for many coexistence scenarios in TV bands. Hence, these databases are currently the dominant dynamic spectrum-sharing approach for TV white spaces. The operation of the reconfigurable radio is shown in Figure 2.1 (right-hand side). Spectrum databases may include information about device locations and their activities, spectrum usage, coverage/interference levels, relevant regulations and policies, services and networks. This information can be used for several environmentally aware network operations such as planning, troubleshooting and radio resource management. Spectrum databases are a way of controlling spectrum sharing where the systems are provided with predictable QoS, which helps in avoiding unstable situations such as unnecessarily frequent channel switching and makes implementation of practical systems easier. This kind of spectrum management is what is usually referred to as reconfigurable radio which operates by acquiring knowledge about the radio environment, internal state, established policies, usage parameters and user requirements. These are discussed in Table **2.1** below:

Table 2.1. Reconfigurable Radio Characteristics for Decision-making.

Reconfigurable radio characteristics for decision-making	
Main Learning Component	Learned Information
Radio Environment	Spectrum usage status, coverage areas available radio systems and allocated spectrum bands, interference levels.
Internal State	Frequency bands and protocols, distribution of traffic load and transmission power levels.
Established Policies	Spectrum bands allowed for CR use under certain conditions, that include maximum transmit power levels in operating and adjacent bands, and the set of rules that CRs should follow to avoid causing harmful interference to PUs and other SUs.
Usage Patterns	Usage patterns that include the behaviour CR systems and other radio systems.
User Requirements	User requirements include the need for high bandwidth, high throughput, low latency, low network costs, fast and stable download times.

However, there are still several challenges left before databases can be applied to CRNs, since the characteristics of primary systems and their signals differ significantly, e.g. regarding received signal power levels and coverage areas.

2.3.2 Software Defined Radio

Given the need for more efficient use of the radio spectrum, the CR uses SDR technology while attempting to manage the radio spectrum. The term SDR is used to describe a viable device, with components quite different from the traditional transceiver, that is primarily defined by software but includes significant hardware components [53]. Thus, an SDR is a design paradigm for CR devices that is an identifier of a class of radios that could be programmed and reprogrammed through software. The operation of the SDR, as shown in Figure 2.1 (i.e., right-hand side), the adaptation and reconfiguration means that the SDR and CR are bound to change the way they communicate. This is made possible through its reconfiguration capability that allows for dynamically and autonomously adjusting the operational parameters and protocols based on the available knowledge and environmental context in order to achieve its objectives. Numerous research works in the field of CRs concentrate on observation and adaptation as the main notions of CR operation. However, in this chapter the researcher would like to view the same notions from a different perspective in order to make them more distinguishable CR networking aspects. These are (i) learning and reasoning, and (ii) inference and decision-making that enable the CR to learn and reason, reach a conclusion based on the results obtained and then make a decision. The discussion on these two aspects, which proceeds in the following subsections, encompasses both the application of game theory and RL techniques to CRNs.

2.3.3 Learning and Reasoning in CRs

The learning and reasoning concept in CRs stems from the field of AI, which fosters optimal resource usage and management that seek to enable numerous secondary spectrum access applications [54]. Game theory, dynamic games in particular, offers the basic mathematical tools to model the way in which the interactions among autonomous players (e.g., CRs devices seeking to maximize their throughput) occur. Modeling this aspect through dynamic games helps in studying how past experiences affect current CR actions, which allows one to be able to model how the CRs learn from their past and also from their neighbors. In CRNs, the application of dynamic games has been in modeling the behavior of SUs competing for spectrum resources, and also in modeling negotiation interactions between PUs and SUs. There are different types of dynamic games, ranging from repeated games studied in [55] to stochastic games studied in [56], [57] (**NB:** also discussed in Section **2.5.4.3**) and evolutionary games studied in [58].

The most improtant aspect to consider in learning mechanisms is in which strategy is the learning performed (i.e., supervised, unsupervised or reinforced). When applying any of these ML techniques in CRNs, the first challenge of the learning process is in avoiding making wrong choice of actions that will lead to unfeasible decisions, especially in autonomous CR learning processes. The second challenge is in defining the learning process within the context of the CR system and the objectives which will contribute to the final decision. In terms of algorithmic design and implementation, the cognitive functionalities that enable the CRs to learn from previous policies and improve their behavior by "reinforcing" those actions that previously led to higher reward, are much complex. The design of such learning algorithms has already presented itself as a huge challenge. This is because the measurements that should be conducted by learning from the environment also introduce other problems in terms of the measurements to be used and how to apply them. However, advances in ML have indicated that RL can be used for both single agents and multiple agents in CR and DSA problems. The RL strategy has been applied to a variety of problems of this nature, including dynamic channel selection studied in [59], transmission power adaptation for spectrum management studied in [60], cooperative sensing in ad hoc networks studied in [61], and multicarrier aggregation studied in [62].

2.3.4 Inference and Decision-making

Inference and decision-making from the information obtained are the most important aspects related to the intelligence of CRs. The primary objective of inference in CRs is in the choice of actions that result to efficient decisions [54]. In CRs, inference is viewed as a decision process that uses both historical and current knowledge of the environmental context to improve CR operation. Furthermore, the learning and reasoning processes need to be enriched with good knowledge based on using ML methods in order to foster increased efficiency in subsequent decision-making in future mobile and wireless networks [63]. Therefore, in the CR sense, there is a tight coupling between inference and reasoning since reasoning is classified into a dichotomy of either instinctive or cognitive. Cognitive reasoning requires the power of cognition which involves the complex interaction between previous and current information and can be better handled using RL. On the other hand, instinctive reasoning applies to biological systems and involves the use of emotions in the entire process of reasoning, inference and decision-making, which may result to a greedy CR operation. At the moment, CRs not being biological systems are stripped of emotions, which makes the application of RL techniques easy. However, with the increase in the intelligentization of CR devices through the use of complex NNs, instinctive learning cannot be ruled out in the near future, given that NNs are derived from biological systems.

State-of-the-art decision-making algorithms in CRs and wireless networks in general involve the use of mathematical programming (e.g., game theory and queueing theory), deep architectures (e.g., computational DL, DNNs, and DRL) and evolutionary algorithms (e.g., swarm intelligence and genetic algorithms). The traditional decision-making in CR systems is the decision on spectrum availability, which involves spectrum sensing, channel selection strategy and the optimization of radio performance [64]. Thus, in case the current channel degrades, an SU can use an appropriate technique from the ones named above in executing one of the following decisions: (i) remain in the current channel, waiting for it to improve (i.e., stay-and-wait); (ii) remain in the current channel, but adjust to the conditions (i.e., stay-and-adjust); (iii) handover to another available and better one that meets its QoS requirements (i.e., spectrum handoff), or (iv) terminate application lifetime, which is undesirable (i.e., channel drop-out) [65].

2.3.5 Analysis of State-of-the-Art Resource Allocation Techniques in CRNs

As SDR technology is maturing rapidly, it is becoming feasible to consider its use in other commercial CR systems. For example, next-generation CRs could enable commercial broadband services to co-exist with government services. By doing so, regulators would be able to shoehorn more services into each band in order to alleviate the spectrum shortage [66]. Currently, more emphasis on CR research has concentrated on the challenges faced by mobile users, i.e., finding the temporary, often fleeting spectrum holes around other users so they can transmit, receive or both. However, fixed BS infrastructure can also benefit greatly from the application of CR techniques and may be able to save network operation energy. BS sleeping mechanisms have been proposed to put idle BSs into sleep mode during low traffic conditions, which can be effectively designed to cognitively decide which BSs should actually go to sleep, using traffic forecasting techniques. There are many immediate cost benefits that can drive the increased use of SDR techniques in BSs, for example, improved market time in the face of evolving telecommunication standards of a single platform for multiple standards. Mobile devices and other client devices with minimal cost implementations, using BSs that follow an

SDR paradigm can support multiple devices within a single wireless access network, thus allowing network operators to offer a wider range of services. This will further drive the deployment of CR technologies into the network infrastructure, which means that users can more reasonably afford the additional processing power and costs required by CR systems.

On the other hand, BSs present a different set of challenges to CR technology, especially that of high computational power consumption. To ease the performance requirements of mobile devices, BSs have a significantly higher performance requirement in terms of sensitivity to weak received signals, transmit power, linearity, and bandwidth. Recently, there has been a convergence of some aspects of BSs and clients: femtocell BSs mean that power and computational resources are more limited; wideband communication schemes mean that clients must now match the BSs in spectral bandwidth [67]. However, the linked issues of linearity and sensitivity remain. A decrease in linearity or sensitivity would have unacceptable impact on overall network performance or mobile client power consumption. From the network perspective, the move toward cell densification (i.e., small heterogeneous cells) to offer increased broadband services to existing mobile clients requires that small cells be able to detect, configure, and self-manage in an independent and distributed fashion. This aspect was noted by Huawei technologies when highlighting the importance of self-organizing networks (SONs) [68] to streamlined radio deployments as well as spectral allocation [69].

The recent breakthoughs in integrated access nodes and back-haul design are required to enable seamless CR operation in dense networking scenarios, especially because of the coming of the 5G and the IoT where plug-and-play will be essential to deployment. Here, integrated access nodes will use their self-organizing capabilities to self-organize and access available spectrum blocks for both access and back-hauling, which are key capabilities for enabling high-frequency spectrum access. This makes SON essential to the imminent future of connected things, since by self-automation, self-configuration, self-optimization and self-healing, some computational and operational resources of the network can be freed up and be deployed elsewhere [70]. Moreover, with the positive explosion of the IoTs, managing and keeping them up require automated approaches that will yield a new set of network assurance challenges that network operators will have to deal with easily [71]. Dealing with so many challenges simultaneously seems to be a very daunting task and these problems can safely be classified as model and algorithmic deficit kind of problems, which are discussed in Section **2.3.7.1** and Section **2.3.7.2**, respectively.

2.3.6 Challenges Facing CRNs in Future Generations

Previous and current generations of mobile and wireless communication systems are based on mathematical models derived from either theoretical considerations or field measurement campaigns. Mathematical models describe quantitatively the effect each system component has on the overall network performance [72], which means that they are at the heart of all phases of wireless communication network design. Mathematical models describe overall network operation using optimization theory from planning to deployment, to resource management and to maintenance and control as well. Therefore, network infrastructure nodes are deployed and managed through optimization theory, which uses a centralized allocation of available resources [73]. Using optimization theory, state-of-the-art OSA algorithms have shown tremendous success in solving spectrum management problems since the proposal of CRs [74]. For example, algorithms for channel selection and optimum spectrum allocation and utilization have shown great improvement using mathematical programming techniques such as game theory, queuing theory, search heuristics such as genetic algorithm, swarm intelligence, etc. RA and energy efficiency are of great importance in CRN communications and many RA strategies have been used to optimize the performance of RA and energy efficiency in CRNs. Noteworthy strategies include mathematical programming (i.e., game theory, queuing theory, etc.) and evolutionary algorithms (i.e., bio-inspired algorithms), however, the implementation of these strategies have proven to be challenging in terms of achieving real-time performance. This is partly because most of these approaches require accurate and timely channel state information (CSI) and other network statistics [75].

In CRNs, spectrum management entails the ability of SUs to measure the channel quality and perform channel selection based on its utilization factor, its packet dropping rate, as well as throughput, which are very important aspects that define the QoS. In terms of defining the QoS, the channel utilization factor itself can be determined from aspects such as spectrum sensing accuracy which include the false alarm rate, and the channel holding time [76], while the packet dropping rate can be determined from the expected delay for an SU that is already in the queue associated with that channel [77]. Flow throughput is, in the context of resource sharing, balanced fairness, which considers the per-flow throughput with a large number of traffic classes. Usually flow throughput uses the decoding-cumulative distribution function [78], along with the prioritized Raptor codes [79].

Spectrum management techniques that combine both resource allocation and energy efficiency have been proposed with the aim of increasing transmission capacity at reduced energy consumption. For example, the authors in [80] proposed fractional programming and convex optimization techniques for energy-efficient optimal power allocation, while the authors in [81] focused on the energy efficiency in non-orthogonal multiple access enabled heterogeneous cloud radio access networks. In another contribution in [82], energy efficiency optimization was studied in a CR massive input massive output (MIMO) network in both centralized and distributed energy efficiency algorithms. In these strategies, optimal solutions are obtained through the use of iterative algorithms, irrespective of spectral efficiency and energy efficiency maximization. Mostly, energy efficiency problems are originally fractional problems that are non-convex; the Dinkelbach's method of fractional programming is employed to transform them into parametric problems and solve the optimal solution iteratively.

However, as wireless technology enters into the era of all "things" connected to the internet, smart devices will require rigid performance such as ultra-low latency and superior energy efficiency [16]. As the IoT continue to steer operations in mobile and wireless communications, new spectrum use cases such as device-to-device, human-to-human, machine-to-machine, vehicle-to-vehicle, and many more have developed, all of which have different performance characteristics and differing resource demands. This presents the opportunity and potential of reshaping and redefining the current wireless communications landscape. However, they all need to have CR capabilities in order to meet their staggering resource requirements because of the spectrum shortage problem. Nevertheless, the above-named design algorithms that need to drive their requirements have also presented themselves as challenges in terms of increased computational complexity with an increased number of devices. For example, the IoT in its current form still provides connectivity to various servers and systems through a centralized server-client model [83], and based on current reports, this is quite efficient for now, given that it is still in its infancy. Unfortunately, the current centralized approach may not be compatible with future distributed deployment scenarios.

The primary issue will arise when the wireless environment becomes much more distributed than what it currently is. As the wireless network becomes more distributed, except for spectrum scarcity problems, enormous amounts of measurements will need to be conducted by the CR learning module, and the decisions that need to be made have already brought about research challenges [84]. The decision module of CR devices will be faced with the vexing problem of creating sufficient capacity and energy-efficient operations, combined with low latency for heterogeneous spectrum use cases. With these challenges comes the issue of network sustainability, which has its roots in network energy consumption. The reduction in energy consumption by optimizing energy efficiency is a pervasive problem in CRNs. Its solution constitutes a vital step in guaranteeing network sustainability and minimizing the cost of network operation and expenditure. This means that the versatility of current solution algorithms needs to be improved, since in their current state they have a few limitations going into the 5G era and beyond.

The solution to these challenges now lies in how well the staggering resource requirements of future use cases and energy efficiency are handled in the realm of spectrum shortage issues. Since in spectrum management issues users take autonomous actions when operating spectrum access, the most convenient model for such network dynamics is from the game theoretic perspective. Taking nothing away from game theory itself, it is a very efficient mathematical programming technique, except that the system is required to perform the same computations even if the problem is recurring. The direct and immediate consequence is a wastage of both time and computational resources, which necessitates finding new ways of reusing previous solutions. One way of doing this is to store the actions that previously led to better system performance and just reinforce them when the need arises, which is a concept used in Chapter **5** of this thesis.

2.3.7 CRs in Future Generations - A Need for Improved Approaches

Since the IoT has emerged as a concept where things have unique identities, physical interfaces and virtual personalities and this paradigm rapidly spread over the community of academia and industry and received close attention as among the most pertinent research topics [85]. As the demand for performing intelligent tasks for mobile devices and network infrastructure with low latency is ever increasing, the requirements of intensive computation and large storage size impede the application of intelligent computational techniques in wireless networking. Deep architectures, however, offer promising solutions by integrating the computational power and storage of edge-processing nodes (i.e., BSs). Thus, in this section, the researcher envisions a joint signal processing and DL spectrum management scheme to improve the efficiency of spectrum management while minimizing the sum of computing power consumption.

Spectrum sensing techniques in CRNs have been carefully designed and have shown great success from their infancy up to 5G, in recovering idle spectrum holes for opportunistic transmissions and exhibiting unique formulations in CSI. However, more efforts are still required in order to break the bottleneck of spectrum sensing errors and energy-efficient spectrum detection [86]. Extensive work has been carried out to reduce spectrum sensing errors and reduce spectrum information feedback overhead by utilizing the spatial and temporal correlations of CSI using compressive sensing (CS) techniques [87]. However, CS algorithms usually face challenges since they require the signals or signal data to be sparse, which is not always the case in CRNs. Real-world wireless data are not exactly sparse such that the convergence speeds of the existing signal recovery algorithms is relatively slow and this has limited the practical applications of CS techniques.

In terms of RA, the objective of intelligent resource management is the allocation of available resources to maximize several network performance metrics. This means that to optimize network throughput, the transmit powers, the frequency bands, the computing power as well as the memory space can be scheduled among the network nodes based on the arising demands, the conditions of the propagation channel. The optimization of network throughput should ensure that all users experience the guaranteed QoS in terms of the required communication latency in an energy efficient manner. The issue of energy efficiency is the most important here, since of all the parameters to be optimized, it is the common denominator. Here, the packet drop rate and flow throughput are the main considerations since one desires to deliver as many packets within their deadlines at the lowest energy possible. Most of the problems faced by traditional techniques in efficiently solving future mobile and wireless network problems are due to model and algorithmic deficits, as discussed below.

2.3.7.1 Model Deficit Problems

In mobile and wireless communication systems, model deficit problems usually occur when wireless systems operate on channels with no well-established mathematical models such as in molecular communications [88]. Even adequate systems in molecular communications are still unable to give accurate models for controlled propagation of carrier molecules. These might include steps from the encoding and the decoding of signal information onto information molecules, to the transmission and reception systems for carrier or information molecules. These are still not available in practical scenarios such as drug delivery systems [89]. Most SP algorithms are often optimized for mathematically convenient models and such models are often linear, while real systems have many imperfections and non-linearities [90]. What exacerbates the issue of mathematically convenient models is the fact that the current generation of communication systems exhibits higher energy cost relative to computation [91]. As a result, model deficit problems make the design of energy-aware algorithms with carefully managed communication strategy a crucial and challenging component of future mobile and wireless communication systems.

Central to most contemporary design and optimization approaches for wireless network applications is the manual customization of the network protocol stack, such that the application programming interfaces (APIs) are consistent with manual customization across nearly all platforms that provide an internet protocol (IP) stack [92]. Manual customization of the network protocol stack, in spite of having so many remarkable and laudable benefits, comes with rigidly defined models, which cannot deal with the biggest challenges of the IoT that need to be taken very seriously. Surprisingly enough, concerns about model deficit problems have emerged from within telecommunications, but have not yet been discussed and addressed. For example, one critical challenge in model deficit emerges when dealing with dense heterogeneous networks, where modeling network node locations is typically difficult to predict deterministically. In the current state-of-the-art methods involving the modeling of network nodes using stochastic geometry, network nodes are treated as random variables with given spatial distributions [93]. However, the primary concern with employing stochastic geometry is that it focusses on the analysis of traditional, non-energy-efficient performance measures such as densification.

The lack of scientific ways to incorporate the trade-off between the impact of densification in energy efficiency and gain saturation as the density of infrastructure nodes increases puts the problem of energy efficiency under model deficit. An ML-based communications system does not require such a rigidly defined model for representation and transformation of information and could be optimized in an end-to-end manner for a real system with harsh realistic effects. For example, at first glance, the Poisson kriging approach [94] presents an attractive alternative to increasingly popular models that use Bayesian spatial models for the following reasons: (i) ease of implementation with less computational complexity, and (ii) avoids limitations that come with the conditional auto-regressive models used in Bayesian models by accounting for the size and shape of geographical nodes.

2.3.7.2 Algorithmic Deficit Problems

Algorithmic deficit problems are more common in wireless communication systems and for the properties and demands of future mobile and wireless networking, it makes the computational requirements of traditional optimization techniques unbearable. This is often the case with wireless communication channels with strong non-linearities, often encountered in optimacal communications and in multi-user networks. For example, the analyses of optimal decoders operating over a number of well-established channel models tend to be computationally complex. The complexity exponents that describe the minimum known complexity that can provably achieve a gap to maximum likelihood performance usualy vanishes in the high SNR limit [95]. Also, given the network energy constraints with end users desiring long-lived systems, the current network-centric philosophy may be acceptable for simple data gathering and querying applications [96]. But this design methodology may not be feasible for large-scale future mobile and wireless networks because of the large number of protocols that have to optimize the design while simultaneously ensuring smooth network operation by concurrently executing application-level functionalities such as medium access control and time synchronization.

However, on the brighter side, it is well known that NNs with their universal function approximation capabilities have remarkable capacity for algorithmic learning [19], [20], and recent research works have demonstrated remarkable capacity in this aspect, a construct that has been shown to be Turing-complete [21]. The execution of NN algorithms can be highly parallelized through the use of data and computationally distributed concurrent architectures, and has shown to work well with small data types conducive to efficient wide single-instruction multiple data operations. There is a lot of hope that such learned algorithms can be executed significantly faster at lower energy budgets than their manually programmed counterparts. On the other hand, recent advancements in computer processing units as well as in distributed data storage make the use of DL strategies even more practical in achieving low-complexity joint-layer strategies. However, research in this area is still in its infancy and there are several open challenges that need to be addressed on methods to integrate DL strategies into the current way wireless communication networks are be operated.

Therefore, to address these problems, the applicability of ML techniques has been explored in the wireless communications domain. ML techniques owe their popularity to providing a general framework for solving complex problems where the model phenomena to be learned seems too complex to be derived and dynamic to summarize in simple mathematical terms [98]. In order to address both model and algorithm deficit problems, several ML techniques have been widely recognized for solving a range of classification and regression problems where no well-defined mathematical model exists. Owing to the recent advances in computing power and the ability to collect and store massive amounts of data, ML strategies have found their way into the wireless networking domain. In addition, CRN problems are often formulated in terms of classification and prediction, ML strategies can provide elegant and more practical solutions. In this context, the application of ML to CRNs seems almost natural and presents a clear motivation. Therefore, the intention of this review is to elicit more research on the integration of DL strategies into CRNs to solve the key challenges associated with the modern mobile and wireless networking systems.

2.4 INTRODUCTION TO MACHINE LEARNING

The objective of this section is to build a solid mathematical foundation leading up to other ML strategies. At the end of this section, the application of supervised learning strategies in mobile and wireless networks is reviewed. In ML, model assumptions are essential prior to building a model that will be used for prediction. If assumptions are met, then the model can accurately reflect the data and is likely result in accurate predictions. Each model has different assumptions that must be met, so checking assumptions is important, both in choosing a model and in verifying that it is the appropriate model to use. An illustrative example of the basic ML objective is shown in Figure 2.2 below.

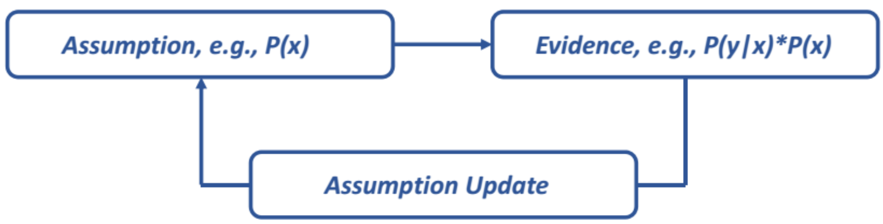

Figure 2.2. Illustrative Example of the Basic Principle of Machine Learning.

The objective of the model in Figure 2.2 above is to predict how the variable y is influenced by changes in the variable x. In statistics, this is often described as the problem of approximating a target function that maps inputs to outputs, which is characterized as searching through and evaluating a candidate hypothesis from hypotheses space.

Problems and tasks in communication systems are easily dealt with using traditional signal processing methods, so, the question is how ML can help to improve the results. The answer to this question is that ML methods can help in cases of model or algorithmic deficit. ML strategies are self-adaptive algorithms capable of yielding better analysis of patterns with experience or with new data and are one of the most common AI techniques used for processing Big Data. ML is usually split into two subcategories, namely supervised and unsupervised learning. However, some problems are solved using approaches such as RL and DL. RL, which in its most fundamental form is viewed as a third and separate subcategory of ML, combines techniques borrowed from behavioral psychology and game theory; thus representative algorithms of this nature will be denoted as such [99].

However, many advanced RL algorithms are usually referred to as "deep RL" (DRL) since they incorporate strategies from supervised and unsupervised learning, as well as DL. For example, a DRL algorithm would use a DNN in an RL framework. The DL strategy, which typically refers to the use of DNNs, is not necessarily viewed as a separate subcategory of ML [100], but as a means of achieving the ends associated with each of the three ML subcategories. The focus of this section is primarily on supervised learning, which is the typical approach in DL. However, a great deal of RL strategies are also considered, taking into account its integration with DL tools which leads directly to the recently introduced deep architecture, i.e., DRL.

2.4.0.1 Activation Functions in Machine Learning

ML algorithms are powerful in learning complex and complicated tasks - all thanks to the use of activation functions. In a neural node, the activation function defines the output of that node or neuron given given input or of a set of inputs [101]. These functions are used in NNs for the computation of weighted sums of inputs and biases, which are used to decide if a neuron can be fired or not. It does so by manipulating the presented data through some gradient processing such as gradient descent to produce an output. In NNs, when performing regression tasks, a linear activation function is usually used at the output layer, i.e., estimation of a real-valued vector. It is thus necessary to introduce some standard activation functions used by ML methods. These are given in Table **2.2** below [102].

Table 2.2. List of Activation Functions Used in Machine Learning.

Name	**Application**	**Range**
Linear	$f(x) = x$	$(-\infty, \infty)$
Rectified Linear Unit (ReLU)	$f(x) = \max(0, x)$	$(0, \infty)$
Sigmoid	$f(x) = \frac{1}{1+e^{-x_i}}$	$(0, 1)$
Hyperbolic tangent (tanh)	$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$(-1, 1)$
Softmax	$f(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$	$(0, 1)$

As seen from Table 2.2 above, activation functions can either be linear or non-linear depending on the representative function. These are used to control the outputs of NNs when applied in different domains, e.g., object recognition, segmentation, scene understanding and description, machine translation, text-to-speech systems, cancer detection systems, fingerprint detection, weather forecast, self-driven cars, etc. Thus, the proper choice of an activation function results in accurate ML computation. For linear models, the mapping of inputs to outputs using activation functions is performed in the hidden layers before the outputs stage. For example, the transformation of the vector **x** is given as follows:

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b, \tag{2.1}$$

where the term **x** denotes the inputs vector to the node, **w** represents the weight weight vector, and b is the bias vector. Non-linear activation functions are transfer functions required to convert linear inputs to non-linear outputs and are applied to the outputs of the linear models to produce a transformed non-linear output that is ready for further processing. Here, the neural node is represented by a non-linear activation function, $\Phi(\cdot)$, that takes the sum of the weighted inputs as illustrated in Figure 2.3 below.

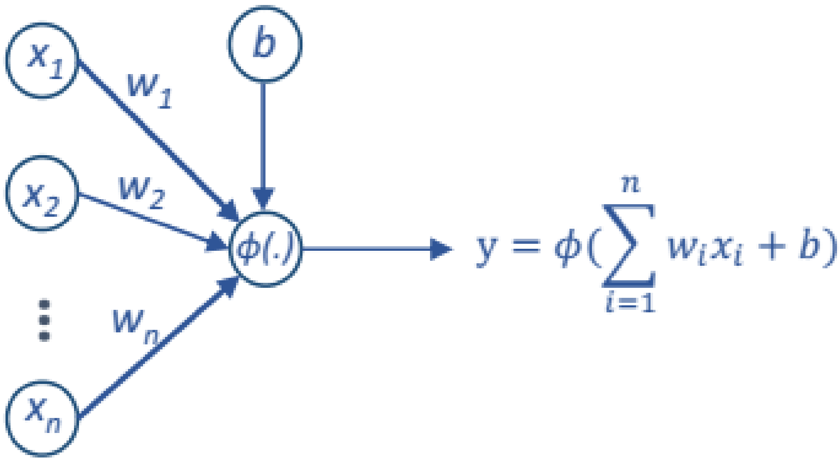

Figure 2.3. Single Neuron Computation with a Non-linear Activation Function.

The activation function of a single computational unit like the one shown in Figure 2.3 is typically calculated using the dot product of an edge weight vector **w** with an input vector **x**, plus a scalar bias b, as shown in (2.2). Thus, the corresponding output of each neural node can be mathematically expressed as follows:

$$y(\mathbf{x}) = \Phi\left(\sum_{i=1}^{n} w_i x_i + b\right) = \mathbf{w}\cdot\mathbf{x} + b, \tag{2.2}$$

where x_i represents the inputs, w_i denotes the weights and b represents the bias value of the neuron [149]. Detailed analyses and derivations of the different activation functions can be found in [102].

2.4.0.2 Loss Functions in Machine Learning

In ML, a loss function or a cost function or an error function is used to calculate the performance of a learning model, which is usually its ability to estimate the relationship between the actual value or the input x and the predicted value or output y. Thus, loss functions are typically expressed as a difference between output and the input values, which are estimated by iteratively running the ML model in order to compare predictions against the ground truth, i.e., the known values of y. ML techniques, therefore, learn by means of loss functions, which evaluate how well a specific algorithm models the given data [103]. However, if the predictions deviate too much from actual results, the loss function would give a large number, but with the help of optimization functions such as gradient methods, the loss function learns to reduce the error. Several examples of loss functions used in ML are given in Table **2.3** below.

Table 2.3. List of Loss Functions.

Name	**Loss** $\ell(x,y)$
Mean squared error (MSE)	$\|\|\mathbf{x}-\mathbf{y}\|\|_2^2$
Root mean squared error (RMSE)	$\max(0,x_i)$
Categorical cross-entropy	$-\sum_j x_j \log(y_j)$
Hyperbolic tangent (tanh), Softmax	(see **Table 2.2**)

In a ML model, the objective is usually to find parameters and weights or structures that minimize a cost function. In this case, the model is said to be learning to minimize a cost function using gradient descent methods, which are efficient optimization algorithms that attempt to find either local or global minima of a given function.

2.4.1 Supervised Learning

Supervised learning is a technique in which both the input data x and the desired output data y are provided during training. Both the input and the output data are generally labelled for purposes of classification in order to provide a learning basis for future data processing [104]. Thus, in supervised learning, a data set always contain examples together with their respective labels. Such an operation is typically be denoted as input data x and its respective label y together, such that one has a training example $(x,y) \in D$, where D represents the data set. During a learning task where the label y must be predicted from its example x, the supervised learning algorithm estimates the underlying conditional distribution $p(y|x)$. In problems of this nature, the objective is usually obtaining a model of the conditional distribution whose parameters are denoted as θ. Assuming that one has a set of independent and identically distributed (i.i.d.) data $D = \{x_1, x_2, \cdots, x_n\}$, $D_i, i \in n$, that has been drawn from the data generating distribution $p_{\text{data}}(x)$, the maximum likelihood estimator of parameters, θ, of a model of the data generating distribution is given as follows [105]:

$$\theta_{ML} = \arg\max_{\theta} p_{\text{model}}(D;\theta) = \arg\max_{\theta} \prod_{i=1}^{n} p_{\text{model}}(x_i;\theta), \tag{2.3}$$

where ML stands for maximum likelihood, and the term p_{model} represents the function space of probability distributions over θ. When taking a logarithm operation on both sides in (2.3), one obtains a more computationally appealing representation as follows:

$$\theta_{ML} = \arg\max_{\theta} \sum_{i=0}^{n} \log p_{\text{model}}(x_i;\theta). \tag{2.4}$$

When the right-hand side of (2.4) is divided by n, the expected value of the log-probability of this model can be obtained over the empirical data generating distribution as follows:

$$\theta_{ML} = \arg\max_{\theta} \mathbb{E}_{x\sim\hat{p}_{data}} \log p_{\text{model}}(x_i;\theta). \tag{2.5}$$

An alternative method of obtaining the above maximum likelihood estimation would be by minimizing the Kullback-Leibler (KL) divergence between both the data generating and the model distributions as follows:

$$D_{KL}(\hat{p}_{\text{data}}||p_{\text{model}}) = \mathbb{E}_{x\sim\hat{p}_{\text{data}}}\left[\log \hat{p}_{\text{data}}(x) - \log p_{\text{model}}(x)\right]. \tag{2.6}$$

However, because the data generating distribution is usually not a function of the model distribution, the same minimization problem can be solved through the minimization of

$$-\mathbb{E}_{x\sim\hat{p}_{\text{data}}} \log p_{\text{model}}(x). \tag{2.7}$$

This operation is thus equivalent to the maximization problem stated in (2.3). Then (2.7) is called the negative log-likelihood of the model distribution, when minimized results in the minimization of the cross-entropy between the two distributions [106]. In supervised learning, this is very important considering that the objective of the model is producing correct labels for data examples drawn from the data generating distribution whose model has never been seen a priori. Therefore, for a complete representation of (2.4), the maximum likelihood estimation of the conditional distribution, which is what is required in order to predict a label y_i from a given independent and identically distributed (i.i.d.) example x_i, considered and is given as follows:

$$\theta_{ML} = \arg\max_{\theta} \sum_{i=0}^{n} \log p_{\text{model}}(y_i|x_i;\theta). \tag{2.8}$$

However, sometimes regularization of the parameters of the model is desirable when a better generalization of the model required. Thus, building on the ML perspective of loss functions, it can be indicated that by adding a regularization function, a prior is induced over the model parameters of the optimization function, which subsequently changes the maximum likelihood estimation to a maximum a posteriori (MAP) point estimate, resulting in the following optimization problem:

$$\theta_{MAP} = \arg\max_{\theta} p(\theta|D) = \arg\max_{\theta} \log p(D|\theta) + \log p(\theta). \tag{2.9}$$

The optimization problem in (2.9) has been obtained through the use of Bayes-Turing factors of product axioms [107], the properties of logarithms, and also the fact that the optimization problem does not depend on the data generating distribution, hence an MAP was obtained. Some popular applications of supervised learning models with associated learning algorithms that analyse data used for classification are discussed in the following subsections.

2.4.1.1 Support Vector Machines

Support vector machines (SVMs) were initially introduced to the ML community to perform binary classification tasks and were then extended to perform regression as well as multi-class classification tasks [88]. In order to explain SVMs, one should begin by defining them in the context of linear regression, which is among the most prevalent linear classifiers in the ML and statistical communities, typically formulated as follows:

$$y_i = \mathbf{w}^T \mathbf{x}_i + w_0, \tag{2.10}$$

A common approach in which (2.10) is solved is through the vectorization of both the output and input variables and solve the normal equations, which results in a closed-form solution for the MMSE. However, (2.10) can also be adapted for classification tasks is through the well-known logistic regression function, given as follows:

$$p(y=1|x;\theta) = \sigma(\theta^T \mathbf{x}), \tag{2.11}$$

where the term $\sigma(x)$ denotes the activation function. The attractive property of logistic regression functions is that they have well-defined probabilistic interpretations that can also be viewed as the maximum likelihood of the conditional distribution $p(y|x)$ [108]. On the other hand, SVMs with kernels perform similar to logistic regression models, the only difference is in the loss functions used. An SVM model minimizes the hinge loss, while logistic regression model minimizes logistic loss and there are two implementations of SVM algorithms, namely soft margin and hard margin. Soft margin SVM uses the hinge loss function $\max\{0, 1-y_i(w^T x_i + b\}$, while the actual objective function it minimizes is as follows:

$$\min_w \lambda ||w||^2 + \sum_i \max\{0, 1-y_i(w^T x_i + b)\}, \tag{2.12}$$

where $||w||^2$ is the regularizer. However, in hard margin SVM, the whole objective function is just $\lambda ||w||^2$. On the other hand, logistic regression produces probabilistic values as opposed to SVM, which produces either a 1 or 0 [109]. The logistic loss function is thus defined as follows:

$$\min_w \lambda ||w||^2 + \sum_i \log(1+\exp(1-y_i w^T x_i + b)), \tag{2.13}$$

which exhibits a similar convergence rate to the hinge loss function discussed above. Since (2.13) is a continuous function, it can be solved using the popular gradient descent algorithm. The hypothesis of a linear regression is as follows:

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \cdots + \theta_j x_j = \theta^T x = \frac{1}{1+e^{(-z)}} = \frac{1}{1+e^{(-\theta^T x)}} \tag{2.14}$$

where

$$h_\theta = \left\{ \begin{array}{ll} 1, & \text{if} \quad z \geq 0 \\ 0, & \text{otherwise} \end{array} \right\}. \tag{2.15}$$

However, if the logistic regression fails to perform classification, one has good reason to believe that their data may not be linearly separable. Another model to try out is an SVM with a non-linear kernel such as a radial basis function (RBF). RBFs are a means of approximating multivariate functions through a linear combination of terms based on a single univariate function, called the RBF [110]. The radialization is done so that it can be used in multiple dimensions. A number of kernels that can be applied in SVM models are the linear, polynomial, radial basis function or sigmoid, and are summarized in Table **2.4** below.

Table 2.4. List of kernel Functions Used in Support Vector Machines.

Name	kernel function	Name
$k(x_i, x_j) =$	$x_i^T x_j$	Linear
	$(x_i^T x_j + 1)^d$	d-th degree Polynomial
	$\exp(-\gamma \lvert\lvert x - I - x_j \rvert\rvert^2)$	Radial basis Function
	$\tanh(\gamma x_i \cdot x_j + C)$	Sigmoid

Here, the formulation $k(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$ basically means that kernelization is a dot product of the input data points that are mapped onto a higher dimensional feature space through the transformation Φ. Thus, the kernelization of SVMs lead to the realization of a kernel SVM. Another algorithm that offers lightweight solutions to classification problems is the decision tree [111], which is applied as a surrogate in Chapter **4** of this thesis.

2.4.2 Application of Supervised Learning in Communication Systems

When applying ML techniques in wireless communications, the very first condition to be considered is the type of problem concerned. Generally speaking, problems addressed in this category of ML are basically regression, classification and clustering problems. In regression problems, the task of

ML is to predict continuous values for the current inputs and identify the class to which the current input belongs in classification problems. For example, content caching may need to acquire content request probabilities of each user. Caching can therefore be seen as a regression problem that takes the user profile as input and the content request probabilities as output, while BS clustering problems can naturally be handled by K-means clustering algorithms.

2.4.2.1 Physical Layer

The focus of this part is on the physical layer, where concentration is on both the transmitter and receiver sides of the communication systems channel, where in the receiver side, the application of channel detection and decoding are reviewed, while on the transmitter side mostly applications that tackle algorithmic deficit problems are considered. On the receiver side, multi-class problems in which an input signal x from the received baseband signal needs to be classified correctly into the transmitted message. In channel decoding tasks, unless wireless channel estimation is incorporated into the learning process, the decoding task at the receiver becomes difficult because of fast-varying channels. Therefore, it is generally beneficial to use data-aided methods in incorporating channel domain knowledge in the hypothesis class definition.

In [112], channel decoding examples and the knowledge of near-optimal message-passing methods for the decoding of sparse graphical codes were used in setting up a parameterized model. These examples borrow the structure of message-passing are trained to decode more general codes. Other tasks that could potentially benefit from ML strategies at the receiver include modulation classification, which is a classification problem that is justified by the complexity of optimal solutions (i.e., algorithm deficit) [113]. Most applications for tackling algorithm deficit problems on the transmitter side are related to the complexity of the non-convex programs that typically underlie power control optimization algorithms. In a contribution in [114], a training set is obtained by running a non-convex solver to produce an optimized output vector for given input channels. However, this approach could not optimize the sum rate directly, which is the performance metric of interest, but instead relied on the assumption that similar inputs - the channel coefficients - generally yield similar optimal solutions, which is the power allocation vector. In duplex transceiver applications, reference is made to [115], where self-interference is cancelled in order to overcome the lack of a well-established transmitter-receiver chain model of non-linearities.

2.4.2.2 Link and Medium Access Control Layers

In the link and medium access control (MAC) layers, some applications of ML strategies usually deal with the problems that lack suscinct mathematical models for complex access protocols and wireless communication environments. In [116] the authors proposed a mechanism for predicting if a channel decoder would be successful in giving accurate outputs during the first few iterations on the iterative decoding process. The success of such a prediction would mean that the predictor could be used to request early re-transmissions to reduce reansmission latency using automatic retransmission request (ARQ). However, at the MAC layer, data-aided approaches can be used instead for the task of predicting the availability of spectrum in CRNs in the presence of interfering incumbent devices with complex activation patterns for CR resource management applications [117].

2.4.2.3 Network and Application Layers

In the network and application layers, well-known applications that are well-suited for ML strategies are the caching of popular contents which is used for reducing network latency and congestion [118]. Traditionally, caching takes place within the core network segment, but has since been moved to the network edge devices (i.e., BSs) because of the advantages of directly catering for the preference of the local population of users. However, this approach usually suffers from reduced hit rate, owing to storage capacity constraints, ML strategies could be used to optimize the selection of content that can be stored at the edge, which can be formulated as a classification problem. This application could greatly benefit from data-driven approaches in order to adapt to the specific features of the local traffic. A summary of the application of supervised learning in mobile and wireless communication systems is tabulated in Table **2.5** below.

2.4.3 Unsupervised Learning

Unsupervised learning is an ML technique where supervision is not required; instead one needs to allow the model to work on its own to discover the required information from unlabelled data sets. This ML technique allows one to perform complex processing tasks, unlike to its supervised counterpart, which is very predictable. As opposed to supervised learning, in unsupervised learning, the training data set consists of unlabelled inputs, which means that the inputs are assigned without

Table 2.5. Comparison of Supervised Learning Approaches in the Protocol Stack.

Layer	**Application**	**Technique**	**Algorithm**	**Refs**
Physical Layer	Channel decoding	Neural BP	SGD	[112]
	Modulation classification	SVMs	Maximum Likelihood Estimation	[113]
	Power control optimization	Supervised DNN	WMMSE	[114]
	Interference cancellation	Supervised NNs	NN Non-linear Canceller	[115]
Link and MAC Layers	Prediction - channel decoder	Supervised AE	Binary Predictor	[116]
	Prediction - spectrum sensing	MLP	Batch BP	[117]
Network and Application Layers	Proactive caching	Echo State	Sublinear Algorithm	[118]

any assigned labels. For instance, if the inputs are two-dimensional, there is no indication provided by the data about the corresponding desired output. Thus, the objective of unsupervised learning is to discover properties of the mechanism generating the data and the most prevalent unsupervised learning problems are associated to clustering.

In clustering, the objective is usually to group input data points that are close to one another, hence assigning cluster indices (i.e., labels) to each input point. Thus, clustering is an unsupervised counterpart of classification and its objective is to categorize input samples $\{\mathbf{x}_i\}_{i=1}^{n}$ into clusters $1, 2, \cdots, c$ without any supervision $\{y_i\}_{i=1}^{n}$. In summary, the data set contains only input features, based on the input features; the algorithm must be able to extrapolate on the statistical structure of the input data set that is required to carry out the desired task [119]. One of the most prevalent applications of unsupervised learning is in outlier/anomaly detection which is aimed at finding irregular samples in a given data set.

It is important to point out that while supervised and unsupervised learning bear some resemblance to the classical problems from decision and estimation theory. However, there is a fundamental difference, classical decision and estimation theory that assumes the existence of the knowledge of the probability distributions of the output vector given the input $p(\mathbf{y}|\mathbf{x})$ and that of the input vector $p(\mathbf{x})$. But in this supervised learning technique, the maximum likelihood operation does not require to make this assumption. It is able to operate based only on some realizations of the underlying distributions, even though these distributions are themselves not known. This procedure is similar to that which is illustrated in Figure 2.2 above.

2.5 REINFORCEMENT LEARNING

This subsection defines the basic concept of the RL strategy from its roots in behavioral psychology and game theory; the different variations and add-ons, i.e., Q-learning and MDPs are also introduced. In order to understand the recent advancements in RL, i.e., how DRL and DQNs operate, one needs to be familiar with RL, Q-learning and MDPs. So, the objective of introducing the different variations is to achieve a smooth build-up to DRL and the recent advancements in the field, i.e., DQNs. In RL, unlike in unsupervised learning, there exist some kind of supervision; however, this kind of supervision comes in the form of "supervision in reinforcement" [120]. "Supervision in reinforcement" refers to a type of supervision that does not emanate from the specified set of desired outputs for every input in the data as in supervised learning. An RL algorithm instead receives feedback from the environment (i.e., the wireless network), which only comes after selecting output for given inputs or observations. This kind of feedback thus indicates the degree to which the output, known as action, fulfills the goals of the RL agent (i.e., BS or user equipment). Therefore, the task of a RL algorithm is to train an agent that interacts with its environment how to behave when it encounters different scenarios, known as states by performing certain actions which eventually result in rewards (i.e., throughput). The agent has the sole purpose of maximizing its total reward and "reinforce" the actions that lead to better rewards using a certain strategy or policy. This basic procedure is illustrated in Figure 2.4 below.

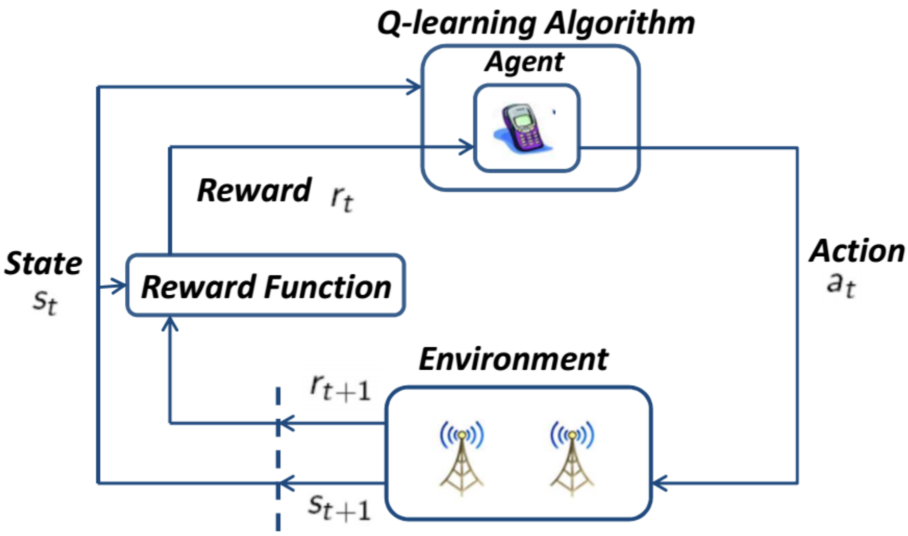

Figure 2.4. The Basic Operation Principle of Reinforcement Learning.

2.5.1 Reinforcement Learning with Q-learning

Q-learning is an online RL algorithm that determines optimal policies or strategies that the agent should follow without any detailed modeling of the system environment. Thus, the basic operation principle of Q-learning is that, if an agent receives environmental rewards through taking a certain action, the number of times that the agent takes that action will increase in subsequent actions. This algorithm is applicable to sequential decision-making problems in which an agent interacts with its environment by sequentially taking actions - the outputs - on the basis of its observations - its inputs - while receiving feedback on each selected action [121]. For example, at a time t, the agent is in state s_t selects an action a_t from a set of possible actions and executes it, then arrives at a new state s_{t+1}, and receives a reward r_t. This is so that this can be expressed as the tuple (s_t, a_t, r_t, s_{t+1}). This technique can thus easily be understood using the concepts defined below.

- **Environment**: In RL, an environment is defined as the space through which an agent makes its moves after taking the current state s_t and action a_t as input, and returns the reward r_t as the output and then its next state s_{t+1}. For example, in wireless communications the agent is usually the algorithm that operates in the BS or the user equipment, depending on the application perspective (i.e., downlink or uplink). In this case, the environment could be the laws of physics that govern communication through electromagnetic waves, and the protocols of the wireless environment that process the actions and determine the consequences (i.e., rewards or penalties). So, the environment can be summarized as the transfer function that transforms an action a_t taken in the current state s_t into the next state s_{t+1} and a reward r_t. However, the function of the environment is not always known and it is just a "black box" where only the inputs and outputs are seen.

- **Agent**: The agent is defined as a function that transforms the new state and reward into the next action. In short, an agent takes actions. One can consider for a moment, a BS allocating resources to mobile users based on their QoS requirements. In this case, the agent is the algorithm that makes the BS to take an action while allocating resources to the different mobile users. As opposed to the environment, the function of the agent can be known.
- **Action**: In the above BS RA example, $a_t \in \mathcal{A}$ denotes a possible action a_t in the set of all possible actions $\mathcal{A}$ that the agent can take at a given state s_t. In mobile wireless technology, actions would include, among others, call admission/rejection/termination, cell handover, channel hand-off, etc.
- **State**: A state, denoted $s_t \in \mathcal{S}$, which is immediate situation that the agent finds itself in. For example, in wireless communications, an instantaneous configuration that the BS or user equipment can find itself in is predominantly the channel conditions which then significantly affect their rewards (i.e., throughput in the network).
- **Reward**: A reward, denoted as $r_t \in \mathcal{R}$, is the measure of the success of an agents' actions. For example, if a user executes certain transmission action in certain channel conditions, it receives a reward in the form of achievable throughput. However, it is important to note that the rewards which effectively evaluate the agent's actions, may be either immediate or delayed.
- **Discount factor**: The discount factor, usually denoted by $\gamma^t \in [0,1]$, is factor that dampens the effect of the rewards on a given choice of action. This is done by multiplying the future rewards as discovered by the agent to make them worth less than immediate rewards. Therefore, the discount factor enforces some kind of short-term satisfaction in the agent.
- **Policy**: The policy, denoted by π, represents the function that maps states to actions that promise better rewards. It is a strategy employed by the agent employs in order to determine the next action based on the current state. A sequence of policies is given as $\pi = (\pi_1, \pi_2, \cdots, \pi_t)$.
- **Value**: The value, denoted by V, is the expected long-term discounted return. Therefore, the term $V^{\pi}(s)$ represents the expected long-term return of the current state s_t under policy π.

- **Q-value**: The Q-value or the action-value function, where Q is almost similar to V except for the fact that it takes an extra parameter in the form of the current action a_t. Then, $Q^{\pi}(s,a)$ denotes the long-term return of state s_t, after taking action a_t by following policy π. Q is thus the function that maps the state-action pairs to rewards.

Therefore, considering $J(\pi)$ as the performance measure, an example of an objective function in RL is given as follows:

$$J(\pi) = \mathbb{E}\left\{\sum_{t=0}^{\infty} \gamma^t r_t(s_t,a_t)|s_0,\pi\right\}, \tag{2.16}$$

where $\gamma \in [0,1]$, s_0 is the start-state and one is summing reward function r_t, which is the reward function for the state s_t and action a_t over t time steps. The time-step term $t \in \{0,1,2,\cdots\}$ refers to the moment at which an action is taken. System observations can be performed either over an infinite horizon, i.e., $t \in \{0,1,2,\cdots,\infty\}$ or over a finite horizon where $t \in \{0,1,2,\cdots,T\}$. The time duration of observing a system is application-dependent. The finite horizon is an intuitive and fundamental formalism for decision-theoretic planning when considering discrete-time problems, typical of telecommunication problems.

2.5.2 Reinforcement Learning and Markov Decision Processes

MDPs are a kind of controlled Markov chains, which constitute the basic framework for dynamically controlling systems that evolve stochastically. The formal definition of the MDPs framework is usually accompanied by the definition of value functions V and policies π. Thus, MDPs are actually a generalization of non-controlled Markov chains, and many useful properties of Markov chains carry over to controlled Markov chains. The key Markovian property in MDPs is that, conditioned on the state, s_t, and action, a_t, at some present time t, the previous state s_{t-1}, and the next one s_{t+1} are independent of one another. Many stochastic planning problems in RL such as robotic control can be successfully modeled using MDPs. Thus, MDPs are the de facto standard formalism for RL decision-making [122]. An alternate definition for (2.16) is given as follows:

$$Q^{\pi}(s,a) = \mathbb{E}\left\{\sum_{k=1}^{\infty} \gamma^{k-1} r_{t+k}|s_t = s, a_t = a, \pi\right\}, \tag{2.17}$$

where the term k denotes the number of steps taken after the start state s_0. In both the definitions, a discounted weighting of states is defined as $d^{\pi}(s)$, which is the discount encountered beginning from s_0, following policy $\pi : d^{\pi}(s) = \sum_{t=0}^{\infty} \gamma^t Pr\{s_t = s|s_0,\pi\}$.

2.5.3 Model-based RL - Controller and System Evolution

RL strategies can be divided into two sub-categories, model-free and model-based methods. The model-free RL strategy has already been discussed in the preceding section. Contrary to model-free RL, in model-based RL the rewards obtained depend on the policy and the system dynamics (i.e., the model) through the definition of a cost function. In Model-based RL, the cost function enables the computation of optimal actions directly from the model. Model-based RL has its roots in control theory, which is why in other literature the system state is denoted as x instead of s and action is denoted as u instead of a. The term normally used for action is control and the controller, by choosing actions at each time unit, has an influence on both the costs and the system evolution. This means that the fundamental difference that comes with MDPs is the use of transition probabilities, which define the system evolution from one state to the next. As it is often the case in control systems, one assumes that the behaviour of the system at any time is determined by the state of the system, as well as the control action. The system then evolves sequentially between different states such that the current state and control action determine the probability of moving from the current state to the next one.

The model-based RL strategy discussed here is applied in Chapter **4** of this thesis to derive control actions for DL. The models usually studied in model-based RL are very special in that not just one, but several objective costs exist such that the controller has to minimize one objective subject to the constraints on the others. As a result, this is a class of MDPs known as the constrained MDPs (CMDPs), which usually arise in situations where the controller has more than one objective to optimize [123], as will be seen in Chapter **5**. In order to make the concept of CMDPs more precise, one must define a tuple $\{\mathcal{S}, \mathcal{A}, \mathcal{P}, c, d\}$ where $\mathcal{S}$, $\mathcal{A}$ and $\mathcal{P}$ have been defined earlier. The term $c : \mathcal{K} \rightarrow \mathcal{R}$ is the immediate cost related to the cost function that needs to be minimized, and $d : \mathcal{K} \rightarrow \mathcal{R}^K$ is a K-dimensional vector of immediate costs, related to K constraints. This kind of control models constitute policies known as "open loop", which are defined as those policies that do not require information on the current system state except for the start state s_0 [124].

2.5.4 Cost Criteria and the Constrained Problem

Cost criteria are frequently used applications of control of MDPs in the Markov decision chains with finite state spaces usually incurring two types of costs (i.e., the operating cost and the holding cost), studied in [127]. Other generalizations of Markov decision chains are the expected average and the discounted costs. Thus, the primary objective of cost-constrained problems is to minimize the expected average operating cost, subject to a constraint on the expected average holding cost, under some existence of an optimal constrained randomized stationary policy for which the two stationary policies differ on at most one state [125]. In such problems, MDPs are used to formulate stochastic optimization problems and solve them online using RL, where the general idea is: for any policy π and an initial distribution $\eth$, the finite horizon cost for a horizon T is defined as follows:

$$C^T(\eth,\pi) = \sum_{t=1}^{T} \mathbb{E}_{\eth}^{\pi} c(s_t,a_t), \tag{2.18}$$

where the term $\mathbb{E}$ is the mathematical expectation operator. A cost that gives less importance to the far future is known as the discounted cost, thus, for a fixed discount factor γ^t, one defines

$$C_{\gamma}^T(\eth,\pi) = (1-\gamma^t)\sum_{t=1}^{T} \alpha^{t-1}\mathbb{E}_{\eth}^{\pi} c(s_t,a_t), \qquad \text{and} \qquad C_{\gamma}(\eth,\pi) = \bar{\lim}_{T\to\infty} C_{\gamma}^T(\eth,\pi). \tag{2.19}$$

The term $\bar{\lim}$ exists because there are infinitely many states and actions, and the term T which represents the finite horizon is put instead

$$C_{\gamma}(\eth,\pi) = (1-\gamma^t)\sum_{t=1}^{T} \gamma^{t-1}\mathbb{E}_{\eth}^{\pi} c(s_t,a_t), \tag{2.20}$$

thus, the expected average cost can be defined as

$$C_{ea}^T(\eth,\pi) = \sum_{t=1}^{T} \mathbb{E}_{\eth}^{\pi} c(s_t,a_t) = \bar{\lim}_{T\to\infty} C_{ea}^T(\eth,\pi). \tag{2.21}$$

Let the term $C(\eth,\pi)$ to denote the above mentioned costs, then, $C(\pi):\mathbf{X}\to\mathbb{R}$ is the vector whose s_t entry is $C(s_t,\pi)$. Then, the cost function related to the immediate costs d are defined as

$$D^{T,k}(\eth,\pi) = \sum_{t=1}^{T} \mathbb{E}_{\eth}^{\pi} d_k(s_t,a_t). \tag{2.22}$$

Therefore, for a K-dimensional vector of real numbers, one can define the constrained control problem as a problem of finding a policy that minimizes $C(\eth,\pi)$, subject to $D(\eth,\pi)\leq V$; where $C(\eth,\pi)$ and $D(\eth,\pi)$ stand for the expected costs defined above. Some typical applications of MDPs in wireless communication networks are described in [126]. This approach has many advantages in wireless communications, because (i) in order for it to determine the joint optimal power control or dynamic power management policies, it does not require a priori knowledge on the traffic arrival and channel statistics; (ii) it is able to exploit partial information about the system such that little information needs to be learned; and (iii) it obviates the need for action exploration, which severely limits the adaptation speed and run-time performance of conventional RL algorithms. Examples treated in this fashion include a packet communication system with reject option and a single-server queue with service rate control [127], where the problem of energy-efficient transmission of delay-sensitive data was considered over fading channels. The objective was to achieve the minimum possible energy consumption, under delay constraints, in the presence of stochastic and unknown traffic and channel conditions. Thus, the stochastic optimization problem was formulated as an MDP and solved online using RL. A joint data admission control-power allocation to maximize the throughput defined as the average admitted rate was studied in [128], where the structural properties of the optimal admission control-power allocation policy are analysed with respect to fading channel, data arrival, and queue length states.

Under no a priori statistical knowledge of system random channel fading and data arrival processes, an online admission control-power allocation algorithm is used. A call control algorithm in which communication links have variable capacity supporting multiple classes of service was also proposed in [129], where the novelties of this approach are: (i) the problem is modelled as a CMDP; and (ii) the CMDP is solved via RL algorithm by using the Lagrangian approach and state aggregation. This approach is capable of controlling class level QoS in terms of both blocking and dropping probabilities. In this thesis, this technique is applied in dynamic and constrained power management in Chapter **5**, Section **5.4.2**.

2.5.4.1 Partially Observable Markov Decision Processes

Partially observable MDPs (POMDPs) are a combination of an MDP to model system dynamics with hidden Markov models that connect unobservant system states to observations. In POMDPs, the agent can perform actions that affect the system and cause the system state to change, with the goal of maximizing a reward that depends on the sequence of system state and the agent's actions. Such a reward is usually a discounted reward, as shown in (2.16). However, the agent cannot directly observe the system state, but at each discrete point in time, it can make observations that depend on the state. The agent uses these observations to form a belief (i.e., a belief state) of the state in which the system is currently in. This belief state is thus expressed as a probability distribution over the states and the solution of the POMDP is a policy prescribing which action is optimal for each belief state.

POMDPs can be used in data aggregation and routing for cooperative multi-hop communications in wireless sensor networks (WSNs) in order to obtain the most energy-efficient sensor alternatives for data exchange and gathering. The decision-making includes different metrics such as transmission delay, energy consumption and expected network congestion. Data query in WSNs serves to disseminate commands from the BS to the intended WSN nodes to retrieve their readings. The authors in [130] proposed a probabilistic scheme for selecting the set of WSN nodes that should respond to user queries. The problem was formulated as a parametric POMDP with average long-term rewards as the optimization metric and the states were defined as data attributes for each sensor node. Sensor nodes would choose actions between answering and not answering to a query.

2.5.4.2 Multi-Agent Markov Decision Processes

Multi-agent MDPs (MMDPs), also referred to as multi-agent planning assumes that there is a set of heterogeneous agents, with each agent having its own set of actions and a given task to be solved. Even though each agent has its own goals, it is assumed that the problem is a fully cooperative one [131], thus the utility of any particular system state is similar for all the agents. Thus, one one hand, MMDPs are often thought of as decision processes rather than games because of the existence of a joint utility function. On the other hand, they are referred to as a general case of stochastic games. In actual fact, they are nothing more than n-person stochastic games in which the pay-off function is the same for all agents.

2.5.4.3 Decentralized Partially Observable Markov Decision Processes and Stochastic Games

Decentralized POMDPs (Dec-POMDPs) are a general extension of the POMDP framework and a specific case of partially observable stochastic games. this is because, as agents are built for ever more complex environments, methods that consider the uncertainty in the system have strong advantages. This uncertainty is common in domains such as autonomous navigation in robotics, inventory management, and in WSNs [132]. Stochastic games are an extension of strategic-form games, which is attributable to the traditional game theory in which the environment changes in response to the players' choices. Stochastic games are generally defined as repeated interactions between several participants in which the underlying state of the environment changes stochastically, and depends on the decisions of the participants [133].

In [134], a random access problem was considered from the stochastic game perspective in a single cell system with one receiver and multiple uplink transmitters sharing a single, slotted, synchronous classical collision channel. Both selfish and cooperative users were allowed to select retransmission strategy-based performance requirements such as throughput, delay and transmission costs. All users were assumed to have packets of the same length and whenever a collision occurred, the users attempted to retransmit their packets in subsequent slots to resolve collision for reliable communication. The state space of the system consisted of the number of packets waiting at each transmit buffer, or the length of time each had been waiting to be transmitted. The reward/cost was derived based on the number of time slots that the transmitted packet waited, on the number of packets that had not been transmitted during that period, and possibly on additional variables. The transition depends on the actions chosen by the players, but it has a stochastic component, which captured the number of new packets that arrived at the various transmitters during every time slot.

In [135] a game theoretical technique for packet forwarding was proposed for relay networks consisting of a source, a relay and a destination node communicating on a common channel. To optimize system performance in terms of throughput, delay and power consumption cost, a stationary Markovian game was utilized. In [136], stochastic games were approached from the queuing perspective where individuals that required service had to choose whether to be served by a private slow service provider, or by a powerful public service provider. In this case, the state space consisted of the current load of the public and private service providers, and the cost was the time required to service a user. This application of stochastic games provided useful insights into achieving wireless Big Data solutions that could help in analysing specific situations and suggesting proper behaviour to the participants. Thus, the complete resource allocation mechanism that utilizes Q-learning and MDPs is illustrated in Figure 2.5 below.

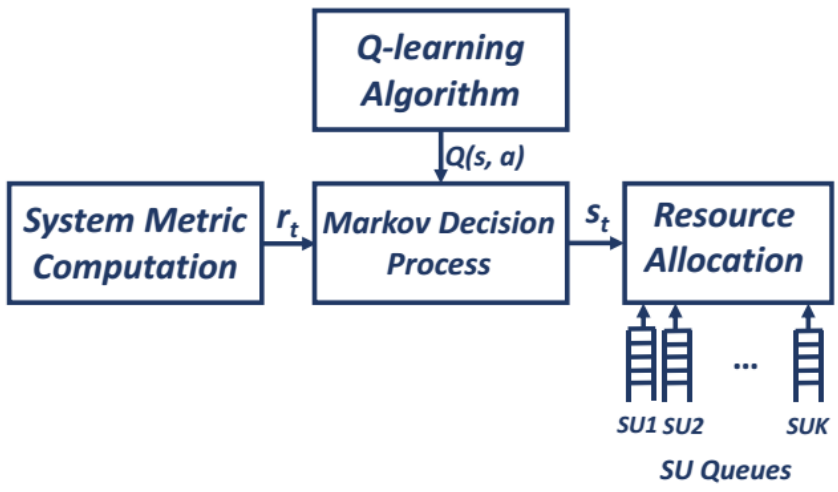

Figure 2.5. Operating Principle of a Resource Allocation Scheme in CRNs.

The model illustrated in Figure 2.5 above is dedicated to the uplink channel perspective where SUs can select parameters contemplated for resource allocation such as transmission powers. Each SU has a queue associated with it and the MDP and Q-learning algorithm has to maximize the average throughput of each connected SU. A generic application of RL in CRNs is illustrated in Figure 2.6 below.

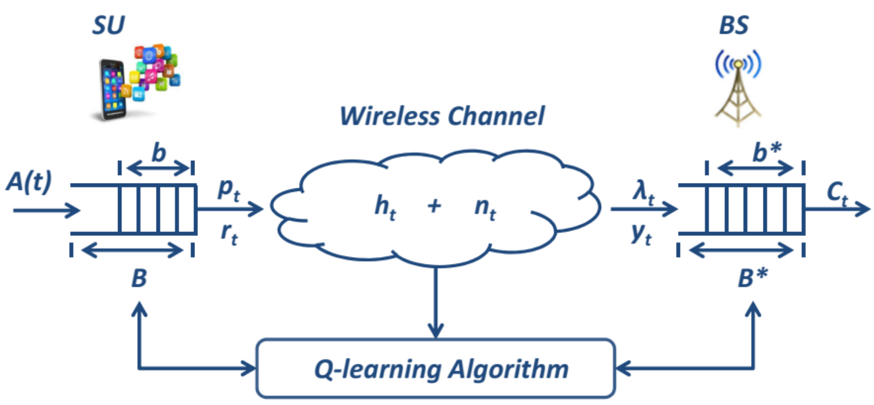

Figure 2.6. Typical Resource allocation Application in CRNs using the RL and Q-learning Concepts.

Figure 2.6 shows the application of RL in performing power management decisions for an SU transmitting through a wireless channel. At time t, the power management decision parameters consist of the allocated transmission power p_t, the buffer state of the sender b, transmission rate r_t, channel state h_t, additive noise n_t, channel output y_t, BS buffer state b^*, and the achievable capacity C_t. The terms B and B^* represent the buffer lengths at the SU and BS, respectively, and feedback and update are done via a Q-learning algorithm.

2.5.5 Deep Reinforcement Learning

DRL uses DL and RL principles to create efficient algorithms that can be applied to problems that previously seemed complicated because of the number of controls that they required, such as robotic control and navigation [137]. In implementing DL architecture (i.e., DNNs) with RL algorithms (i.e., Q-learning, actor critic, etc), a powerful DRL model can be created that is capable of scaling to problems that were previously unsolvable [120]. That is because DRL usually uses raw signals (i.e., sensor or image) as input and can receive the benefit of end-to-end RL as well as that of convolutional NNs (CNNs), as is evident in DQNs for ATARI games [138]. In DRL, NNs are referred to as the agents that learn to map state-action pairs to rewards by using coefficients to approximate the function relating the inputs to the outputs. The learning objective is to find the right coefficients (i.e., weights) by iteratively adjusting them along gradients that promise less error. In wireless communication networks, CNNs can be utilized to recognize an agent's state, i.e., the wireless channel condition, by performing signal recognition. An implementation of resource allocation in CRNs using DRL is illustrated in Figure 2.7 below.

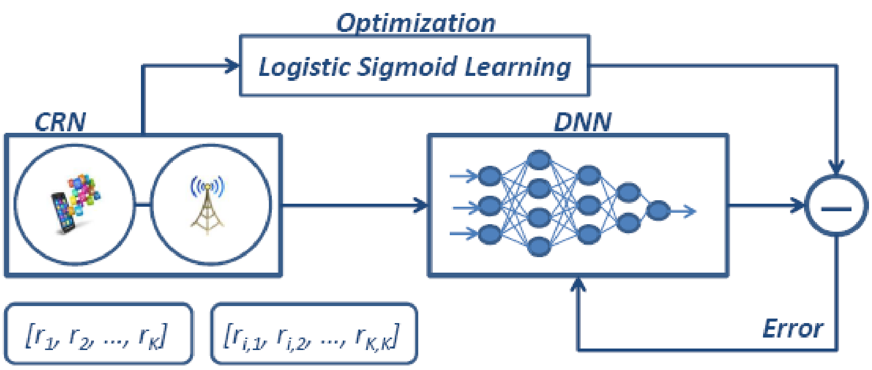

Figure 2.7. Resource Allocation in CRNs using DRL via DNNs and Logistic functions.

In Figure 2.7 above, the circles around the components of the CRN represents plurality, $r_1, r_2, \cdots, r_K$ represent the transmission rates allocated to SUs, while $r_{i,1}, r_{i,2}, \cdots, r_{K,K}$ represent the available rates offered by the BS. The optimization is carried out using a DNN via logistic sigmoid activation functions and the error is propagated back to the first hidden layer to improve the optimization process. However, it should be noted that CNNs and DNNs perform differently in such tasks, so it is wise to choose the best NN that fits one's application. CNNs derive different interpretations from signals in RL when applied in supervised learning problems. For example, in supervised learning, CNNs apply labels to signals and apply matching, which is prevalent in image processing applications where labels are matched to pixels and rank the labels that best fit the image in terms of their probabilities.

When performing the same image-processing task in RL, which is usually in computer vision (i.e., robotic navigation), a given image represents a state. Taking an action depends on future actions and states and what the CNN does is rank the actions possible to perform in that state and choose the best action. What the policy agent then does is to map a state to the best action (i.e., $a = \pi(s)$). Recall the function of Q, which is a distinct example of the Q-function that maps state-action pairs to rewards. Here, Q actually maps state-action pairs to the highest combination of immediate reward with all future rewards that might be harvested by later actions in the trajectory. The general equation for Q-function is thus given as follows:

$$Q(s_t, a_t) \leftarrow (1-\alpha_t)\underbrace{Q(s_t,a_t)}_{\text{old value}} + \underbrace{\alpha_t}_{\text{learning rate}} \underbrace{\left(\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma^t}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1},a)}_{\text{estimate of future value}}\right)}_{\text{learned value}}. \tag{2.23}$$

After assigning values to the expected rewards, the Q-function can selects the state-action pair that will result in the highest Q-value. At the beginning of the RL, the initialization of the NN coefficients can be done in a random/stochastic manner. Then, through the use of feedback, the NN can use difference between its expected reward to adjust its weights and improve its interpretation of state-action pairs. The feedback from the environment is a scalar number that is sent in response to each new action and the rewards returned by the environment can be varied, delayed or affected by unknown variables, thus introducing noise to the feedback loop. This process results in a complete expression that takes into account not only the immediate rewards produced by an action, but also the delayed rewards that may be returned several time steps deeper into the sequence of the trajectory.

When the Q-function is called on any given state-action pair it nests the Q-function to predict the value of the next state, which in turn depends on the Q-function of the state after that, and so on. This characteristic of DRL makes it promising in applications where agents need to act on time series data for time series modeling where recurrent NNs (RNNs) are one of the state-of-the-art models present. One popular architecture of the RNN used in time-series applications is the long short-term memory (LSTM), which is able to store information over extended time intervals [139]. This resolves the decaying error back-flow issue for many time-series problems where accurate predictions are required and traditional architectures such as multi-layer perceptron (MLP) fail. For example, a DRL algorithm to derive an optimal time scheduling policy for the gateway by backscattering was proposed in [140], in order to deal with large state and action spaces.

2.5.5.1 Deep Q-learning with Experience Replay and the Use of Policy Gradients

Deep Q-learning is a new Q-learning approach based on the use of DNNs to approximate the action-value function Q. Thus, DQNs with experience replay, also known as deep Q-learning with policy gradients, are an extension of the traditional Q-learning method, which is aimed at reducing training time by storing a number of transitions to be sampled later for the agent to learn from. Thus, experience replay is a key technique behind many recent advances in DRL [141], whose objective is to learn a strategy that leads to the best possible reward and the standard learning framework for the agent to use to learn the optimal behaviour and perform action selection is the action-value function. When using Q-learning, the technique becomes a DQN. Using a DQN is like taking some random actions and learning from them through the Q value function; and it is a regression problem (i.e., ℓ_2-loss is used) where two networks are used for training. Its primary objective is to find solutions to address high-dimensional and continuous control problems effectively through the use of DNNs as powerful function approximators. The flowchart of this procedure is illustrated in Figure 2.8 below.

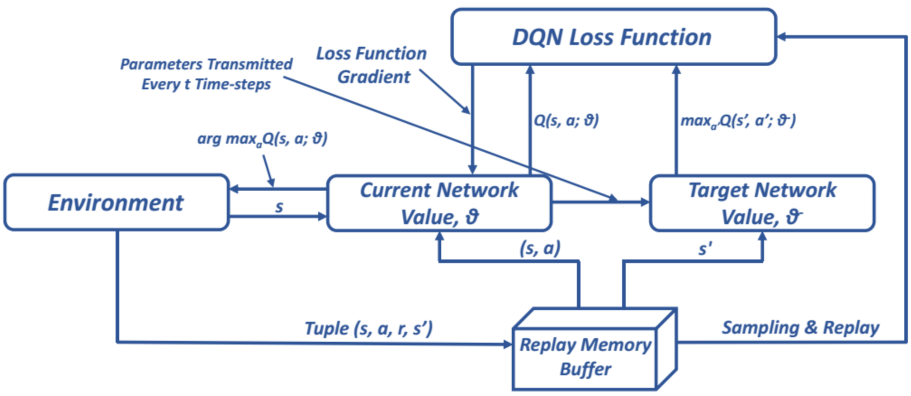

Figure 2.8. Flowchart of a DQN Algorithm with Experience Replay.

As seen in Figure 2.8 above, the DQN has been developed from the foundations of Q-learning, which is a classical algorithm used in RL. The update of the algorithm is as follows:

$$Q_{t+1}(s_t,a_t) \leftarrow Q_t(s_t,a_t) + \alpha_t[r_t + \gamma' \max a' Q_t(s',a') - Q_t(s_t,a_t)], \tag{2.24}$$

where α_t, γ' and r_t have been defined earlier. The DQN algorithm combines Q-learning with NNs, therefore the DQN uses a DNN as the Q-value network to compute the mean-square error (MSE) is used to define the loss function in Q,

$$L(\theta) = \mathbb{E}[(\hat{Q}(s,a;\theta) - Q(s,a;\theta^-))^2], \tag{2.25}$$

where θ is defined as the network parameter, and the target Q-value $\hat{Q}$ is defined as

$$\hat{Q} = r_t + \gamma' \max_{a'} Q(s',a';\theta). \tag{2.26}$$

As illustrated in Figure 2.8 above, the loss function is determined based on the second term of (2.24). Therefore, to obtain the gradient of the loss function L with respect to the parameter θ, an NN is trained using a SGD approach that updates the parameters to obtain the optimal Q-value. in terms of using policy gradients, one must recall that the objective is to maximize the expected cumulative discounted reward; the gradient of the objective is given by [142] as follows:

$$\triangledown_\theta \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right] = \mathbb{E}_{\pi_\theta}\left[\triangledown_\theta \log \pi_\theta(s,a) Q^{\pi_\theta}(s,a)\right]. \tag{2.27}$$

Here, $Q^{\pi_\theta}(s,a)$ is the expected cumulative discounted reward from deterministically choosing action a in state s while following the policy π_θ. The key idea in the use of policy gradient methods is to accurately estimate the gradient by observing the trajectories of executions that are obtained by following π.

2.5.5.2 Policy Gradients and the Monte-Carlo Method

In the simple Monte-Carlo method, the agent samples multiple trajectories and then uses an empirically computed cumulative discounted reward, v_t, as an unbiased estimate of $Q^{\pi_\theta}(s_t,a_t)$. This then updates the policy parameters via SGD as follows:

$$\theta \leftarrow \theta + \alpha \sum_t \triangledown_\theta \log \pi_\theta(s_t,a_t) v_t, \tag{2.28}$$

where α is the step size or learning rate. The equation results of the well-known REINFORCE algorithm are reported in [143]. Using deep Q-learning with experience replay, the experience of the agent can be stored in memory and randomly replayed at each time step for training. The input state s_t, which might be a clip of time series, is mean-value-normalized. The value of Q for a given state-action pair is the estimated expectation of total future rewards (i.e., profit and loss for the present task), discounted at the current step. The first step initializes the network $\mathbf{P}$ with random parameters θ, and policy π. From the fifth step, the procedure predicts the action probabilities $\mathbf{P}(A|A;\theta)$ and samples an action following policy π. The objective of the policy gradients is to maximize $\sum_t \log P(y_t|x_t;\theta)$, where x_t and y_t are training examples at time step t.

$$J(\theta) = \sum_t \log P(y_t|x_t;\theta) \cdot \hat{A}_t \tag{2.29}$$

is performed at every time-step until the last episode, at the last hidden layer, the error $\frac{\partial}{\partial\theta}J(\theta)$ is propagated to the first hidden layer for an update.

$$\bigtriangledown J(\theta) = \sum_t \underbrace{\bigtriangledown \log P(y_t|x_t;\theta)}_{\text{Actual - Predicted}} \cdot \hat{A}. \tag{2.30}$$

Here, the term $\hat{A}_t$ is defined as the advantage, which when positive pushes up the probabilities for all actions, otherwise it pushes them down. Using policy gradients is like learning optimal behaviour directly from experience (i.e., no value function) and it is a classification problem (i.e., maximum log likelihood is used with a some minor change).

2.5.6 Application of Deep Reinforcement Learning in CRNs

In CRNs, each SU acts as an agent and independently executes the DQN algorithm to select and access a licensed channel. The set of all actions $\mathcal{A} = \{a_1, a_2, \cdots, a_n\}$ that each SU can select at time t, where n denotes the number of licensed channels. The set of all states, represented as $\mathcal{S} = \{s_1, s_2, \cdots, s_n\}$, denotes the channel selected by the SU and the utility obtained after channel access. When executing the above procedure, the agent learns to perform resource allocation tasks and obtains high rewards by choosing better actions. At each time step, the agent is given observations in the past time-steps and then it chooses an action, which results in a reward. The adjustment parameters are transmitted every t time steps and after T time steps, an episode ends. This procedure can be performed based on either univariate or bivariate methods. In the univariate case, the model is tested on whether it can capture the underlying dynamics, while in the bivariate case the model is tested on whether it can utilize the hidden relationship among inputs. For both cases, the inputs are positive values. A typical application of DQN and experience replay in a CRN environment can be done as illustrated in Figure 2.9 below.

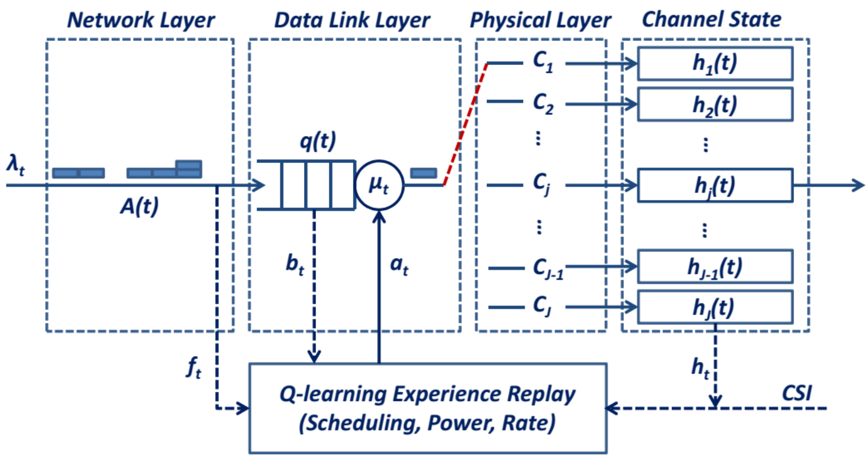

Figure 2.9. The Orchestration of RA using MDPs and Experience Replay in CRNs.

In Figure 2.9 above, λ_t is the packet arrival rate, $A(t)$ is the packet arrival process, $q(t)$ is the queue length, μ_t is the service rate, b_t is the buffer state, a_t is the current action, C_t and h_t have already been defined earlier. The transition probabilities f_t, the buffer state and the CSI from the receiver are fed into the algorithm which will decide on a scheduling mechanism, transmission power, and transmission rate, using experience replay.

Numerous architectures exist in literature that use deep Q-learning in other fields, but for the purpose of demonstrating the usefulness in communication networks, the researcher focuses on baseline models. A DRL application was investigated for solving resource management problems in network slicing in [144], where resource slicing and priority-based slicing were considered in the core network. In [145], a traditional RL-based approach was proposed to solve the missing gradient problem when training the transmitter, where the source data was converted into transmit symbols. Here, both the channel and the receiver were considered as the environment while the transmitted data was regarded as the state that is being observed by the transmitter. Then, for every time step t, the transmitted signals were considered as the actions executed by the transmitter. Then, the end-to-end loss for each sample was evaluated at the receiver using the policy gradient algorithm, and was sent back to the transmitter as a reward to learn from and optimize the loss without requiring channel gradients. Applications of RL and DRL in wireless communication systems are tabulated in Table **2.6** below.

Table 2.6. Application of RL and MDPs in Wireless Communication Networks.

Application	**Problem**	**States**	**Actions**	**Rewards**	**Refs**
MIMO transmission control	Minimizing transmission power	Channel and traffic statistics	Power allocation and rate allocation	Transmission costs and energy efficiency	[126]
Wireless communs.	Power control	Channel and traffic statistics, buffer state, power management	Power management	Packet holding and overflow cost	[127]
Wireless communs.	Control-power allocation	CSI, data arrival, queue length	Power management	Maximum throughput	[128]
Wireless communs.	Admission and call dropping control	Link capacity, Arrival frequency	Admission decision	Class-level QoS	[129]
WSNs	Data acquisition	Node attributes	Query or no query	Confidence levels	[130]
Wireless communs.	Power and rate adaptation	Buffer occupancy, SIR, data rate	Transmission strategy	Received packets, energy	[134]
Relay networks	Routing	Sample arrivals, Tx queue and distance	Send, Wait and route selection	Energy, E2E delay and data volume	[135]
CRNs	Time scheduling	CSI, energy state	Scheduling policy	Maximize throughput	[140]
RL	Policy learning	Resource demand	Scheduling	Min. average slowdown	[142]
Wireless communs.	End-to-end learning	SNR in AWGN and RBF channels	Transmitter normalization	Minimize convergence time	[145]
Network slicing	RA for priority-based slicing	Resource demand, & availability	Resource scheduling	Min. scheduling delay	[144]
CRNs	Spectrum access	Spectrum occupancy	Sense, occupy spectrum	Ultra-low power	[146]

2.6 DEEP LEARNING

This section discusses the different components of DL, their mathematical background and then their applications in mobile and wireless communication systems. DL, also known as DNN, as discussed earlier is capable to learn unsupervised from either unstructured or unlabeled data sets through the use of a hierarchical level of artificial NNs (ANNs) [148]. Even though there is no unified definition definition of DL, a more pragmatic and realistic definition of DL is that it is a process that not only learns the relationship among several variables, but also the knowledge governing the relationship, as well as the knowledge that makes sense of the relationship [147]. DL algorithms are trained using SGD, which is a very popular and efficient algorithm in training DNNs and other non-convex models. Owing to its low computational complexity per iteration, an SGD has been used to solve a variety of ML and SP optimization problems in a variety of ways.

At each iteration, SGD firstly calculates a gradient based on a randomly selected sample and updates the model parameter along the negative gradient direction of the current iterate, a process known as backward propagation (BP). Theoretical properties of SGD are well understood for optimizing both convex and non-convex objectives, the latter of which can be related to other assumptions on objective functions, e.g., error bound conditions and the Polyak-Lojasiewicz conditions [150]. Indeed, there is a huge gap between the theoretical understanding of SGD and its very promising practical behaviour in non-convex learning setting, as exemplified in the setting of training highly non-convex DNNs. In this thesis, this training technique is applied in spectrum occupancy reconstruction in Chapter **3**.

2.6.1 Feedforward and Feedback Control

In DL, the problem of synthesizing NNs that can serve as state and output feedback laws to achieve control objectives that can be specified as reachability of a target set, region of stability of a target set around an equilibrium point or principle, logical formulas written in a temporal logic can be used for training the weights of a feedforward-feedback network from samples. The hierarchical function of a feedforward-feedback DL function is illustrated in Figure 2.10 below.

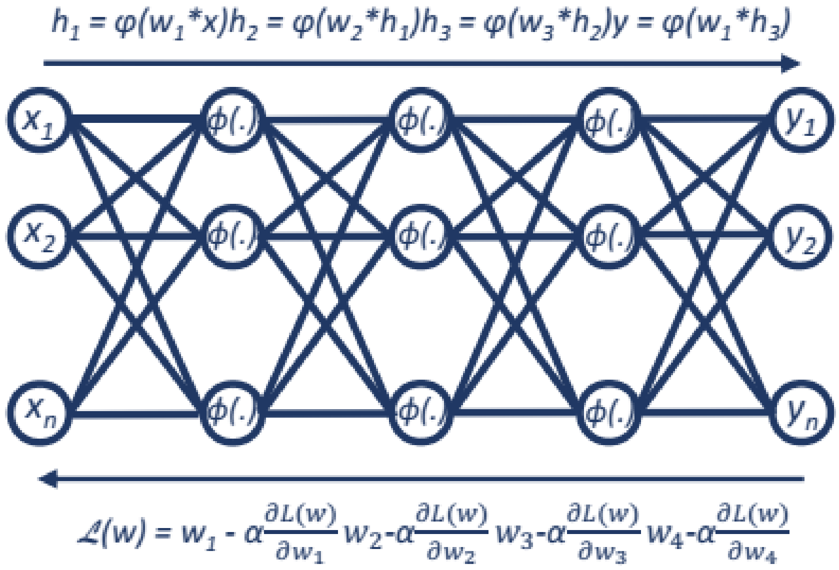

Figure 2.10. The Learning of a Four-layer Convolutional NN Through Inference (forward-passing) and Learning (BP) processing.

In Figure 2.10, the principle of inference (feed-forward) and learning (BP) processes on an NN are illustrated. Here, a 2-D CNN has been described in mathematical terms where the convolutional NN (CNN) weights are learned through the minimization of the loss $\mathcal{L}$ via gradient descent methods. As described earlier, the term $w_{(.)}$ denotes the weight of each hidden layer, α is called the step size or the learning rate and $(\cdot * \cdot)$ denotes the convolution operation. During feedforward or forward propagation, as the name suggests, the input data are fed in the forward direction through the network. Each hidden layer accepts the input data, processes it as per the activation function and passes it to the successive layer. This process is carried out until the output is obtained [151]. The CNN illustrated in Figure 2.10 above consist of an input layer with inputs $\mathbf{x} = \{x_1, x_2, \cdots, x_n\}$ that are prosessed by a convolutional layer through the operation $h_1 = \Phi(w_1 * \mathbf{x})$, where h_1 is the output of the first hidden layer [152]. This output is subsequently fed as input to the next hidden layer until the output $\mathbf{y}$ is achieved at the end.

BP is actually a technique from supervised learning that is based on the GD method that minimizes the error of the network by reducing the gradient of the error curve [153]. To train a CNN, the loss function $\mathcal{L}(w)$ is used as a measure of the difference between the output $\mathbf{y}$ and the actual ground truth $\mathbf{y}^*$. Therefore, the training objective is based on obtaining the best weights $\mathbf{w}$ that minimizes the $\mathcal{L}$ function through BP. During the BP operation, the gradient of $\mathcal{L}$ is computed over the weight of the last hidden layer, and the weights are updated as follows:

$$w_4 = w_4 - \alpha \frac{\partial \mathcal{L}(w)}{\partial w_4}. \tag{2.31}$$

The process repeats until the GD eventually results to a set $\mathbf{w}$ that minimizes $\mathcal{L}$ [151]. This technique performs parallel training for improving the efficiency of the multilayer perceptron (MLP) network, discussed in Section **2.7.1**.

2.7 APPLICATIONS OF DEEP LEARNING IN WIRELESS COMMUNICATIONS

DL will play an important role in CR, which promises a broad prospect in spectrum monitoring and management with coming applications for the 5G and IoT networks. A great deal of research and exploration on DL techniques has been done and a series of effective schemes has been developed. In this section, a brief overview of DL strategies and approaches is presented; more specifically DL schemes are extended from SP to wireless technology with the continuous evolution of wireless communication systems.

2.7.1 The Multilayer Perceptron

The traditional MLP consists of at least three layers with densely connected units in each layer, and only an MLP with more than one hidden layer is considered as a DL structure [153]. Through the use of the BP algorithm, MLPs have become popular and effective in learning models for complex, multi-layered networks, as a result they are widely integrated into more complex architectures such as the output layer in CNNs in classification tasks and the analysis of continuously changing mobile environments. For example, a key generation and certification technique using MLP was proposed in [154] to wirelessly communicate data to a receiver. Here, both the sender and receiver used identical MLPs and both perceptrons began synchronization by exchanging some control frames. Also, a complex MLP model was used for multi-user detection in [155], where channel approximation and multi-user detection were simultaneously performed in a space division multiple access-orthogonal frequency division multiplexing (SDMA-OFDM) system. This technique was proposed to minimize the high computational complexity of classical techniques such as maximum likelihood, and the MMSE, which yields poor detection performance when the number of users exceeds the number of receiving antennas.

An application of the Levenberg Marquardt training algorithm using MLPs for mobile positioning was proposed in [156], designed for GSM operations in urban environments. Here, using the received signal strength, the key performance metrics such as accuracy and cost were used in the evaluation of the technique. In medical applications, the MLP was applied for QoS-aware, content-aware, and device-aware non-intrusional medical QoE prediction model over small cell networks in [157]. Here, the MLP was used for medical QoE prediction and acted as a platform for maintaining and optimizing diagnostic quality. Unknown data sets of QoS, content features and display characteristics were used as training inputs to the MLP and the QoE results were produced in the form of the mean opinion score (MOS) as medical QoE. The efficiency of this model was evaluated through subjective tests carried out by medical experts and the prediction accuracy obtained via the correlation coefficient and root mean squared error (RMSE) indicates that the proposed model succeeds in measuring medical QoE closer to the visual perception of the medical experts.

2.7.2 The Restricted Boltzmann Machine

The RBM, which defines the point where DL meets physics, was originally designed as an unsupervised learning technique. It is essentially an energy-based undirected graph model, which consists of one visible layer and one hidden layer, where each unit can assume only binary values. The architecture of RBMs is discussed in detail in [158], while the application of the energy-based model and the probabilistic model can be found in [159]. However, in a nutshell, the probability that a binary state of a visible neuron i is set to 1 is given as:

$$P(h_i = 1|v) = \frac{1}{1+e^{-\mathbf{W}\cdot\mathbf{v}}+\mathbf{b}_i}, \tag{2.32}$$

where h, and x are respectively the hidden and visible units and the visible units are conditionally independent from the hidden units, and vice versa. This technique was applied in spectrum occupancy reconstruction [159], where spectrum occupancy data was sampled from an MRF. The RBM was trained using the Metropolis-Hastings algorithm through multiple steps of Gibbs sampling. The authors used latent factors for prediction and after the training phase, the goal was to predict the binary values of the spectrum results that had not been received. This application is reported in Chapter **3**, Section **3.5** of this thesis.

A stack of RBMs is referred to as a DBN, which achieves superior performance in time-series forecasting. DBNs usually consist of probabilistic generative NNs composed of multiple layers of RBMs [160]. An application of RBMs for time-series prediction was proposed in [160], where a three-layer deep network of RBMs was used to capture the features of the input space of time-series data. After pre-training of the RBMs using their energy functions, GD training was used to fine-tune the connection weights between the visible and hidden layers. A novel approach for broadcasting time-critical traffic information in vehicular communications was proposed in [161]. The proposed method for predicting the traffic flow volume in the internet-of-vehicles (IoVs) uses DBNs with multiple layers of RBM AEs where the time-series from roadside units is used by a three-layer DBN to extract and learn key input features for constructing a model to predict traffic flow. A firefly algorithm was used in learning rate parameters and also optimize the DBN topology during the training process.

2.7.3 Long Short-Term Memory

An LSTM network is typically a RNN that is extremely successful in dealing with time-dependent SP problems such as speech recognition, joint source-channel coding design, machine translation, and time-series forecasting [162]. The design objective of the LSTM was to address the vanishing and exploding gradient problems associated with conventional RNN architectures (e.g. autoregressive integrated moving average (ARIMA)) [139], another architecture used for time-series purposes. Thus, it is a special kind of a DL scheme that successfully overcomes the obstacle of long-range dependencies in time-series using three regulators of the flow of information inside the LSTM unit (i.e., an input gate, an output gate, and a forget gate). An illustration of a single memory block of an LSTM architecture is shown in Figure 2.11 below.

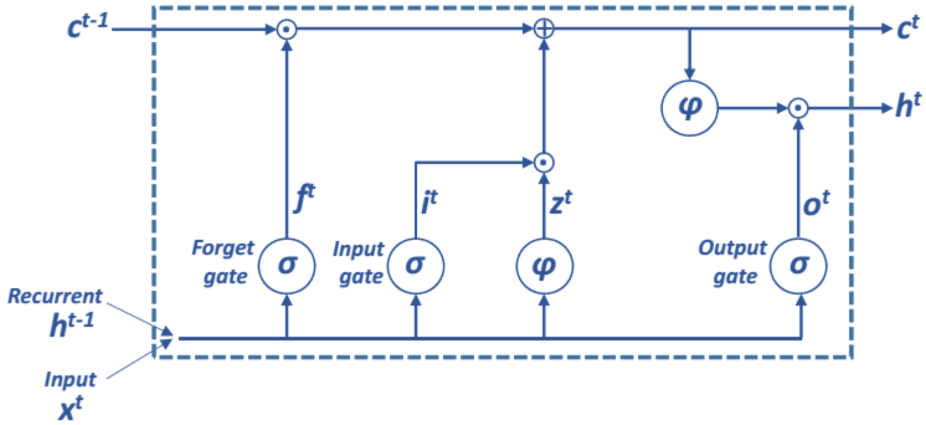

Figure 2.11. One Memory Cell LSTM Architecture.

Because of its need for memory, an LSTM is used with forget units, f^t, which give the memory cells the ability to determine when to forget certain information, thus determining optimal time lags. Since 2015, the LSTM has been used for traffic flow prediction and the structure of its complete architecture is given in [56]. The LSTM cell operates as follows: The input gate, i^t, takes a new input point from outside and processes newly available data. The memory cell input gate takes input from the output of the LSTM cell in the last iteration. The forget gate decides when to forget the output results and thus selects the optimal time lag for the input sequence. Then, the output gate, o^t, takes all the computed results and generates an output for the LSTM cell. Thus, LSTM is more effective in the utilization of model parameters in the training of prediction models for large-scale traffic matrix predictions and converges quickly to give state-of-the-art prediction performance. In modeling the LSTM for time-series applications, a linear regression layer is applied on the output layer of the LSTM cell. For example, one can denote the input time-series as $X = (x_1, x_2, \cdots, x_n)$, the hidden state of the memory cells as $H = (h_i, h_2, \cdots, h_n)$, and the output time-series as $Y = (y_1, y_2, \cdots, y_n)$. Thus, the computation procedure of an LSTM cell is as outlined in [163], with the MSE objective function used, given by the following formula:

$$MSE = \sum_{t=1}^{n} (y_t - p_t)^2, \tag{2.33}$$

where y_t represents the real output and p_t represents the traffic flow predicted through projection. In order to minimize training error and meanwhile avoid local minimal points, Adam optimization [164], which is a modification of SGD optimization with adaptive learning rates, is applied for BP through time (BPTT). The LSTM was used for proactive caching in mobile edge computing in [165], where the authors used it to improve the prediction accuracy of content popularity, which is typically unknown and changes over time.

In mobile and wireless communication networks, the prediction of traffic series and mobility traces could directly benefit from the capabilities of LSTM. In [162] an LSTM was applied to spectrum prediction together with a module generator for a latency-optimized field-programmable gate array. The proposed scheme obtained superior results in radio frequency spectral prediction compared to other time-series prediction techniques such as the ARIMA, which uses a naive predictor. Time series prediction by extracting useful information from historical records to determine future values was considered in [166]. A random connectivity LSTM (RCLSTM) model was used to reduce the considerable computing cost associated with LSTM and was tested for predicting traffic and user mobility in telecommunication networks. RCLSTM is based on stochastic connectivity between neurons which achieves significant breakthroughs in the architecture formation of NNs compared to LSTM. As a result, RCLSTM exhibits a certain level of sparsity, which satisfyingly reduces the computational complexity and is hence suitable for latency-stringent application scenarios. However, this improvement demonstrates that the prediction accuracy of RCLSTM is comparable to that of the conventional LSTM, no matter how much the number of training samples and the length of input sequences are changed.

An energy-aware and adaptive management (ENAAM) algorithm was proposed in [167], which is an online optimization based on foresight control policies. Here, the BSs and virtual machines (VMs) could be switched ON/OFF dynamically in order to achieve energy saving and QoS provisioning by exploiting short-term traffic load and harvested energy forecasts using LSTM. This contribution was inspired by the convergence of communication and computing, which led to the emergence of multi-access edge computing (MEC), where computing resources at the edge of the network are supported by VMs.

2.7.4 Auto-Encoder

AEs were also initially designed to slove problems in unsupervised learning where inputs are copied to outputs, as discussed earlier. The primary objective of the AE is learning compact data representations for the purpose of dimensionality reduction [168]. An AE is an unsupervised learning technique that also poses as a supervised learning technique in disguise, while at the same time it is also an unsupervised-*ish* DL technique. A typical application of an AE in an end-to-end communication system is illustrated and discussed in [23]. The AE is considered a supervised learning technique since it does have a target value (i.e., original inputs), thus it is considered to have some degree of supervision. On the other hand, it is considered an unsupervised learning technique since the target value is not in addition to the input data. Yet another characteristic of an AE is that it belongs to the family of NNs, but also very closely related to principal component analysis (PCA) [169]. Thus, (i) it is an unsupervised ML algorithm similar to PCA; (ii) it minimizes the same objective function as PCA; (iii) it is an NN; and (iv) an NN's target output is its input.

Since an AE in DL is used to map original data from the input layer into a code and can recover the data that closely matches the input data at the output layer, it is a DL technique [170]. In summary, AEs are unsupervised learning in design, but the depth of their architecture renders them DL techniques. An AE application in end-to-end communications, where the transmitter is modelled as a feed-forward NN with several layers, communicates with a receiver over a time-varying wireless channel. Here, both the transmitter and receiver were represented with fully connected DNNs and were jointly optimized over an additive white Gaussian noise (AWGN) channel. The time-varying frequency response of the wireless channel characterizes the channel in terms of time delays, Doppler frequency shifts and gains, all of which vary randomly in the modelling. The wireless channel is represented by a single layer which provides a transfer function operation $p(y|x)$ between the transmitter and receiver. The receiver is also implemented as a feed-forward NN, which receives the channel output signal y and processes it to the received signal r via several transformations.

In mobile and wireless communication applications, expert knowledge is incorporated into the AE-based communication syatem to avoid the prevalent "black box" architecture. An example of this is in [23], where the communication transceiver design was considered as an AE task. This AE-based system was extended for multi-user communications in [171], where communication over interference channels in OFDM systems with multipath channels was considered. Also, a data-driven DL approach that uses an AE to optimize transceivers without CSI knowledge was studied in [172]. There are also other variations of the AE, namely the de-noising AE, which is a basic AE architecture that takes a partially corrupted inputs randomly to address the identity-function risk, which the AE has to recover or de-noise [173]. Another variant is the sparse AE, which can learn sparse features through which a small number of hidden neurons will respond to a specific feature of the data set.

2.7.5 Stacked Auto-encoder

The last of the DL applications uses the SAE, which consists of multiple layers of sparse AEs that employ greedy-wise training. An SAE has since been used with a Softmax activation layer to fine-tune a sub-band power allocation model in [174]. This hierarchical model of AEs was used to extract high-level features and correlations from the input data through multiple layers of non-linear processing units, which improves the accuracy of prediction. DL architectures built in this way reduce computational and time complexity, since a trained model can achieve multiple objectives without the need for retraining and can make inferences within milliseconds. As modulation identification and classification of the transmitted signals remain a challenge in modern intelligent communication systems such as CRs, the computation of distinct features from input data sets becomes easy with the SAE compared to other well-known methods. Recently, the SAE gained significant attention in the pattern recognition of complex data owing to its superior performance, and has since been applied in spectrum sensing problems.

In [175] a spectrum sensing algorithm for OFDM signals based on DL and covariance matrix graph was proposed. The DL advantages in image processing were applied to the spectrum sensing of OFDM signals by firstly building the spectrum sensing model of OFDM signals and then analyzing the structural characteristics of the covariance matrix. After normalizing the covariance matrix and transforming it into a gray-level presentation, the gray-scale map of the covariance matrix was established. A CNN designed based on the LeNet-5 network was used to learn the training data to obtain more abstract features hierarchically. The test data were input into the trained spectrum sensing network model based on which spectrum sensing of OFDM signals was completed. Another DL SAE application based on time domain signals in spectrum sensing problems is shown in Figure 2.12 below [176].

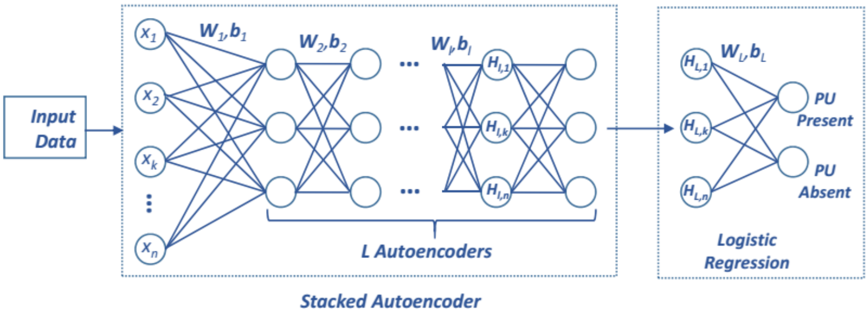

Figure 2.12. Illustration of a SAE Implementation for Spectrum Sensing Using an Input Data Set Consisting of Time-domain Signals.

In this publication, the authors proposed a novel spectrum sensing framework for OFDM signals to address the issues of noise uncertainty, time delay and carrier frequency offsets suffered by conventional OFDM. The SAE was designed to extract the hidden features of OFDM signals and to use these features to classify the OFDM user's activities. In another implementation, the authors proposed to improve the spectrum accuracy under low SNR conditions using time-frequency domain signals as shown in Figure 2.13 below. Higher spectrum sensing accuracy was achieved compared to

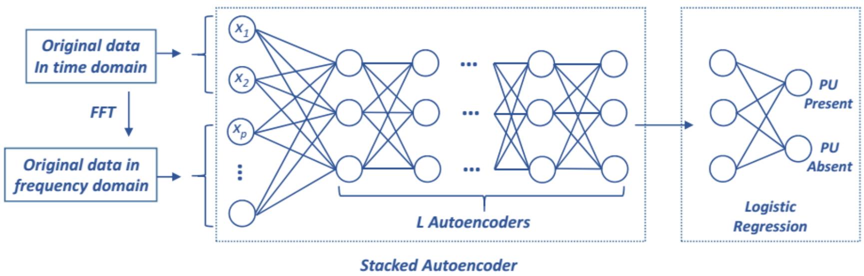

Figure 2.13. Illustration of a SAE Implementation for Spectrum Sensing Using an Input Data Set Consisting of Time-frequency Signals.

the time domain implementation, albeit at the cost of higher computational complexity. In addition, a blind spectrum sensing method based on DL was studied in [177] to improve spectrum sensing in low SNR situations where prior information of the licensed user is lacking. In this application, three kinds of NNs were used together: a CNN, LSTM and fully connected NNs; resulted in improved performance compared to the traditional energy detector, especially in low SNR regimes. The effect of different LSTM memory layers was also analyzed and it was explored why the DL-based detector could achieve better performance. The motivation for using several LSTM layers is to establish a more efficient model of the probability distribution of the observed sequence of HMMs, extract the timing features of the signals and distinguish the signal and noise from the timing regularity of the input data.

The LSTMs consist of several NN layers and a regularization layer whose role is to speed up training the model and improve the regularization capability of the model. The fully connected NNs use a stack of multilayer NNs to form a DNN, which refines the output features of the LSTM and attenuates the influence of task-independent features on the decision results. The last layer is the decision layer of the entire network using a linear NN. A comparison of the algorithms discussed is shown in Table **2.7** below.

Problem	**Application**	**DL Method**	**Technique**	**Refs**
Image reconstruction	Image processing	MLP	Generalized delta rule	[153]
Cryptography	Network security	MLP	Key generation and MLP	[154]
Channel approximation	Wireless communications	Complex MLP	Differential evolution	[155]
Mobile location estimation	Wireless communications	Robust MLP	Levenberg-Marquardt	[156]
QoE prediction	Telemedicine	MLP	RMSE	[157]
Spectrum occupancy reconstruction	CRNs	RBMs	SGD	[159]
Time-series forecasting	Chaotic time-series	DBNs	GD	[160]
Time-critical information broadcasting	Self-organizing IoV	DBNs	Firefly algorithm	[161]
Spectral Prediction	CRNs	LSTM	MSE	[162]
Time-series prediction and user mobility	Wireless communications	Random Connectivity LSTM	RMSE	[166]
RA and energy	Wireless	LSTM	Adaptive	[167]

harvesting	communications		management algorithm	
Phonetic labeling of acoustic frames	Speech recognition	LSTM	Asynchronous SGD	[163]
Proactive caching	IoT	Bidirectional deep RNN	Adam optimization	[165]
Network intrusion detection	Network security	Sparse AE and PCA	BP	[168]
End-to-end signal reconstruction	Wireless communications	AE	BP	[23]
Discriminative learning	Image processing	Stacked denoising AE	BP	[173]
Multicarrier systems	Wireless communications	AE	SGD	[171]
Channel agnostic end-to-end Learning	Wireless communications	DNNs	GD	[172]
RA	Multi-cell networks	SAE	SGD	[174]
Spectrum sensing	CRNs	SAE	SGD	[176]
Spectrum sensing	CRNs	CNN, LSTM, Fully connected NNs	Tensorflow backend	[177]
Spectrum sensing	CRNs	CNN	LeNet-5 network	[175]
Constellation design	Radar communications	AE	BP	[180]
Spectrum sensing	CRNs	Ensemble classifier	AdaBoost algorithm	[186]
Compressive training	Embedded systems	CNNs	SGD	[189]
Energy and performance efficient computation offloading	Mobile cloud computing	DNNs	Tensorflow	[190]

Table 2.7. Tabulation of DL Approaches in Wireless Communications.

The numerous advantages of DL are discussed in the following subsections. The computational prowess of DL techniques has gained immense popularity and has so far attained remarkable achievements in terms of performing complex tasks such as in the fields of computer vision, natural language processing and communication network domains [178]. Other recognized advantages of DL in addressing mobile and wireless networking problems are summarized below:

2.7.6 Multimodal Information Understanding

DL, with its rich family of methods, encompassing NNs, hierarchical probabilistic models enable the development of fundamentally new ways to rethink mobile and wireless communication systems design. With the field of mobile and wireless communications being rich in expert knowledge in terms of different ways of channel modeling to compensate for various hardware imperfections, using DL strategies may ensure the desing of optimal signalling and detection schemes to ensure reliable information transfer [179]. Using multimodal information understanding, DL can implicitly capture the intricate structures of large-scale data. While it might seem a rather little impossible task to write robust algorithms that can handle multiple tasks and handle large amounts of data, the combination of DL with RL (i.e., DRL) makes the implemetation of algorithms that can learn to accomplish such tasks with reasonable accuracy possible [180].

2.7.7 Multivariate Representation

DL techniques have been found to be able to handle geometric mobile data using a rebranded technique called geometric DL [181], which is the niche field under the umbrella of DL that aims to build NNs that can learn from non-Euclidean data. This is actually a conundrum since the traditional DL models are performed on Euclidean data, which is either 1D or 2D. However, in reality, everything that can be observed exists in 3D and the data should actually reflect that. Therefore, since the use of flying platforms such as unmanned aerial vehicles is rapidly growing in wireless networks, the time has come for DL approaches reach that level [182]. Using the geometric data for the multivariate representation of the coordinates, the topology, the metrics and mobile data, user location and network connectivity can be represented using point clouds and graphs. Therefore, this important property of geometric data representation bears great potential to revolutionarize geometrical analysis of mobile data using geometric DL [183].

2.7.8 Feature Extraction and Pattern Recognition

Feature extraction is mainly associated with dimensionality reduction and entails the conversion of given input data into a set of features such that DL models can extract high-level features from complex structured data with inner correlations automatically [184]. DL feature extraction algorithms are capable of automatically extracting features through DNNs of different layers and depths. The algorithms begin with an initial set of consistent data and develop borrowed values, also called features, expected to be descriptive, and to simplify the consequent learning and observed steps [185]. The importance of feature extraction in mobile and wireless communications is emphasized in the context of PU signal classification in spectrum sensing [186]. Since mobile and wireless data are generated by heterogeneous sources and often exhibit non-trivial spatio-temporal patterns, whose labeling requires outstanding effort, DL can be beneficial for this task [187].

The advantage of DL in this case is that it can reduce the cost of expensive feature engineering techniques in the processing of heterogeneous and noisy mobile Big Data, which makes it more attractive in mobile and wireless communication networks than other ML methods. The effectiveness of DL strategies in regognizing patterns in unlabelled data, which is an exceptional capability it shares with supervised learning is its other attractive advantage. However, in the execution of pattern recognition tasks, supervised learning only thrives with labeled data. Since current mobile and wireless communication systems generate unlabeled or semi-labeled data, this presents a quantitatively different state of affairs for supervised learning [188] because it cannot utilize unlabelled data explicitly. DL strategies also provide a variety of useful techniques for pattern recognition tasks such as RBMs, which also perform the same task pretty well unsupervised learning applications.

2.7.9 Compressed Representation

Performing compressed representation of data using other ML techniques such as linear regression and random forests is a huge challenge, but DL is able to accomplish this task through the use of DNNs with different layers and depths such as stacked LSTM or SAE. Because of the vast amounts of data that need to be operated continuously in wireless networks, regardless of storage and access distribution and quich response to new information, DL is beneficial in model compression and accelerated retrieval using NNs [189]. Compressed representation is also beneficial in reducing overheads in data storage and processing in mobile networks by compressing the size of data while maintaining their utility. Thus, using DL models, a single model could be trained to fulfill multiple objectives without requiring the model to be retrained for each different objective. Therefere, it is argued that DL is an essential tool for future mobile wireless networking problems, owing to its advantage in reducing the huge requirements for computational and memory requirements of mobile systems when performing multi-task learning applications [190].

2.7.10 Big Data Exploitation

Telecom operators currently have huge amounts of wireless Big Data, far beyond what they are able to grasp and make sense of. Ranging from the continuous information streaming from their customers and their usage behaviour, to the thousands of infrastructure elements and millions of handsets, the data generated are huge. It all makes for an exciting exercise to understand what the data are all about and how can these be used for greater value creation. If one adds this to the paradigm shift that the IoT seeks to bring about, multiplying exponentially the number of data points and devices, the result is gigantic volumes of data. With IoT forming a key component of next generation strategy in telecommunications and the greater demand and willingness to understand customer needs and ensure targeted customer offerings, while maintaining best quality networks despite increasing costs, it is imperative to look at predictive analytics in a new light and seek to develop greater understanding of its benefits.

The use of predictive analytics could be beneficial to mobile operators in obtaining new levels of insights into how their networks are used and how they could be improved. Key strategies include segmentation, next best offer, churn prevention, maximizing lifetime value of customers, capturing customer sentiments, cross-selling/up-selling, predictive maintenance, optimization of customer care, process optimization. Since future wireless network generations will become more sporadic, which would be evident in terms of time, subscriber, location and application; network providers would be able to examine subscriber and location to find out which users consume more data of all the data in the network. Significant performance gains could be achieved through SONs and network operators could then apply predictive analytics that would reveal which users are actually consuming how much bandwidth - and where are they located. This level of foresight would be key not only in unlocking the potential of SONs, but also to maximize the return on investment for software-defined networking (SDN) and network function virtualization (NFV) in the core.

Integration of Big Data analytics was explored with network optimization to reach the objective of improving user QoE in [191]. The first objective involved Big Data-driven framework mobile network optimization. Then, Big Data characteristics collected from both the user equipment and the wireless network were presented together with several techniques in data collection and analytics are discussed from the viewpoint of network optimization. In [192] a framework for transforming heterogeneous networks to smart networks by leveraging wireless Big Data, CR, and NFV techniques was proposed. The CR and NFV support resource slicing in spectrum, PHY layers, and network layers, while wireless Big Data is used for designing intelligent mapping of resources and traffic prediction through AI methods.

2.8 CONCLUSION

In this chapter, a systematic review of DL techniques in mobile and wireless networking and the way in which they can play an increasingly important role in the CRN domain was conducted. The first objective of this review was to discuss the unacceptable impact on the overall network performance as SDR and CR become more feasible. This objective was met by analysing the state-of-the-art CR solutions and it was realized that SONs are essential to the imminent connected future because of their automatic reconfiguration, optimization and self-healing capabilities, which would free some operational resources to be deployed elsewhere in the network. The second objective was to analyse the need for improved spectrum management in the IoT era. Here, some problems faced by traditional techniques in efficiently solving wireless networking problems were discussed categorically in terms of model and algorithmic deficit problems and it was motivated that more research should focus on integrating DL techniques into CRNs. The third objective of this chapter was explaining why DL strategies are seen as promising solutions for future wireless networking problems in a broader context. To address this objective, a comprehensive review of ML techniques, together with their mathematical fundamentals and recent work that lies at the intersection between DL and mobile networking, was provided. Then the basic concepts, advanced principles of various models and research works specific to mobile and wireless networks across different application scenarios were reviewed. The fourth objective was concerned with how SP and DL techniques could be combined to achieve better spectrum management schemes for future CRN deployments. In this objective, the tailoring of SP and DL models into general mobile and wireless network applications was discussed, which is an aspect that has been overlooked in previous review works. The fifth objective was to discuss the cutting-edge DL methods that are relevant for future mobile and wireless network environments. In the hope that this review would become a definite guide to researchers and practitioners, applications of DL in wireless communications were discussed and DL was advocated as an indispensable design tool for the complexity of future mobile and wireless communication networks. This was done by discussing the advantages of DL strategies in recent technological advancements in the fields of multimodal information understanding, multivariate representation, feature extraction, pattern recognition, and compressed representation. The last objective was to discuss the most successful applications of DL in the mobile and wireless networking domain. In this case, the use of DL strategies was made even more practical by discussing the Big Data exploration in enabling predictive analytics and NFV in mobile and wireless networks.

However, what transpired from this review is that there are several obstacles that need to be overcome in order to derive more satisfactory performance from DL models in mobile and wireless networking. The first obstacle is that most DL models are data-driven and for this reason, large amounts of data are required to train NNs to ensure satisfactory performance. In such data-driven approaches, the communications system is treated as a black box and trained by using a huge volume of data. Training DL models in this way requires sufficient computing resources and extensive computation time, both of which are rarely found in communication devices. Another challenge is that the required training data are difficult to obtain in mobile and wireless networks, because these data require extensive measurement campaigns that are very expensive to conduct. One of the most promising ways to overcome this obstacle is through the hybridization of model-based (i.e., SP techniques) and data-driven approaches such that with the joint use of these two approaches, transfer-learning methodologies can be developed. The combination of these two approaches allows wireless communication domain knowledge to be incorporated into the available data, which can reduce the demand for computing resources and training time.

CHAPTER 3 SPECTRUM OCCUPANCY RECONSTRUCTION IN DISTRIBUTED COGNITIVE RADIO NETWORKS USING DEEP LEARNING

3.1 CHAPTER OVERVIEW

Spectrum occupancy reconstruction is an issue that is often encountered in collaborative spectrum sensing in distributed CRNs. It often arises in practical spectrum sensing scenarios owing to imperfect reporting channel conditions or other specific collaborative spectrum sensing schemes. The SSD contributed by SUs usually have gaps of missing entries which results in incomplete spectrum occupancy results that degrade the performance of spectrum detection, especially when the amount of missing entries is large. In order to circumvent this predicament, numerous missing data imputation algorithms such as matrix completion have been proposed and have shown great promise in the reconstruction of the spectrum occupancy data matrix. However, these algorithms often achieve lower spectrum occupancy reconstruction resolution owing to the usage of the standard singular value decomposition (SVD) algorithm, which performs well in more general matrices.

In this chapter, a DL-based spectrum occupancy prediction that uses DBNs, which are a kind of generative NN composed of RBMs, is proposed. Here, the spectrum observations are represented as a magnetic state recovery problem on an MRF, and connections between SUs are represented using link energy functions from the Ising model. This is a data mining approach where the RBM captures the feature of the input space of spectrum occupancy data into a matrix; which after pre-training of the RBM using their energy functions, a spectrum occupancy data matrix is obtained using the Metropolis-Hastings algorithm. Then, a new training principle for DL representations based on the idea of making the learned representations robust to partial corruption of the input data pattern is introduced using matrix factorization. In order to decide on the size of the samples and learning rates, Gibbs sampling is adopted during the training process and the missing entries are processed and learned using a scaled SGD algorithm. The simulation results obtained in this chapter indicate that the SGD algorithm has a superior spectrum occupancy reconstruction performance than the standard SVD algorithm.

3.2 INTRODUCTION AND BACKGROUND

The distributed spectrum sensing technique in CRNs was introduced in order to achieve more reliable SSD sets contributed by individual SUs for more efficient RA purposes. It entails a collaborative form of spectrum sensing, where individual SUs contribute their spectrum sensing observation results with their neighbours in order to ascertain the time granularity of spectrum usage states. Thus, in distributed spectrum sensing, each SU receives and processes the spectrum occupancy data contributed by its neighbors to obtain more accurate spectrum occupancy information. However, there are certain hindrances that limit the performance of this novel specrum sensing approach.

Such hindrances include the challenges associated with energy limitations and partial network failures, which result is spectrum occupancy measurements from all SUs not being available at all times [193]. These challenges may include, among others, (i) imperfect/poor conditions of the reporting channel due to wireless channel fading [194], whereby the local detection performance of SUs become unreliable [195]; (ii) the use of energy-saving collaborative spectrum sensing techniques whereby SUs with no information to transmit in the next time slot may decide not to collaborate their spectrum sensing observations [196]; and (iii) byzantine attacks where network adversaries may launch malicious attacks resulting in SSD falsification. [197].

Because of the above named challenges, the overall quality of distributed spectrum sensing performance deteriorates and hence inadequate to provide sufficient and direct the occupied/unoccupied states of channels. If these challenges are not well handled, they may result in wrong statistical inferences that may lead to the interruption of PU activities and consequently reduce the spectral efficiency of SUs. As a way of circumventing this problem, new motivations were proposed in order to recover the whole SSD matrix from a sampling of the available data entries and obtain accurate spectrum occupancy states. In this way, qualitative and precise information regarding the current spectrum occupancy states could be obtained, as well as accurate inferences on the current SU context could be made. As a result, numerous studies were conducted for spectrum occupancy reconstruction problems in centralized, decentralized and distributed collaborative CRN architectures. In all these architectures, the most prevalent spectrum occupancy reconstruction technique was matrix completion, which is a low-rank matrix completion method utilizing SVD in processing, learning and imputing the missing entries under the low-rank constraint assumption [198].

In the centralized collaborative spectrum sensing approach, which is infrastructure-based, SUs individually observe the concerned radio spectrum bands through spectrum sensing, then sends their spectrum observations to the fusion center (FC). The FC uses a combining technique to process and analyse the spectrum observations contributed by individual SUs, then takes a global decision that regards the occupancy status of the spectrum bands concerned. This information is then relayed back to the SUs [199]. In the case of missing observations, it is not difficult to perform spectrum occupancy reconstruction using the centralized architecture. When encountered with the problem of missing entries, the FC performs the spectrum occupancy reconstruction task by leveraging the low-rank nature of the SSD matrix using the general matrix completion method.

Authors in [200] also used matrix completion to perform spectrum occupancy reconstruction and impute the missing entries using the available entries collaborated by a subset of SUs. In order to model an incomplete SSD matrix, the authors had to make sure that the FC receives spectrum observations from a subset of the SUs in the CRN. This was done by allowing some SU collaborations to fail, while allowing others to go through. The problem was formulated as a joint-sparsity reconstruction problem, and the reconstructed SSD matrix, the status of the channels were obtained using matrix completion. A similar approach was applied in [201] where the SSD matrix was assumed to be correlated and with the low-rank constraint. Numerous other research contributions were proposed to solve the centralized spectrum occupancy reconstruction problem and interesting results were obtained, for example, the OR fusion rule technique in [202], the compressive sensing technique in [203], and the multi-dimensional correlation technique in [204].

In the distributed spectrum sensing approach, which is an infrastructure-less scheme, SUs individually observe the radio spectrum bands and share their observations among themselves.

Distributed spectrum sensing is a particularly useful approach for wide area ad hoc CRNs where only a subset of SUs can detect the energy from PU transmissions and others would benefit from the collaborated spectrum sensing results [205]. In this way, the spectrum occupancy information in one location is shared with nearest neighbors such that even remote SUs who obtain that knowledge can be able to spatially reuse any unoccupied spectrum bands. However, when encountering the problem of missing spectrum observations, dealing with them in a completely distributed approach is always tricky, because, as opposed to the centralized architecture, SUs have to perform the spectrum occupancy reconstruction individually. This means that, each SU has to perform some data mining into the SSD and accurately impute the missing data entries in the SSD matrix in order to obtain accurate spectrum occupancy states. As a result, the case of missing value imputation in the distributed scenarios requires high computational capabilities per SU, hence it has not been extensively studied.

However, there have been ways that were introduced in order to alleviate the computational burden from SUs. A support fusion-based distributed compressive sensing approach was proposed in [206], where a local compressed reconstruction and adaptive learning of support knowledge among SUs was applied. The novelty of this approach is that, during each iteration of the proposed algorithm, the local sparse spectrum occupancy reconstruction would be achieved via a truncated l_1-minimization technique and then incorporated the support information from previous iterations. In this way, each SU would obtain local support detection via a thresholding of its local reconstruction and then exchanged this information with its nearest neighbors. After several rounds, each SU would also receive fused support information from the other SUs in the CRN and in this way the spatial diversity and reliable spectrum occupancy information would be obtained.

The distributed spectrum occupancy reconstruction problem was also approached from the empirical measurements perspective and several contributions have been published. In [207] the problem was modeled by characterizing spectrum maps via spatial statistics and MRFs, where the mathematical premises presented outlined the generation of useful statistics from measurement data. In [208], the authors used MRFs to model spectrum occupancy where the spatial correlation of spectrum usage was described using spatial statistics and a semi-variogram. Multiple wireless technologies were then used to extract extra parameters in real-life measurement scenarios in order to give better insights regarding the extraction of useful spectrum sensing parameters.

In addition, simulation studies were conducted on distributed spectrum occupancy reconstruction using

MRFs in [209], where the entire CRN was modeled as a plenary grid. Here, the joint distributions of spectrum occupancy was modeled by employing both one-dimensional and two-dimensional Ising models and the phase transitions of CRN connectivity were studied via the random fields model. Then, the spectrum occupancy reconstruction was performed using matrix completion and the obtained results indicated the prominence of Ising models in the prediction of network connectivity. Another study that used multiple dimensions (i.e., 2D and 3D) was reported in [210], where a spectrum situation reconstruction problem was considered in which the spectrum occupancy problem was viewed as an image recovery problem. The spectrum occupancy reconstruction was solved using total variation inpainting, which takes advantage of SSD correlations in multiple dimensions. This technique proved to be effective compared to the belief propagation technique which considers only the two-dimensional orientation. The practical applicability of the discussed techniques depend on the amount of available information that each SU could acquire regarding the state of spectrum occupancy.

In all the studies discussed above, it can be observed that the evaluation of both the spectrum occupancy and spectrum occupancy reconstruction problems was considered using either probabilistic or statistical approaches. In as much as probabilistic and statistical techniques offer useful insights in such problems, they are often limited to the assumptions that are required in their derivations. As a result, when using these approaches in spectrum occupancy reconstruction problems, one has to first determine whether the problem to be solved consists of random variables or random processes. ML techniques on the other hand have received much attenction as powerful tools to solving problems of this nature. Even though ML techniques are also probabilistic and statistical in nature, due to their heuristic background, they do not require any assumptions on the data or even any prerequisite information about the problem to be solved. But, they just proceed to give better results compared to their probabilistic and statistical counterparts. However, there has been very little work done in spectrum occupancy using ML strategies, one example being a contribution on spectrum occupancy variation and traffic prediction in [211].

The most efficient ML technique is spectrum occupancy reconstruction problems is DL owing to its use of networks capable of learning unsupervised from unstructured and unlabeled data through the use of a hierarchical level of ANNs [212]. However, there are a few notable applications of DL strategies recently, including PU classification studied in [213], where interesting results with improved accuracy in PU agents' classification and recognition were obtained through the use of DBNs. Also in [214] PU modulation classification was studied as a way of gaining knowledge of the different technologies used by PUs. DBNs were applied for pattern recognition and classification in automated modulated classification, which achieved high accuracies regarding modulation detection and classification in the presence of noise. As a results, high accuracy classification was obtained for 4 Frequency Shift Keying, 16 Quadrature Amplitude Modulation, Binary Phase Shift Keying, Quadrature Phase Shift Keying techniques in various environments.

In addition, the authors in [215] used spectrum sensing, DL and dynamic optimization in designing a high-throughput SU system that will effectively coexist with PUs. Their proposed method involved training a deep CNN model that performs classification directly from spectrograms. The detection of PU scenarios was treated as an image classification problem and was solved using CNNs to obtain calibration tables for SU transmitters.

3.2.1 Motivation

In recent developments in spectrum occupancy reconstruction, matrix completion-based strategies have indicated the possibiltity of reconstructing the entire spectrum SSD from a few available in [216]. Using matrix completion, it has been proved that if the SSD matrix that needs to be reconstructed adheres to the low-rank constraint and also satisfies the incoherence property, it can be reconstructed accurately from just a few available entries [217]. However, it has also been observed that the matrix completion approach achieves lower specrtum occupancy reconstruction resolution owing to the use of the standard SVD, which performs well for general matrices.

Therefore, the motivation of this study is to solve the spectrum occupancy reconstruction problem by using stochastic matrix factorization instead of the standard SVD method. The stochastic matrix factorization technique that is proposed in this study reconstructs the SSD matrix using scaled SGD, which is an algorithm used in DL that learns through the BP technique. As discussed in [218], SGD is a prominent strategy for training DL algorithms using the stochastic approximation of a gradient optimization, and has been proven to generalize well [219], and converges to points of similar loss for a given size of network [220]. As a result, this work uses a DL strategy to find a new way of spectrum knowledge discovery.

3.2.2 Summary of Contributions

The method proposed in this chapter seeks to provide: (i) accurate link physics models that precisely model and characterize the set of distributed CRN link states, and (ii) a distributed and low complexity algorithm that dynamically optimizes the performance of spectrum occupancy reconstruction in CRNs. The summary of contributions for this chapter are discussed as follows:

- Since communication links need to be available for SUs to collaborate on their spectrum sensing results, a mathematical framework in which these can be modeled as physical systems has been developed based on the MRF model. An energy model that characterizes the communication links between SUs in an MRF using link energy functions founded on the canonical Ising model is introduced. The canonical Ising model is used to study the phase transitions of SU observations using the electron spin model [$\uparrow$, $\downarrow$] [221], which are used to represent the binary logic (i.e., the [1, 0] or ON/OFF channel status) of spectrum observations. Then, the CRN network connectivity regarding interactions between SUs have been found through the use of molecular models and were characterized by the potential energy stored in the bonds that form a molecule. The initial spectrum observations of spectrum occupancy for the SSD matrix were created using the Metropolis-Hastings algorithm and then the sampling of spectrum observations is done using the Gibbs bivariate exponential distribution. Then, the creation of the spectrum occupancy data matrix with missing values was accomplished by allowing some SU contributions to fail while others were allowed to succeed.
- Then, the whole spectrum occupancy reconstruction problem was formulated as a matrix factorization problem and the applicability of the SGD algorithm was subsequently explored in solving and reconstructing the SSD matrix. The proposed SGD algorithm was scaled in order to resolve the problem of being trapped in the local minimum and this was also assisted by scaling the spectrum occupancy data matrix in small ranks which deals with the convergence problems associated with SVD. This whole technique resembles computational DL owing to its capability to allow for some addition of depth into the SSD matrix, which is a very important step in discovering the missing matrix entries. Thus, the use of SGD in improving the accuracy of spectrum occupancy reconstruction and the convergence speed of missing value imputation. the results obtained in this chapter indicate the prominence of the SGD algorithm over the traditional SVD algorithm.

3.3 SYSTEM MODEL

3.3.1 The Proposed CRN Model

In this section the CRN model and topology assumptions regarding SU pairwise interactions are introduced for the proposed spectrum occupancy reconstruction problem. Considering an SU-PU coexistence in a CRN where there are K SUs randomly deployed, forming a two-dimensional Poison point process as shown in Figure 3.1 below:

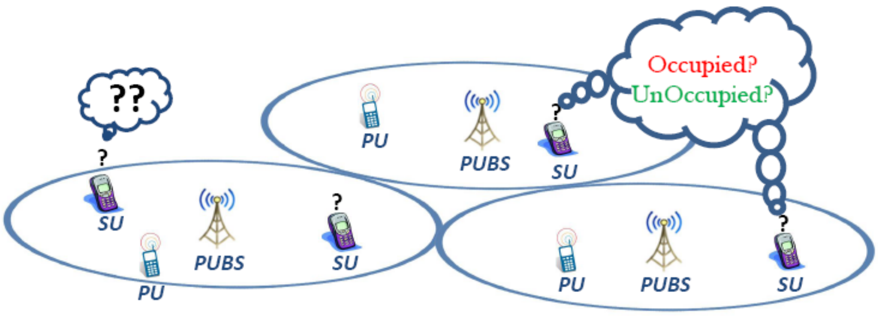

Figure 3.1. Illustration of a CRN Showing a Random Deployment of SUs and PUs.

Figure 3.1 above shows SUs and PUs, where SUs perform spectrum sensing in order to share their spectrum sensing observations with one another and safely access the licensed PU spectrum. PUs are assumed to interrupt only a small area in order to enable for the application of MRFs. This is to allow for the use of the first-order Markov property that comes with the MRF and Ising model, which hold rigorously in small PU interruption ranges. Moreover, a time-slotted CRN system is assumed such that spectrum occupancy observations are shared every time slot.

3.3.2 The CRN Topology, Pairwise Interactions and Communication

An MRF model is used to represent the CRN topology, where SUs the distributed on a plenary grid can be factorized into a cluster within a 1500 m $\times$ 1500 m geographical area. This area is further divided into 50 $\times$ 50 sub-areas, resulting in a 30 m $\times$ 30 m square area, as shown in Figure 3.2 below:

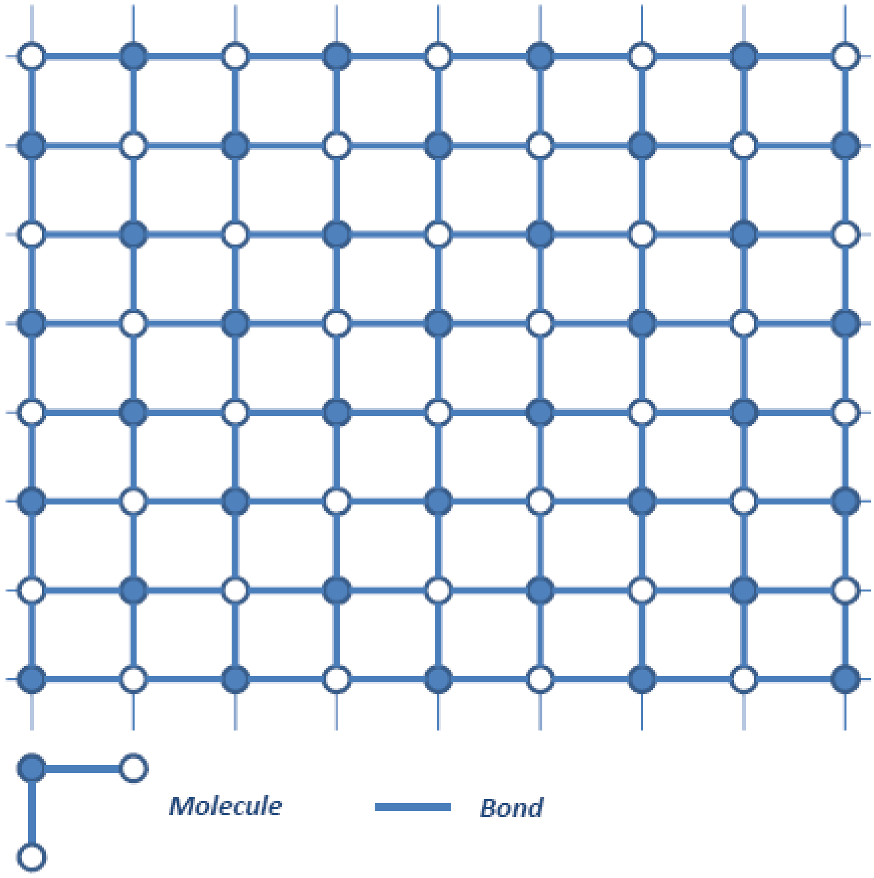

Figure 3.2. CRN Topology on an 1500 m $\times$ 1500 m Area. Solid circles represent tagged SUs, while their neighbors are represented by open circles and connected by edges to exhibit two-point cliques.

In Figure 3.2 above, the solid circles represent the tagged SUs while their neighbors are represented and the edges connecting these exhibit two-point cliques. In this case, the prominent features of this CRN model can be realistically captured by a MRF model such that the whole CRN resembles magnetic lattice. Thus, in this representative model, each SU has four nearest neighbors positioned on the four immediately ajacent grids and their interactions are represented as edges called two-point cliques. It is assumed that when SUs share their spectrum occupancy observations, they exchange electromagnetic energy, which justifies the use of the magnetic lattice topology. The spatial correlation of the spectrum occupancy is represented using the classical Ising model, where SUs, their neighbors, their cliques and interactions are modeled as molecules. In this case, the spatial distribution of this electromagnetic energy used in communication together with its temporal variations are used to model the flow of energy between neighboring SUs.

This type of representation has been adopted from [223] where each grid point in the lattice possesses either a spin-down ($\downarrow$) or a spin-up ($\uparrow$), representing the binary logic of channel status. This analogy is representative of what is illustrated in Figure 3.2 above where the small circles indicate the SU positions the same way in which lattice points are arranged on the surface of a square ferromagnet. The idea behind this is to realized the pairwise interaction between SUs using an undirected graph $\mathcal{G}(\mathcal{V},\mathcal{E})$, where $\mathcal{V}$ denotes the set of K SUs, where $v \in \mathcal{V}$ represent position of each SU node, $\mathcal{E}$ denotes the set of all connections between SUs, where $e \in \mathcal{E}$ defines pairwise interactions between SUs forming a social network that defines the collaboration paths in the CRN.

3.4 PROBLEM FORMULATION

Assuming that each SU observes a single channel at any given time slot, then each grid represents one of the two possibilities, either a $\downarrow$ or $\uparrow$. This is such that at a specific time slot, the series of grid points detail the spatio-temporal spectrum occupancy. It is also assumed that SUs are honest about their observations and share their spectrum sensing observations in such a way that each SU constructs its own SSD and when faced with the missing value problem it performs an individual spectrum occupancy reconstruction. Thus, in the context of spectrum occupancy in CRNs, a spectrum situation matrix is defined as $\mathbf{X}$, where the matrix entries denoted as $X_{k,t}$ represents the spectrum observation entry contributed by the k^{th} SU at time slot t and occupies the matrix entry position (k,t). The spectrum situation matrix $\mathbf{X}$ is thus composed of T time slots, resulting in the SSD represented by the random process $\mathbf{X}^{K\times T}$, given as follows:

$$\mathbf{X} = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,T} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,T} \\ \vdots & \vdots & \ddots & \vdots \\ X_{K,1} & X_{K,2} & \cdots & X_{K,T} \end{bmatrix}, \tag{3.1}$$

which represents the SSD matrix, where $X_{k,t} \in [0,1]$, $k = 1,2,\cdots,K$ and $t = 1,2,\cdots,T$ represent the sensing reports from the k^{th} SU at sensing slot t. Then, considering a probability distribution on $\mathbb{X} = \{0,1\}^{\mathcal{V}}$ where the variables take the form 0 and 1, the random variable X stands for spectrum occupancy observation contributed by each SU located at the vertex $v \in \mathcal{V}$ of the graph $\mathcal{G}$.

3.4.1 Causes of Missing Spectrum Sensing Results

Even though the SSD gives a more complete view of the spectrum usage pattern and deeper understanding of some hidden patterns behind channel state evolution and usage, some SSD are inherently uncertain because of challenges related to channel conditions as well as energy-saving spectrum sensing schemes.

Reporting Channel Conditions: This situation usually occur due to poor reporting channel conditions such that SUs are not able to detect the SSD sent at time slot t. As a result, the SSD matrix entry $X_{k,t}$ is dependent on the reporting channel condition and independent on the other missing values as well as the available entries such that $X_{k,t} = 0$. Thus, the missing value mechanism is defined as missing completely at random (MCAR).

Collaborative Spectrum Sensing Schemes: This occurrence usually manifest when some SUs in the CRN assume energy-saving spectrum sensing schemes. In this case, the SSD matrix entry $X_{k,t}$ depends only on the transmission probability of that particular SU and not on the values of the SSD matrix. Thus, the missing SSD entry is dependent on the value of $X_{k,t} = 0$ such that the missing value mechanism is defined as missing not at random (MNAR). This means that in the case of MNAR, the missing values are known and they can be imputed by $X_{k,t} = 0$. Thus, the problem remains with the MCAR mechanism and the focus of this chapter is on the pre-processing of SSD matrix that is MCAR, which, according to [224] is a strong case of missing at random (MAR) because the missingness of the data is still random but it is due entirely to the observed variables. Therefore, the SSD matrix $\mathbf{X}$ in (3.1) can be rewritten as follows:

$$\mathbf{X} = \begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} & \cdots & X_{1,T} \\ X_{2,1} & X_{2,2} & \emptyset & \cdots & X_{2,T} \\ X_{3,1} & \emptyset & X_{3,3} & \cdots & X_{3,T} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ X_{K,1} & \emptyset & X_{K,3} & \cdots & X_{K,T} \end{bmatrix}, \tag{3.2}$$

where there are missing entries at matrix positions $X_{3,2}$, $X_{2,3}$ and $X_{K,2}$ represented by $\emptyset$.

3.4.2 The Optimization Problem

The objective of spectrum occupancy reconstruction, as it is with missing value imputation, is actually not to find what the missing value is, but rather to restore the important characteristics of the SSD set as a whole by accurately estimating the missing SSD matrix entries [225]. Thus, at this point, it is safe to assume that the spectrum observations contributed by SUs are dependent on one another since the SSD is generated based on the same ground truth, which is the status of the PU channels. Assuming that the spectrum reconstruction is performed within the spectrum sensing phase of a single cognitive cycle so that it does not impact negatively on the performance, a matrix factorization technique is introduced which will approximate the SSD matrix $\mathbf{X}$ using a product of two smaller matrices $\mathbf{Y}$ and $\mathbf{Z}$ in a way that the SSD matrix $\mathbf{X}$ is accurately reconstructed from these smaller matrices, as follows:

$$\hat{\mathbf{X}} \approx \mathbf{Y}\mathbf{Z}^T, \tag{3.3}$$

where the smaller matrix $\mathbf{Y} \in \mathbb{R}^{k \times r}$ is the component matrix whose columns $\{Y_{:,t}\}_{t=1}^{r}$ are the vertices in a hypercube $[0,1]^k$ and the other smaller matrix $\mathbf{Z} \in \mathbb{R}^{r \times t}$ is the matrix of coefficients whose distribution does not depend on the available or the missing entries. This phenomena is as illustrated in Figure 3.3 below.

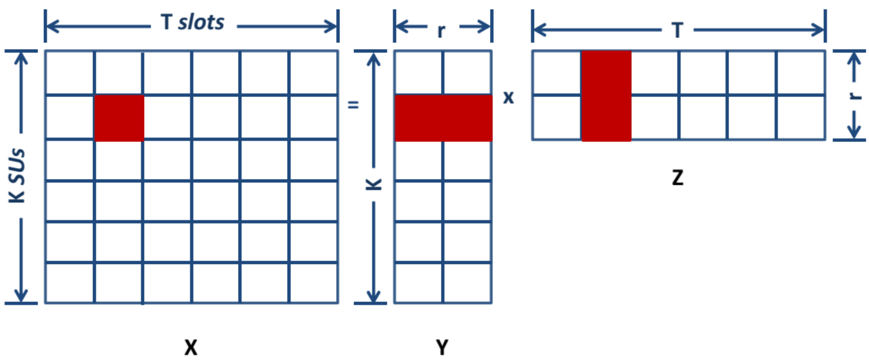

Figure 3.3. An Illustrative Example of a Matrix Factorization Approach.

In the context of the SSD illustrated in Figure 3.3 above, the matrix $\mathbf{X}$ represents the partially observed SSD matrix; where $\mathbf{Y} \in \mathbb{R}^{|K| \times r}$ denotes a sub-matrix where each row k represents a vector containing r latent factors where $r << |K|, r << |T|$ describing the contributions made by SU k; and $\mathbf{Z} \in \mathbb{R}^{|T| \times r}$ denotes the sub-matrix containing r latent factors describing time slot t. Therefore, both $\mathbf{Y}$ and $\mathbf{Z}$ can be referred to as latent matrices that can be learned through the optimization of an objective function given the criteria of the RMSE, given in (3.16). Therefore, the factorization of $\mathbf{X}$ into the product $\mathbf{YZ}$ entails an approximation problem that seeks to find

$$\mathbf{Y} \in \{0,1\}^{k \times r}, \qquad \mathbf{Z} \in \mathbb{R}^{r \times t}, \qquad \mathbf{Z}^T \mathbf{e}_r = \mathbf{e}_t, \qquad r \ll \min\{k,t\}, \tag{3.4}$$

where the term r is the optimal rank assumption implying that the matrix rank constraint correlates the known entries with the unknown ones, $\mathbf{Z}^T \mathbf{e}_r = \mathbf{e}_t$ denotes that the columns of matrix $\mathbf{X}$ are affine as opposed to being linear combinations of the columns of $\mathbf{Y}$, which is also known as the signal direction. This assumption is imposed in order to avoid the origin from being treated differently from the other vertices of the hypercube $[0,1]^k$, therefore, the objective function is represented as a fixed-rank optimization problem as follows:

$$\hat{\mathbf{X}} = \min_{\mathbf{X}\in\mathbb{R}^{k\times t}} ||\mathbf{X}-\mathbf{YZ}||, \qquad \text{subject to} \qquad \text{rank}(\mathbf{X}) = r, \tag{3.5}$$

where the rank constraint r is from the assumption that the optimal matrix rank has to be known a priori.

3.5 PROPOSED SOLUTION FORMULATION

In this section, the SSD matrix is synthesized using a generative modeling technique based on probabilistic DL discussed in [226], where a time-series spectrum occupancy prediction technique that uses DBNs is obtained. This is a kind of probabilistic generative NN composed of an RBM as discussed in Chapter **2**. Here, the RBM is used to capture the input feature space of the spectrum occupancy data from an MRF using link energy functions from the Ising model. Thus, this model captures the underlying distribution of the data as well as the mechanisms used to generate synthetic data during its generation process.

Therefore, the probability distribution of SUs can be factorized into a cluster within 1500 m $\times$ 1500 m geographical area leading to a Bayesian network representation of a CRN. Then, the 1500 m $\times$ 1500 m geographical area is divided into a 50×50 grid and an SU cluster is formed by a collaborating group of SUs located at each of the grid points such that after pre-training of the RBM using their energy functions, a spectrum occupancy data matrix is obtained using the Metropolis-Hastings algorithm. To decide on the size of the samples and learning rates, Gibbs sampling is adopted during the training process and SGD is used to process and learn the missing entries.

3.5.1 Ising Model and Markov Random Fields

In a sample space Ω, the Ising model defines a probability measure on the set of all possible configurations defined as a collection of all possible realizations of random variables, X, which stands for the CRN spectrum occupancy. This is such that an energy function corresponding to a specific spectrum occupancy $X = \{x_0, \cdots, x_i, \cdots, x_n\} \in \Omega$ can be defined in terms of the Hamiltonian

$$\mathcal{H} = -I \sum_i x_i - J \sum_{i<j} A_{ij} x_i x_j, \tag{3.6}$$

where $I \sum_i x_i$ describes the effect of the exogenous inputs, where I represents the external effects and $J \sum_{i<j} A_{ij} x_i x_j$ represents the endogenous effect such that $\sum_{\langle i,j \rangle} x_i x_j$ means that the exchange of energy is counted only for neighboring SUs, where J represents the energy exchange between neighboring SUs. Here, each variable x_i and x_j represent a grid in the topology and has two possible values representing the configuration of the spectrum occupancy states, i.e., $x_i = -1$ denotes an occupancy state while $x_i = 1$ represents an idle state. Thus, the terms I and J are two parameters of the Ising model such that the Ising model can intuitively address the fact that the spectrum occupancy in the CRN depends on the natural (i.e., exogenous) and manufactured (i.e., endogenous) inputs.

The second summation in (3.6) is taken over all pairs i, j of points that are neighbors such that the Ising model makes the simplifying assumption that only interactions between neighboring points need to be taken into account, which is referred to as a first order Markovian property. The term A_{ij} is the entry of the adjacency matrix indicating the existence of an edge $e \in \mathcal{E}$ between SU_i and SU_j forming a full duplex communication between neighbors represented by $A_{ij} = A_{ji} = 1$, and 0 otherwise.

3.5.2 Markov Random Fields and Gibbs Distribution

In order to solve this problem, the SU observations of the ON/OFF behavior of the PU channel are modeled using the Gibbs function, where it is assumed that each SU positioned at each vertex $v \in \mathcal{V}$ of the random graph $\mathcal{G}$ exhibits the ON/OFF behavior through spectrum sensing results $X^{k \times t}$. Also assuming that the spectrum occupancy is based on the $[1,0]$ binary logic on a two-dimensional lattice modeled using the electron spin model, the intrinsic angular momentum on the magnetic field with two orientations. Therefore, combining these magnetic orientations with the ON/OFF states' binary logic yields a CRN system with two states.

However, given the random variables on an MRF, determining the probability of any labeling is not trivial because determining what set of conditional probabilities to use on an MRF is not straightforward. Also, an arbitrary set of conditional probabilities for different sites and neighborhoods may not be mutually consistent such that it is not obvious how to determine this. In this case, the the most likely MRF labeling can be made easier through the use of Gibbs distributions, which turn out to be equivalent to MRFs and much more easier to manipulate since the cliques capture the dependencies between neighborhoods [227]. Therefore, at this point, a set of sites $\{v_1, v_2, \cdots, v_k\}$ that form a clique is defined, given a probability distribution defined for a set of sites and labels if the distribution takes the form:

$$P(X) = \frac{e^{-E(X)}}{\sum_{x \in X} e^{-E(X)}}, \qquad \text{where} \qquad E(X) = I \sum_{i=1}^{N} x_i + J \sum_{\langle i\ j \rangle} x_i x_j, \tag{3.7}$$

where the denominator denotes the normalizing value that is required to make all probabilities sum up to 1, and $E(X)$ denotes the energy function. Using the idea of PU interruption ranges defined earlier, which allows for the use of some useful properties that can be quantified by the Ising model [207], Gibbs distributions are employed to allow the MRF probability distribution to be written in the form of (3.7) above. Now, one can be able to define the sample space Ω for specific spectrum occupancy such that the random variable $X \in \Omega$ be assigned to a probability according to the probability distribution defined by the Gibbs distribution in [227] for the two-dimensional Ising model.

Now, suppose that the system can be changed from a state X to the state $X^{'}$ by toggling the value of state X_j, the resulting transition function can be represented as follows:

$$T(X \rightarrow X^{'}) = \frac{P(X^{'})}{P(X)} = e^{-(E(X^{'})-E(X))} = e^{-\Delta E(X)}, \tag{3.8}$$

can be used, where the transition function $T(X \rightarrow X^{'})$ denotes the probability that the system transitions from state X to $X^{'}$, the term $\Delta E(X)$ represents the change in system energy when a transition from X to $X^{'}$ is made. Therefore, in order to be able to sample spectrum occupancy data from the whole 1500 m $\times$ 1500 m CRN, initial states for each SU observation are artificially created, an this becomes the initial state of the CRN. Then, after combining the Gibbs distribution function $P(X)$ with the transition function $T(X)$ one needs to specify the amount of time that each state $X \in [0,1]$ has to last in order to avoid the time series from toggling between states.

To achieve this, the Metropolis-Hastings algorithm is used, based on the idea that it can search the states through an ergodic Markov chain and use the transition function $T(X \rightarrow X^{'})$ that satisfies the relation [228], given as

$$P(X)T(X \rightarrow X^{'}) = P(X^{'})T(X^{'} \rightarrow X), \tag{3.9}$$

which expresses the idea of state equilibrium in the reversible transition $X \leftrightarrow X^{'}$. However, the initial state of the system may be far from the equilibrium that is being sought when the distribution function $P(X)$ is large. This means that during the simulation procedure, the algorithm has to go through a number of steps at first before measurements can be taken (i.e., before reaching the burn-in period). Since equilibrium is an essential part of the process, in order to apply the Metropolis-Hastings algorithm, a transition function that satisfies the balance equation must be found. In order to achieve this using the canonical Ising model, suppose that the initial sampled state is X^t, and later proceeded by a new proposal state X^* are drawn with the probability $Q(X^*|X^t)$ and then the ratio α is computed as follows:

$$\alpha(X^t, X^*) = \frac{P(X^*)Q(X^t|X^*)}{P(X^t)Q(X^*|X^t)}. \tag{3.10}$$

Here, the quotient $P(X^*)/P(X^t)$ denotes the likelihood between the proposed sample X^* and the initial/previous sample X^t, while the quotient $Q(X^t|X^*)/Q(X^*|X^t)$ represents the ratio of the proposal density in two directions. Each time the new state X^{t+1} is chosen according to the following rules: when $\alpha \geq 1$, one assigns X^* to X^{t+1} and when $\alpha < 1$, $X^{t+1} = X^*$ with probability α, and $X^{t+1} = X^t$ with probability $(1-\alpha)$. Thus, once the I and J for the Ising model are specified, one can obtain the sequence of random samples from (3.7) above, which represents the spectrum occupancy during a period of time. This procedure is outlined in Algorithm **1** below.

3.5.3 Modeling Incomplete Spectrum Sensing Data

Using the Metropolis-Hastings algorithm discussed above, the spectrum sensing data can be sampled from the whole CRN and the incomplete SSD can be modeled by allowing SUs to share their spectrum sensing results. Here, some collaborations are allowed to succeed while other collaborations made to fail so that each SU has an incomplete SSD matrix. The number of failed collaborations can be varied depending on the amount of missing entries required to test the spectrum occupancy reconstruction algorithm. Then, given that the observed SSD matrix $\mathbf{X}$ is incomplete, each SU has the task of reconstructing it by imputing the missing entries of the observation matrix (3.1) in a way that resembles a matrix factorization problem depicted as follows:

$$\left[\begin{array}{c|c|c|c|c|c} & & & & & \\ \hline & ?_{2,2} & & & & \\ \hline & & & & & \\ \hline & & & x_{i,t} & & \\ \hline & & & & & \\ \hline & & & & & x_{k,t} \end{array}\right] \approx \left[\begin{array}{c} \mathbf{y}_1^T \\ \hline \mathbf{y}_2^T \\ \hline \vdots \\ \hline \mathbf{y}_i^T \\ \hline \vdots \\ \hline \mathbf{y}_k^T \end{array}\right] \times \left[\begin{array}{c|c|c|c|c|c} \mathbf{z}_1 & \mathbf{z}_2 & \cdots & \mathbf{z}_r & \cdots & \mathbf{z}_t \end{array}\right]. \tag{3.11}$$

Algorithm 1 Procedure for Generating Spectrum Occupancy Using the Metropolis-Hastings Algorithm.

Inputs: Number of SUs K, Number of time slots T

Outputs: Incomplete SSD matrix $\mathbf{X}$

Begin with a complete graph with un-directed edges between each node pair

Initialize iteration $t = 1$

For $t < T$ **do**

    Each SU collects spectrum sensing measurements to generate initial value u, and

Set $X{-}^{t} = u$ and broadcast to one-hop neighbors

    Draw a sample X^* from a proposal distribution $Q(X^*|X^t)$.

    Evaluate acceptance probability [229] $\alpha = \min\left\{1, \frac{P(X^*)Q(X^t|X^*)}{P(X^t)Q(X^*|X^t}\right\}$

    Generate u from a uniform distribution [0, 1]

    If $u \leq \alpha$ **then**

        Accept X^* as the next sample X^{t+1} with probability $\alpha(X^t|X^*)$ and keep X^t as the next sample X^{t+1} with probability $1 - \alpha(X^t|X^*)$

    Else

        Set $X^t = X^{t-1}$

    End If

    Each SU has an SSD matrix

    Update iteration number $t = t + 1$

End For

Equation (3.11) above shows the incomplete spectrum occupancy observation matrix $\mathbf{X}$, where $\mathbf{X}$ is being approximated with a product of two smaller matrices $\mathbf{Y}$ and $\mathbf{Z}$. The matrix entries $x_{i,k}$ and $x_{m,k}$ are MNAR, while entries denoted by $?_{2,2}$ are assumed to be of the MCAR type. The reconstruction of SSD matrix $\mathbf{X}$ can be performed is by applying a data mining technique that approximates $\mathbf{X}$ from the product of $\mathbf{YZ}$ with high resolution of correctness. Then, using the objective function in (3.5), one minimizes the cost function defined by the difference $\mathbf{X} - \mathbf{YZ}$ subject to the matrix rank constraint r.

3.5.4 Spectrum Occupancy Reconstruction Using a Missing Data Imputation Technique

In this subsection, a scaled iterative procedure called the SGD is used to reconstruct the incomplete spectrum occupancy data matrix $\mathbf{X} \in \mathbb{R}^{k \times t}$, which will be scaled by a rank r. The matrix factorization is performed to obtain two matrices $\mathbf{Y} \in \mathbb{R}^{k \times r}$ and $\mathbf{Z} \in \mathbb{R}^{r \times t}$ such that $x_{i,t} \approx \mathbf{y}_i^T \mathbf{z}_t$, $\forall i,t \in I$ with a probability closest to 1. Here, the term I now denotes the indices of the existing elements in matrix $\mathbf{X}$, such that the reconstruction problem can be represented using the following convex form:

$$\hat{\mathbf{X}} = \min_{Y,Z} \frac{1}{2} \mathcal{P}_I ||\mathbf{X} - \mathbf{YZ}||_F^2, \tag{3.12}$$

which is one half of an averaged squared Frobenius norm of (3.5), and the term I denotes the complete set of indices $(i,j) \in I$ such that $\{(i,j) : i \in \{1,\cdots,k\}, j \in \{1,\cdots,t\}\}$, such that the operator $\mathcal{P}_I(\mathbf{X}_{i,j})$ behaves as

$$\mathcal{P}_I(\mathbf{X}_{i,j}) = \left\{ \begin{array}{lll} X_{i,j}, & \text{if} & (i,j) \in I \\ 0, & \text{otherwise} & \end{array} \right\}. \tag{3.13}$$

In the stochastic set-up of the SGD for solving (3.12), β known entries are picked at a time and then a GD step that updates the matrices $\mathbf{Y}$ and $\mathbf{Z}$ is taken. Because of the cost structure, this means that the technique ends up updating only a maximum of β rows of $\mathbf{Y}$ and $\mathbf{Z}$ at a time. The spectrum occupancy reconstruction procedure is outlined as a missing value imputation technique in Algorithm **2** below.

3.5.5 SGD Algorithm Description

When performing the SSD matrix update, it is better to let $\beta_{\mathbf{Y}}$ rows of $\mathbf{Y}$ and $\beta_{\mathbf{Z}}$ to be updated when β known entries are picked, where $\beta_{\mathbf{Y}} \leq \beta$ and $\beta_{\mathbf{Z}} \leq \beta$ [230]. This is shown in **Step 05**. Then, in **Step 06**, the sub-matrices need to be defined; firstly let $\mathbf{Y}_\beta$ be the sub-matrix corresponding to $\mathbf{Y}$ with $\beta_{\mathbf{Y}}$ rows. Similarly, let $\mathbf{Z}_\beta$ be the sub-matrix corresponding to $\mathbf{Z}$ with $\beta_{\mathbf{Z}}$ rows. At each iteration step β known entries are picked such that one arrives at a sub-problem of completing matrix $\mathbf{X}_\beta$ of size $\beta_{\mathbf{Y}} \times \beta_{\mathbf{Z}}$.

Therefore, having β known entries at indices I_β, which need to be approximated by $\mathbf{Y}_\beta \mathbf{Z}_\beta^T$. In **Step 07**, a residual matrix for updating the SGD algorithm is defined and denoted by $\mathbf{S}$. Therefore, if $\mathbf{S}_\beta$ is the residual matrix of this sub-problem, then the partial derivatives at $(\mathbf{Y}_\beta, \mathbf{Y}_\beta)$ are $(\mathbf{S}_\beta \mathbf{Z}_\beta, \mathbf{S}_\beta \mathbf{Y}_\beta)$.

Algorithm 2 Proposed SGD method for Distributed Spectrum Occupancy Reconstruction in CRNs.

Inputs: $\mathbf{X} \in \mathbb{R}^{k \times t}$: step size, Δ: weight factor, ϖ, matrix rank, r, learning rate, α_t

Outputs: $\hat{\mathbf{X}} \in \mathbb{R}^{k \times t}$

01: Partition $\mathbf{X} \in \mathbb{R}^{k \times t}$ into $\mathbf{Y} \in \mathbb{R}^{k \times r}$ and $\mathbf{Z} \in \mathbb{R}^{r \times t}$

02: Initialize step size Δ and weight factor ϖ.

03: Find indices for non-missing entries

04: **While** termination condition not reached **do**

05: Pick β known entries with their indices.

06: Set up a completion sub-problem by finding the indices corresponding to the sub-matrices $\mathbf{Y}_\beta$ and $\mathbf{Z}_\beta$ which need to be modified. Consequently, find the subset I_β of indices out of the $\beta_\mathbf{Y}\beta_\mathbf{Z}$ indices.

07: Compute the residual $\mathbf{S}_\beta = \mathcal{P}_{I_\beta}(\mathbf{Y}_\beta\mathbf{Z}_\beta - \mathbf{X}^*_\beta)$.

08: Given a step size θ, update $\mathbf{Y}_\beta$ and $\mathbf{Z}_\beta$ as $\mathbf{Y}_{\beta+1}$ and $\mathbf{Z}_{\beta+1}$.

09: Fix $\mathbf{Y}$, find $\mathbf{Z}$ that minimizes $\frac{1}{2}||\mathbf{X} - \mathbf{Y}\mathbf{Z}||_F$

10: Fix $\mathbf{Z}$, find $\mathbf{Y}$ that minimizes $\frac{1}{2}||\mathbf{X} - \mathbf{Y}\mathbf{Z}||_F$

11: Update $\mathbf{Y}^T\mathbf{Y}$ and $\mathbf{Z}^T\mathbf{Z}$

12: Compute the MSE using (3.16).

13: **If** termination condition is reached

14: **Go to** 18:

15: **Else**

16: **Go to** 04:

17: **End If**

18: **End While**

Here, $\mathbf{S}_\beta = \mathcal{P}_{I_\beta}(\mathbf{Y}_\beta\mathbf{Z}^T_\beta - \mathbf{X}^*_\beta)$ is of size $\beta_\mathbf{Y} \times \beta_\mathbf{Z}$, such that the updates for the proposed SGD are:

$$\mathbf{Y}_{\beta+1} = \mathbf{Y}_\beta - \theta\mathbf{S}_\beta\mathbf{Z}\left(\frac{\beta\mu}{\min(k,t)}(\mathbf{Z}^T\mathbf{Z}) + (1-\mu)(\mathbf{Z}^T_\beta\mathbf{Z}_\beta)\right)^{-1} \tag{3.14}$$

and

$$\mathbf{Z}_{\beta+1} = \mathbf{Z}_\beta - \theta\mathbf{S}_\beta\mathbf{Y}\left(\frac{\beta\mu}{\max(k,t)}(\mathbf{Y}^T\mathbf{Y}) + (1-\mu)(\mathbf{Y}^T_\beta\mathbf{Y}_\beta)\right)^{-1}, \tag{3.15}$$

where θ is the step size, $\frac{\beta}{\min(k,t)}$ is the normalization constant, and $\mu \in [0,1]$ is a non-negative scalar that is used as a weight in order to weigh $\mathbf{Y}^T\mathbf{Y}$ and $\mathbf{Y}^T_\beta\mathbf{Y}_\beta$ differently. The normalization constant term $\frac{\beta}{\max(k,t)}$ ensures that the Frobenius norm $\frac{\beta(\mathbf{Y}^T\mathbf{Y})}{\max(k,t)}$ and $(\mathbf{Y}^T_\beta\mathbf{Y}_\beta)$ are of the same order as well for the terms $\mathbf{Z}^T\mathbf{Z}$ and $\mathbf{Z}^T_\beta\mathbf{Z}_\beta$. In this case, it has been chosen to compute $\mathbf{Y}^T\mathbf{Y}$ and $\mathbf{Z}^T\mathbf{Z}$ after every update until the whole matrix had been reconstructed; that is, until the minimum of the cost function had been reached.

Both SVD and SGD are matrix factorization algorithms, the traditional matrix completion uses the SVD technique, while the proposed technique uses the SGD technique. In cases where the SSD matrix has a lot of missing entries, which is typical of collaborative spectrum sensing scenarios, SVD can be applied with the assumption that the missing entries are known to be zeros. However, this assumption is intuitively incorrect in practical collaborative spectrum sensing scenarios, since it implies that certain SU observed that the channel is not occupied, which creates some bias. The general idea of using SVD is to approximate a low-rank matrix $\mathbf{X}$ by decomposing it to three low-rank matrices, which can then be approximated as the minimization of the Frobenius norm $||\mathbf{X}-\hat{\mathbf{X}}||$. On the other hand, the primary objective of SGD is to use a technique almost similar to SVD for low-rank approximation of $\hat{\mathbf{X}}$ by finding $\mathbf{Y}$ and $\mathbf{Z}$ that minimizes the function $f(\mathbf{Y},\mathbf{Z}) = ||\mathbf{X}-\mathbf{YZ}||$. Thus, the difference between the two techniques is that SGD only considers the entries of $\mathbf{X}$ that are available in order to approximate the missing ones, which is not a convex function to minimize. However, SGD still works well, since it makes this function convex by fixing alternately $\mathbf{Y}$ and $\mathbf{Z}$ and optimizing for $\mathbf{Y}$ and $\mathbf{Z}$ respectively before computing the final reconstruction error. This is shown in Steps **09** to **11** in Algorithm **2**.

3.6 SIMULATION RESULTS AND DISCUSSION

In this section, the simulation parameter set-up section contains simulation parameters used in this chapter, the preliminary results subsection checks the sanity of the proposed algorithm, while the results subsection presents graphical simulation results and also analyses the impact of the matrix size, matrix rank and step size on the performance of both imputation algorithms, exemplified using convergence tables. A series of simulations were conducted using $MATLAB^{TM}$, on an Intel(R) Core(TM) i5-4590 @ 3.30GHz processor with 8.00 GB RAM, and resolution of 1920 $\times$ 1080 GPU.

3.6.1 Simulation Parameter Set-up

The SSD data set that is used for simulation has been created artificially using MRFs, Gibbs distribution and the Metropolis-Hastings algorithm, as discussed in Section **4.6**. A fraction of SU nodes p_c is randomly picked at a time within the set of SUs to combine their spectrum sensing results with those of neighbors. By allowing some transmissions to succeed while others fail, a fraction p_c of successful collaboration, which means that there are $1-p_c$ latent values, is obtained. Therefore, $p_c = \mathbf{X}$ represents the SSD matrix and forms the the input SSD matrix $\mathbf{X}$, which is very sparse in its composition and is decomposed and reconstructed by the two algorithms matrix completion and SGD. The decomposition/factorization of $\mathbf{X}$ results in factor matrices $\mathbf{Y}$ and $\mathbf{Z}$, which provide the $1-p_c$ latent factors for the missing values, learned by the two algorithms separately.

The two algorithms are evaluated by reconstructing the input data matrix with the help of their learned latent factors and finding the MSE of the original spectrum sensing data matrix with the constructed ones. The MSE is given by:

$$MSE = \sqrt{\frac{\sum_{k,t,x\in X}(x_{k,t}-\hat{x}_{k,t})^2}{|X|}}. \tag{3.16}$$

By letting $y_{k,r}$ and $z_{r,t}$ be the elements of $\mathbf{Y}$ and $\mathbf{Z}$, respectively, a missing entry at matrix position SU k and time slot t is predicted by

$$\hat{x}_{k,t} = \sum_{r=1}^{R} y_{k,r} z_{t,r}. \tag{3.17}$$

The performance of these two algorithms is tested for different values of latent values t, with different matrix ranks r, and with different step sizes (learning rates) α_t. The rest of the simulation parameters are listed in Table **3.1** below.

Table 3.1. Simulation Parameters Used for Setting up the CRN.

Parameter	**Value**
Frequency, f_c	2.1 GHz
Total bandwidth	8 MHz
Maximum number of SUs, K	300
Base station transmission power, P_{tx}	24 dBm [250 mW]
Communications channel (Noise)	Gaussian (Additive white)
Noise power density	-174 dBm
Receiver noise figure	9 dB
Path loss exponent, α	3
Minimum duration of each state, Δt	1 min
Average percentage of busy channels	5%
Test matrix ranks, r	4, 5, and 6
Scalar weight, ϖ	0.5
Learning rate, α_t	0.01, 0.1

The total bandwidth of 8 MHz is the bandwidth allocated to CRNs in South Africa. Because the CRN has been allowed to adjust slowly and carefully, with the $\Delta t = 1$ minute, the small learning rates of 0.01 and 0.1 were carefully chosen because of cautiousness not to allow the CRN to toggle rapidly between states. Thus, in the simulations, a probabilistic channel modeling framework, which characterizes the distribution of a sample of the channel, as well as its spatial correlation is used.

3.6.2 Preliminary Results

The CRN is simulated such that it enables one to sample uniformly from the whole network formed by a 50×50 grid. This is shown if Figure 3.4, where the yellow spots on the grid represent sites occupied by PU activities. As the sampling is carried out, it can be noticed that the sample mean of the fraction of occupied sites to the unoccupied sites on this graph decreases with the increase in network size. For example, Figure 3.4 shows a 50×50 network with a sample mean of 0.1129 compared to 0.0838 when a 1000×1000 grid is simulated. The result in Figure 3.4 above is obtained from toggling the

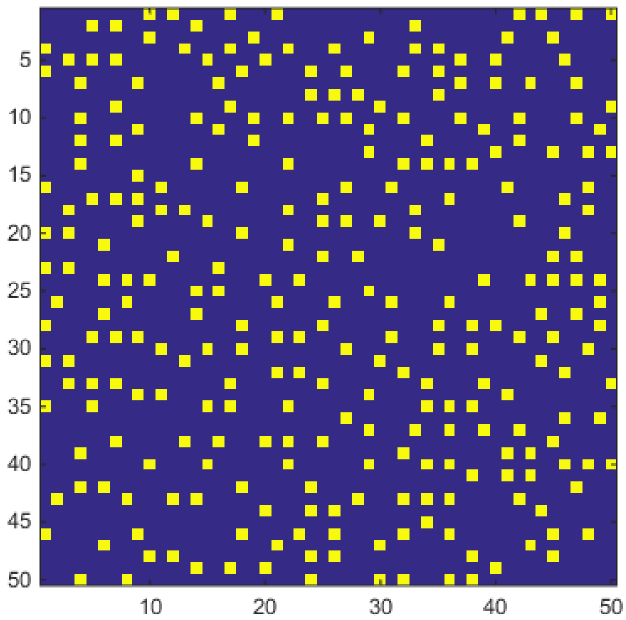

Figure 3.4. Results Show SU Opinions About Spectrum Occupancy.

states between ON and OFF and thus shows different sites on the grid showing spectrum occupancy states, denoted by yellow (bright) spots indicating sites occupied by SUs having an opinion of an occupied spectrum. The blue (dark) patches indicate the sites that SUs perceive as vacant and open for opportunistic transmission. As can be seen in Figure 3.4, the Metropolis-Hastings algorithm together with the Ising model is able to sample from the whole CRN with a probability closer to 1. Therefore, a proper target distribution in the form of an exponential bivariate distribution, defined inside a hyper-cube (i.e., four-dimensional cube), is used.

Figure 3.5 indicates that the algorithm is able to sample from the whole proposal distribution, even at its tails, which shows that all the sites of the grid are covered by the exponential bivariate distribution. It is worth noting that the states sampled from the Markov chain are also samples drawn from the target distribution using the proposal distribution where the proposals X_1 and X_2 are sampled from a uniform (0,8) distribution. That is, the sampling proposals for $\mathbf{X}$ are done from within a box $[0,10]^k$ as stated in Section **5.4**.

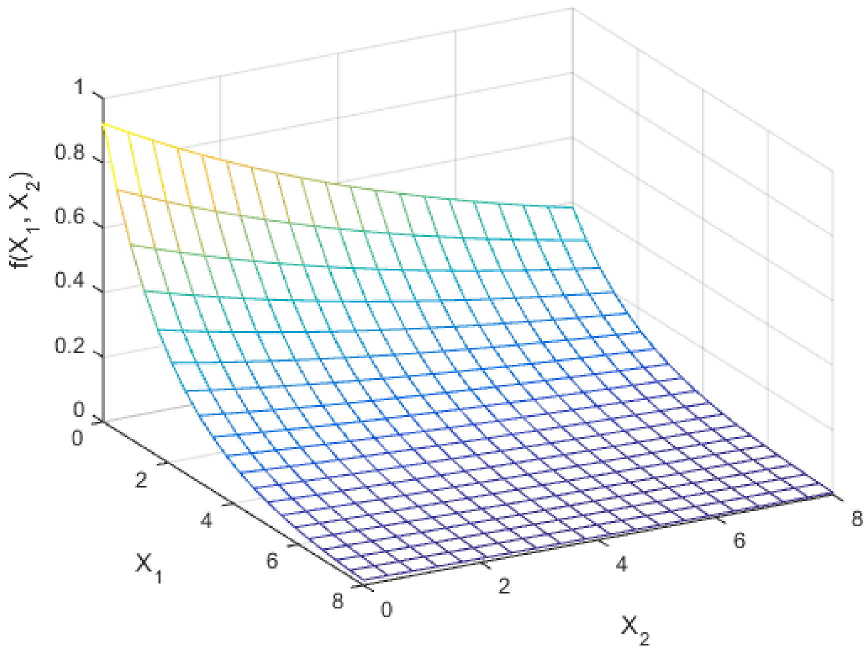

Figure 3.5. Target Distribution Using the Bivariate Exponential Distribution.

3.6.3 Simulation Results and Discussion

In this subsection, the simulation of the performance of the two algorithms is done, where it is assumed that there are 300 SUs distributed in the CRN grid, where each SU gathers the samples from other SU nodes once per time slot. Thus, the sampling rate is defined as the ratio of the collected measurements to the total number of SUs. For example, suppose that an SU receives spectrum occupancy measurements from 49 other SUs, which makes it 50 opinions. This makes $p_c = 50/300 = 16.67\%$, which is then treated as the sampling rate. Similarly, where 249 measurements have been received, the sampling rate is 83.33%. However, care must be taken not to approach much closer to the total number of SUs in the CRN because the idea of missing values will be lost and the purpose of this work will be defeated.

In Figure 3.6, as expected, it can be observed that the curves become steeper as the sampling rate, p_c, increases. When the sampling rate is 16.67%, which is the lowest chosen value, the estimation performance is poor, as it reaches 100% at a distribution mean of 55. As the sampling rate increases, the probability of correct estimation also increases. For example, when the sampling rate is at 83.33%, the estimation reaches 100% at a distribution mean of 49.

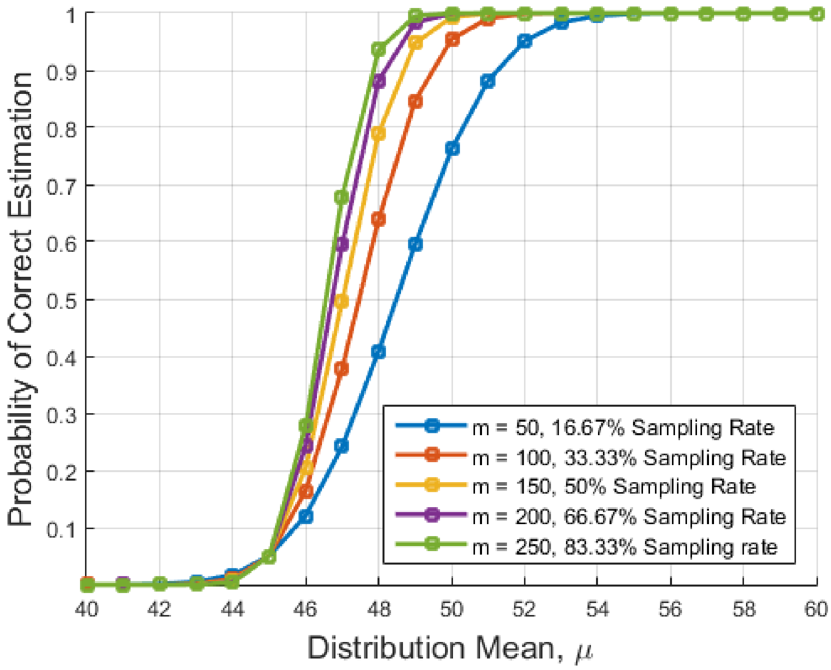

Figure 3.6. Probability of Correct Estimation of Missing Values.

Now, assuming that the observation matrix has been evaluated to be MAR as the mechanism of missingness, the simulations for the proposed spectrum occupancy reconstruction and learns the missing values using an SGD algorithm are conducted. The estimation error in (3.5) is used as an evaluation metric to test the estimation performance of the SGD method. The overall size of the spectrum usage data matrix is $K \times T$, with partial entries of spectrum usage known owing to the limited number of SUs reporting.

The result in Figure 3.7 has been plotted as the amount of error in computing the gradient of the algorithm for different values of matrix ranks r, varying between 4 and 6. These are the results for computational error when using the SGD method for spectrum occupancy reconstruction. It can be observed from Figure 3.7 that with a smaller rank (e.g., $r = 4$), the computational error is very small compared to $r = 6$. However, both computations take the same number of iterations to converge. When taking $r = 5$, the computational error initially falls between $r = 4$ and $r = 6$, but eventually becomes much better than both of them and converges earlier.

For the same values used in the simulation of Figure 3.7, the simulations for the cost error of running the algorithm are conducted and this result is shown in Figure 3.8, where it is observed that

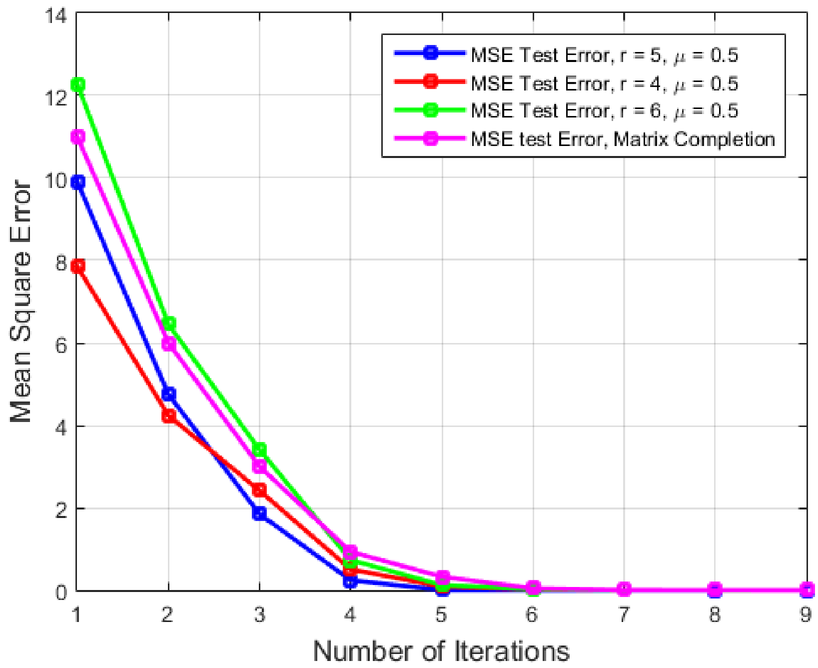

Figure 3.7. Convergence of Algorithm with Computational Test Errors.

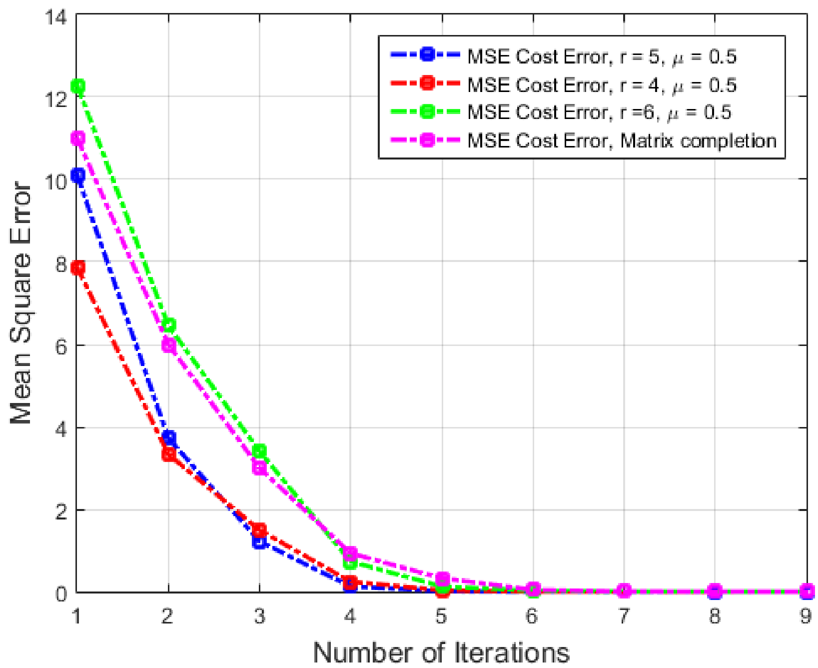

Figure 3.8. Convergence of Algorithm with Computational Cost Errors.

using rank $r = 6$, the behavior is the same as the computational error shown in Figure 3.7. However, the same behavior seen in Figure 3.7 could not be repeated for $r = 4$ and $r = 5$. The demand for space in memory becomes almost the same for both $r = 4$ and $r = 5$. From the results illustrated in Figure 3.7 and Figure 3.8, we can conclude that in order to perform reconstruction using the stochastic gradient descent, the rank of the sample matrix should optimally be $r = 5$, not more. The parameter Δ is fixed to a minimum of 1 minute to avoid the spectrum occupancy status from changing rapidly. The sampling rate is varied from 10% to 83.33%. The sampling rate is referred to as the fraction $p_c = \mathbf{X}$ and the crossover probability between H_0 and H_1 is fixed 47.50%, as obtained in Figure 3.6. The case where the number of SUs collaborating on spectrum occupancy observations is considered is between 30 and 250 out of a total of 300 SUs and every spectrum occupancy state lasts for a maximum of 3 minutes to keep it from toggling between ON and OF before the algorithm finishes computing.

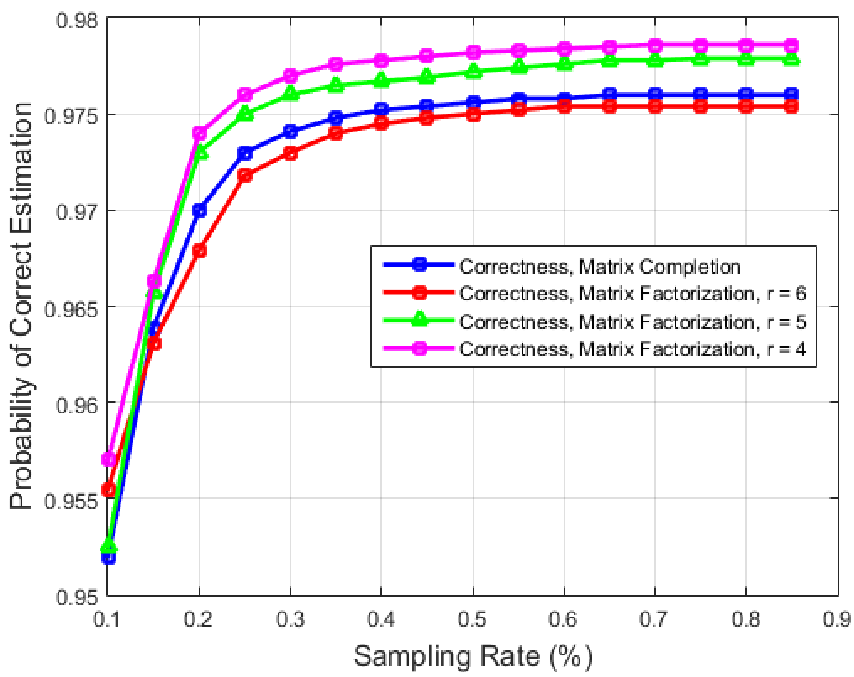

Figure 3.9. Performance of Correct Estimation in Reconstruction Spectrum Occupancy Matrix Against Sampling Rate.

In Figure 3.9 and Figure 3.10, it can be seen that the proposed algorithm can reconstruct the spectrum occupancy matrix with a high correctness rate even at a 10% sampling rate. As the sampling rate increases, there is a corresponding decrease in the misdetection rate. As seen in Figure 3.9, when the sampling rate reaches 60%, the occupancy data matrix reconstruction correctness rate has already

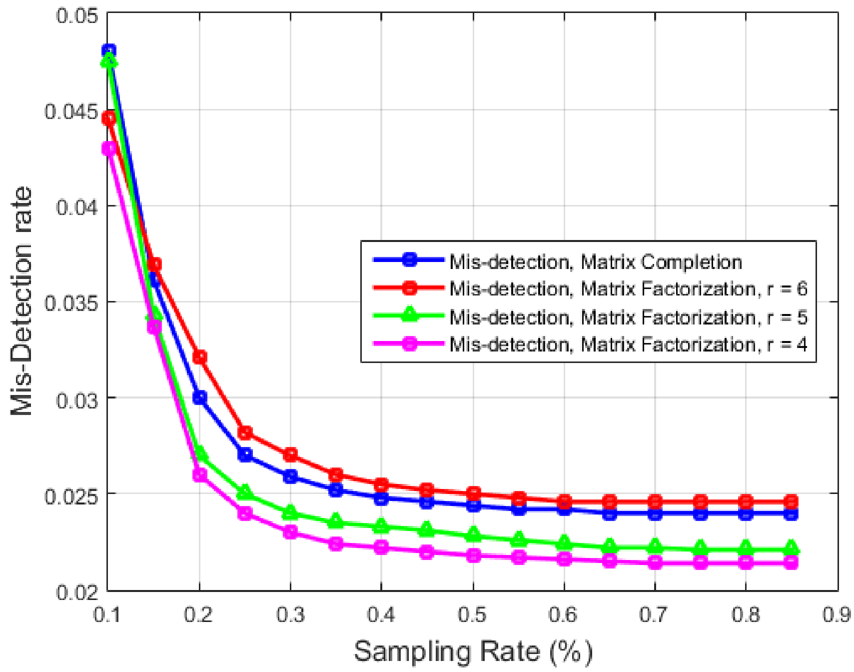

Figure 3.10. Probability of Incorrect Estimation in Reconstructing Spectrum Occupancy Matrix Against Sampling Rate.

reached 97.86% when $r = 4$. A corresponding decrease in the misdetection rate of 2.14% is seen in Figure 3.10. This means that as the sampling rate reaches 60%, the proposed algorithm guarantees a misdetection rate that is less than 3%, when the reconstruction rate $r = 4$.

There is a noticeable performance improvement with the matrix reconstruction rate; for example, when $r = 6$, the proposed algorithm performs worse than matrix completion, but when the value of r decreases, the performance improves. This can be explained in terms of introducing errors and reducing the errors. When the matrix rank $r = 6$, more errors are introduced into the algorithm than when the matrix rank r is equal to 5 or 4. There is always a price to pay in terms of sacrificing accuracy when increasing the matrix rank, however, every improvement in terms of reducing the matrix rank means accurate reconstruction.

3.6.4 Overall Performance Comparison

The following presentation of the convergence tables seeks to summarize the performance of both algorithms. Firstly, the size of the spectrum occupancy matrix $k \times t$ and the matrix ranks r are varied, while keeping the learning rate α_t constant and evaluating the total error and convergence time. The number of iterations is kept at 10 for all the investigations

Table 3.2. Convergence Results for Matrix Size 3000×10000.

Algorithm	k	t	r	α_t	MSE	time
SGD	3000	10000	4	0.1	3.59e-08	2.59 sec
SGD	3000	10000	5	0.1	2.35e-07	3.52 sec
SGD	3000	10000	6	0.1	4.80e-07	3.12 sec
SVD	3000	10000	*	0.1	3.55e-07	3.20 sec

Table 3.3. Convergence Results for Matrix Size 300×1000.

Algorithm	k	t	r	α_t	MSE	time
SGD	300	1000	4	0.1	NaN	0.55 sec
SGD	300	1000	5	0.1	NaN	2.13 sec
SGD	300	1000	6	0.1	NaN	0.21 sec
SVD	300	1000	*	0.1	NaN	1.05 sec

The results shown in Table **3.2** and Table **3.3** show that the computation error (i.e., MSE) is dependent on the size of the spectrum occupancy matrix. Reducing the size of the spectrum occupancy matrix by a factor of 10 actually improves the reconstruction performance in terms of the MSE and convergence time. Now, the learning rate is reduced from 0.1 to 0.01 and evaluates the response of the algorithms in terms of reconstruction performance.

Table 3.4. Convergence Results for Matrix Size 3000 × 10000.

Algorithm	k	t	r	α_t	MSE	time
SGD	3000	10000	4	0.01	1.65e-01	3.46 sec
SGD	3000	10000	5	0.01	2.27e-01	2.89 sec
SGD	3000	10000	6	0.01	2.23e-01	3.26 sec
SVD	3000	10000	*	0.01	2.25e-01	3.15 sec

Table 3.5. Convergence Results for Matrix Size 300 × 1000.

Algorithm	k	t	r	α_t	MSE	time
SGD	300	1000	4	0.01	NaN	4.08 sec
SGD	300	1000	5	0.01	NaN	2.11 sec
SGD	300	1000	6	0.01	NaN	0.56 sec
SVD	300	1000	*	0.01	NaN	1.30 sec

NB: * The abbreviation **NaN** in Table 3.3 and 3.5 is a formulaic representation of real numbers in terms of floating-point numeric that stands for **Not a Number**, usually used as an approximation to support a trade-off between range and precision. ** In SVD sampling with matrix rank is not performed; matrix rotation about an arbitrary axis is performed instead.

The results shown in Table **3.4** and Table **3.5** indicate that reducing the learning rate increases the reconstruction error (i.e., MSE) but reduces the convergence time for both algorithms. This behavior is due to weight decay, which is easier to understand from intuition. Weight decay means that when a smaller learning rate is used, more and more epochs are used in training and the training weights are suppressed more and more. This suppresses the result as well, owing to too much training. When this suppression happens, the model becomes weak, such that it under fits the model and both training and testing loss gets larger.

3.7 CONCLUSION

In this chapter, a solution for spectrum occupancy reconstruction in distributed spectrum sensing in CRNs was envisioned. Spectrum occupancy reconstruction is a problem often encountered in distributed spectrum sensing when the spectrum sensing data contributed by SUs has gaps of missing entries. Because the lack of available spectrum sensing data with missing spectrum observations, the spectrum occupancy reconstruction problem was formulated as a magnetic excitation state recovery problem using a plenary grid model on an MRF, where each grid point possessed either a spin-down or a spin-up. Each of these magnetic spin states was used to represent the binary logic of channels, i.e., 0 - idle and 1 - busy, which led to a realization of the spatio-temporal spectrum usage in CRNs. Communication between SUs was envisioned using pairwise interactions where SUs would share their spectrum observations by exchanging electromagnetic energy whose spatial distribution was used to model the flow of energy between SUs.

The spectrum occupancy reconstruction problem was then mathematically formulated as a matrix factorization problem and solved using the SGD algorithm. The simulation results indicate that SGD is able to learn and impute the missing values with a low reconstruction error compared to SVD. The results also indicate that with more collaborating SUs (i.e., fewer missing entries), the reconstruction has a high resolution, while they also indicate that the convergence time for both algorithms is dependent on the matrix size, rank and learning rate. However, one interesting observation is that the SGD algorithm performs better than the SVD algorithm when the matrix rank is not greater than 5. The results also show that in as much as factorizing a matrix with missing entries can be done easily using SVD, when an optimization technique is needed to solve the factorized matrix, SGD becomes important as long as the matrix rank is kept as low as possible. This implies considering the fact that when the data matrix grows (i.e., number of SUs), the complexities to be dealt with also increase; for optimal results the relation $r \ll min\{k,t\}$ in (3.4) should always hold.

CHAPTER 4 QOS PROVISIONING AND ENERGY SAVING SCHEME FOR DISTRIBUTED COGNITIVE RADIO NETWORKS USING DEEP LEARNING

4.1 CHAPTER OVERVIEW

One of the major challenges facing the realization of CRs in future mobile and wireless communications is the issue of high energy consumption. Since future network infrastructure will host real-time services requiring immediate satisfaction, the issue of high energy consumption will hinder the full realization of CRs. This means that offering the required QoS in an energy efficient manner, resource management strategies need to allow for effective trade-offs between QoS provisioning and energy saving. To address this issue, this paper focuses on single BS management, where a resource consumption efficiency is obtained by solving a dynamic RA problem using bipartite matching. A DL predictive control scheme is used to predict the traffic load for better energy saving using a SAE. Considered here was a BS processor with both processor sharing (PS) and first-come-first-served (FCFS) sharing disciplines under quite general assumptions about the arrival and service processes. The workload arrivals a defined by a Markovian arrival process while the service is general and the possible impatience of customers is taken into account in terms of the acceptable delays. In this way, the BS processor is treated as a hybrid switching system that chooses a better packet scheduling scheme between mean slowdown (MS) FCFS and mean slowdown (MS) PS. The simulation results presented in this paper indicate that the proposed predictive control scheme achieves better energy saving as the traffic load increases, and also the processing of workload using MS PS achieves substantially superior energy saving compared to MS FCFS.

4.2 INTRODUCTION AND BACKGROUND

Because of the increase in data rate requirements and the level of heterogeneity, the rate of change in network traffic expected in future mobile and wireless communication networks poses new challenges related to spectrum management and energy consumption [42]. The increase in multimedia services and other mission critical applications that are hosted by network infrastructure, which demand immediate satisfaction in terms of QoS requirements has led to even newer challenges in energy efficiency [233]. These has led to escalating concerns regarding energy efficiency in terms of network operational costs such that research on energy consumption and saving have gained huge attention. In mobile and wireless communication networks, BSs consume much of the power and its power consumption varies over time [234].

The general BS operation mode can be described as a finite state machine which can be explained in terms of a two-state Markov model, i.e., idle (OFF) and active (ON). Hence the energy consumption of a general BS depends on its mode of operation. When the BS is in its active mode, it has to process traffic streams from all its associated users, hence its computational components are faced with maximizing user satisfaction demands, while simultaneously minimizing their energy consumption. On one hand, maximizing user satisfaction by upholding user QoS requirements through an increase in transmission power has a substantial effect on energy efficiency, while on the other hand reducing transmission power degrades QoS performance [141]. Therefore, quantifying the trade-off between energy consumption and the required QoS are key parameters in BS energy consumption that require balanced optimization.

Researchers in both industry and academia have proposed workable solutions to reduce network operational costs through the installation of energy efficient hardware [236], but this approach does not seem to solve the problem completely. Reducing network operational costs by installing more energy efficient hardware somehow requires a compromise between the energy cost and the user coverage drop. Thus, this approach somehow falls short in addressing the energy efficiency problem since it might result in wireless access networks being almost invariably over-provisioned and under-provisioned with respect to the user traffic demands [237]. Even though resource over-provisioning may lead to better QoS provisioning in terms of negligible packet losses and transmission delays, it comes at non-negligible operational costs [238].

It is thus clear that resource over-provisioning needs to be replaced by optimal or even near-optimal energy efficient solutions that allocates adequate network infrastructure based on current resource demand. Adequate provisioning, however, requires a better understanding of the relationship between resource demand, available capacity and the transparency between real-time and best effort traffic streams. In light of the above, newer and more efficient resource management schemes, capable of controlling how much network resources are allocated at a certain time can be extremely effective and provide quite large network operational cost reductions. As the energy efficiency of BSs is receiving more attention due to several factors, there are also concerns about meeting the international mobile telecommunications (IMT)-2020 and beyond requirements, as well as challenges facing cognitive radio networks (CRNs).

In the overall energy efficiency network design and evaluation process, the primary importance is an adequate metric that is directly related to the optimized decisions across all the protocol layers. The most popularly used metric is the bits-per-Joule, which is defined as the system throughput for unit energy consumption [239]. A great quantity of information-theoretic results for energy efficiency at the link level based on this metric set the limitation on transmission power is set as a constraint, and it has been proven that the upper bound channel capacity per unit energy can only be achieved by utilizing an unlimited number of degrees of freedom per information bit [240].

Analyzing the bits-per-Joule capacity at network level proved that capacity increases with the number of nodes in the network, implying that large-scale energy-limited networks may only be suitable for delay-tolerant applications. For example, this metric has been widely used as the utility function in game-theoretic approaches for energy saving in wireless networks, where the energy consumption models only consider the transmission power associated with data transmission rate [241]. However, the transmission power is only a part of the overall energy budget and when the energy consumption of other parts (e.g., circuit power consumption of the transceiver) are taken into account, the energy efficient schemes described in literature might not be appropriate to meet the IMT-2020 and beyond requirement specifications.

4.2.1 Meeting IMT-2020 and Beyond Requirement Specifications

Nowadays, global mobile data traffic is increasingly dominated by delay- and loss-intolerant traffic streams, which means that traffic congestion in the core network is an inevitable occurrence. This can quickly lead to overall network performance degradation resulting from the moderate-to-high traffic levels [242]. The consequence of this is the catastrophe of heavy burst losses as the whole network might degenerate into chaos. The daily operations of network components that are pushing data and multimedia traffic into the internet require a significant increase in network energy consumption or no user will be able to use the network properly. However, despite the intense research efforts by both academia and standardization bodies in the quest to meet the requirements specified in the IMT-2020 [231], there is still considerable controversy concerning the definition of QoS and its direct influence on network resource provisioning. This is because huge research efforts have focused on the optimization of sum-rate to support high data transmission rates [243].

Apart from energy efficiency, there is a variety of objectives that have been put forward in the preparation of the next generation of mobile and wireless communications. Other than achieving high data rates, these objectives are based on network metrics such as improved coverage with uniform user experience, higher reliability and lower latency, better energy efficiency, lower cost user devices and services, better scalability with the number of devices [91]. However, these objectives have to be realized simultaneously, and the challenge is that they are often coupled in a conflicting manner. Achieving improvements in one objective leads to degradation in the other. Consequently, the design of future mobile and wireless networks requires new optimization tools that are capable of properly handling both the existence of these objectives and intelligent trade-offs between them.

In CRNs, QoS provisioning requires a cross-layer design, numerous studies have proposed to allocate resources reasonably by considering QoS requirements for secondary users (SUs). For example, the authors in [244] considered the energy effect on QoS provisioning and proposed an energy efficient channel hand-off strategy using POMDPs. In the development of the hand-off strategy using POMDPs, the authors considered imperfect channel sensing and residual energy of SUs, which considers beliefs about the operating and backup channels and the residual energy at the SU. An α-retry policy was proposed as a spectrum access strategy to enhance QoS for SUs in [245], where a preemptive priority queue is built as a 2-dimensional DTMC and the blocking rate, the forced dropping rate, the throughput and the average delay of the SU packets were analyzed. From a spectrum management perspective [246] proposed a queuing theory-based analytical framework to analyze QoS for SUs. The spectrum resource management problem was considered for co-located SUs with both streaming and intermittent data and seamless end-to-end service was ensured by effectively identifying the number of backup channels. In another contribution, the authors in [11] proposed the use of priority queues as a RA model for different traffic classes. They used packet priority to explore the relevance and implications of various heterogeneous classifications. The RA model incorporate the essential concepts of heterogeneity which were developed with weight attached to differentiate between different traffic classes.

Meeting the IMT-2020 and beyond requirement specifications within the CRN space has presented itself as a huge challenge to researchers since balancing the above approaches in an energy-efficient manner depends on several factors. Such factors are the degree of cooperation between PUs and SUs and the reliability of spectrum sensing [248] exacerbate the problem of energy efficient CRN operations. Because of the intermittent nature of SU connections in the primary networks, it is a difficult task to achieve energy efficient CRN operations due to the preemptive priority of PUs. Preemptive priority gives PUs absolute rights to use their licensed spectrum such that while SUs try to exploit transmission opportunities in free spectrum bands, PUs can resurface at any time and force SUs to terminate their transmissions. In that case, SUs are forced to leave the current spectrum band and handoff their activities to another channel that is deemed vacant through spectrum sensing. These transmission terminations lead to subsequent transmission delays and high energy consumption, which affect the latency and energy efficiency respectively, especially when the probability of forced terminations is high [249]. Thus, it seems as if there are apparently too many different requirements that need to be kept in mind when attempting to design lasting solutions. Unfortunately, there is no easy short cut to achieving a long-lasting solution that balances energy-efficient resource allocation and QoS in CRNs. This means that these two objectives cannot be treated separately because they are coupled - sometimes in a consistent fashion, but more often than not in conflicting ways implying that improvements in one objective leads to deterioration in the other.

4.2.2 Technological Challenges Facing CRNs

The 5G era which is driven by the IoT introduces the requirements of high data rates, low latency, efficient use of spectrum resources and coexistence of different network technologies. Because of the existing spectrum quagmire, all future wireless devices need to be CR-capable, which makes the CR technology the biggest enabler of 5G networks and beyond. Apart from the major concern of spectrum scarcity, there are several other issues and technological challenges that hinder the design, implementation and realization of a fully functional CRN. All future mobile and wireless network technologies will be deployed in a distributed fashion in order to exhibit significant gains in network capacity maximization. Wireless technologies deployed in this manner require significant coordination among network devices which results in high communication overheads.

Beyond the level of heterogeneity introduced by CRNs into the traditional wireless network, the other spectrum use cases that are supposed to enormously benefit from the CR technology also have heterogeneous requirements. For example, the IoT and its other derivatives such as the cognitive internet of people, services, data and things and several other variants of the IoT, with bandwidth-demanding services such as video streaming and video-on-demand, all pose great challenges to the issue of spectrum management. Because of the challenges that come with the dynamics of channel availability, uncertainty of spectrum sensing and spectrum access, as well as PU activities, real-time SU transmissions might suffer. On one hand, the network designer is faced with numerous design trade-offs, diversified network dynamics and limited resources, while on the other hand SUs with bandwidth-consuming services with stringent QoS constraints require resources to be allocated immediately. Thus, this necessitates a holistic cross-layer design approach to exploit the CR technology optimally, which calls for the development of intelligent and invisible paradigms for programmable and controllable networks to satisfy future requirements.

Regarding the intelligent use of spectrum resources, the CR technology stands to benefit massively from the incorporation of AI into its operation. From a CR perspective where the issues of QoS and energy efficiency have put enormous pressure on top of the existing spectrum shortage, achieving energy-efficient operations is a daunting task. With the interconnection of heterogeneous devices posing numerous challenges that may include high energy consumption, data rate requirements and intermittent connections because of SU mobility and PU activities, the incorporation of AI will ensure efficient decision making. A perfect solution to this problem will require some kind of automated approach that achieves both QoS and energy efficiency, while also yielding a new set of network assurances such as reliability.

However, any automation that network designers can come up with has to be within the perimeters of good energy consumption. One promising way is to use the massive amount of data that is generated by the great quantity of network equipment to develop predictive measures to deal with the energy efficiency problem. AI strategies such as deep architectures (i.e., DL, DRL) can be used to analyze these data, extract the relevant patterns, make sense of the data and then prescribe energy efficient actions to be taken by network equipment. This may lead to a realization of more effective and efficient energy saving models for CRNs. Existing energy saving schemes in traditional cellular networks are discussed in the following section.

4.3 EXISTING ENERGY SAVING SCHEMES

4.3.0.1 The EARTH Model

The first energy-saving technique was proposed in the Energy Aware Radio and network TecHnologies (EARTH) project, which is a concerted effort to achieve energy efficiency in wireless networks. The primary objective of the project was the reduction in energy consumption in mobile networks as environmental concerns such as global warming were gaining momentum [250]. To quantify the energy savings in a wireless network, the power consumption of the entire system needs to be captured and an appropriate energy efficiency evaluation (E3F) needs to be defined. The E3F is applied to provide an assessment of the BS energy efficiency of a 3GPP LTE network deployed where BSs are switched ON/OFF based on traffic load [251]. This model is based on a finite-state machine model consisting of two operational states, P_{on}, which denotes the static/load-independent power consumption figure; P_{tx}, which denotes the dynamic/load-dependent power consumption figure whose consumption trajectory is scaled according to traffic load.

The EARTH model addresses power consumption through flexible, load-adaptive adaptations by embedding duty-cycle scenarios over traffic load whose optimization strategy relies on reassigning traffic load from overloaded BSs to other BSs depending on their current traffic. The operational strategy of the EARTH E3F framework is illustrated in Figure 4.1 below [251].

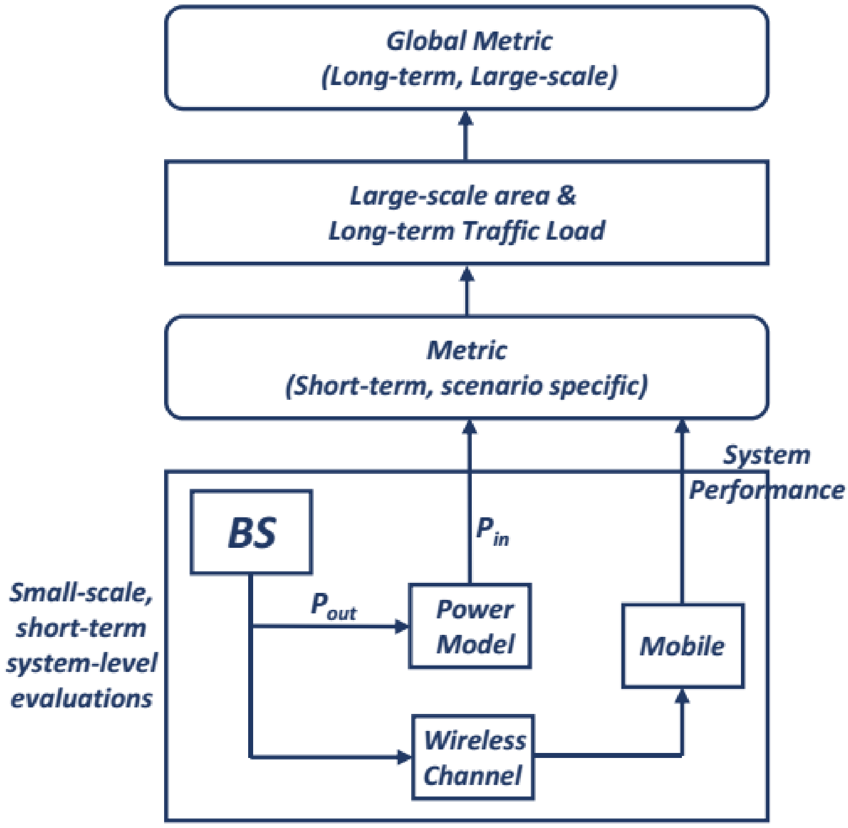

Figure 4.1. Illustration of the EARTH E3F Framework.

This technique offers energy efficient communication in low traffic regimes and plays an important role in reducing overall network power consumption [253]. However, in terms of meeting the IMT-2020 requirement specification, one striking drawback is that when a BS is switched ON/OFF, there is a switching cost incurred in terms of the energy and the time to transition from ON-OFF when there is little or no channel activity and vice versa, which is a significant amount that cannot be ignored. Except the evident and noticeable reductions of the operational costs faced by mobile operators it also adapts the level of resource over-provisioning by re-associating traffic to moderately served BSs [254]. There is a considerable amount of energy and time is spent in turning on BS components, user data management and user re-association; which affects the QoS due to server response times which also has a consequence in packet delays and losses. There is also limited support for newer systems and technologies and the (de)activation information is not defined.

4.3.0.2 Green Cellular Network Model

The second energy-saving technique is the green cellular network model, which was proposed to address the shortcomings of the EARTH model and reduce carbon dioxide (CO_2) emissions. The green cellular network incorporates the use of both grid power supply and energy harvested through solar and wind supply, as illustrated in Figure 4.2 below [167].

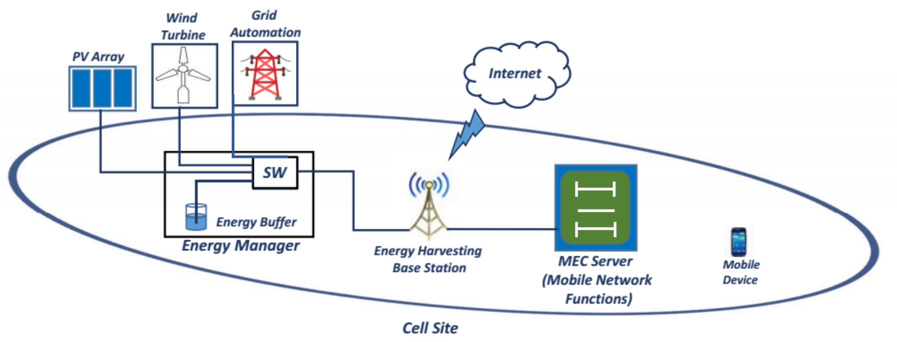

Figure 4.2. Illustration of a Green Cellular Network Setup.

However, the basic structure of the EARTH model was not discarded, but was maintained and reused - meaning that the BS power consumption remained unaltered. Through this technique, BSs can selectively switch between grid and harvested energy in order to reduce network operation energy costs. In this technique, an energy-harvesting BS is co-located with a multi-access edge computing (MEC) server to reduce energy consumption further. the radio network and energy-level information is communicated periodically to the MEC server through the radio network information services (RNIS) and the energy manager. The RNIS entity is responsible for selecting the appropriate energy source to fulfill the energy buffer and for monitoring the energy levels of the system.

The authors in [256] investigated an environment-aware framework for CRNs where PU and SU networks collaborate together. The collaboration between the two networks is to maximize their profits and meeting PU QoS and the total CO_2 emissions. In terms of energy balancing, [177] proposed an energy balancing strategy where the key technique is that each BS maintains two parameters. These parameters contain the trend of its previous energy consumption and then predicts its future quantity of energy which is defined as the BSs potential energy capacity. Using this concept offers better solutions, but the simplifying assumptions made may often introduce inaccuracies when the switching is not optimized. In terms of green cellular networking, it is being argued that energy saving can be achieved through the adoption of renewable sources of energy to make communication networks more energy efficient. However, from a communication perspective, a more energy-efficient system is created where the power consumption of the BS is optimized rather than adding several more power sources.

4.3.0.3 The One-step-ahead Predictive Model

The final energy-saving strategy is the step-ahead predictive model, which has the potential of playing a decisive role in boosting energy efficiency in future mobile and wireless networks through the use of predictive analytics. This technique does not optimize the network operation only in terms of energy consumption, but also with respect to balancing rates and power consumption, which accounts for overhead signaling and circuit power. The use of predictive analytics through one-step-ahead network traffic prediction puts the system ahead of the normal time and enables it to take proactive decisions. Network predictive analytics entail the extraction and analysis of patterns in network traffic trends and user behavior to predict future network behavior for better resource allocation. Through this technique systems can learn traffic profiles and automatically tune their computational parameters to accommodate future demands, thus data analytics promises to be a possible pathway towards achieving both QoS and energy efficiency objectives.

There are several prediction techniques that are suited for training and testing CR systems such as the bio-inspired meta-heuristic in [253], which is inspired by swarm intelligence and has been seen to achieve better energy saving for 5G networks. The authors proposed the separation of the control and data planes and the use of particle swarm intelligence to handle the interaction and operation of users to achieve better energy efficiency and lower aggregate delays. But, the integration of real-time practical learning capabilities using such a soft computing technique is a challenge in CRNs [248]. However, an investigation of applications from deep architectures (e.g., computational DL, DNNs, DRL, etc.) has shown some significant benefits in enabling decision-making through prediction in the absence of real-time information. For example, artificial neural network techniques studied in [258], and in [259] proved to improve the accuracy and decrease the complexity of traffic prediction and resource allocation, respectively. However, from both these works, it is clear that recent resource allocation problems have become dynamic and require the enhancement of ML architectures in order to improve their inference capabilities.

As just pointed out, [260] proposed an inference engine using fuzzy techniques to improve resource allocation in CRNs using an improved channel allocation that considered signal strength as the decision variable for the channel access priority of SUs. A genetic algorithm (GA)-based RA technique was studied in [261] where the impact of both the available spectrum resource size in terms of the population size. Here, GA was used to define the radio in the form of chromosomes and genes, where the users' QoS requirements were given as the input of the GA algorithm. The impact of both the available spectrum resource size were analyzed in terms of the population size and the number of defined chromosome's genes in spectrum allocation efficiency.

In another contribution, an improved LSTM was used in [262] to obtain accurate and fast traffic flow forecasting in intelligent transportation systems. Also, a time series prediction for extracting useful information from historical records to determine their future values was studied in [166], where an RCLSTM model was used to reduce the computational complexity associated with LSTM and was tested and verified for traffic prediction and user mobility in wireless networks. The RCLSTM was found to exhibit a certain level of sparsity, which appealingly reduces the computational complexity making it suitable for latency-stringent applications. An online optimization algorithm called ENAAM, based on traffic prediction and foresighted control policies was proposed in [167]. Here, the BSs and VMs are dynamically switched ON/OFF to effect energy saving and QoS provisioning by exploiting short-term traffic load and harvested energy forecasts using LSTM.

This contribution was inspired by the convergence of communication and computing has led to the emergence of MEC, where computing resources supported by VMs are distributed at the edge of the mobile network. BSs aiming at ensuring reliable and ultra-low latency services, are equipped with an energy-harvesting system to reduce energy consumption. One short-coming of this traffic prediction technique is that it relies on the obvious seasonality of traffic that has been aggregated with the granularity of "hours of day". Longer time granularities, based on hours can have significant short-comings in today's network traffic since it does not exhibit the same seasonal behavior, but rather up-down linear trends. The traffic pattern in modern cellular networks has changed drastically since the emergence of smart phones due to the many applications hosted by wireless networks.

Applications for social networking (e.g., Facebook), for internet telephony (e.g., Skype), for micro-blogging (e.g., Twitter, Posterous, FriendFeed, etc.), for instant messaging (e.g., WhatsApp, Facebook Messenger, WeChat, Viber, etc.), consist mostly of real-time content and hence exhibit different traffic patterns compared to the traditional voice, text messaging, emails and web surfing. Now the traffic behavior is mostly application-specific, and may present some anomalies with trends that require a new traffic matrix each time slot in order to be effective. Due to this, there are significant changes in individual packet sizes, burst sizes, packet inter-arrival times and the behavior of inactive periods. For this reason, a system capable of learning the behavior of traffic, predict its future behavior, and dynamically adjusts the resources assigned to traffic relations may turn early research into usable solutions.

4.3.1 Motivation and Contributions

The application of data analytics in wireless data promises to be a possible pathway towards achieving both QoS and energy efficiency objectives. This will help in obtaining tentative operating points for network equipment to achieve energy efficiency and network sustainability, which is an essential step towards managing the high level of heterogeneity that comes with future networks. Using the predictive technique, PRBs can be allocated to real-time traffic flows, with each traffic flow having a queue associated with it, as shown in Figure 4.3 below.

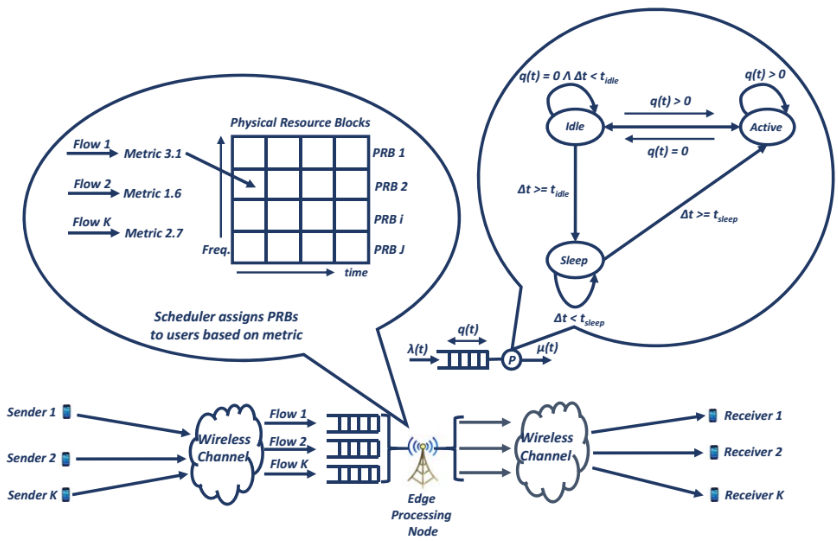

Figure 4.3. Illustration of a Step-ahead Predictive BS Model.

Figure 4.3 above shows a single BS with both the opportunistic spectrum access and the opportunistic computing part, where the actions of the BS processor are based on the state of each queue. Here, $q(t)$ represents the length of the queue in the buffer, $\lambda(t)$ is the packet/job arrival rate, $\mu(t)$ is the packet/job service rate, and Δt is the time elapsed in each state. The transitions between the BS operating modes is triggered by either an event or by the passage of time. For example, for an empty queue, i.e., $q(t) = 0$ the processor is idle and saves power. However, when events arrive, the processor is switched to active mode with little time overhead. Moreover, if the BS processor stays idle mode beyond a predefined threshold duration, it is then put to sleep mode. However, while the processor is in this state it registers no external arrivals. The processor may then transition back to its active mode if an event of an arrival occurs. The work done in this chapter is motivated by this model together with successes in DL with regard to the consideration of traffic classification and prediction of both spatio-temporal characteristics of traffic flow.

The application of data analytics in wireless data promises to be a possible pathway towards achieving both QoS and energy efficiency objectives. This will help in obtaining tentative operating points for network equipment to achieve energy efficiency and network sustainability, which is an essential step towards managing the high level of heterogeneity that comes with future mobile and wireless networks. Then, depending on the applied network functions, the question regarding the balance between QoS provisioning and energy efficiency rests on the packet-service discipline to achieve better energy saving. To solve this problem, an interesting contribution in [263] is followed, however, the approach proposed in this paper differs from this since it involves a separation principle that decouples the design of RA, energy consumption, and service QoS provisioning and makes the problem manageable. The major contributions of this paper are summarized as follows:

- Firstly, a distributed dynamic RA based on uplink (UL) power allocation and SU resource reservation protocol is proposed. Resource reservation is a transport layer protocol designed to reserve resources across a network for QoS using the integrated services model and is adopted in this paper to give SUs a high probability to complete their transmissions. The resource reservation problem is solved using geometric programming (GP) and a resource percentage threshold (RPT) serves as the portion of resources reserved for SUs, and otpimal RA solution is obtained through a weighted bipartite matching from graph theory with a polynomial complexity of $\mathcal{O}(K^3)$ compared to the $\mathcal{O}(K!)$ of integer programming. The weighted bipartite matching scheme is only used as a bridge from the traditional optimization to the DL cost function, it is not a solution.

- Secondly, a resource consumption efficiency obtained from the bipartite graph solution is then used to solve as weighted cost function in which power consumption is added together with different weights reflecting their contribution to BS power consumption. This initiates a DL predictive control scheme with control actions derived to drive a stacked auto-encoder (SAE) in making future traffic predictions as well as computing performance measures.
- Finally, the control actions are applied and using this formulation and relevant parameters of the operating environment such as workload arrival patterns are estimated and used by the model to predict the future behavior over a finite horizon, T. Using the output of the SAE, which are just a regression between the previous and current system states, appropriate packet processing schemes is chosen between mean slowdown first come first served (MS FCFS) and and mean slowdown processor sharing (MS PS). The belief that mean slowdown (MS) is important in packet-by-packet processing as a measure of the systems energy efficiency is proven by the results that different traffic flows consume significantly different amounts of resources and the choice of a processing scheme determines the overall energy efficiency of the system.

4.4 PROPOSED SYSTEM MODEL

Consider a single-cell spectrum sharing scenario where a PU transmitter and a PU receiver coexist with a set $\mathcal{K} : k = 1, 2, \cdots, K$ SUs in an energy-constrained CRN. It is assumed that SUs are running real-time traffic and performing opportunistic transmission on the shared spectrum consisting of $\mathcal{J} : j = 1, 2, \cdots, J$ physical resource blocks (PRBs). Perfect channel state information (CSI) is assumed for all K SUs uniformly distributed in a cluster such that the distance between them and the BS is characterized with channel gains $g_{j,1}(t), \cdots, g_{j,K}(t)$ as illustrated in Figure 4.4 below [159].

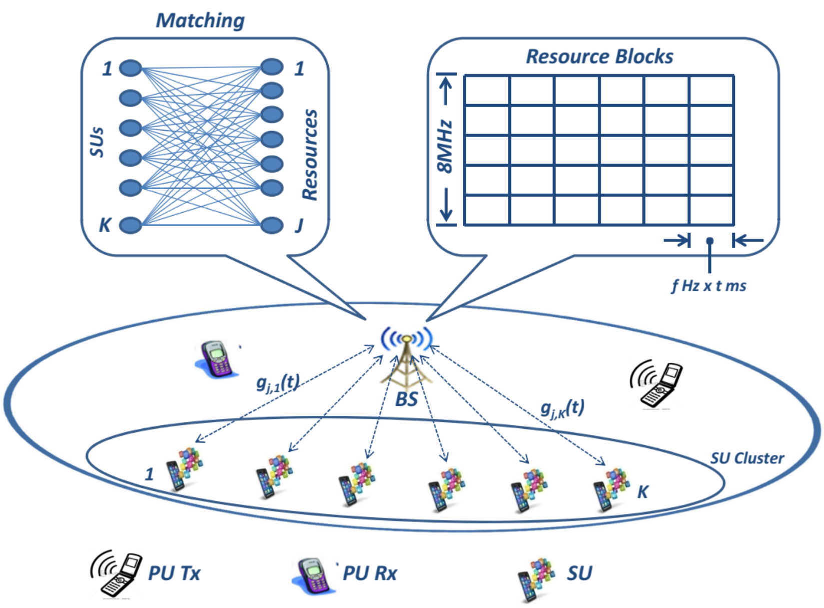

Figure 4.4. Resource Allocation Characterized as a Weighted Bipartite Matching Problem.

As shown in Figure 4.4 above, resources are allocated using a mapping technique through an undirected bipartite graph. Since the CR technique is performed by SUs, and the transmission access and gateway is done by the BS, a bipartite matching strategy in graph theory is applied to optimize the RA. The BS is assumed to utilize a hybrid access scheme where SUs can connect when there are free RBs and employs resource reservation for bandwidth estimation admission control. An effective interference alignment technique to combat the interference among SUs and PUs is assumed. Therefore, the proposed system model is split into two parts which are discussed in the following subsections.

4.4.1 Channel Acquisition and Spectral Efficiency

Since the provision of satisfactory QoS is still a challenging issue due to error-prone wireless channel conditions, user mobility and channel contention, a resource reservation solution is proposed for all SU sessions. The purpose of the resource reservation scheme is to ensure that all QoS sensitive sessions can get sufficient bandwidth in order to sustain high performance. Each SU k can transmit data through PRB j such that the transmission link is represented by $k \to j$, which is denoted as follows:

$$d_{jk} = \sqrt{(x_k - x_j)^2 + (y_k - y_j)^2}, \tag{4.1}$$

which is a Euclidean distance measure where the k^{th} SU and the BS are located at (x_k, y_k) and (x_j, y_j), respectively. In order to increase the chances of SUs completing their transmissions, a resource reservation strategy is proposed where the BS reserves a fraction of the J PRBs for opportunistic SU transmissions, which the SUs can exploit with an optimal transmission power $P_{j,k}$ that is closely controlled by PUs activities. Thus, the spectral efficiency of the k^{th} SU with access to the j^{th} PRB is formulated as follows:

$$R_{j,k} = \log_2(1+\gamma_{j,k}), \quad \text{where} \quad \gamma_{j,k} = \frac{P_{j,k} g_{j,k}}{\sigma^2 + \mathfrak{I}_k}, \tag{4.2}$$

where $R_{j,k}$ is the Shannon bound of the spectral efficiency, and $\gamma_{j,k}$ is the signal-to-interference-plus-noise ratio (SINR) obtained through the designated transmission power $P_{j,k}$, the channel gain $g_{j,k}$, the noise spectral density σ^2, and the interference caused by simultaneous transmission $\mathfrak{I}_k$. The channel gain $g_{j,k}$ is defined as a function of the distance measure in (4.1) and other terms such as the path-loss coefficient $n_{j,k}$, the channel fading coefficient $h_{j,k}$ and the path-loss exponent α_{pl} as given in [264]. In terms of multiple access interference originating from other SU using the same access technique as the k^{th} SU, a linear SINR prediction that employs constant first-order derivatives across adjacent transmissions is assumed. Therefore, according to the proposed resource reservation strategy, the UL rates achieved by SUs can be obtained as follows:

$$R_{j,k} = \frac{\xi \varsigma^{th}}{\lambda} \log_2\left(1+\gamma_{j,k}\right), \tag{4.3}$$

where $\frac{\xi \varsigma^{th}}{\lambda}$ is the fraction of resources reserved for SUs which represent the admission condition for each SU that requests a connection. The terms ξ and ς^{th} are computed as follows:

$$\xi(\gamma) = \log(1+\gamma) - \left(\frac{\gamma}{1+\gamma}\right)\log\gamma, \quad \varsigma^{th}(\gamma) = \frac{\gamma}{1+\gamma}, \tag{4.4}$$

where the term γ is the most influential parameter as a factor selected to adapt the channel quality conditions to the QoS.

4.4.2 The BS Server Opportunistic Computing

The BS is assumed to use the OpenFlow software defined networking (SDN) standard defined in [177], which is used to monitor network information and subsequently decide on best configurations to be applied in the entire CRN. The design objective is to maximize the CR system performance with respect to throughput while reducing infrastructure energy consumption. For the objective of this work, short period prediction is a relevant due to the nature of current day multimedia applications. Thus, the tuning of parameters of the proposed prediction model is very crucial to achieving accurate short period prediction. For example, the incoming input rate (e.g., Megabit per second), the ARIMA, LSTM, and SAE models are used for predicting. These parameters are the amount of data needed to identify the model (i.e. training set), the number of last observations of the throughput (i.e. lag) needed as inputs for the model, the data granularity, variance and packet size distribution. The high-level representation of traffic classification and decision-making life cycle is shown in Figure 4.5 below.

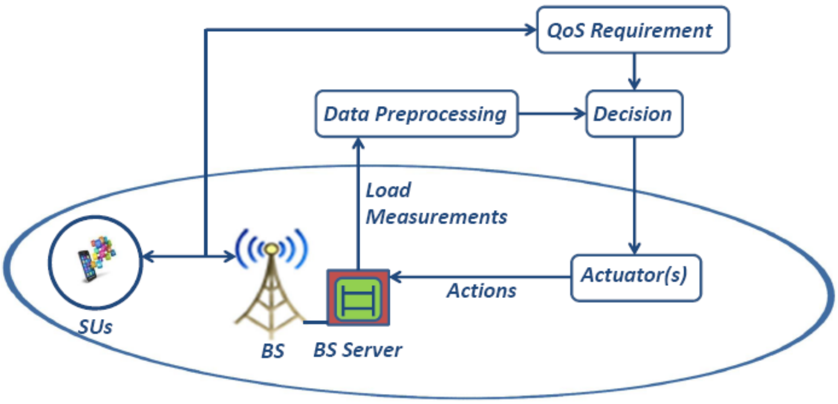

Figure 4.5. A High-level View of the Decision-making Life Cycle as a Model-based RL Formulation.

In Figure 4.5 above, the circled SUs represent all the SUs in the cluster where the BS server operates using model-based RL, with an actuator that estimates the state of the processor using MDPs. The traffic streams from SUs consist of a variety of QoS requirements and the current load measurements are the inputs used by the actuator to make energy-efficient decisions and the subsequent transmission actions. This completes a model-based RL strategy whose process feeds the RL portion of the algorithm, which determines, in a single look-ahead, which possible scheduling scheme would provide the most effective energy saving, QoS, cost and response. The optimization of the system performance and energy consumption involves performance specifications that are measured such as the traffic load measurements and QoS requirements which form the system state $x \in X$. Therefore, as seen in Figure 4.5 above, the BS has to make energy-saving decisions based on the following dynamic equation

$$\hat{x} = \phi(x(t), u(t)), \tag{4.5}$$

which describes a continuous-time non-linear input affine system, where $x(t)$ is the state of the system defined by the load-dependent power consumption $P(\upsilon, \rho, t)$, where υ denotes the BS switching mode, ρ is the system utilization, and the term $u(t)$ represents the control input to the system, which are the decisions made on which scheduling scheme to be used. In order to achieve the objective of the study, one must be cognizant that the main cause of high energy consumption in wireless networks is BS operation, whose process has proved difficult to manually optimize because of the complexity of the interactions of the equipment necessary for its operation. The operation of model-based RL was discussed in Chapter 2, Section **2.5.2** and Section **2.5.4**.

4.5 MATHEMATICAL PROBLEM FORMULATION

Since the rules and heuristics needed for every scenario that ensures everything from efficient RA to energy efficiency are difficult, particularly when interactions with the immediate environment are considered. In this chapter, each of the parameters required to ensure energy-efficient RA will be formulated individually and then and then used in the DL predictive control scheme. Thus, the optimization problem for SU UL capacity maximization is formulated with the resource reservation technique whereby the BS dynamically reserves resources for SUs based on the number of PUs currently being served. The formulation of the optimization problem is as follows:

$$\mathbf{P1} = \arg\max_{\mathbf{P}} \sum_{j=1}^{J}\sum_{k=1}^{K} R_{j,k}, \quad \text{subject to} \tag{4.6}$$

$$\mathbf{C1}: \sum_{j=1}^{J}\sum_{k=1}^{K} P_{j,k} \leq P_{max}, \quad \forall k \in K \tag{4.7}$$

$$\mathbf{C2}: Pr\left\{\sum_{k=1}^{K} P_{j,k}|g_{j,k}|^2 > I_{th}\right\} \leq \delta, \quad \forall k \in K \tag{4.8}$$

where $\mathbf{P}$ is the set of all individual UL transmission powers $P_{j,k}$, which for all $k \in K$, is limited by a power constraint P_{max} given in (4.7). This is a maximum allowed power which is set individually for each SU due to the power falloff of $d_{jk}^{-\alpha_{pl}}$ with distance d_{jk}. The constraint in (4.8) denotes that the probability that the interference threshold I_{th} is exceeded must not exceed δ.

4.5.1 Traffic Load and Power Consumption

Every BS activity has required power consumption implied in its energy consumption, thus, it is assumed that the power consumed by the BS belongs to the following classes: $P_{on}(t)$, which represents the load-independent power consumption, $P_{tx}(t)$, which represents the load-dependent total transmission power, and $P_{server}(t)$, which is the load-dependent computational power consumption of the server. Therefore, the total BS power consumption is obtained as a combination of these classes as follows:

$$P(\upsilon,\rho,t) = \upsilon(t)P_{on}(t) + P_{tx}(t) + P_{server}(\rho,t), \tag{4.9}$$

where $\upsilon(t) \in \{\varepsilon, 1\}$, $\varepsilon \neq 0$ is the BS switching status indicator; 1 for active mode and ε for power saving mode, $\rho(t)$ is the maximum server utilization factor at time slot t. In fact ε is the operational, load-independent power consumption representing the normal BS power expenditures which consist of baseband processing, conversion, cooling, etc. The term $P_{tx}(t)$ represents the load-dependent total transmission power from the BS to the served SUs. The term $P_{server}(\rho,t)$ denotes the load-dependent computational energy consumption of the server which is defined as follows:

$$P_{server}(\rho,t) = P_{idle}(t) + \rho(t)P_{comp}(t), \tag{4.10}$$

which describes the QoS part where the BS server dynamically adjusts its computational power based on the current traffic load and demand. The term $P_{idle}(t)$ is the server load-independent operational component, and $P_{comp}(t)$ is the maximum power consumed by the server when operating at full power. Assuming that the BS computational resources can be tuned, the term $P_{comp}(t)$ is linearly scaled with respect to $\rho(t)$. The term $\rho(t)$ denotes the slope of the trajectory that quantifies the load dependence. Therefore, $P(\upsilon,\rho,t)$ weighs the energy consumption due to the BS transmission and server computation. Since a single BS is considered, an optimization weight α is employed and the corresponding weighted cost function is defined as follows:

$$J(\upsilon,\rho,t) \triangleq \bar{\alpha}P(\upsilon(t),\rho(t),t) + \alpha(\varphi(t) - \rho(t))^2, \tag{4.11}$$

where $\bar{\alpha} \triangleq 1-\alpha$, $0 \leq \alpha \leq 1$, the quadratic term $(\varphi(t)-\rho(t))^2$ accounts for the QoS cost, where $\varphi(t) = \ell(t)/\ell_{max}$ is the approximation of the normalized BS load at time slot t, as given in [265]. Hence, over the finite horizon $t = 1, \cdots, T$, the following optimization problem is defined as follows:

$$\mathbf{P1}^* : \min_{\upsilon,\rho} J(\upsilon,\rho,t), \quad \forall t \in T \tag{4.12}$$

subject to

$$\begin{aligned} &\mathbf{C1}^* : 0 \leq \rho(t) \leq 1, \\ &\mathbf{C2}^* : \upsilon(t) \in \{\varepsilon, 1\}, \\ &\mathbf{C3}^* : I_{max} \geq I(t), \end{aligned} \tag{4.13}$$

then the vectors υ and ρ contain control actions for the considered time horizon T, i.e., $\upsilon = [\upsilon(1), \upsilon(2), \cdots, \upsilon(T)]$ and $\rho = [\rho(1), \rho(2), \cdots, \rho(T)]$. The constraint $\mathbf{C1}^*$ specifies the server utilization factor bounds, $\mathbf{C2}^*$ specifies the BS operation status, $\mathbf{C3}^*$ forces the required number of VMs, I_{max}, to be always greater than or equal to a minimum number $I(t) \geq 1$.

4.5.2 The Overall Optimization Problem

In order to achieve an optimal RA strategy, the transmission collisions between PUs and SUs may result when the interference caused by SUs exceed δ as shown in (4.8). In order to meet this condition, the RA problem has to be reduced to a bipartite matching problem. The advantage of this formulation is that globally optimal power allocations can be effectively computed for a variety of system-wide objectives and SU QoS constraints. Therefore, a single objective bipartite graph to realize the bipartite matching technique is constructed as illustrated in Figure 4.6 below [266].

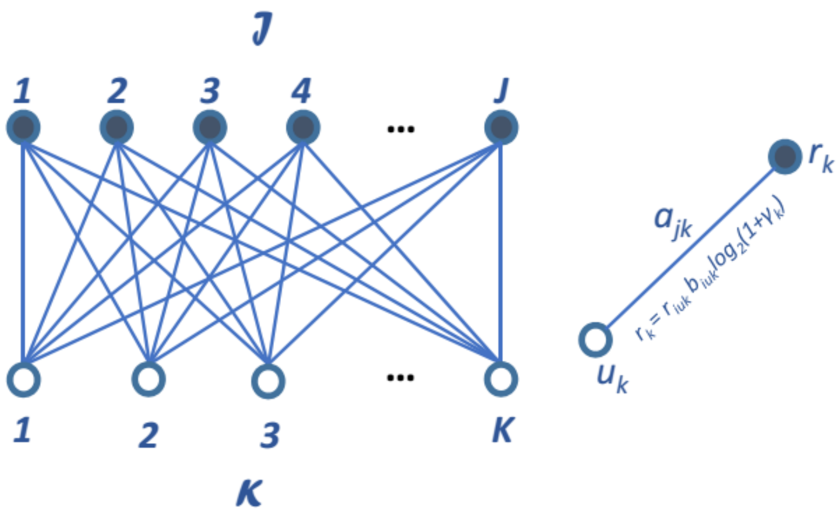

Figure 4.6. An Unconstrained Weighted Bipartite Graph Representation of Resource Allocation.

As shown on Figure 4.6 above, on the left is the bipartite graph showing the whole matching process, while on the right is a graph edge showing individual allocation. In this illustration, the K SUs form a set $\mathcal{K} = \{1, \cdots, K\}$ and on the opposite side, the J PRBs form a set $\mathcal{J} = \{1, \cdots, J\}$; $\mathcal{K} \leq \mathcal{J}$. This constitutes a two-dimensional mapping problem where each SU may want a number of PRBs such that the inputs to the system are the number of SUs and the number of PRBs. Thus, during the RA process, it is paramount to consider certain parameters such as rate requirements and the weight a_{jk} of the link which will be explained in detail in Section **4.6.1**. Therefore, since the spectrum resource is ξ, the formula for data rate R_{jk} can be formulated to the system data rate as follows:

$$R_{sys} = \sum_{i=1}^{K} R_{jk}. \tag{4.14}$$

In order to maximize (4.14), it is required that all the possible $K!$ combinations between SUs and PRBs are tried out. At this point, the weight of each connection link between the SU and the BS is calculated using a bipartite graph as follows:

$$w_{jk} = \xi \left\{ J \times K \times \frac{R_{jk}}{\sum_{j=1}^{J} \sum_{k=1}^{K} R_{jk}} \right\}, \tag{4.15}$$

where $J \times K$ is used to normalize the mean value of w_{jk}. The higher the value of w_{jk}, the higher the data rate attained, thus the optimal RA can be described by the following optimization problem:

$$\mathbf{P1}^{**} : \arg \max_{j \in \{1,2,\cdots,J\}} \left(\sum_{k=1}^{K} w_{jk} \right) \tag{4.16}$$

subject to

$$\begin{aligned} &\mathbf{C1}^{**} : J \geq K, \quad \forall q_i, q_j \in \{1,2,\cdots,K\}, \\ &\mathbf{C2}^{**} : \{q_i, q_j\} \leq \{r_1, \cdots, r_J\} \end{aligned} \tag{4.17}$$

whereby assuming that the bipartite graph is perfectly symmetric, the original graph can be solved using a Hungarian matching algorithm.

4.6 PROPOSED SOLUTION FORMULATION

When an SU requests an UL connection, the BS checks that if by accepting the new SU connection its meets its admission control condition, availability of resources and interference constraints. Transmission resources are afforded to SUs only if these conditions are met, but mostly this resource allocation is determined by the number of SUs already in the system. To obtain the new admission condition for a newly arriving SU, the number of SUs is increased by 1 and this new admission condition is compared with an admission bandwidth threshold, ξ^{su}, $\xi_{new} \geq \xi^{su}$, which is set to a minimum equi-spaced sub-channel per SU. This admission constraint is imposed to protect the integrity of PU transmissions, then the admission control probability ϕ for SUs within the BS coverage can be represented as follows:

$$\phi = \left\{ \begin{array}{ll} 1, & \xi_{new} \geq \xi^{su} \\ 0, & \text{Otherwise}, \end{array} \right\} \tag{4.18}$$

with $\xi_{new} = \frac{\xi \varsigma^{th}}{\lambda+1}$, where the denominator is the arrival rate λ in (4.3) increased by 1. Then, the total number of SUs attempting to connect to the BS can be given by

$$\lambda = \frac{\xi \varsigma^{th}}{\xi_{new}} - 1 = \xi \varsigma^{th} (\xi^{su})^{-1}. \tag{4.19}$$

So, if $0 \leq \frac{\xi \varsigma^{th}}{\xi_{new}} - 1 \leq \xi \varsigma^{th} (\xi^{su})^{-1}$, it means the BS is under-loaded, thus the admission probability equals 1. If $\frac{\xi \varsigma^{th}}{\xi_{new}} - 1 > \xi \varsigma^{th} (\xi^{su})^{-1}$, the BS is overloaded and any new SU connection request will be rejected. This results in a new admission probability of $\frac{\xi\varsigma^{th}}{\lambda+1}$. In this case, the optimal transmission power and channel power gains per SU are obtained using GP.

4.6.1 The Bipartite Matching Strategy for RA

This formulation constitutes a two-dimensional mapping problem where each SU may want a number of PRBs such that the inputs to the system are the number of SUs and the number of PRBs. Thus, during the allocation process, it is paramount to consider certain parameters such as the weights of rate requirements and a graph data structure is considered for this problem, as outlined below.

1. Denote the $\mathcal{K}$, $\mathcal{J}$ as bi-partition sets, $\mathcal{K} = \{1, 2, \cdots, K\}$ and $\mathcal{J} = \{1, 2, \cdots, J\}$ are considered as two independent sets as discussed above.
2. Initiate labels u_j and v_j by $u_j = \max_{\forall i} w_{ji}$, that are supposed to support the SU data rates are labeled $r_1, r_2, \cdots, r_J$, for the satisfaction of the k^{th} SU, the bandwidth demand b_i can be matched to one or more of the available radio units as illustrated in Figure 4.7 (a) below:

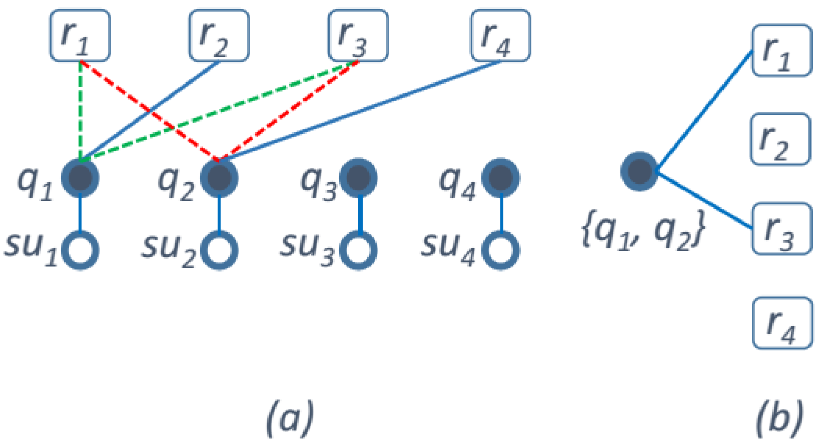

Figure 4.7. Multi-objective Oriented Bipartite Graphs and Evolving of Concepts to Facilitate RA.

This formulation is supported by Hall's theorem [267], where based on the SU (or traffic) demands RA by satisfying different design criteria, without loss of generality, q_i and q_j. In order to achieve RA for the the bandwidth demands q_i and q_j, an example of the solution is shown in

Figure 4.7 above, where the objectives are collected to form vectors of rate request, $\{q_i, q_j\}$ for SU traffic k, where the candidate PRBs must satisfy both criteria.

3. Construct a subgraph as in Figure 4.7 (b), which is a straightforward generalization of the bipartite graph matching in Figure 4.6 to a multi-dimensional matching, where Figure 4.7 (a) shows the RA using a bipartite graph and Figure 4.7 (b) is a solution of the bipartite matching problem by constructing a regenerative bipartite graph.

4. The weight $a_{kj} \in [0,1]$ is then added to the links connecting the SU to the corresponding PRBs, such that the number of PRBs consumed by each SU is represented as follows:

$$r_k = \sum_{j \in J} a_{kj} b_{iu_k}, \qquad b_{iu_k} = \frac{b_{min}}{b_{max}}, \tag{4.20}$$

where the term b_{iu_k} is the number of consumed PRBs, b_{min} and b_{max} are the minimum required long-term rate which represents the rate QoS class requested by SU k and the maximum required rate, respectively.

5. The number of PRBs consumed is regarded as the load efficiency, which is the resource consumption efficiency, using (4.15) and (4.20) is given as

$$\rho(t) = \frac{R_{jk}(t)}{r_k(t)}, \tag{4.21}$$

which is the long-term rate achieved by SU k.

4.6.2 Efficiency of the Bipartite matching Algorithm

The bipartite matching algorithm, which is an efficient matching approach in graph theory, provides an optimal RA solution with a polynomial complexity of $\mathcal{O}(K^3)$ compared to the $\mathcal{O}(K!)$. Given a finite bipartite graph $\mathcal{G}$ with vertices that can be partitioned into two disjoint sets $\mathcal{K}$ and $\mathcal{J}$, such all of its edges $\mathcal{E}$ connect a vertex in $\mathcal{K}$ to one in $\mathcal{J}$. A matching is a subset of edges $K \subseteq \mathcal{E}$ no two members of which share a common vertex, where K is the cardinality of the node set of the bipartite graph.

For example, if data rates r_1, r_2, and r_3 satisfy demand q_i, and data rates r_1, r_3, and r_4 satisfy demand q_j, the request vector $\{q_i, q_j\}$ for SU_1 and SU_2 traffic only connects to r_1 and r_3. This is the intersection of sets formed by PRB matching both criteria for SU_1 and SU_2 as shown in the regenerative bipartite graph in Figure 4.7 (b). In this way, a regenerative bipartite graph is obtained for multi-objective RA, using a technique called preference requirement which states that for any application that the SU is running, there is a preference requirement of data rate from the corresponding vertex in set $\mathcal{K}$ to set $\mathcal{J}$.

4.7 MODELING THE PREDICTIVE CONTROL SCHEME USING DEEP LEARNING

Supposing that a naive method is applied at the current time t to predict the traffic load and then decide an appropriate trade-off between QoS provisioning and energy saving to be applied at time $t+1$. Referring back to Figure 4.5, let the BS server be able to save the traffic profile for the previous and current time slots, $t-1$ and t, respectively, for it to be able to predict the traffic load for the time $t+1$ by curve-fitting and approximates its moving trend. Assuming that the BS' static energy and server utilization can be tuned to scale the server dynamic power consumption in proportion to the expected traffic load for the next time slot, it can make the necessary trade-off between QoS provisioning and energy saving.

Using the predicted value of the traffic load together with the values associated with the maximum BS capacity $\varphi(t)$, the energy consumption cost function $J(\upsilon,\rho,t)$ in (4.12) can be solved. To solve the cost function $J(\upsilon,\rho,t)$, an online management technique is employed whereby the cost function is treated as a Lyapunov candidate [268]. Therefore, using the Lyapunov technique, the BS processor is treated as a hybrid switching system where $J(\upsilon,\rho,t)$ is associated with a search of an optimum operating state, which when reached is maintained until parameters are updated. Then, a time series prediction technique can be applied to learn the function that maps a sequence of past observations as input to an output observation. But before that, the control actions that need to drive the time series prediction model have to be derived.

4.7.1 Derivation of Control Actions

In this subsection, using model-based RL and the Lyapunov techniques, the control actions that will drive the time series prediction model are derived. Assuming that the control actions to drive the model in the next time slot $t+1$ are $\rho(t) \triangleq (\upsilon(t),\rho(t))$, the system state vector which contains the inputs to the time series model at time slot t is denoted by $x(t) = (\rho(t),\varphi(t))$. Here, the cost function $J(\zeta,\rho,t)$ is associated with reaching a certain state and maintaining it until the duration Δt elapses.

This means that, in the sequel, the presentation $(\upsilon(t), \rho(t), t)$ is dropped to make way for $(x(t+\Delta T))$ in the sequel. Now, if the upcoming time slot is represented as $t+1 = t+\sum_{t=1}^{T}\Delta t$, which denotes the next transmission time interval where the predicted system state is $x(t+1)$, becomes the current state information. Taking the optimal cost function $J(x(t))$ as a Lyapunov candidate and the input trajectory as $u(t)$, then $\Delta t = \sum_{t=1}^{t}\Delta t < T$, $1 \leq t \leq T$ is denoted as the time between two decision time steps, i.e., $d(t)$ and $d(t+1)$, as exemplified by the illustration in Figure 4.8 below.

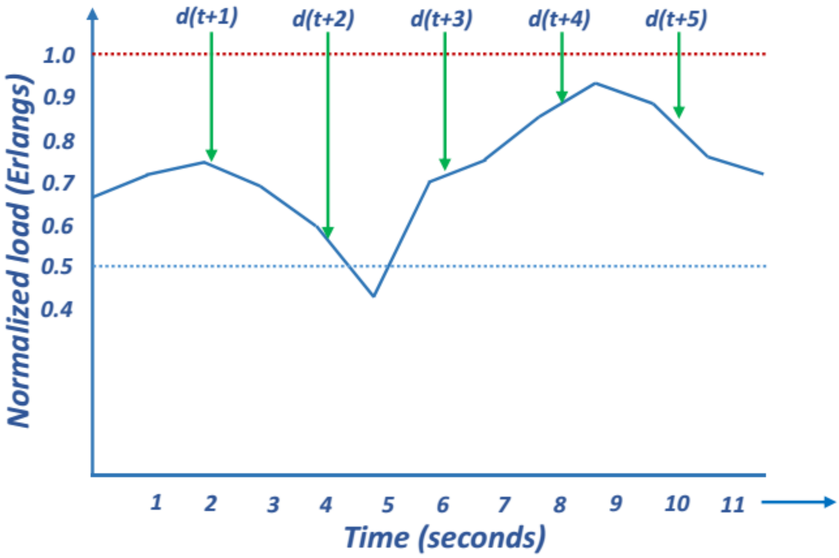

Figure 4.8. Graphical Representation of the Time Series Prediction and Decision Steps.

With reference from the graphical representation of the auto-scaling in Figure 4.8 above, the system has to take serious concern when the load exceeds the 0.5 mark which is indicated by the broken line. Assuming that the probability that the traffic flow information of the space-time points causes the future traffic flow, then given the traffic load $\rho(t)$ at time step t, the decision-making step be given as $\tau(t)$. The decision for the next transmission time is performed, the control $\varsigma(t)$ is applied at the beginning of the next time slot, whereas the offered load $\rho(t)$ is accumulated during the time slot and its value is only known at the end of each time slot. Then, it means that the decision for the next time slot $t+1$ is made at the end of time-slot t. The estimated system state for time-slot $t+1$ is given as follows, with more details in **Appendix A**.

$$x(t+1) = \phi(x(t), \upsilon(t)), \tag{4.22}$$

where $\phi(t)$ is the behavior model that captures the relationship between $(x(t), \upsilon(t))$ and the next state $x(t+1)$.

4.8 EXPERIMENTAL SET UP

The experimental set up step, which is similar for the auto-regressive integrated moving average (ARIMA), the LSTM and the proposed SAE are summarized as follows:

- Load and normalize the data set.
- Split data set into training and testing.
- Reshape input to be in the form of samples, time steps, features.
- Create and fit the predictive network.
- Make predictions and calculate performance measures.

A detailed analysis of the set up steps is outlined in the following subsections.

4.8.1 Data Collection, Pre-processing and Normalization

In this section, a traffic flow data set was obtained using a traffic flow simulator in [270], aggregated it into data points separated by 1 second for 90 seconds (each for IN and OUT traffic data). The traffic streams for K SUs were represented in the form of a traffic flow matrix with a history of n time slots as follows:

$$\mathbf{S}^f = \begin{pmatrix} S_1 \\ S_2 \\ \vdots \\ S_K \end{pmatrix} = \begin{pmatrix} S_1(t-n) & \cdots & S_1(t-1) \\ S_2(t-n) & \cdots & S_2(t-1) \\ \vdots & \vdots & \vdots \\ S_K(t-n) & \cdots & S_K(t-1) \end{pmatrix}. \tag{4.23}$$

whose rows represent the historical traffic flow data during the previous n time slots. Thus, this traffic flow matrix, together with the system state and controls are used as the input data for the predictive model to generate predictions for the time horizon T, i.e., $t, t+1, \cdots, t+T$.

4.8.2 Data Pre-processing

In this section, an MLP with a nonlinear activation function, the logistic sigmoid, which has the squashing role in restricting from a node to (-1, 1), was used. This is is represented using attention matrix **A**, given as follows:

$$\mathbf{A} = \Phi\left(W(\mathbf{S}^s) + \zeta\right), \tag{4.24}$$

which can be interpreted as the probability that the traffic flow information of the space-time points causes the behaviour of future traffic flow. The term Φ is the sigmoid activation function between outputs and the hidden layers of the neural network, which is monotonic, continuous and differentiable, given as follows:

$$\Phi(b) = \left(1 + e^{\kappa(b_{max} - b_{min})}\right)^{-1}, \tag{4.25}$$

where $\kappa \in [1, -1]$ is a constant that determines if the function is increasing or decreasing, and b_{min} is the QoS metric of a given service and determines the absissa shift of the function, whereby the absissa is the QoS metric b_{max}, both given in (4.20). The term W is the weight that acts as a projection and vectorization of the speed matrix $\mathbf{S}^{\mathbf{s}}$ between the input and the hidden neurons, the speed matrix $\mathbf{S}^s$ has the same size as $\mathbf{S}^f$ in (4.23); and the term ζ is the bias variable without which a given layer will not be able to produce output in the next layer that differs from 0. Then, the traffic flow matrix $\mathbf{S}^f$ is point-wise multiplied by the attention matrix **A** to obtain a weighted traffic flow matrix $\mathbf{S}^{\mathbf{A}}$ for the application of the DL procedures. Then using feature mining, we obtain server instances of the applications. By server instances, we mean a set of features/attributes representing a specific occurrence of the problem.

4.8.3 Training, Testing and Validation

During training and testing, the data set was split into 30% training and 70% testing data sets, respectively. The variations of the traffic load were aggregated in seconds on average of kilobits per second. Then, it was reshaped, services and jobs put into a job queue, extracted, processed and the service priorities made available in the priority queue and decisions for the next transmission slot are performed based on these priorities. The goal of the SAE is to use a speed matrix $\mathbf{S}^s$ of the space-time points of $\mathbf{S}^f$ to learn attention-weight matrix **A**. Thus, given the traffic load $\rho(t)$ at time stamp t, the decision-making step be given as $\tau(t)$ leads to a decision that should make sure that enough resources are allocated to serve the traffic until the next decision step. This is the number of virtual machine instances that need to be turned on to served the traffic classes, such that the number of virtual machines allocated define the instance of a class. Here, a fully connected SAE, consisting of two encoders, each encoder consisting of a single hidden layer as shown in Figure 4.9 below.

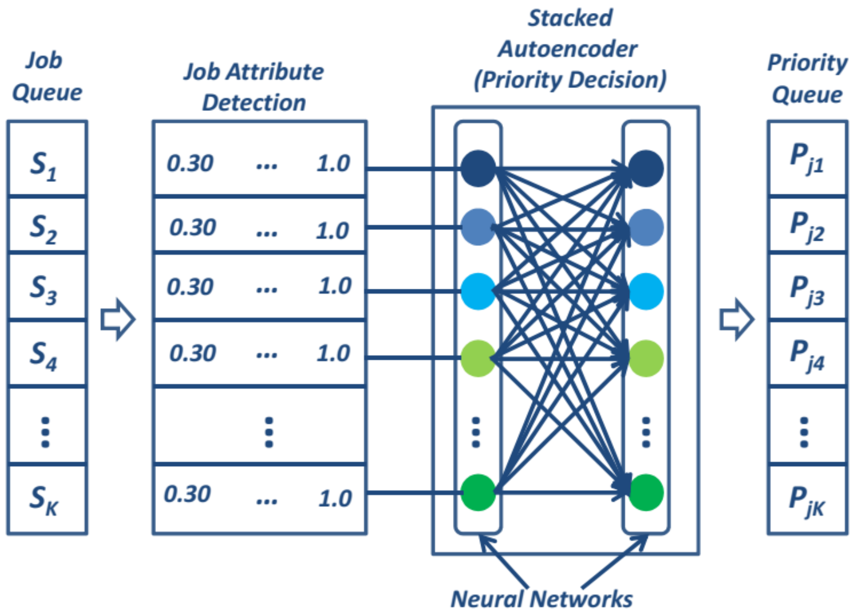

Figure 4.9. SAE with Job Queue Attribute Detection and Priority Queue.

The service attributes are obtained from the job queue, i.e., resources (b_{iu_k}, b_{max}), deadline, and processing time. The scheduler performs all sets of combinations of the application environment mentioned in the service attributes. All attributes per service per user are entered with calculated weights and placed into priorities. Assign the current value of $\rho(t)$ into a system state $x(t) = [\rho(t), q(t)]$ as the input vector that drives the system behavior at time t. Initialize the cost function $J(\upsilon, \rho, t)$ to zero; then begin a breadth-first search by building a tree of all future states up to the prediction depth T, i.e., $\hat{x}(t+1), \cdots, \hat{x}(t+T)$. Accumulate the cost as the search for future states travels through the tree, accounting for predictions and past outputs. Create a state-space $\mathcal{S}(t+n)$ using the set of states reached in every prediction depth $t+n$. For every prediction depth $t+n$, the search continues from the set of states $\mathcal{S}(t+n-1)$ reached at the previous step $t+n-1$, exploring all possibilities of obtaining the next system state. Update the accumulated cost as the result of the previous accumulated cost, plus the cost associated with the current time step $t+n$. When this exploration has finished, select the action at time $t+T$ that leads to the best final accumulated cost as the optimal operating value, as done in **Appendix A**.

4.8.4 Calculation of Performance Measures

Two ways of exploring the state space are used in this paper, one is the random technique, which uses the randomness of a random tree, while the other one uses exploration technique from RL technique. The **Random Technique** proceeds as follows: Let t be the current time, and $\rho(t+n-1)$ be the predicted traffic load in time slot $t+n-1$, $n = 2,\cdots,T$. It then performs a random prediction as follows: If the expected load difference $\rho(t+1)-\rho(t) > 0$ then the offered load in the next time slot is randomly selected in the range $[0.5,1]$, otherwise it is selected evenly from the range $(0.0,0.5)$. This is an exploration kind of behavior.
The **RL-based Technique** substitutes the randomness of the exploration of the random tree with an exploitation technique from RL, which is a very naive technique that only cares when the traffic load exceeds 0.5. This means that it selects its prediction in the range $[0.5,1]$, which saves a lot of time compared to the exploration technique.

4.9 PRIORITIZATION AND ENERGY CONSUMPTION PER SERVICE

Once the prediction horizon is fully explored, a unique sequence of reachable states $\hat{x}(t+1),\cdots,\hat{x}(t+T)$ with minimum cumulative cost is obtained. Two decisions have to be made when generating the class values, one prioritizes the QoS while the other prioritizes the energy saving; they are described as follows:

- **QoS prioritization:** This approach prioritizes QoS provision over energy saving, which means it allocates more resources to guarantee QoS. The allocation of more resources reduces the energy efficiency since more server instances will have to be launched at that given time. In this case, QoS requirements are guaranteed while energy saving suffers. In order to guarantee QoS, the decision $d(t)$ taken at time step t considers future traffic changes until time step $t+1$. The service class is generated as follows:

$$d(t) = \max(QoS(\lambda(t))), \forall t \in \{\tau(t), \tau(t+1)\}. \quad (4.26)$$

  The $QoS(.)$ function takes the traffic load measured at time t as input and outputs $I(t)$ which is the number of virtual machines that are required to serve the measured traffic rates without violating the QoS.

- **Energy saving prioritization:** This approach takes priority of energy saving by ignoring short-lived bursty traffic between steps t and $t+1$ to save on the energy consumption that comes with launching many virtual machines, thus accepting short-lived QoS degradation due to the under provision of virtual machines. The service class is generated as follows:

$$d(t) = \max(QoS(\lambda(\tau(t))), QoS(\lambda(\tau(t)))). \tag{4.27}$$

This gives the number of processing units by specifying the scheduling mechanism as the idea is to concentrate the computational resources by choosing the appropriate scheduling mechanism for each service that will not drain too much of the systems' energy. Thus, for the energy consumption per application, the following equation is used

$$EE(\upsilon, \rho) = \frac{\upsilon(t)P_{on}(t) + P_{tx}(t)}{\lambda} + \rho \cdot P_{comp}(\rho, t), \tag{4.28}$$

where λ is the packet arrival rate, $\rho = \lambda\varphi(t)\mathbb{E}[s]$ is load factor, $\varphi(t)$ is the normalized traffic load, $\mathbb{E}[.]$ is the mean service rate of the server, and s is the service time per job. Equation (4.28) supports the principle of concentrating computation on a small number of processing units in order to minimize the server power consumption per application [181]. A simple analytic model that uses the combined energy-QoS cost function includes in its first part the well known Pollaczek-Khintchine formula for $M/G/1$ queue [272] for the average response time, based on Poisson arrivals of jobs and general service time distributions, and in its second part the energy consumption per job, given as follows:

$$C(\upsilon, \rho) = a\mathbb{E}s\left[1 + \frac{\rho(1+C_s^2)}{2(1-\rho)}\right] + b\left[\frac{\upsilon(t)P_{on}(t) + P_{tx}(t)}{\lambda} + m\mathbb{E}[s]\right]. \tag{4.29}$$

Here, $C_s^2 = \sigma_s^2/(1/\mu)^2$ represents the squared coefficient of variation of the service time, where σ_s^2 denotes the variance of the service time and μ is the mean service rate; the constants a and b describe the relative importance placed on QoS and energy consumption, respectively. This allows for the computation of the value of the arrival rate that minimizes $C(\upsilon, \rho)$. Then, the result shown in (4.30) below indicates the optimum setting of the load $\rho^* = \lambda^*\mathbb{E}[s]$ and its dependence on $\upsilon(t)P_{on}(t) + P_{tx}(t)$ and on the ratio b/a, given as follows:

$$\rho^* = \frac{\sqrt{\frac{2b(\upsilon(t)P_{on}(t)+P_{tx}(t))}{a(1+C_s^2)}}}{1 + \sqrt{\frac{2b(\upsilon(t)P_{on}(t)+P_{tx}(t))}{a(1+C_s^2)}}}. \tag{4.30}$$

Equation (4.30) above gives a simple rule of thumb for selecting system load for optimum operation, depending on how the system weighs the importance of energy consumption with respect to average response time. It is also evident that ρ^* increases with the ratio $\frac{b(\upsilon(t)P_{on}(t)+P_{tx}(t))}{a(1+C_s^2)}$, which indicates that the optimum load should increase with the expression $\upsilon(t)P_{on}(t) + P_{tx}(t)$ of the system; the relative importance that one places on energy, and with the squared coefficient of variation of service time. The algorithm for this process is shown in Algorithm **3** below:

Algorithm 3 Proposed Algorithm to Exhaustively Evaluate Operating States Within the Prediction Horizon T.

Inputs: Current state $x(t)$, Prediction horizon T

Outputs: $\varsigma(t)$, $EE(\upsilon,\rho)$, $C(\upsilon,\rho)$

Initialize all inputs: $\mathcal{S}_t = x(t)$
Assign $J(\hat{x}(t)) \leftarrow J(\hat{x}(t-1))$
For $t = 1 : T$ **do**
  Set $\mathcal{S}_{t+1} \neq \emptyset$
  For $x \in \mathcal{S}_t$ **do**
    Predict environment parameters for $t+1$
    Estimate the next state, $\hat{x} = \phi(x,u)$
    Set $\mathcal{S}_{t+1} = \mathcal{S}_{t+1} \cup \{\hat{x}\}$
    Predict the cost $J(\hat{x})$ using (A.3)
    Find $x_{min} \in \mathcal{S}_T$ with minimum cost $J(x)$
    (i.e., $\hat{u}(t) =$ input leading from $x(t)$ to x_{min})
    If $J(\hat{x}) = J(x) - J(\hat{x})$ **do**
      Execute QoS priority using (4.26)
    Else
      Execute energy saving using (4.27)
    End If
    Compute service cost using (4.30)
  End For
  Predict the cost $J(\hat{x})$ using (A.3)
  Return $\boldsymbol{\varsigma}$**(t)**
End For

4.10 SIMULATION RESULTS AND DISCUSSION

To validate our main findings of this chapter, a series of simulations were conducted using $MATLAB^{TM}$, on an Intel(R) Core(TM) i5-4590 @ 3.30GHz processor with 8.00 GB RAM, and resolution of 1920

$\times$ 1080 GPU. The services considered are tabulated in Table **4.1** below.

Table 4.1. CRN Traffic Profiles Considered in this Chapter.

CRN Applications' Traffic Profiles			
Application/Job	**Desired Delay**	b_{min} **bps**	b_{max} **bps**
Audio or VoIP Service	180 ms	30 K	64 K [44]
Online Gaming	150 ms	1 M	4 M [274]
Buffered Video	2 ms	3 M	25 M [275]
Video Conferencing	300μs	256 K	20 M [273]

The features tabulated in Table **4.1** above are used to compute the resource consumption efficiency for each service run by the CR devices using (4.20). Simulation parameters are as shown in Table **5.2** below.

4.10.1 Resource Reservation and Allocation

In this section, it is assumed that some resources that are left for SUs using the resource percentage threshold (RPT), which gives the amount of resources remaining for SUs when PUs are active in their channels. The bandwidth required by PUs is calculated and their required capacity is set to find the bandwidth percentage to be allocated to SUs, which gives the RPT for SUs. After computing the RPT for SUs, the performance on the achievable capacity is evaluated using two algorithms, (i) Dynamic-RPT through GP, and (ii) Dynamic-RPT and bipartite matching through GP with the number of PU changing from six to nine in each case. The achievable capacity is illustrated in Figure 4.10 below. Figure 4.10 above shows results for the RPT for SUs, given the number of PUs and resources allocated to them. It can be observed that the achievable UL capacity for SUs decreases with an increase in the number of SUs being admitted, which suggests that their performance might be limited by interference from other users (PUs and SUs). However, by combining dynamic resource percentage threshold with bipartite matching achieves better performance through the GP solution obtained in (4.6), a better performance is achieved. It is shown that as the number of PUs is increased from six to nine, the RPT reduces from 46.64% to 19.97%, which is a 26.67% difference.

Table 4.2. Simulation Parameters Used in this Chapter.

Simulation parameter	**Value**
Carrier frequency, f_c	2.1 GHz
System bandwidth [SA CRN Standard], ξ	8 MHz
Maximum number of RBs, J	100
BS operating power, P_{on}	40.25 dBm [10.6 W]
BS Transmission power	46 dBm [40 W]
Dynamic maximum power, P_{serv}	56.7 dBm [472.3 W]
Energy consumption at P_{idle}	3 J
Path-loss model	$34.46 + 20\log_{10}(d_{jk})$
Noise power spectral density, σ	2×10^{-11} watts/Hz
Symbol duration, T_s	500×10^3 symbols/sec
Symbol rate	$1/T_s$
Maximum packet arrival rate, λ	60 packets/sec
Minimum and maximum BS load, φ	$[5, 10]$ MB
Maximum decision interval, Δt	1 sec
Minimum number of VMs, $I(t)$	1
Maximum number of VMs, I_{max}	30
Number of input layers	1
Hidden layer activation function	Sigmoidal
Learning rate	0.3
Number of output layers	1

With six PUs and only one SU admitted into the system, the achievable capacity is 318.56 Mbps and 306.16 Mbps at an RPT of 46.64%, which indicates that Dynamic-RPT and bipartite matching performs 3.54% better than Dynamic-RPT alone. As the number of SUs admitted into the system, the achievable capacity decreases such that at eleven SUs, the achievable capacity is 28.96 Mbps and 27.83 Mbps, which is a 0.3% difference between the two algorithms. When the number of PUs is increased to nine, the achievable capacity 136.36 Mbps and 131.05 Mbps with one SU in the system, however, there is only a difference of 0.483 Mbps between the two algorithms at eleven SUs. However, with nine PUs, a more asymptotic behavior is observed as the number of SUs increases,

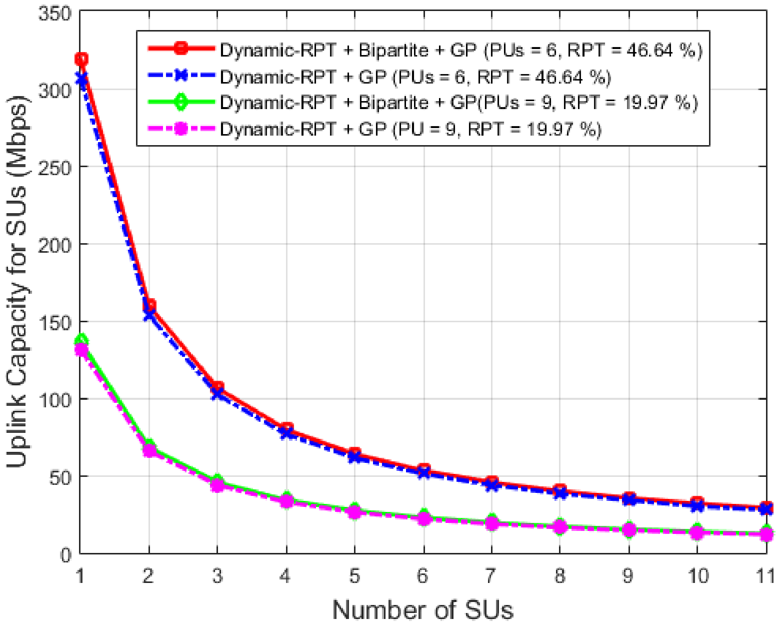

Figure 4.10. UL Achievable Capacity for SUs.

which suggests that the RPT algorithm always allow for resources to be reserved. The percentage decrease in the RPT as a function of an increasing SINR is illustrated in Figure 4.11 below.

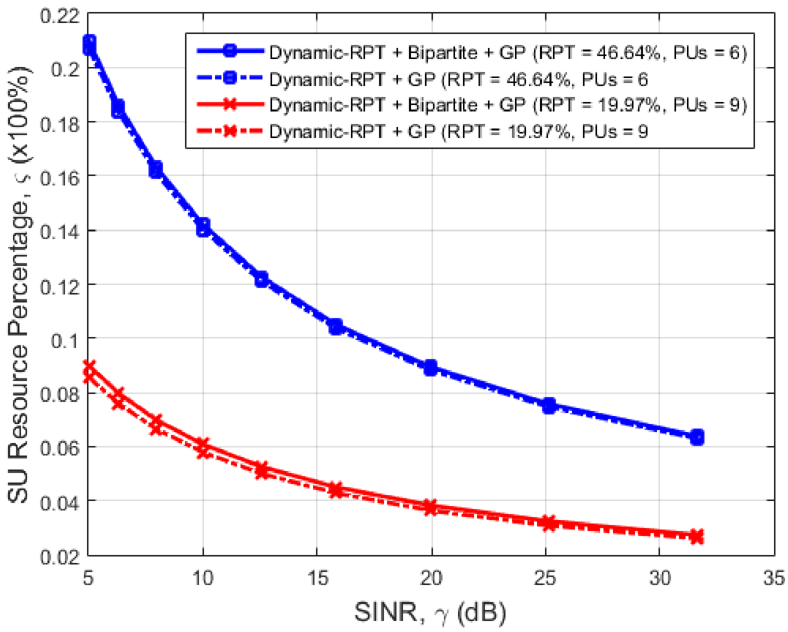

Figure 4.11. Resource Percentage Threshold vs. SINR.

Figure 4.11 above shows the decline in the RPT as the SINR γ increases, and what can be observed is that the RPT decreases rapidly when there are fewer PUs in the system than when there are more PUs. This observation can be explained as follows: with an increase in the number of available channels and good channel conditions, the probability of SUs' channel access increases, which also increases the possibility of SU collisions and is a problem that still needs to be addressed. Furthermore, with a decrease in the number of available channels, even if the channel conditions improve, there are fewer transmission opportunities for SUs due to PU activities and an increase in the collision intensity among SUs. In the same fashion, now the achievable UL capacity per SU as the SINR increases is evaluated in Figure 4.12 below.

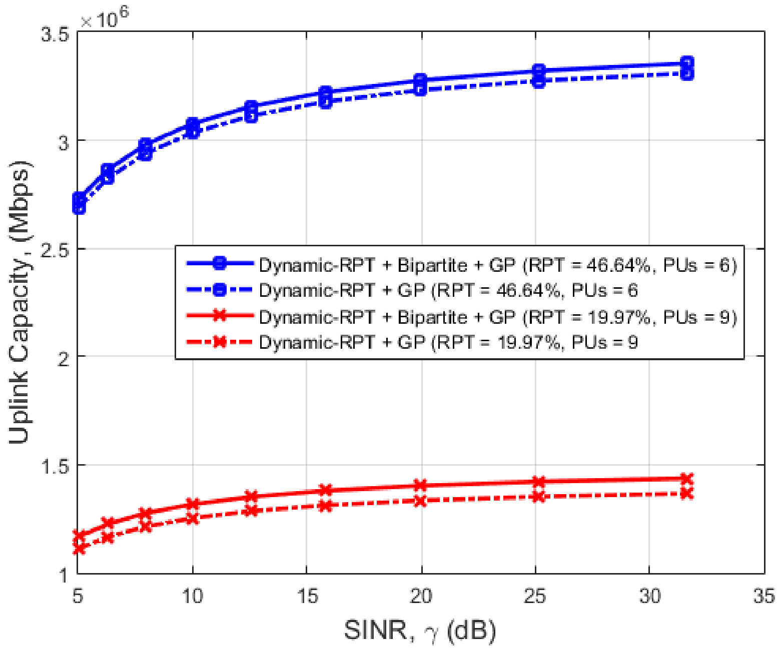

Figure 4.12. UL Achievable Capacity per SU vs. SINR.

In Figure 4.12 above, it is shown that the performance of the UL capacity per SU as a function of an increasing SINR γ; however, there is a performance difference as the number of PUs is increased from six to nine. As it has been the case with the performance shown in Figure 4.10, it is also shown here that the achievable UL capacity per SU decreases. However, the difference between the two algorithms is less when the number of PUs is kept constant. This shows that, even though SUs might be expected to exploit much of the available capacity and achieve higher transmission throughput, this is affected by number of available PUs. This can be explained using the response of the effective system bandwidth as a function of an increasing SINR, which is shown in Figure 4.13 below.

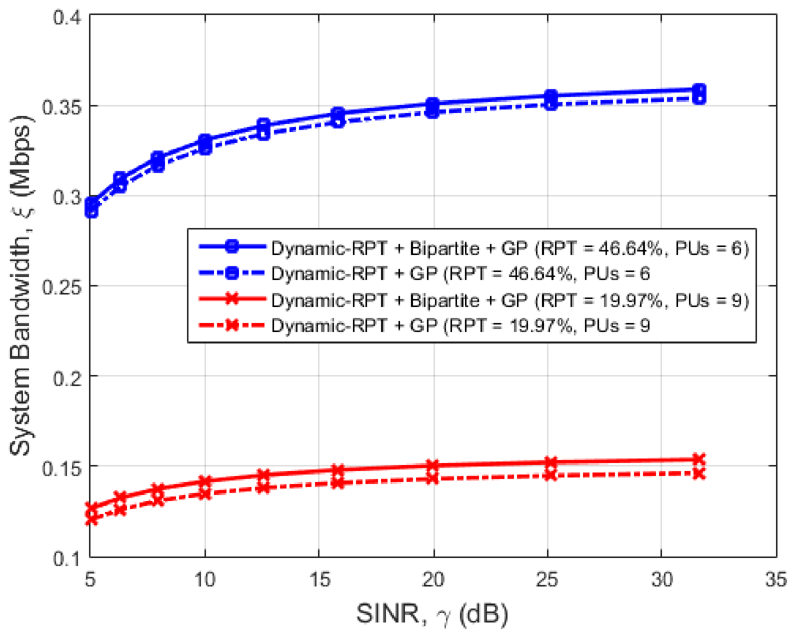

Figure 4.13. System Bandwidth vs. SINR.

Figure 4.13 above shows the response of the system bandwidth as a function of an increasing SINR γ for the two algorithms and changing number of PUs. The system bandwidth decreases drastically with and increase of PUs from six to nine. This result serves to confirm a well-known notion that SUs can have a higher instantaneous throughput when there are more available channels, as more PUs reappear to reclaim their licensed channels and the PU activities together with the increased contention among SUs, creates high liability, since there might not be available channels for some SUs.

4.10.2 BS Traffic Load Prediction

Table **4.3**, Table **4.4** and Table **4.5** below present convergence comparison of the training and validation for three NN architectures used for forecasting/prediction purposes. All three prediction schemes are trained over 110 epochs, each epoch consisting of 2 000 individual training trials, and the RMSE is used as a performance measure of the models. The training and validation results of the ARIMA in performing the same prediction task are shown in Table **4.3** below.

Table 4.3. Training and Validation Results for the ARIMA.

ARIMA											
Training Epoch	10	20	30	40	50	60	70	80	90	100	110
Training loss	0.51	0.41	0.33	0.25	0.21	0.16	0.13	0.12	0.12	0.11	0.11
Validation Loss	0.46	0.33	0.30	0.24	0.20	0.15	0.13	0.12	0.12	0.11	0.11

The training shown in Table **4.3** above was performed over 110 epochs and the training convergence was observed at a loss of 0.11. The same performance evaluation was done for the LSTM as shown in Table **4.4** below.

Table 4.4. Training and Validation Results for the LSTM.

LSTM											
Training Epoch	10	20	30	40	50	60	70	80	90	100	110
Training Loss	0.40	0.32	0.27	0.21	0.19	0.14	0.12	0.12	0.11	0.10	0.10
Validation Loss	0.38	0.30	0.25	0.19	0.16	0.12	0.11	0.11	0.11	0.10	0.10

In Table **4.4** above, it can be observed that the LSTM begins with a training error that is lower than that of the ARIMA and a loss of 0.10 is observed at the end of the training.

Table 4.5. Training and Validation Results for the SAE.

SAE											
Training Epoch	10	20	30	40	50	60	70	80	90	100	110
Training Loss	0.31	0.28	0.23	0.17	0.15	0.12	0.11	0.11	0.11	0.10	0.10
Validation Loss	0.28	0.25	0.21	0.15	0.14	0.12	0.11	0.11	0.11	0.10	0.10

Compared to the ARIMA and LSTM, the SAE architecture performs well with an initial training error of 0.3115 (RMSE), which is a 19.79% and 9.21% superiority over ARIMA and LSTM, respectively. Therefore, compared with the other two architectures, in the sequel, the SAE becomes the architecture of choice because of the lowest training error. The training of the SAE in traffic load prediction is performed where a careful choice of the learning rate is made to be $\alpha = 0.3$ to make the SAE training more reliable. Even though the training and subsequent optimization might take a little longer than it would with a higher learning rate, this is a better choice of the learning rate. A training rate higher than this one proved to be faster, while the convergence is poor because the weight updates become big such that the optimizer overshoots the minimum and makes the training loss more worse. The results for BS load prediction are illustrated in Figure 4.14 below.

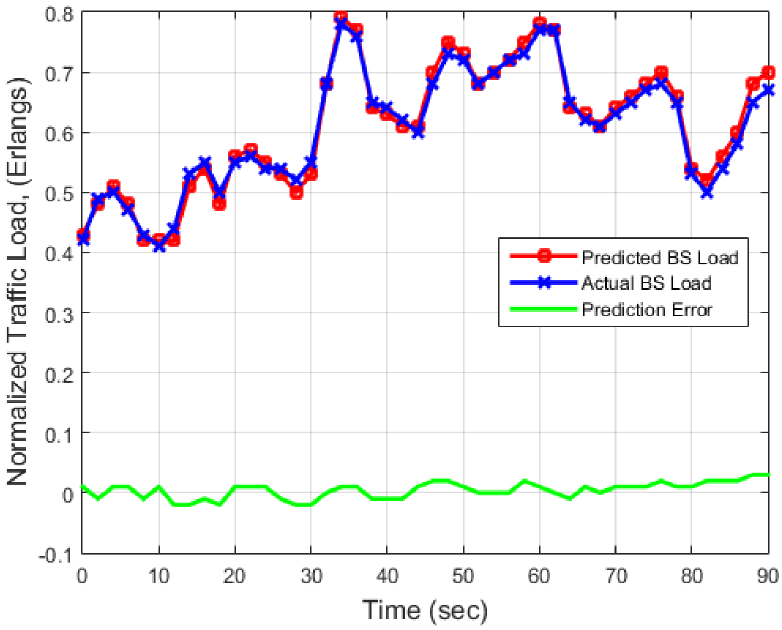

Figure 4.14. BS Load Prediction Using an SAE Architecture.

As shown in Figure 4.14, the BS load pattern prediction results based on the SAE architecture for a time horizon of $T = 90$ seconds. The prediction error shows some stabilization as the training time increases, which indicates a reliability of the training results to be obtained.

4.10.3 Energy Saving and QoS provisioning

When the SAE is use in energy saving with respect to the optimization weight, α, the RL-based technique is compared with the random technique with the load varied from 5 MB and 10 MB, the performance results are shown in Figure 4.15 below.

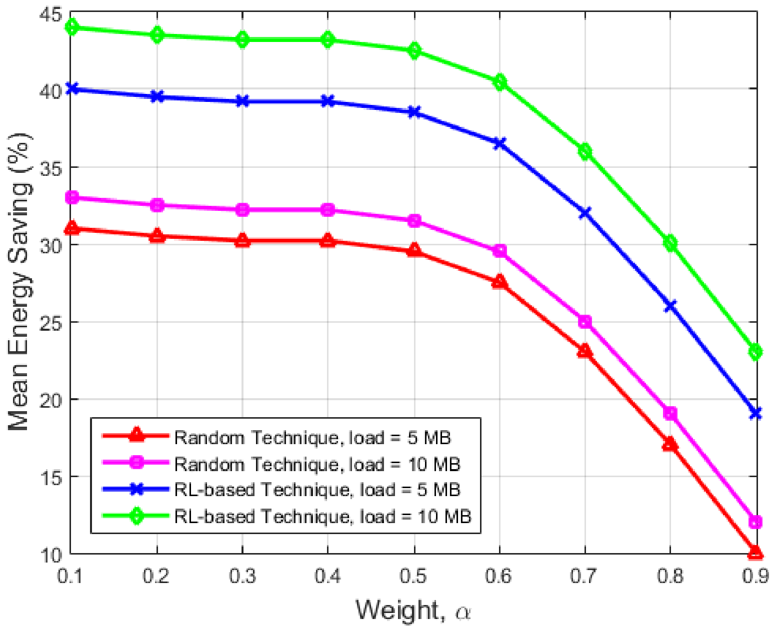

Figure 4.15. Mean Energy Saving as a Function of Optimization Weight α Using SAE.

As shown in Figure 4.15, the performance of energy saving in terms of the traffic load with respect to the optimization parameter α. In the range $\alpha = [0.1, 0.4]$, the impact of the traffic load is clearly visible that as the load is increased (i.e., 10 MB), more energy is being saved compared to when there is less traffic load (i.e., 5 MB). This shows that as more load is allocated to each VM, energy saving increases due to the reduction on the number of VMs required to be turned on. This means that concentrating the load in fewer VMs saves energy than when the load is distributed over more VMs, i.e., when load is 5 MB. Another notable observation is that there is more gain in energy saving with the RL-based technique compared to the random technique when the load is increased from 5 MB to 10 MB. However, in the range $\alpha = [0.4, 1]$, the energy saving drops for both algorithms, which indicates that as $\alpha \to 1$ the priority is placed on QoS than on energy saving and the system can allow for short-lived drops in energy saving.

4.10.4 Effect of Traffic Load on Server Response Times

In this subsection, the evaluation is based on the effect of the traffic arrival on server response times and server energy consumption. This evaluation is, in principle, concentrating on the server computational units' energy consumption. At first, the traffic load is varied, while the service rate is kept constant; then the traffic load is kept constant while the service rate is varied. In both cases the energy per packet decreases as the packet arrival rate increases. The results obtained are summarized in Table **4.6** and Table **4.7**, respectively.

Table 4.6. Energy Consumption per Service, Varying *a* and Keeping *b* at its Minimum.

Energy Consumption per Packet (Joules/bit)									
Traffic Load	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
a = 0.6, b = 0.1	63.556	31.972	21.454	16.187	12.990	10.781	9.075	7.606	6.202
a = 0.7, b = 0.1	63.607	32.025	21.510	16.243	13.040	10.814	9.075	7.549	6.057
a = 0.8, b = 0.1	63.658	32.079	21.566	16.210	13.090	10.847	9.074	7.491	5.912
a = 0.9, b = 0.1	63.710	32.133	21.622	16.356	13.140	10.880	9.074	7.434	5.767

Table 4.7. Energy Consumption per Packet, Keeping *a* at its Minimum and Varying *b*.

Energy Consumption per Packet (Joules/bit)									
Traffic Load	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
a = 0.1, b = 0.6	379.551	189.954	126.756	95.156	76.190	63.533	54.471	47.643	42.288
a = 0.1, b = 0.7	442.801	221.604	147.873	111.006	88.880	74.117	63.549	55.593	49.361
a = 0.1, b = 0.8	506.051	253.254	168.990	126.856	101.570	84.610	72.628	63.543	56.433
a = 0.1, b = 0.9	569.301	284.904	190.106	142.706	114.260	95.283	81.707	71.493	63.505

Table **4.6** and Table **4.7** above show the energy consumption per packet vs. normalized traffic load as a function of the priority placed on QoS and energy-saving, respectively. Table 4.6 shows the performance of the system when *a*, which is the importance placed on QoS, is varied and b, which is the importance on energy consumption, is kept at a minimum of 0.1; the energy consumption is low. This attests to the result obtained in (4.30) in Section **4.9**, which shows that the optimum setting of the load $\rho^* = \lambda^*\mathbb{E}[s]$ will depend on $\upsilon(t)P_{on}(t) + P_{tx}(t)$ and on the ratio b/a.

Table **4.7** above shows the performance of the system when the priority is placed on energy consumption b and the QoS priority is at its minimum. These results illustrate that when the priority is placed on QoS, the energy consumption will increase since the system needs to guarantee QoS by allocating more resources, thus increasing energy consumption and reducing on energy saving. From these results, it can be noted that when the priority is placed on QoS, energy saving decreases as the traffic load increases. This is because when the traffic load increases, the resource consumption efficiency increases as the computational resources have to stay ON for a long time which increases energy consumption, thus reducing the energy saving. However, when the priority is shifted to energy saving, the energy consumption is low as the system ignores the increase in traffic load and allows some minor QoS degradation. Therefore, fixing one parameter, either a and varying b or vice versa allows one to scale the system's response time and the energy consumption per application.

4.10.5 Cost of Energy Consumption with Server Mean Slowdown

Here, the performance of the variation of energy consumption cost as a function of the mean arrival rate λ for both FCFS and PS scheduling mechanisms with server mean slowdown (MS) is evaluated. The number of channels and the price per joule of energy are kept constant while the mean service rate μ is varied. The traffic profiles in Table **4.1** are considered as the QoS parameters, where b_{max} represents the size of the request which is the maximum required bandwidth, and b_{min} represents the benchmark minimum required bandwidth as defined in (4.20).

4.10.5.1 Mean Slowdown First Come First Served

The performance evaluation results for the MS FCFS scheduling mechanism for processing rates $\mu = 0.6$, $\mu = 0.7$ and $\mu = 0.8$ are shown in Figure 4.16, Figure 4.17 and Figure 4.18, respectively.

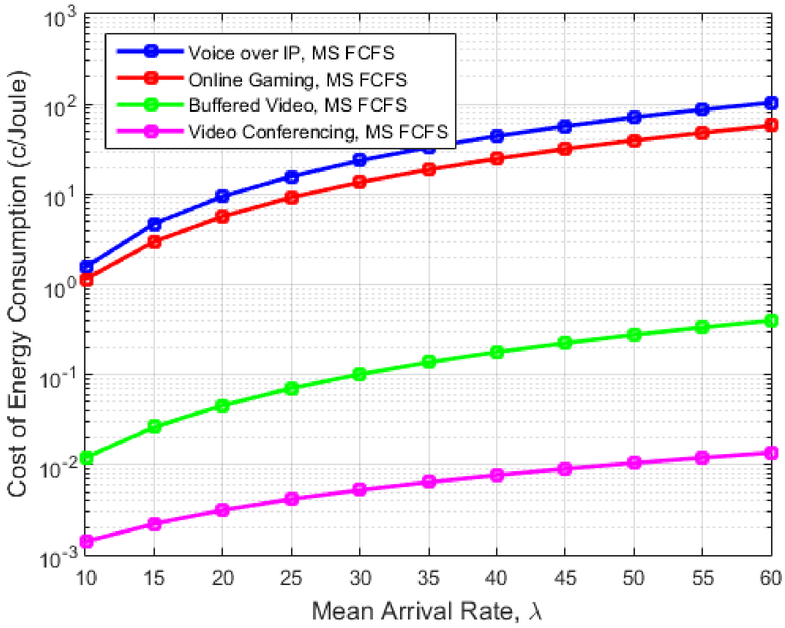

Figure 4.16. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.6$.

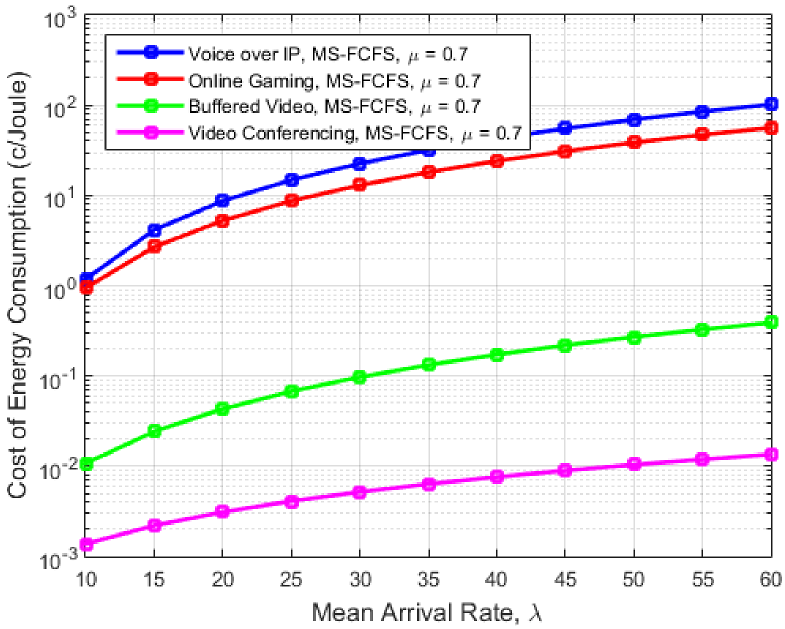

Figure 4.17. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.7$.

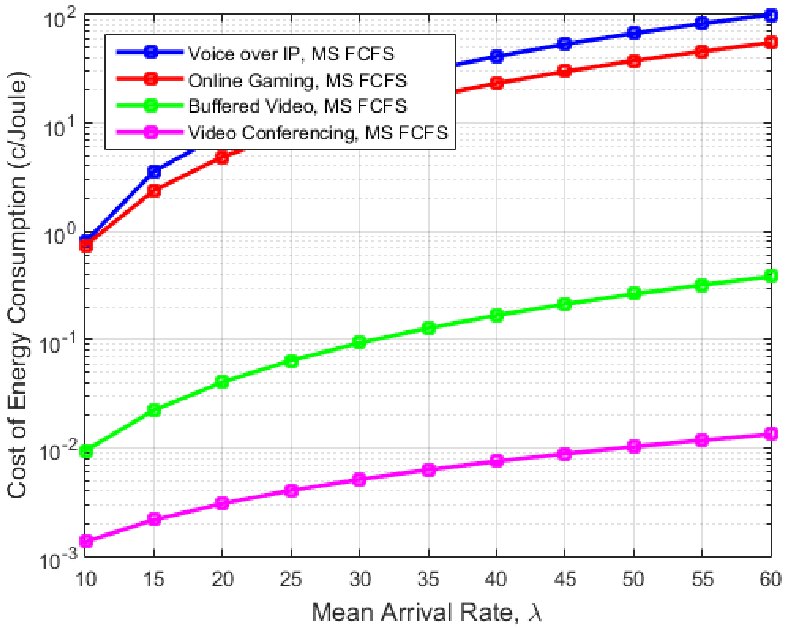

Figure 4.18. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.8$.

In Figure 4.16, Figure 4.17 and Figure 4.18 above, there is a noticeable increase in the energy consumption since as the rate of job arrivals increase, more energy is expended in pushing them out of the system within their deadlines. The video conferencing application has the lowest cost on the system compared to the other applications, owing to the difference between the b_{min} and b_{max} as seen in Table **4.1**. The greater the difference between these values gives a large value of $b_{i_{u_k}}$, the lesser the value of the resource consumption efficiency, as can be seen in (4.21). However, the cost of energy consumption decrease by 0.43% when the service rate is increased from $\mu = 0.6$ to $\mu = 0.8$. Thus, for FCFS scheduling scheme with server mean slowdown, the cost of energy consumption varies inversely with the service rate. , which shows the variation of the cost of energy consumption as a function of arrival rate for MS FCFS scheduling mechanism. The effect of increasing the arrival rate on energy consumption is investigated with the number of VMs, service rate, and size of requests constant.

4.10.5.2 Mean Slowdown Processor Sharing

The performance evaluation results for the MS PS scheduling mechanism for processing rates $\mu = 0.6$, $\mu = 0.7$ and $\mu = 0.8$ are shown in Figure 4.19, Figure 4.20 and Figure 4.21, respectively. When the scheduling scheme used is the MS PS, the cost of energy consumption is reduced as shown in Figure 4.19 below.

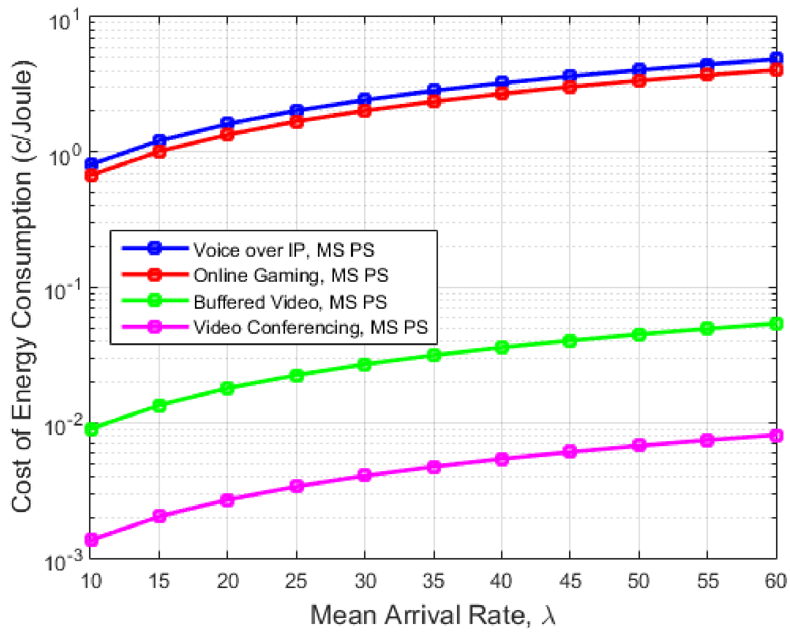

Figure 4.19. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.6$.

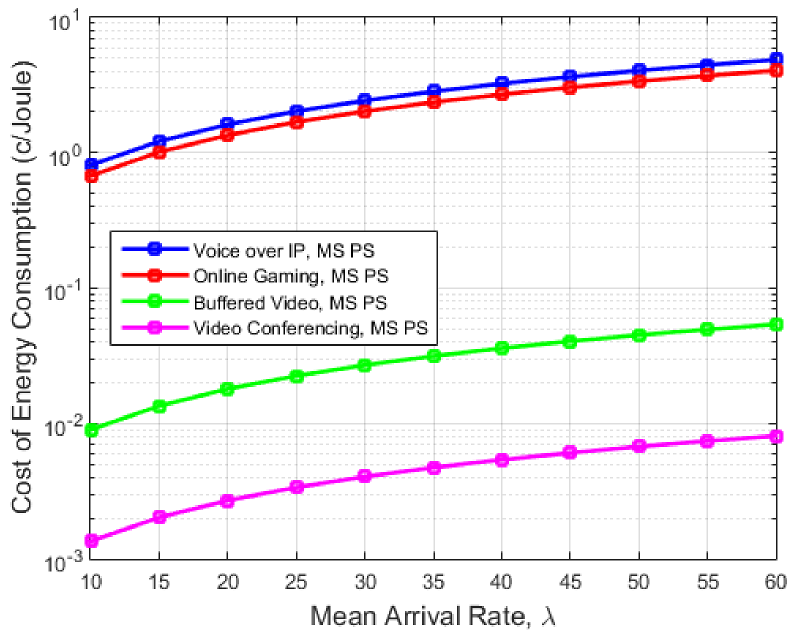

Figure 4.20. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.7$.

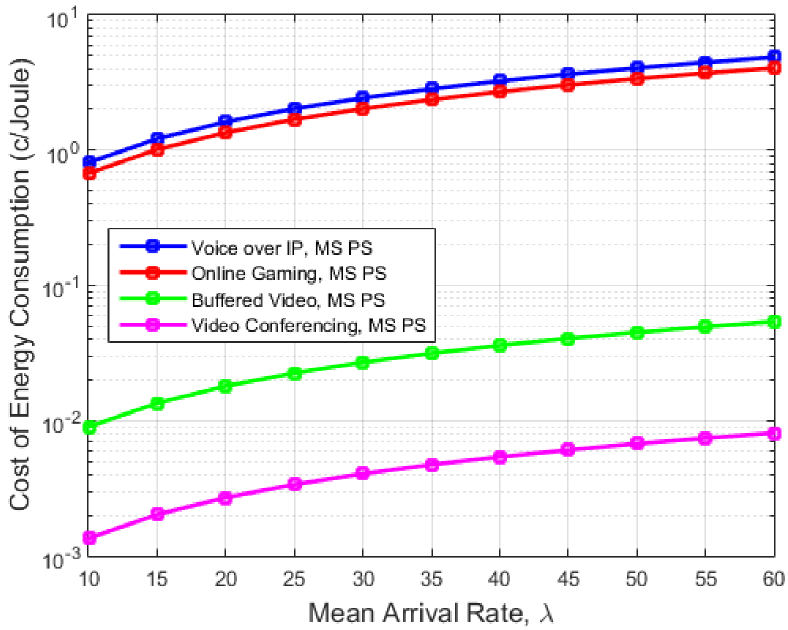

Figure 4.21. Cost of Energy Consumption as a Function of Mean Arrival Rates with $\mu = 0.8$.

Figure 4.19, Figure 4.20 and Figure 4.21 above show the variation of the cost of energy consumption as a function of an increasing arrival rate for the PS scheduling mechanism with mean slowdown. The energy consumption and its cost increase as the mean arrival rate increases, but it is not sensitive to a changing processing rate, hence the performance remains the same. It can be observed that there is a substantial energy saving with the MS PS compared to the MS FCFS, with MS PS achieving higher energy saving with the cost of energy consumption not increasing much as the mean arrival rate increases. This is because there is almost no queuing delay with the PS system since all packets are served simultaneously such that the due to traffic load as the incoming traffic rate is never greater than its outgoing capacity. Except for the fact that MS is more energy efficient, which applies for both FCFS and PS, PS is an even more efficient scheduling scheme since it concentrates job processing to within a few computational units instead of distributing the workload across all processing units. With a few VMs commissioned, less energy is consumed and hence there is substantial energy saving when the MS PS scheduling mechanism is used compared to MS FCFS.

Detailed results for energy saving between the two workload scheduling schemes are tabulated in Table **4.8** and Table **4.9** below.

Table 4.8. Percentage Energy Saving Using MS FCFS Scheduling Mechanism.

CRN Services												
λ Pkts per sec	10	15	20	25	30	35	40	45	50	55	60	
VoIP	0.12	0.41	0.87	1.48	2.53	3.18	4.27	5.52	6.92	8.48	10.20	%
Online Gaming	0.10	0.27	0.53	0.87	1.30	1.81	2.40	3.08	3.85	4.70	5.63	%
Buffered Video	0.0	0.0	0.0	0.01	0.01	0.01	0.02	0.02	0.03	0.03	0.04	%
Video Conf.	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	%

Table 4.9. Percentage Energy Saving Using MS PS Scheduling Mechanism.

CRN Services												
λ Pkts per sec	10	15	20	25	30	35	40	45	50	55	60	
VoIP	0.81	1.22	1.62	2.03	2.43	2.84	3.24	3.65	4.05	4.46	4.86	%
Online Gaming	0.68	1.01	1.35	1.69	2.03	2.36	2.70	3.04	3.38	3.71	4.05	%
Buffered Video	0.01	0.01	0.02	0.02	0.03	0.03	0.04	0.04	0.05	0.05	0.05	%
Video Conf.	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	%

The results in Table **4.8** and Table **4.9** above show that a decrease in the b_{iu_k} is also a decrease in the percentage of energy saving. This can be explained based on the fact that as the difference between the required long-term resources b_{min} and the the maximum required bandwidth becomes large, more packet processing units are required. However, one striking observation is that as the packet arrival rates increase, the system's energy efficiency also increases. This means that the energy consumption per packet decreases, as observed in Table **4.6** and Table **4.7**. However, in this case it is the packet-service discipline that improves the performance. Regarding packet-service discipline, the central issue that has been addressed is the notion of fairness by realizing that different packets consume energy differently. In essence, the question is what packet-service discipline is deemed energy-fair? The answer to this question remains in the server itself, where traffic flows require different services and the desired scheduling properties that are required.

4.11 CONCLUSION

In this paper, a QoS provisioning and energy saving scheme for single BS management where CRs run real-time services was proposed with design expectations in line with the IMT-2020 and beyond requirement specifications. Since QoS constraints need to be satisfied, energy efficiency becomes a critical concern and energy management issues need to be addressed from a single BS perspective. Firstly, a distributed dynamic RA based on resource reservation protocol was proposed to reserve resources to give SUs a high probability to complete their transmissions. The optimal RA solution was obtained through a weighted bipartite matching with a polynomial complexity of $\mathcal{O}(K^3)$ compared to the $\mathcal{O}(K!)$ of integer programming. A resource consumption efficiency obtained through bipartite matching was then used to solve as weighted cost function in which power consumption was added together with different weights reflecting their contribution to BS power consumption. Using the derived control actions, workload arrival patterns were estimated and used by the DL model to predict the future behavior over a finite horizon, T. Using the output of the SAE, which are just a regression between the previous and current system states, appropriate packet processing schemes is chosen between MS FCFS and MS PS. The simulation results obtained indicate that the predictive control achieves better energy saving as the traffic load increases, owing to few number of VMs commissioned to serve the traffic load. This shows that concentrating the workload on a few computational resources saves energy due to fewer VMs being turned on, which is also seen using the MS PS packet processing achieves an average of 6.89924% superior energy saving compared to MS FCFS. This shows that communication systems with PS represent adequate models for resource sharing, e.g., a bandwidth of communication systems and can be adopted for balancing QoS provisioning and energy saving in future mobile and wireless network design.

CHAPTER 5 QOE-DRIVEN RESOURCE MANAGEMENT IN ENERGY-CONSTRAINED CRNS USING DEEP HIERARCHICAL REINFORCEMENT LEARNING

5.1 CHAPTER OVERVIEW

CRs have limited and strict energy constraints, which makes energy efficiency a critical problem in the realization of CRNs in future mobile and wireless networks, whose primary network deployment strategy will exhibit more distributed and autonomous behavior. To address the problem of energy efficiency further, distributed QoE-driven RA strategies need to be formulated to ensure user satisfaction in an energy-efficient manner. However, one has to be cognizant that with QoE being a function of QoS, energy-efficient CRN operation will also require that the problems related to high energy consumption and transmission latency be addressed. It has been noticed that the state-of-the-art cognitive approaches that deal with both QoE and QoS problems, in as much as they do exhibit a certain degree of intelligence, actually lack the necessary intelligence to deal with highly distributed and autonomous networks. The use of RL is associated with problems of poor convergence due to the lack of deep exploration of the state-spaces, consequently resulting in a requirement of many cycles to converge or even large amounts of training data, which consume too much system energy.

In order to address this, this chapter proposes to improve QoE evaluation using the perfect docition to demonstrate the possibility of handling dynamic RA problems using an innovative user-centric and context-aware technique. In this regard, SUs experiencing heterogeneous traffic use the mean opinion score (MOS) to measure QoE in a distributed manner and the RA problem is solved using DRL. In improving the CRN-level QoS, an energy-efficient dynamic RA scheme that minimizes power consumption, subject to transmission delays using MDPs was proposed. A constrained power minimization technique that chooses optimal and energy-efficient decisions for each application-allocated resource and penalizes packet losses was used. The solution was achieved by reformulating the constrained power management problem into an online-based learning strategy with known and unknown system dynamics and solved via factored MDP (FMDP). Lastly, both solutions were combined into a single network architecture using a DQN framework, which uses experience replay to estimate the state-action value solutions without supervision, and consequently finding a quicker way to achieve energy-efficient CRN operation.

5.2 INTRODUCTION AND BACKGROUND

In the view of the sporadic nature of future mobile and wireless networks, which is evident in different forms, such as application/service, the aim of providing real-time services for delay-sensitive applications is paramount [278]. Current smartphones with enhanced capabilities have fostered an increase of multimedia and interactive bandwidth-demanding services over wireless networks, resulting in heterogeneity of network traffic [279]. The unprecedented increase in smartphone usage with increased service diversity, generating huge amounts of data, has brought about new challenges that make intelligent processing and computation a necessity [280]. This traffic heterogeneity, coupled with user behavior, results in heterogeneity in network management, which means service providers need effective data analysis tools to enable faster response to user QoE expectations. Heterogeneity in network management has its own specific challenges due to the numerous variables and actions that have to be taken into account [281]. With CRNs adding extra diversity of operation to the already existing heterogeneity, the need for even better data analysis tools has become urgent.

In order to enable CR systems to access the available spectrum resources in the best possible way, certain variables associated with QoS require particular actions to be taken; however, each action may have performance implications. For example, a channel handoff action due to service disruption might lead to performance degradation due to loss of packets. In its current state, the CR concept does not have the required intelligence to deal with service disruption, but there are signs that this will be solved as 5G systems continue to develop. The application of ML techniques in 5G systems will apply not only to physical network elements, but also to enabling traffic balancing and supporting traffic steering, as well as DSA [282]. This will result in a proactive network framework that will: (i) explore and learn the network dynamics by capturing user and content-dependent resource provisioning, and (ii) optimally exploit the limited resources based on limited knowledge to provide unprecedented predictive capabilities to user-specific services.

In order to achieve this, the identification of multimedia application requirements that are not only related to QoS but also to user context is crucial. The user's QoE demands are required to derive these requirements, which can be better understood through the mapping between their subjective ratings to the objective QoS [283]. However, because CRN resource management problems are often difficult online decision-making tasks, where appropriate solutions depend on the understanding of both the workload and the environment, QoE evaluation approaches have to be slightly different and even more dynamic than those applied in traditional wireless networks. As opposed to centralized RA, distributed RA in CRNs requires that network entities make decisions locally in order to obtain optimal sub-bands and power levels without having to wait for global information. Performing local decision-making incurs less communication overheads. However, because of the uncertainty of the wireless environment, devices are prone to resource and energy wastage [284].

To improve the QoS, resource wastage due to un-attempted time slots and energy wastage due to packet re-transmissions caused by collisions need to be reduced. Therefore, for service providers, understanding of the degree of influence of various QoS parameters on user QoE is a priority. Thus, in order to improve their service quality, it is paramount for them to understand user experience requirements [285]. However, as the scale of wireless networks increases in terms of heterogeneity and service diversity, the amount of information that BSs have to deal with at a particular instant also increases. As a consequence, the state and action spaces that the BSs have to map in order to improve RA decision-making also increase, which leads to increased convergence time. To circumvent this issue, BSs need to be able to learn information from their previous experiences and to compare that information with current information for quicker decision-making and better performance. QoE-driven adaptation strategies for optimising content provisioning and network planning for multimedia applications over wireless networks have been studied for some time in the context of traditional wireless networks.

RA problems involving QoE evaluation are usually solved using RL through discrete-time MDP modeling, which is an effective strategy in RA problems. RL strategies have also proven to be paramount in solving RA problems for dynamic systems such as CRNs. Recent QoE performance evaluation studies in CRNs include QoE maximization through the use of the Hungarian method for optimal channel assignments studied in [286], a heuristic sub-band assignment that takes the MOS as a MOS utility function studied in [287] and a convex optimization solution approach using hyper-graph formulation for QoE that guarantees power consumption minimization studied in [288]. These contributions focused on deriving closed-form solutions for QoE provisioning from the SU satisfaction perspective. An intelligent spectrum handoff algorithm that used RL was studied in [65] where SUs adaptively use the evolutionary conditions to maximize their expected MOS. The spectrum handoff decisions were modeled using a MDP scheme wand the MOS was the immediate reward. A proxy-based content-aware scheme to perform RA decisions for the BS was studied in [289]. In this regard, the optimizer estimated available resources by retrieving channel quality information.

QoE-based video delivery where two modules were used, a traffic engineering module and a traffic management module. In wireless networking problems, QoE evaluation is divided into two categories (i.e., subjective and objective) and both of them use the MOS as an evaluation tool. Interested readers can refer to [290]. Current docitive approaches for QoE evaluation use the RL technique and exhibit some degree intelligence when dealing with multimedia traffic, but they suffer from inefficient state space exploration. This is primarily because the ε-greedy exploration exploration use in traditional RL strategies suffers from poor convergence. The ε-greedy method in incapable of carrying out deep exploration of the state space, which consequently requires many cycles to converge or even huge amounts of training data. Another problem related to RL-based docitive approaches is that it becomes unfeasible to apply the in distributed networks containing many nodes, hence performing user experience evaluation on the fly becomes a challenge. This is because on-the-fly user experience evaluation requires the collection and correlation of a mixture of variables on the network conditions, on the service itself, as well as on the user. Moreover, because of the heterogeneity in network management, new challenges emerge in terms of the increase in the number of variables and actions that need to be taken into account [281].

5.2.1 QoE Evaluation Challenges in CRs

As the scale of the mobile and wireless networks increase, more distributed and autonomous ways to ensure SU QoE satisfaction need to be put in place before 5G networks are commissioned. This means that, in order to enable seamless CR operation within such heterogeneity, specific network variables that require certain actions need to be handled, using more advanced techniques. However, the biggest challenge in achieving this is that CRs have limited and strict energy constraints, making energy efficiency a critical problem in the dynamic and distributed autonomous deployments of future wireless networks. It is thus required that CRs be equipped with an improved technique that will be more energy-efficient than those currently available. The currently accepted cognition technique for QoE evaluation is not energy-efficient because of its lengthy and complex learning process. However, it has since received immense research attention, despite the fact that its complexity increases with the increase in the observation space.

The docitive paradigm, which is an extension of the cognition strategy, requires shorter learning processes and has better convergence and is hence more energy-efficient. Thus, as an emerging framework to improve CR operation, docition relates to radio entities that teach better policies to network new comers. Docition is achieved by means of radio entities facilitating knowledge dissemination and propagation with the non-trivial aim to facilitate learning [291]. CRs that use the docitive approach, as opposed to those that only use the cognitive paradigm, do not only have to teach one another the end result of spectrum sensing such as channel occupancy, but rather elements of the methods of achieving them. As a result, the docition concept is expected to yield significant benefits for CR algorithms, resulting in more efficient RA. QoE evaluation using the docitive paradigm is applied to the common MOS metric on dissimilar traffic; however, it introduces new challenging questions. One such question refers to how the learned environment-action adaptation experience differs between CRs carrying dissimilar traffic. It has been found that docition performs well using RL are able to handle users with dissimilar traffic, however their performance deteriorates as the network becomes distributed. Thus, to select actions for any given function, the RL agent performs either of two strategies, i.e., exploration or exploitation, discussed in [292]. The consequence of exploration and exploitation in RL is that with excessive exploration it is difficult to maximize rewards, while with excessive exploitation the agent may lose the chance of discovering better alternatives that will lead to even higher reward [293].

Therefore, in order to improve CR cooperation using the docitive paradigm, more expert CR nodes are encouraged to share their knowledge in a cooperative fashion, which significantly speeds up the learning process, increases precision and minimizes the energy spent in delivering information to other SUs. The different levels of docition are elaborately discussed in [294], [295]. In [296], an innovative user-centric and context-aware QoE evaluation for SUs where multiple SUs opportunistically transmit real-time video content over channels licensed to PUs was proposed. The authors applied the MOS as a common measurement scale for the QoE evaluation of SUs with different types of traffic to allow seamless integration of different traffic types. The objective of their work was to improve the convergence of the perfect docition technique when the number of SU nodes increased, which they achieved with 6.67% improvement compared to the other learning techniques.

5.2.2 Motivation and Contributions

Inspired by the recent advances in DRL, the researcher considers building a system that learns to manage resources online directly from previous experience. It is believed that in using a hierarchical DRL architecture, useful and insightful ways to conduct a fundamental reconsideration of a feasible energy-efficient system design in energy-constrained environments can be achieved. As a first step in addressing this problem, a RA scheme using the MOS for heterogeneous traffic is considered. Therefore, this chapter demonstrates the possibility of combining deep exploration with DNNs using DQNs to achieve faster learning and convergence. A detailed presentation of the main contributions is outlined as follows:

- **Perfect Docition:** The possibility of effectively handling dynamic RA problems using an innovative user-centric and context-aware technique for measuring the QoE for SUs experiencing heterogeneous traffic is demonstrated. Here, SUs use the MOS measurement tool to perform QoE evaluation in a distributed manner. Once the SUs have been admitted into the CRN and have completed their initial learning, they can then store the optimal operational policies in a memory buffer for replay so that they can teach best policies to newly admitted SUs through transfer-learning.

  The results obtained in this part indicate that the perfect docition transfer-learning technique actually achieves better performance than other transfer-learning schemes as the number of

SUs increases. The best policies are then communicated among SUs using the perfect docition paradigm, which is in principle treating the distributed CRN as a centralized architecture where each SU seems to have complete network information. This approach leads to optimal solutions, however, when both the state and action spaces increase with the addition of more SUs in the CRN, this approach is rendered unfeasible using the traditional RL strategy owing to its lack of deep exploration of the spaces [291]. In order to circumvent this shortcoming, extra communication among SUs is allowed and then uses the DRL-based perfect docition approach.

- **Model-based RL:** On a network level, an energy-efficient dynamic RA scheme that minimizes power consumption, subject to transmission delays is studied using MDPs. A constrained power minimization technique that chooses optimal and energy-efficient decisions for each application-allocated resource and penalizes packet losses was proposed. This technique was based on reformulating a constrained power management problem into an online-based learning strategy with known and unknown system dynamics. The resultant solution, obtained via FMDP, proved that the system can learn unknown system dynamics from known dynamics through Lagrange approximation. The simulation results prove that the solution obtained achieves minimum power consumption subject to delay constraints.
- **Deep Q-learning:** Both above solutions were then combined into a single dual network architecture using a DQN framework. A DQN is a modification of Q-learning that uses CNNs to approximate an action-value function [138]. The DQN, by using experience replay memory, combines both estimated solutions via a special aggregating layer, automatically producing a single state-action value solution without any extra supervision and consequently finding a quicker way to achieve energy-efficient CRN operation. The results of this section indicates that DQNs achieves better average buffer delays, average power consumption, and better average rewards than the traditional RL strategy.

5.3 PROPOSED SYSTEM MODEL

As is common in the literature, the system model is centered with the customary assumptions and simplifications of a single-cell energy-constrained CRN where SUs coexist with PUs for spectrum sharing on a primary network of radius R_{max}. Adopting the system model described in [296], with the r_1 of each secondary network, it is assumed that a set of $\mathcal{K} = \{1,2,\cdots,K\}$ consists of a subset of $\mathcal{L} = \{1,2,\cdots,L\}$ SUs running data applications, while the other $K-L$ are running multimedia applications. SUs are assumed to perform QoE evaluation using the MOS metric and adopt docitive strategies to teach newcomers optimal policies through transfer-learning in a distributed manner. The QoE results are loaded into the algorithm as new settings by the macro-cell BS (MC-BS), which uses the OpenFlow SDN standard [297] for monitoring network-level information and perform updates network configurations from time to time. The detailed system model explaining this process is illustrated in Figure 5.1 below.

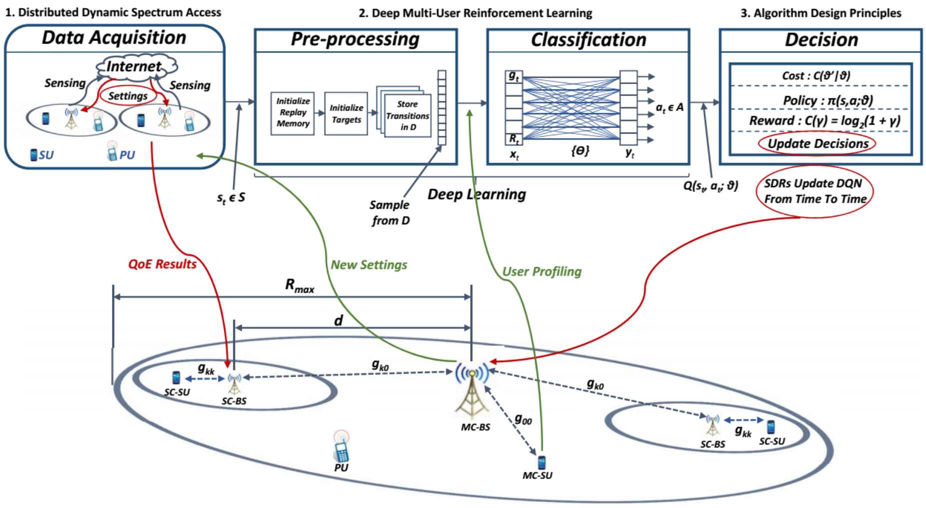

Figure 5.1. Illustration of a Fully Distributed Learning CRN Architecture.

In Figure 5.1 above, the first module is the DSA module which acquires data through spectrum sensing. The second module is the multi-user RL, where the combination of pre-processing and classification completes a DL task, and the third module makes decisions based on the flow of data. The user profile data are indicated by the green arrows, while the monitoring data are indicated by the red arrows. The data acquisition process includes channel occupancy information described in [159] and contributes to historical channel occupancy learning. It is assumed that the IEEE P1900.5 standard [298] that presents the functional definition and requirements relevant to a CR engine is used, and an effective interference alignment technique to combat the interference among SUs and PUs is assumed.

The system creates a profile based on learning using NNs where, based on a sample set of data items that have already been judged relevant by the user, an NN may be trained. The inputs of the NN are the channel gains and the required resources, and the outputs are the relevance judgements of the users. The NN algorithm performs some classification that gives ranks based on the estimated QoE for each input, which may serve as the user profile for future filtering. In each BS, a database contains user profiles with the content that users frequently use. This information can be used to predict impending channel hand-offs to ensure that QoE is maintained in the case of a service interruption.

5.4 PROBLEM AND SOLUTION FORMULATION

The small-cell BS (SC-BS) runs an RA algorithm described in [299] and sends tuning information to SUs in the form of a SINR requirement, which translates to PU interference constraints. When the PU channel is busy, an SU can still access that channel in underlay mode (i.e., with limited power so that PU transmission is not interfered with). The wireless channels for both types of users are characterized by path-loss combined with fading such that the term $g_0^{(pu)}$ denotes the channel gain between the PU and the primary MC-BS, the term $g_k^{(pu)}$ represents the channel gain between the k^{th} SU and the MC-BS, and $g_0^{(su)}$ represents the channel gain between the PU and the j^{th} SC-BS. Also, the terms $\gamma_0^{(pu)}$ and $\gamma_k^{(su)}$ represent the SINR at the MC-BS and the SINR for the k^{th} SU associated with the SC-BS and are respectively represented as follows:

$$\gamma_0^{(pu)} = \frac{P_0 g_0^{(pu)}}{\sum_{k=1}^{K} P_k g_k^{(pu)} + \sigma_0^2}, \quad \text{and} \quad \gamma_k^{(su)} = \frac{P_j g_j^{(su)}}{\sum_{j \neq k} P_k g_k^{(su)} + P_0 g_0^{(su)} + \sigma_0^2}, \tag{5.1}$$

where P_0 is the PU transmission power, P_k represents the transmission power of the k^{th} SU, and P_j represents the transmission power of the j^{th} SU, and σ^2 is the additive white Gaussian noise power. The dynamic RA is assumed to use a DNN architecture that nudges the system towards a fair distribution of resources and drives it to higher resource utilization. Its output is the power allocations for the SUs required to achieve their QoS levels, expressed as

$$P_k = \frac{\vartheta_k(\sigma^2 + g_0^{(su)} P_0)}{g_j^{(su)}(1 - \sum_{k=1}^{K} \vartheta_k)}, \quad \vartheta_k = \left(1 - \frac{1}{\Gamma_k}\right)^{-1}. \tag{5.2}$$

Then, for the optimal power allocation strategy that maximizes SU spectral efficiency while protecting the QoS of PUs, the average interference power constraint, is imposed as

$$\gamma_0^{(pu)} \geq \Gamma_0, \quad \gamma_k^{(su)} \geq \Gamma_k, \quad k \in K, \tag{5.3}$$

where the terms Γ_0 and Γ_k represent the PU and SU SINR thresholds respectively. Therefore, in order for the power allocation to be safe for PU operation, the $1 - \sum_{i=1}^{K} \vartheta_k > 0$ must be met. Then, when one replaces the SU transmission powers in equation (5.2) into equation (5.1), (5.3) is represented as follows:

$$\sum_{j=1}^{K} \alpha_j \vartheta_j \leq 1, \quad \text{where} \quad \alpha_j = \frac{g_j^{(pu)}(\sigma^2 + g_0^{(su)})P_0}{g_j^{(su)}(g_0^{(pu)} P_0/\Gamma_0 - \sigma^2)} + 1. \tag{5.4}$$

As earlier assumed, Γ_0 is constant, then Γ_k is the one parameter that each SU has to adjust so that it meets both (5.2) and (5.4). Assuming that the optimal transmit power in (5.2) is obtained using the algorithm described in [299]. Then, through adapting the relationship between the bit rate and the SU SINR, the transmission rate R is represented as follows:

$$R = \xi \log_2(1 + m\Gamma_k), \tag{5.5}$$

where the term ξ represents the system bandwidth, and the expression $(1 + m\Gamma_k)$ denotes the number of information bits per modulation symbol, while the term m is a constant that relates to the maximum BER requirement. Thus, in this case each SU has to slect a target SINR Γ_k so that all SUs can cooperatively meet the SINR constraints. Therefore, by adjusting the value of Γ_k and consequently R, then the modulation scheme is also adjusted. Based on the qualitative relationship that describes the impact of QoS on QoE given in [290], the researcher turns the focus to an independent type of metric whereby the network accounts for the QoE reductions between the reference and outcome.

In order to evaluate the QoS, a packet-level evaluation results and converts these to the QoE is used, thus in order to select the appropriate measures to keep user-perceived QoE above an acceptable threshold for data traffic, the service provider must be cognizant about the network-level QoS parameters that translate to user-level QoE. To address this issue of QoS in data traffic, one needs to monitor the average packet delay and loss for SUs. Thus, the QoS for data traffic is represented as follows [301]:

$$QoS_D = a \log_{10}(bR(1 - p_{e2e})), \tag{5.6}$$

where p_{e2e} stands for the end-to-end packet loss probability. **Note:** In this chapter, as opposed to the previous one, the parameters a and b are computed based on the maximum and minimum data quality as perceived by the SU.

In terms of video traffic, a QoS-driven RA scheme that maximizes video transmission quality considers the impact of the application as well as network-level parameters on the end user satisfaction, which requires MOS evaluation. In this case, maximizing the peak signal-to-noise ratio (PSNR) of video transmission requires that the information be decoded at the receiver [302]. However, the direct application of PSNR is actually infeasible because the PSNR of the reconstructed video changes as a function of the bit rate as well as the video sequence itself. Therefore, owing to the unfeasibility of PSNR application, one can analyze its effect using the MOS as follows [65]:

$$MOS = \frac{\tau_1 + \tau_2 FR + \tau_3 \ln R}{1 + \tau_4 PER + \tau_5 (PER)^2}, \tag{5.7}$$

where $PER = \frac{1}{1+e^{\eta(\gamma_k-\zeta)}}$ denotes the packet error rate, and the parameters $\tau_1, \cdots, \tau_5$ can be obtained through linear regression, FR represents the frame rate, the parameters η and ζ are respectively the modulation scheme and coding scheme, which are both depend on the packet length. After the MOS for video traffic has been obtained then the objective metric characterizing the relationship between the MOS and the QoE of the video application is represented as a logistic function as follows:

$$QoS_V = \frac{\beta^*}{1+e^{-\kappa(MOS-f)}}, \tag{5.8}$$

where β^*, κ, and f are the parameters of the logistic regression function. Using the fundamental relationship between QoS and QoE, which can be stated as $QoE = f(QoS)$, one can integrate the RA for heterogeneous traffic as follows:

$$QoE = \frac{1}{K}\left(\sum_{i=1}^{L} QoS_D + \sum_{i=L+1}^{K} QoS_V\right). \tag{5.9}$$

where L represents the number of SUs experiencing data traffic only, while $K-L$ is the number of SUs experiencing video traffic. Then with the parameters Γ_k and R properly adjusted, the optimization problem is formulated as follows:

$$\hat{\Gamma}_k = \arg\max_{\Gamma} \sum_{i=1}^{K} QoE_i(\Gamma_k), \tag{5.10}$$

subject to

$$\sum_{i=1}^{K} \vartheta_k(\Gamma_k) \leq 1-\varepsilon, \qquad \text{and} \qquad \sum_{j=1}^{K} \alpha_j \vartheta_j(\Gamma_j) \leq 1. \tag{5.11}$$

In computing the solution, one needs to define the possible set of actions is the set of transmission power levels, which are defined by the candidate SINRs. This means that, during the agents' interaction with the environment, the agent must take actions that maximize its QoE but do not cause too much interference with PUs. Maximizing the QoE without causing too much interference with PUs represents a positive reward; a negative reward is one that incurs cost.

5.4.1 The Action and State Space and Reward Function

In this subsection, the action-space at each time interval t when each agent takes an action $a_t \in \mathcal{A}$, while it is in environmental state $s_t \in \mathcal{S}$ can be denoted as

$$a_t = \{\Gamma_1, \Gamma_2, \cdots, \Gamma_k\} \in \mathcal{A}, \tag{5.12}$$

where $\mathcal{A}$ and $\mathcal{S}$ represent the discrete set of possible actions and set of possible system states, respectively. The action-space is finite discrete space of candidate SINRs, where the choice of an SINR (action) is only considered for the non-channel switching (i.e., no channel handoff) scenario. In the non-switching case, it is either the PU that does not return to claim its channel or the SU that continues to transmit in underlay mode. Thus, each SU agent can conduct a for an optimal solution, from this finite discrete space of candidate SINRs, that not only meets the SINR constraints, but also maximizes is primary objective. Therefore, each SU agent observes a finite state-space $s \in \mathcal{S}$ defined as follows:

$$s_t = \{I_t^k, \mathcal{I}_t\} \in \mathcal{S}, \qquad \text{where} \qquad I_t^k = \left\{ \begin{array}{ll} 0, & \text{if} \quad \sum_{i=1}^{K} \vartheta_k(\Gamma_t^{(i)}) < 1 \\ 1, & \text{otherwise} \end{array} \right\} \tag{5.13}$$

is a binary indicator that specifies whether or not the SU is causing aggregate interference above or below the predefined threshold, and

$$\mathcal{I}_t = \left\{ \begin{array}{ll} 0, & \text{if} \quad \sum_{j \neq k} \alpha_j \vartheta_j(\Gamma_t^{(j)}) \leq 1 \\ 1, & \text{otherwise} \end{array} \right\}, \tag{5.14}$$

is a binary indicator showing the reflection of the aggregate interference caused by other j SUs, except the k^{th} (i.e., $j \neq k$). Since at this point the system assumes the use of a decentralized Q-learning algorithm, in which the information in (5.13) and (5.14) is supposed to be communicated to the small cell BS (SC-BS) by the SUs together with spectrum sensing information. Then, the SUs themselves take appropriate action $a_t \in \mathcal{A}$ in terms of selecting the optimal powers to a policy $\pi(s_t, a_t)$, $\pi \in \mathcal{A}$. Here, the policy/strategy $\pi(s_t, a_t)$ represents hes to be followed by each SU agent to obtain reward $r_t = r(s_t, \pi(s_t))$, given as follows:

$$r_t^{(i)} = \left\{ \begin{array}{ll} c, & \text{if} \quad I_{t+1}^k + \mathcal{I}_{t+1} > 0 \\ QoE^{(i)}, & \text{Otherwise} \end{array} \right\}, \tag{5.15}$$

where $c \in C$ denotes the cost or penalty incurred by following a strategy or policy that results in the violation of the interference constraints, and $QoE^{(i)}$ denotes satisfying the interference constraints for the received traffic, hence the immediate reward.

5.4.2 Dynamic and Constrained Power Management

In this subsection, the energy efficiency of the CRN illustrated in Figure 5.1 is analyzed in terms of power management, with the standard method relating to BS decision-making being FMDPs. The use of FMDPs in establishing a comprehensive evaluation and analysis of energy management of CRNs entails the learning of an energy-aware dynamic RA for the proposed system model. Thus, considered in this subsection is the application of model-based RL in minimizing CRN energy consumption while completing SU transmissions within their acceptable deadlines. This constrained power minimization framework, which is based on MDPs, uses the Softmax-based policy in (5.20) and chooses energy-efficient decisions for each application-allocated resource with the objective of minimizing transmission delays and energy consumption.

This formulation adopts Bayesian probabilities to predict the transition probabilities among licensed channels, which in the end, boils down to model-based DRL in which the SU agent collects QoS information which means that the QoS status of SU links can be be mathematically inferred to support quality evaluation [303]. Thus, in this constrained power minimization problem, it is considered that the learning agent interacts with an MDP whose state-action spaces, as well as the reward function at each time slot t are respectively denoted as $s_t \in \mathcal{S}$, $a_t \in \mathcal{A}$, and $r_t \in \mathcal{R}$. At this point, the environmental dynamics, which are represented using transition probabilities are defined as follows [143]:

$$\mathcal{P}^a_{ss'} = Pr\{s_{t+1} = s'|s_t = s, a_t = a\}. \tag{5.16}$$

In the sequel, $\{\prime\}$ denotes $t+1$, with explicit notation dropped to reduce clutter. Thus, the expected reward is represented as

$$\mathcal{R}^a_s = \mathbb{E}\{r'|s_t = s, a_t = a\}, \quad \forall s, s' \in \mathcal{S}, a \in \mathcal{A}, \tag{5.17}$$

where $\mathbb{E}$ denotes the mathematical expectation operator, which denotes an expectation over a sequence of states $\{s_t : t = 1, 2, \cdots, \infty\}$. Then, the decision-making procedure of the agent is characterized by the policy

$$\pi(s, a; \theta) = Pr\{s_t = s|a_t = a, \theta\}, \quad \forall s \in \mathcal{S}, a \in \mathcal{A}, \tag{5.18}$$

where the term $\pi(\cdot)$ denotes the policy mapping environmental states into actions in such a way that $a_t = \pi(s)$, the term $\theta \in \mathcal{R}$ is a parameter vector such that π is assumed to be differentiable with respect to it, i.e., $\frac{\partial \pi(s,a)}{\partial \theta}$ exists. The state-space, the action-space and the reward model are discussed in the following subsections.

5.4.2.1 The State and Action Spaces

From the BS system perspective, the full observation at time slot t includes the channel state, γ_t, the transmission buffer state, b_t, and state of the wireless transmission card x_t. Instead of the BS collecting all such information and distributing it to individual SUs, which results in high signaling overheads in practice, it is assumed that the state of each SU is only determined by its local observation of the system, upon which each SU will perform QoE evaluation, share information with other SUs using the perfect docition technique and finally select an action independently from the other SUs. Therefore, at the beginning of each time slot the BS possesses the SINR from the previous time slot, γ_{t-1}, the buffer state b_t, such that the state-space can be defined as the vector

$$s_t = (\gamma_{t-1}, b_t, x_t) \in \mathcal{S}. \tag{5.19}$$

Then, based on the above observed state-spaces, the BS agent needs to take an appropriate actions, which includes the selection of an optimal transmission power. This entails searching for an optimal transmission power over a discrete action-space consisting of possible transmission powers that will guarantee better power management and throughput. This results in a power management action $a_t = y_t$ which should achieve better bit-error probability (BEP_t) for a transmission rate R_t. Therefore, y_t is the transmission action, R_t is the transmission throughput that quantifies the performance of the transmission action at constant transmission rate R, and this completes the action space.

5.4.2.2 The Reward Function

In formulating the reward function, it is assumed that the BS agent collects QoS status information in the form of SU link quality for mathematical inference to support quality evaluation. The overall BS computation cost will then be counted by both the total energy consumption cost and the penalty on buffering delay. The advantage of using model-based RL instead of model-free RL for this task is that it is sample-efficient. With model-based RL, if it is appropriate to approximate the dynamics as locally linear, it will take significantly fewer samples to learn the model and once the model and the cost function are known, it becomes easier to plan the optimal controls without any further sampling. Therefore, at each time step, the BS agent selects a transmission action a_t based on its current state s_t using the policy $\pi(s_t)$, which is defined by a Gibbs Softmax function as follows:

$$\pi(s_t, a_t) = \mathcal{P}(a_t|s_t) = \frac{e^{Q(s,a)/\tau}}{\sum_{a' \in \mathcal{A}} e^{Q(s,a')/\tau}}, \tag{5.20}$$

where the term $Q(s,a)$ represents the tendency to select an action a_t at a given state, s_t, which will be discussed later in (5.25). Here, the term τ represents the temperature parameter of a Boltzmann distribution that controls the expected reward for the probability of taking a certain action. Thus, for a higher value of τ, the equally probable all available actions become, while on the contrary, actions are selected based on the maximum value of $Q(s,a)$. Therefore, using this policy, the BS and its co-located server can improve both the SUs' computational experience and its power consumption, which is represented as a finite state consisting of three operational states, given as [127],

$$P([\gamma_{t-1}, x_t], BEP_t, y_t, R) = \left\{ \begin{array}{lll} P_{\mathrm{ON}}(t) + P_{tx}(\gamma_t, BEP_t, R), & \text{if} & x = \mathrm{ON}, y = \mathrm{ON} \\ P_{\mathrm{OFF}}(t), & \text{if} & x = \mathrm{OFF}, y = \mathrm{OFF} \\ P_{\mathrm{tr}}(t), & & \text{Otherwise} \end{array} \right\}. \tag{5.21}$$

where the term P_{tx} represents the transmission power, P_{ON} and P_{OFF} are the powers consumed by the BS in its ON and OFF states, respectively. The term P_{tr} represents the power consumed by the BS during a transition from the ON state to the OFF state, and vice versa. This is actually an extension of the power consumption model used in the previous chapter. Then, having defined the BS power consumption in (5.21), to improve the QoS, the packet goodput distribution is defined as ρ^* and a buffer cost function that will reward the system for minimizing queuing delays given by the queue state q, is given as follows:

$$q([b,x],BEP,y,R) = \sum_{\lambda=0}^{\infty}\sum_{\rho^*=0}^{R} \mathcal{P}^{\lambda}(\lambda)\mathcal{P}^{\rho^*}(\rho^*|BEP,R)\{ \underbrace{[b_t-\rho^*]}_{\text{buffer holding cost}} + \underbrace{\varsigma^* \max([b_t-\rho^*]+\lambda-B,0)}_{\text{buffer overflow cost}}\}, \tag{5.22}$$

which represents the expected sum of the buffer holding cost $b_t-\rho^*$ with respect to the packet/job arrival rate λ_t. To simplify this analysis, one assumes that the traffic arrival pattern has a Poisson distribution with rate λ_t. The term b_t is the current buffer state of a finite buffer with capacity B [309] such that the buffer holding cost represents the number of packets that were not transmitted in the previous time slot. Since the packets not transmitted in the previous interval may miss their delivery deadlines, the penalty ς^* is defined to penalize every packet that is lost, and is given as follows:

$$\varsigma^* = \sum_{t=t_0+1}^{\infty} \mathbf{1}(\gamma^{t_1-t_0}) = \frac{\gamma^t}{1-\gamma^t}, \tag{5.23}$$

where the term γ^t, $0 \leq \gamma^t < 1$ is a discount factor defined later, the value $\mathbf{1}(\gamma^{t_1-t_0})$ is a statistical indicator function that states that: given $\gamma^{t_1-t_0}$, any relation between random variables holds almost surely with respect to the underlying probability measure [306]. Assuming that $\gamma/(1-\gamma)$ is useful in creating an optimal upper bound as γ approaches 1, the theorem and a detailed proof that supports (5.23) are provided in [307].

Considering a finite buffer, the buffer overflow can be defined as a catastrophe resulting from a sudden increase in transmission delay which is caused by either poor channel conditions such as fading or bursty traffic arrivals [308]. In case of poor channel conditions and traffic bursts, the BS might want to transmit at high power in order to make sure that more packets are pushed out of the system as much as possible to avoid transmission delays. In order to avoid this, the optimality of dropping packets when transmitting at low power must always be ensured, which can be achieved by making the term ς^* large enough and every dropped packet must be penalized as much as it would cost the system to admit it into the transmission buffer. To simplify this analysis, a stable buffer is assumed, i.e., buffer overflow catastrophes do not occur, such that the buffer holding cost becomes proportional to the queuing delay given by Little's theorem [310].

With the above assumption, the term ς^* becomes zero and second term in (5.22) vanishes and the QoS of the system increases. However, in the case of an unstable buffer (i.e., with a finite probability of buffer overflow occurrences), then Little's theorem will not hold. In this case, the cost of buffer overflow imposes a penalty ς^* for every dropped packet as defined in (5.23). Therefore, in order to improve the QoS, one must ensure that queuing delays are minimized so that the system is protected against buffer overflows. Since the only information available is the initial system state, the reward function is formulated using the start-state s_0 where one only cares about the initial state and the long-term reward obtained from it. Thus, the BS agent's objective can be formulated using the definition

$$J(\pi) = \mathbb{E}\left\{\sum_{t=1}^{\infty} \gamma^{t-1} r_t | s_0, \pi\right\} \tag{5.24}$$

where $J(\pi)$ is the performance measure, which uses the actual returns $r_t = \sum_{t=1}^{\infty} \gamma^{t^*-1} r_{t+t^*}$ as an approximation for each on-policy action-value function $Q^{\pi}(s_t, a_t)$, where

$$Q^{\pi}(s,a) = \mathbb{E}\left\{\sum_{t^*=1}^{\infty} \gamma^{t^*-1} r_{t+t^*} | s_t = s_0, a_t = a, \pi\right\}, \tag{5.25}$$

where t^* represents the number of steps taken after start state s_0. In this formulation, it is of interest to define a discounted future state distribution $d^{\pi}(s)$, which is a discounted weighting of states that is encountered when starting at the initial state s_0 as discussed in chapter 2. This allows one to express the difference in performance between two policies π' and π compactly [311]. The derivation results of this formulation can be found in [143].

5.4.2.3 The Optimization Problem and Solution Formulation

The solution to this dynamic and constrained power management problem can be attempted using a CMDP formulation and function approximation. Using the CMDP strategy, the power cost can be minimized while the delay and packet overflow constraints are kept below a certain prescribed value. In this case, the sequence of states is modeled using controlled Markov chains with the transition probabilities determined from the conditionally independence of the channel state γ_{t-1}, the buffer state b_t and the power management state x_t. Thus, the transition probabilities are expressed as follows:

$$\mathcal{P}_{ss'}(s'|s,a) = \mathcal{P}^{\gamma_t}(\gamma'|\gamma_t)\mathcal{P}^{b}(b'|[\gamma_t, b_t, x_t], BEP_t, y_t, R)\mathcal{P}^{x}(x'|x_t, y_t). \tag{5.26}$$

Then, at this point, the objective can be stated as the minimization of the infinite horizon of discounted power cost, subject to an infinite horizon of discounted transmission delays, the expectation on the discounted power cost as well as on the discounted transmission delays can be defined as follows:

$$\bar{P}^{\pi}(s)=\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^{t}P(s_t,\pi(s_t))|s_0=s\right],\qquad \text{and}\qquad \bar{D}^{\pi}(s)=\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^{t}q(s_t,\pi(s_t))|s_0=s\right]. \quad (5.27)$$

The reasoning behind the choice of the discounted criterion over the average time criterion is that discounted criteria are popular for optimizing stochastic dynamical systems such as telecommunication systems and also lead to reasonable solutions, while the average time criteria do not [312]. Thus, if one lets Ψ denotes the finite set of all possible policies, then beginning from the initial state s_0, and following the policy $\pi\in\Psi$, the objective function of the constrained power management problem can be formally presented as

$$J(\pi)=\min\bar{P}^{\pi}(s),\quad \text{subject to}\quad \bar{D}^{\pi}(s)\leq\delta,\quad \forall s\in\mathcal{S}, \quad (5.28)$$

where δ is the delay constraint. As it is stated in [126], solving a CMDP problem is equivalent to solving an unconstrained MDP with its Lagrangian dual problem, one can reformulate (5.28) using the unconstrained MDP approach through the introduction of the Lagrange multiplier $\Lambda\geq 0$, which is associated with the delay constraint δ. This results in a cost function that has been reshaped using the Lagrange multiplier as follows:

$$c^{\Lambda}(s,a)=P(s,a)+\Lambda q(s,a), \quad (5.29)$$

where the term $P(s,a)$ represents the power cost previously defined in (5.21), while the term $q(s,a)$ represents the buffer cost defined in (5.22) such that the optimal value of the unconstrained CMDP problem can be calculated as follows:

$$J_{\delta}^{\hat{\pi},\hat{\Lambda}}(s)=\min_{\pi\in\Phi}\max_{\Lambda\geq 0}V^{\pi,\Lambda}(s)-\Lambda\delta=\max_{\Lambda\geq 0}\min_{\pi\in\Phi}V^{\pi,\Lambda}(s)-\Lambda\delta, \quad (5.30)$$

where the MDP is augmented with the cost function in (5.29) on the on-policy value function as follows:

$$V^{\pi,\Lambda}(s)=\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^{t}c^{\Lambda}(s_t,\pi(s_t))|s_0=s\right], \quad (5.31)$$

and then a new policy $\hat{\pi}$ is optimal for the unconstrained CMDP if the Lagrange formulation in (5.30) above solves to

$$J^{\hat{\pi},\hat{\Lambda}}(s)=\max_{\Lambda\geq 0}V^{\hat{\pi},\hat{\Lambda}}(s)+\Lambda\delta. \quad (5.32)$$

Then using a fixed value of Λ, the solution of the rightmost minimization in (5.30) can be obtained by solving the following dynamic programming equation

$$V^{\hat{\pi},\hat{\Lambda}}(s)=\min_{a\in A}\left\{c^{\Lambda}(s,a)+\gamma^{t}\sum_{s'\in S}V^{\hat{\pi},\hat{\Lambda}}(s')\right\},\forall s\in\mathcal{S}, \quad (5.33)$$

where the value $V^{\hat{\pi},\hat{\Lambda}} : \mathcal{S} \to \mathbb{R}$ represents the optimal state-value function which is defined under the optimal action-value function $Q^{\hat{\pi},\Lambda} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and satisfies the formulation

$$Q^{\hat{\pi},\Lambda}(s,a) = c^{\Lambda}(s,a) + \gamma^t \sum_{s' \in \mathcal{S}} p(s'|s,a) V^{\hat{\pi},\Lambda}(s'), \qquad \text{where} \qquad V^{\hat{\pi},\Lambda}(s') = \min_{a \in A} Q^{\hat{\pi},\Lambda}(s',a). \quad (5.34)$$

In other words, $Q^{\hat{\pi},\Lambda}(s,a)$ represents the infinite horizon discounted cost that is achieved after taking an action a in the state s and following $\pi^{\hat{\pi},\Lambda}$. Thereafter, the model-based RL problem is in an unconstrained CMDP under known system dynamics as

$$\pi^{\hat{\pi},\Lambda}(s) = \arg\min_{a \in A} Q^{\hat{\pi},\Lambda}(s,a), \qquad \forall s \in \mathcal{S}. \quad (5.35)$$

At this point, if it is optimal to select the action a with the policy $\pi(s)$ being followed, this then corresponds to a new optimal policy $\pi^{\hat{\pi}}(s)$, and the Q-function has to be minimized with respect to the current state s_t. However, the assumptions in (5.33), (5.34), and (5.35) are that the knowlwdge of the costs and the transition probability functions is available a priori, in which case they can be numerically solved through the use of value iteration defined in [143]. However, in practical scenarios this is not always the case since one is often faced with problems in which the a prior knowledge regarding the costs and the transition probability functions either exist partially or completely non existent. As a consequence, the values of the value function, $V^{\hat{\pi}}$ and the new policy, $\pi^{\hat{\pi}}$ cannot be obtained through the use of the value iteration algorithm; but instead, they may be learned online based on experience.

To this end, it is safe to adopt a Q-learning approach in order to learn the values of $V^{\hat{\pi}}$ and $\pi^{\hat{\pi}}$ online without the need to first establish the estimates of the unknown cost and the corresponding transition probability functions. The task of Q-learning algorithm in this case would be to determine $\pi^{\hat{\pi}}$ without knowing $c(s,a)$ and the transition probability $\mathcal{P}^{a}_{s,s'}(a)$, which makes it well suited to CRN problems such as power allocation. Therefore, at any time step t, the transition of states and the subsequent rewards are assumed to be stochastic with Markovian properties, which allows one to make use of the well-known Bellman's optimality criterion from [313]. When applying the Bellman's criterion in this situation, one has to find an intermediate minimum of $Q(s,a)$, which is denoted by $Q^{\hat{\pi}}(s,a)$. Then, the intermediate evaluation function for every possible next state-action pair (s',a') should be minimized and the optimal actions be performed with respect to each next state s'. In summary, the Q-learning process tries to recursively obtain the $Q^{\hat{\pi}}(s,a)$ by utilizing the available information, which is (s,a,c,s'). Then, using the traditional Q-learning update rules, the Q-values can be updated relative to the agent as follows:

$$Q(s,a) \leftarrow Q(s,a) + \alpha_t \left[c + \gamma^t \min_{a} Q(s',a) - Q(s,a) \right], \quad (5.36)$$

where $0 \leq \alpha_t \leq 1$ represents the algorithm learning rate. Due to the stochastic nature of channel state transitions and the traffic in the CR system representation, imprecise probabilities that represent incomplete beliefs about the transition probabilities are introduced. This is because some components of the state-space, together with their transition probabilities may be known while others are unknown, which increases the complexity of the system, especially when dealing with larger state-spaces. If both these known and unknown system dynamics are independent of one another, then they can be factored to derive a more compact CRN representation, which then results in a FMDP [314].

5.4.3 Factored MDPs for Imprecise Transition Probabilities

Factored MDPs (FMDPs) are a kind of MDPs with imprecise transition probabilities, designed specifically for situations where the precise knowledge of the transition probabilities cannot be defined [315]. Then, instead of using the traditional transition probability measure $\mathcal{P}^a_{ss'}$, one has to define a set of transition probability measures that will precisely account for incomplete system component knowledge. By introducing the notion of FMDPs and exploiting their structure to derive more compact CRN representations of the global state to obtain more optimal policies and efficient transition probabilities, the transition probability function can be factored into both its known and unknown components. Here, the known components should account for the transition from the current state of the system to the FMDP, i.e., $s \rightarrow \hat{s}$, while the unknown components account for the transition from the FMDP to the next state of the system, i.e., $\hat{s} \rightarrow s'$. The formal presentation of the transitions is as follows:

$$\mathcal{P}(s'|s,a) = \sum_{\hat{s}} \mathcal{P}^u(s'|\hat{s},a)\mathcal{P}(\hat{s}|s,a), \tag{5.37}$$

where the superscript u denotes the unknown transitions. Using FMDPs, these function-specific and content-specific independences can be exploited to compactly represent the transitions and the cost function using a linear approximation. Thus, the cost function can be similarly be factored as follows:

$$c(s,a) = c(s,a) + \sum_{\hat{s}} \mathcal{P}(\hat{s}|s,a)c^u(\hat{s},a). \tag{5.38}$$

Then, the factorized transition probabilities and cost functions in (5.37) and (5.38) have been reduced to factorized transition probabilities and cost functions, respectively. Then, when $c^u(\hat{s},a) = 0$ and the transition from the current state s_0 to the FMDP is deterministic, i.e., $\mathcal{P}(\hat{s}|s_t,a)$, i.e., both the known and unknown CR system dynamics generally depend on both the action a_t and the FMDP. However, in the CR system studied here, the unknown system components depend only on the FMDP, i.e., $\mathcal{P}^u(s'|\hat{s},a) = \mathcal{P}^u(s'|\hat{s})$ and $c^u(s'|\hat{s},a) = c^u(s'|\hat{s})$, which implies that the exploration of actions is not required in learning the optimal policy. Thus, both the known and unknown transition probability functions can specifically be represented as follows:

$$\mathcal{P}(\hat{s}|s,a) = \mathcal{P}^{x}(\hat{x}|x,y)\mathcal{P}^{\rho^*}(b-\hat{b}|BEP,R)I(\gamma'|\gamma_t), \tag{5.39}$$

and

$$\mathcal{P}^{u}(s'|\hat{s}) = \mathcal{P}^{\gamma_t}(\gamma'|\hat{\gamma}_t)p^{\lambda_t}(b'-\hat{b})I(x'=\hat{x}), \tag{5.40}$$

where the term $I(\cdot)$ represents the indicator function, which assumes a value of 1 if its argument is true and assumes a 0 value otherwise. Therefore, both the known and unknown cost functions can also be represented as follows:

$$c(s,a) = P([\gamma_t,x],BEP,y,R) + \Lambda \sum_{\rho^*=0}^{R} \mathcal{P}^{\rho^*}(\rho^*|BEP,R)[b-\rho^*], \tag{5.41}$$

and

$$c^{u}(\hat{s}) = \Lambda\varsigma^* \sum_{\lambda_t=0}^{\infty} \mathcal{P}^{\lambda_t}(\lambda_t)\max(\hat{b}+\lambda_t-B,0). \tag{5.42}$$

Then, the new optimal value function, which is now denoted by $\hat{V}$, can be expressed as a function of the optimal state-value function as follows:

$$\hat{V}(\hat{s}) = c^{u}(\hat{s}) + \gamma' \sum_{s_{t+1}} \mathcal{P}^{u}(s'|\hat{s})\hat{V}(s'), \quad \text{where} \quad \hat{V}(s) = \min_{a\in\mathcal{A}}\left\{c(s,a) + \sum_{\hat{s}}\mathcal{P}(\hat{s}|s,a)\hat{V}(\hat{s})\right\}. \tag{5.43}$$

Now, given this optimal value function, the optimal policy of the model-based RL problem, represented as a factored MDP can be computed as follows:

$$\hat{\pi}(s) = \min_{a\in\mathcal{A}}\left\{c(s,a) + \sum_{\hat{s}}\mathcal{P}(\hat{s}|s,a)\hat{V}(\hat{s})\right\}. \tag{5.44}$$

Contrary to the traditional Q-learning which uses the sample average of the state-value function in order to approximate $\hat{Q}$, the procedure employed here uses the sample average of the value function to approximate V^*. However, because the value function V can be directly computed from the value function $\hat{V}$ using known dynamics and (5.43), the learning algorithm only needs to learn the unknown dynamics in order to learn the optimal value function $\hat{V}$ and optimal policy $\hat{\pi}$. Thus, the optimal value of the Lagrange multiplier Λ in (5.29), which depends on the delay constraint, can be learned online using stochastic dynamics as follows:

$$\Lambda' = \Lambda^*\left[\Lambda_t + \alpha_t(q_t - (1-\gamma')\delta)\right], \tag{5.45}$$

which is in agreement with the assumptions made in [316]. Here, Λ^* projects Λ_t onto the range $[0,\Lambda_{max}]$, where q_t represents the buffer cost with expectation $q(s_t,a_t)$, and the $(1-\gamma')\delta$ term converts the discounted delay constraint δ to an average delay constraint. The procedure for solving this problem is outlined in Algorithm **4** below.

Since the energy problem is a continuous phenomenon, learning good energy-efficient strategies

Algorithm 4 FMDP-based Online Learning for Constrained Power Management Algorithm

Inputs: T, Buffer state b_t, Channel state γ_{t-1}, Power management state x_t, BEP_t, π

Inputs: Power management action y_t, Transmit actions R_t, Number of states $|S|$,

Outputs: Power management policy, Transmission policy, Cost c.

01: Initialize slot counter $t = 1$, Lagrange multiplier $\Lambda = 0$, queue length $q_t = 0$

02: Initialize the value function $\hat{V}(\hat{s}) = \hat{V}(q_t, \gamma_t) = 0$

03: **For** any time slot $t \in T$ **do**

04: Take the greedy action $a_t = \arg\min_{a \in \mathcal{A}} \left\{ c^k(s_t, a_t) + \sum_{\hat{s}} p^k(\hat{s}|s_t, a_t)\hat{V}(\hat{s}) \right\}$

05: Observe the FMDP experience tuple $\hat{\sigma}_t = (s_t, a_t, \hat{s}_t, c_t^u, s_{t+1})$

06: Compute the value of state s_{t+1}: $V^t(s_{t+1}) = \min_{a \in \mathcal{A}} c^k(s_{t+1}, a) + \sum_{\hat{s}} p^k(\hat{s}|s_{t+1}, a_t)\hat{V}(\hat{s})$

07: At time t, update the PDS value function using information from steps 04 and 05 as $\hat{V}^{t+1}(\hat{s}_t) \leftarrow (1-\alpha_t)\hat{V}^t(\hat{s}_t) + \alpha_t\,[c_t^u + \gamma^t V^t(s_{t+1})]$

08: Update the Lagrange multiplier Λ using (5.45)

09: Update counter $t \leftarrow t+1$

10: **End For**

11: Go to step 03

in continuous action-spaces is important and this problem requires that the above obtained solution be extended to a continuous action-space. However, it was stated in [304], that problems involving high-dimensional state-spaces can successfully be solved using the DQN framework; only discrete and low-dimensional action-spaces can be successfully solved using DQNs. To circumvent this predicament, one hss to allow the algorithm ot perform a policy search with an efficient stochastic continuous action search on top of a policy that discretises the state-space but lifts the discretisation in the action search [305]. At this point, with the state-, action-space and the cost function explicitly defined, a way of taking advantage of experience replay in training the distributed policies using DQNs is presented in the following section.

5.5 THE PROPOSED DEEP HIERARCHICAL RL ARCHITECTURE

In this section, a deep hierarchical RL (DHRL) approach is specifically employed to minimize the CRN energy cost in terms of energy consumption, subject to transmission delays and penalizes against packet losses. This is a dynamic computation policy that is learned independently by SU agents, which select actions upon the observation of its own perspective. It is assumed that each SU has no a priori knowledge of the system, which means all SUs and job arrival rates and the CRN environment are initially unknown. Thus, in order to handle the processing required in Figure 5.1, the DHRL architecture employs a DQN with experience replay illustrated in Chapter **2**, Figure 2.8 is employed to minimize policy search time. Then, the DHRL-based CRN architecture can be represented as illustrated in Figure 5.2 below:

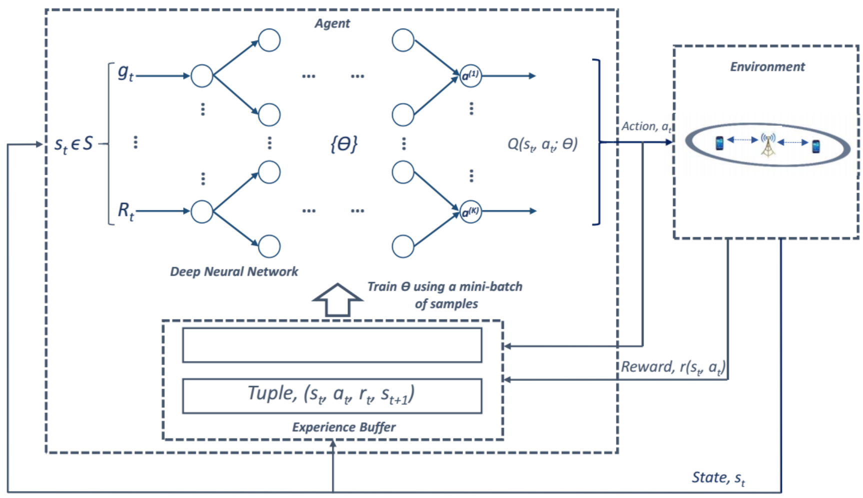

Figure 5.2. An illustration of a DQN.

Figure 5.2 above, shows the proposed DHRL model which deals with power consumption minimization while maintaining the SU QoS requirements by employing an FMDP. In this way, an energy efficiency framework that gives a reward model, which takes into account QoS metrics such as bandwidth and delays can be realized. The advantage of this implementation is that it produces a result that can quickly identify the accurate action during policy evaluation as redundant or similar actions that simultaneously maximizes both RA and energy efficiency. The required overall state-action value $Q(s,a)$ is a combination of both RA and power minimization implementations, almost the same way the value $V(s,a)$ and advantage $A(s,a)$ functions are combined in [300].

Letting $\pi(s,a)$ represent the steady-state probability of the CR system, the experience replay algorithm first initializes a replay memory D, which contains a pool of transition tuples (s_t, a_t, r_t, s') that are randomly generated through the use of an ε-greedy policy [317]. At each time step t, the DQN algorithm randomly samples a mini-batch of transitions of size I from the experience memory buffer D to train the DQN, i.e., $D_i(t) = e_i(1), \cdots, e_i(t)$, where $e_i(t) = (s_i(t), a_i(t), r_i(t+1), s_i(t+1))$. The

Q-values obtained from the training are then used to obtain new transition experiences, which will also be stored in D. The DQN algorithm representation takes advantage of a DNN parameterized by θ in approximating the Q-values $Q(s_t, a_t)$, hence resolving the problem of instability associated with the traditional RL framework in RA.

The DQN algorithm utilizes two separate MLP networks as Q-network estimators, where one estimates the action-value function, $Q_i(s,a;\theta_i)$, while the other estimates the target action-value function, $Q_i(s,a;\theta_i^-)$, as shown in Figure 2.8 in Chapter 2. Therefore, at each time step, the network parameters θ_i of each agents' action-value function are updated through a mini-batch of random samples of transitions $p_i = (s_i, a_i, c_i, \hat{s}_i)$ from the replay buffer memory D_i [60]. Then, the corresponding loss function to be minimized is presented as follows:

$$J(\theta) = \mathbb{E}\left[c(s,a) + \gamma^t \min_{a' \in \mathcal{A}} \hat{Q}(\hat{s}, \hat{a}; \theta^-) - Q(s_j, a_j; \theta)\right]^2. \tag{5.46}$$

The loss function in (5.46) above is used to update the network parameters θ and θ^- through the use of an SGD algorithm with the BP, which is derived according to chain rule as follows:

$$\frac{\partial J(\theta)}{\partial \theta} = \mathbb{E}\left[\left(c_i + \gamma^t \min_{a'} Q(s', a'; \theta^-) - Q(s,a;\theta)\right)\frac{\partial Q(s,a;\theta)}{\partial \theta}\right]. \tag{5.47}$$

In order to ensure sufficient exploration of actions, the use of a Softmax action selection strategy, i.e., $\pi(s|\theta) = \arg\min_a Q(s,a|\theta)$, which is an off-policy approach, defined as the Gibbs Softmax function in (5.20). Then, with this procedure, one can acquire the optimal $Q(s,a)$ for a specific i^{th} function. Otherwise, the traditional action selection procedure is an ε-greedy approach, which is also discussed in Chapter 2 is employed. The experience replay component utilizes stochastic prioritization, which generates the probability of choosing certain actions for replay, thus avoiding the greedy prioritization of services that comes with the ε-greedy approach. The network parameters are updated through a BP technique that follows the update rule

$$\theta' \leftarrow \theta_t + \alpha_t \left[c' + \gamma^t \min_{a \in \mathcal{A}} \hat{Q}(s', a; \theta_t) - Q(s_t, a_t; \theta_t)\right] \nabla_{\theta_t} Q(s_t, a_t; \theta_t), \tag{5.48}$$

where $\hat{Q}$ denotes the action values estimated by the second network, which is updated less frequently for stability purposes. For the purpose of achieving the objective of this work, which is to avoid greedy prioritization of services, only the experience replay component is used. The procedure that summarizes the steps required to compute the loss function in (5.46) for each SU in order to implement the DQN learning mechanism is outlined in Algorithm **5** below.

Algorithm 5 DQN-based Online Learning Algorithm for Constrained Power Management with Experience Replay

Inputs: $\tilde{\sigma}$, Minibatch size I, MC, horizon T.

Outputs: $|A|$, $|c|$, θ, $\hat{Q}$

For each SU $k \in \mathcal{K}$ **do**

  Randomly generate initial state $s_1 = s_t$

  Initialize parameters γ', α_t, $Q \leftarrow \theta$, $\hat{Q} \leftarrow Q$, $\hat{Q} \leftarrow \theta$, $\forall s \in \mathcal{S}, a \in \mathcal{A}$

  Initialize replay memory buffer D_i to memory capacity Ω

  For each time slot $t \in T$ **do**

    Reset SU-BS simulation parameters and monitor the state-space $\mathcal{S}$

    Determine transmit power by selecting action according to $a_{k,t} = \varepsilon\left[\arg\min_a Q(s_t, a|\theta)\right]$

    Execute action $a_{k,t}$ independently, receive reward $r_{k,t}$ and observe next state $s_{k,t+1}$ and with probability ε reject signal

    If reject $= 1$ **then**

      Rerun DQN to get a new action a'

      If $a_t \neq a'$ **then** replace a_t with a'

      End If

    End If

    Set $s' = (s_t, a_t, x')$ and store transition (s_t, a_t, c_t, s') in D

    Randomly sample a mini-batch of I tuple $\{s_j, a_j, c_j, s_{j+1}\}_{j=1}^{I}$ from D

    Update network by minimizing $J(\theta)$ using SGD in (5.47)

    Update network weights θ using $\theta_i' \leftarrow J(\theta_i)$ in (5.48)

  End For

  Train evaluation network and decrement ε by 0.05

  Update target action-value function $\hat{Q} \leftarrow Q$

End For

5.6 SIMULATION RESULTS

In this section, the performance of the proposed algorithms are presented and a summary of parameters used in the simulations are as shown in Table **5.1** and Table 5.2 below. A series of simulations were conducted using $MATLAB^{TM}$, on an Intel(R) Core(TM) i5-4590 @ 3.30GHz processor with 8.00 GB RAM, and resolution of 1920 $\times$ 1080 GPU.

Table 5.1. Simulation Parameters Used in SU QoE Evaluation Algorithm.

Simulation parameter	**Value**
Max. number of SUs, K	42
SU SINR space, γ_k^{su}	$\{-5,-3,-1,1,3,5,7,9,11,13,15\}$ dB
Channel states, γ_{t-1}	[-18.82, -13.79, -11.23,-9.37, -7.8, -6.3, -4.68, -2.08] dB
MOS parameters	$a = 1.3619$, $b = 0.6780$, $\beta = 6.6431$, $f = 30.4264$, $\kappa = -0.1344$

Table 5.2. Simulation Parameters Used in FMDP and the DQN Algorithms.

Simulation parameter	**Value**
BS operating power range, P_{on}	$\{0.32, 0.16, 0.08, 0.04\}$ W
Symbol rate	$f_s = 1/T_s$ 500e3 symbols/sec
Minimum arrival rate, λ_t; Buffer size, B	2 packets/time slot; 30 packets
Hidden layer neurons; Exploration rate, ε	256; 0.9
Hidden layer activation function	Logistic sigmoid
Output later activation function	Gibbs Softmax function
Minibatch size; Replay memory size	32; 10000
Target Q-network update frequency; Finite iteration horizon, T	1000; 2000
Learning rate, α_t; Discount factor, γ^t	0.5; 0.98

5.6.1 SU QoE Evaluation

In this subsection, the performance evaluation results for the QoE evaluation are presented, where four learning mechanisms used by SUs are compared in terms of performance, i.e., the convergence rate and the achievable bit rate. This evaluation shows how quick the environmental awareness experience can be transferred to newly arriving SUs by other SUs already in the CRN. The convergence performance is illustrated in Figure 5.3 below.

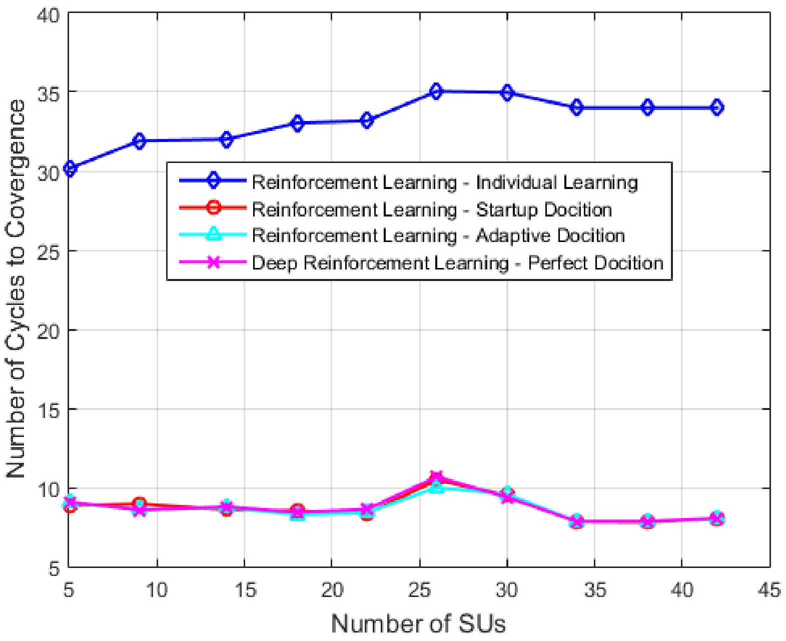

Figure 5.3. Convergence of Individual Learning Compared to the Docitive Paradigms.

Figure 5.3 above shows a comparison of the convergence performance of four learning mechanisms used by SUs. As it can be seen from the above figure, the individual learning mechanism exhibits poor convergence compared to the other three docitive mechanisms. The other three learning mechanisms prove the efficiency of using the docitive learning mechanism by obtaining superior convergence compared to individual learning. Start-up docition, adaptive docition and perfect docition exhibit an almost similar convergence behaviour of taking fewer cycles to converge compared to individual learning. Individual learning converges at 55% more cycles than the other docitive approaches with only five SUs in the CRN, and its performance continues to deteriorate as the number of SUs increases, reaching 70% more cycles with forty-two SUs in the CRN.

The performance evaluation of the four learning schemes in terms of the achievable bit rate is illustrated in Figure 5.4 below. Figure 5.4 above compares the RA performance of the diffent learning mechanisms in terms of the achievable bit rate. As it can be seen from the figure, as the number of SUs in the CRN exceed twenty, the performance of individual learning, startup docition and adaptive docition deteriorates in a similar fashion, while the perfect docition mechanism assumes an almost steady behavior. When the number of SUs in the CRN reaches thirty-four, individual learning, startup docition and adaptive docition have almost the same behavior, while perfect docition continues

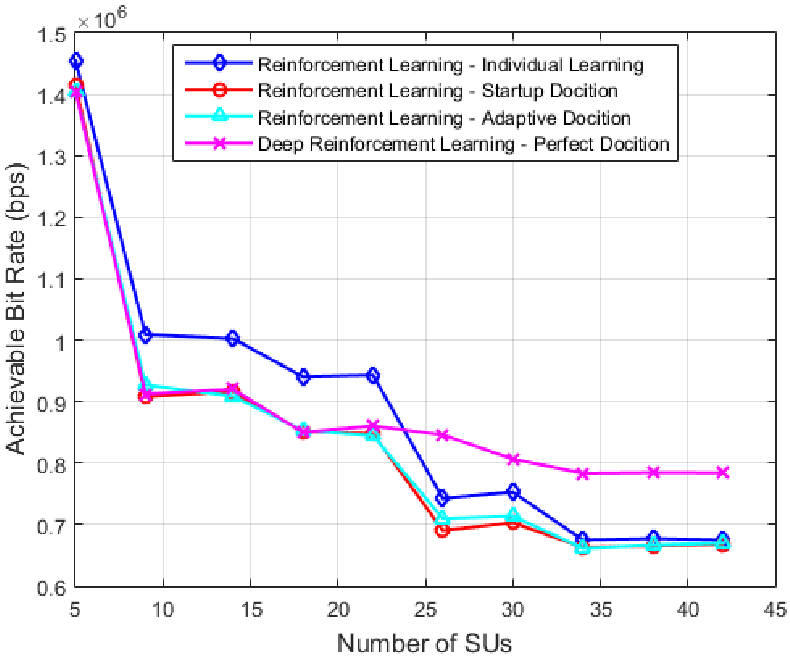

Figure 5.4. Achievable Bit Rate Comparison Between Individual Learning and the Docitive Paradigm.

to dominate with a 6.67% superiority. This behavior is a contradiction to earlier results reported in [291] that perfect docition becomes infeasible for CR problems with many SU nodes owing to the increase in the joint state-action space caused by the increase in SU nodes. However, that was a report associated with the use of the traditional RL algorithm, in this chapter the perfect docition ws used as a completely distributed scheme using DRL, which perfectly justifies the contradiction with earlier results.

5.6.2 CRN Power Management Using the FMDP Algorithm

In this subsection, the performance of the constrained power management algorithm described in Table **4**, is analyzed. In this simulation, the value of ε is decreased from 0.9 by 0.05 in each iteration and to ensure better QoS, the discount factor γ' is made high enough so that ς^* imposes heavy penalties on every dropped packet as the packet arrival rate λ_t is varied from two packets per time slot to four packets per time slot.

Figure 5.5 and Figure 5.6 show the power and average packet holding cost trade-off when the value of P_{ON} is varied. When P_{ON} is high, the power decreases with an increase in the average packet holding

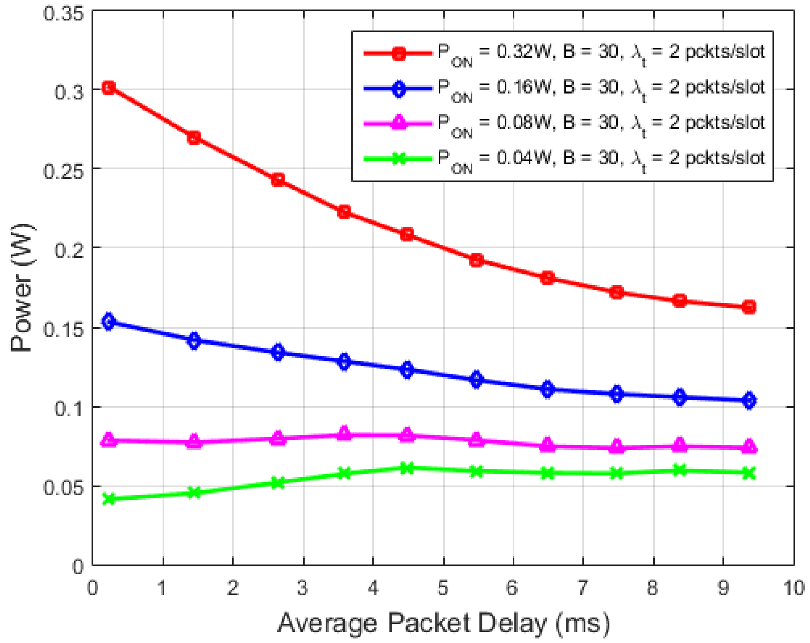

Figure 5.5. Power and Average Packet Holding Cost Trade-off at $\lambda_t = 2$ Packets per Time Slot.

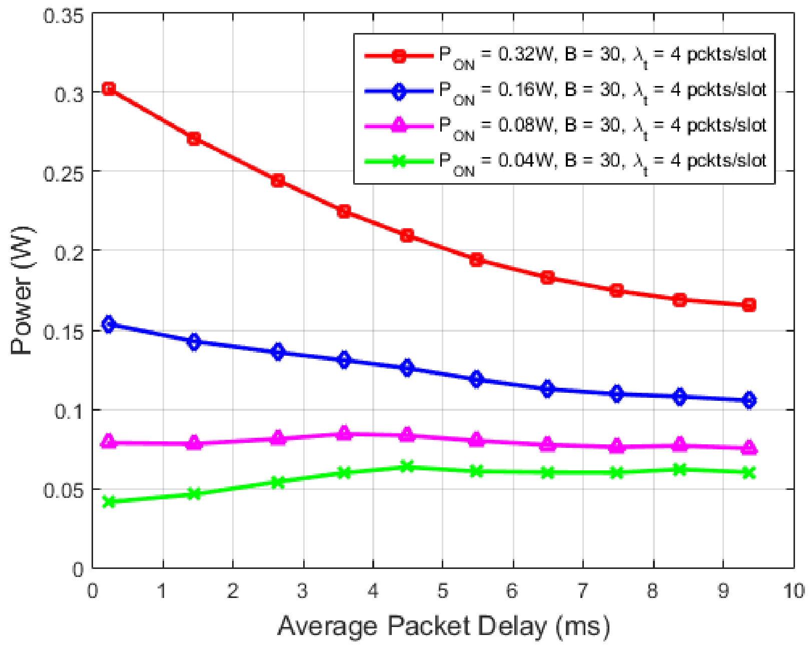

Figure 5.6. Power and Average Packet Holding Cost Trade-off at $\lambda_t = 4$ Packets per Time Slot.

cost, since with a high value of P_{ON}, an increase in average packet holding cost does not affect the packet deadlines, as the power is already high and more packets will be transmitted out of the system per time slot. So, the system just searches for an optimal power per single increase in the holding cost. When the value of P_{ON} is set at 0.08 W, optimal behaviour can be observed. However, when the value of P_{ON} is reduced to 0.04 W, the power increases gradually when the average packet holding cost increases to avoid more costs. Another observation from the figures is that when the packet arrival rate is increased from two to four packets per time slot, only a 0.076% increase in power is noticed at every value of P_{ON}. Even though this percentage increase seems insignificant, it can be inferred that when more packets arrive per time slot more power is expended in order to transmit them without violating the delay constraints.

5.6.3 CRN Power Management using the DQN Algorithm

In this subsection, power management is learned using the DQN algorithm outlined in Table **5** above, with the number of simulation episodes set at $MC = 2000$. The advantage that the DQN has in this section is that almost all the parameters that it needs have already been populated by the MDP algorithm studied in the previous section. So, instead of beginning the whole learning process from scratch, it just loads the population into the DNN as training data and then applies experience replay to learn best policies. However, what needs to be noted is that in this section the statistical packet arrival rate (i.e., packets per time slot) used in the previous section is replaced with higher packet arrival rates that are equivalent to video traffic, hence referred to as job arrival rates ranging from 1.0 Mbps (i.e., equivalent to 200 packets per second) to 3.5 Mbps (i.e., equivalent to 700 packets per second). For comparison purposes, the traditional RL strategy is used as a baseline. Figure 5.7 shows the average buffer delay against the job arrival rates. The DQN algorithm performs better than its RL counterpart by a margin of 5.7% less buffer delays when the arrival rate is at 1.0 Mbps. It can also be observed that as the job arrival rate increases, the average buffer delay shows a linear increase until 2.0 Mbps for both strategies, after which it begins to drop. However, RL continues to show poor performance such that at the set maximum of 3.5 Mbps, DQN performs even better with 12.1% superiority.

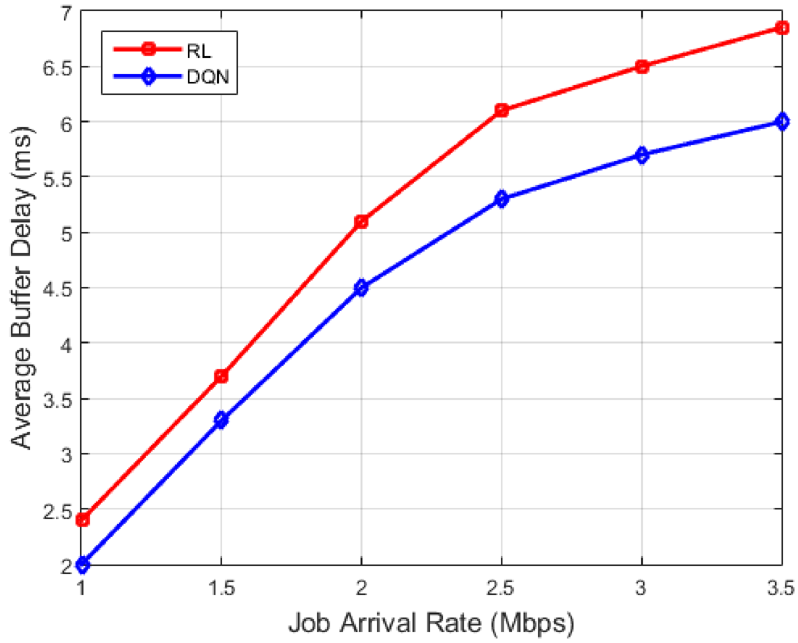

Figure 5.7. Average Buffer Delay vs Job Arrival Rate.

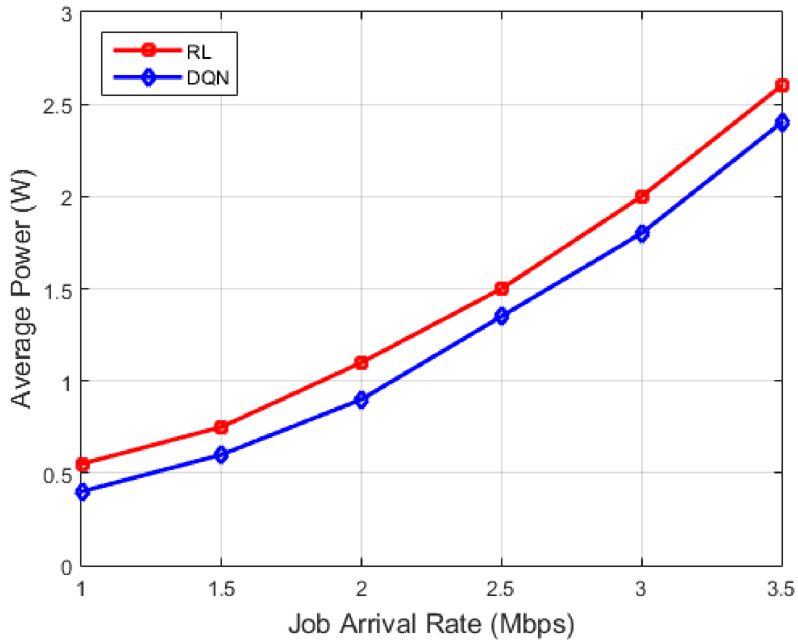

Figure 5.8. Average Power vs Job Arrival rate.

Figure 5.8 shows the average power against job arrival rates which explains the behaviour observed in

Figure 5.7 better. Here, it is observed that the average power consumption increases with an increase in job arrival rates, which suggests that there is a large computation demand, as there are many jobs in the buffer. Because of a higher buffer delay, more computation demand in order to transmit jobs in time leads to high power consumption, hence a negative reward (i.e., cost) as shown in Figure 5.9 below.

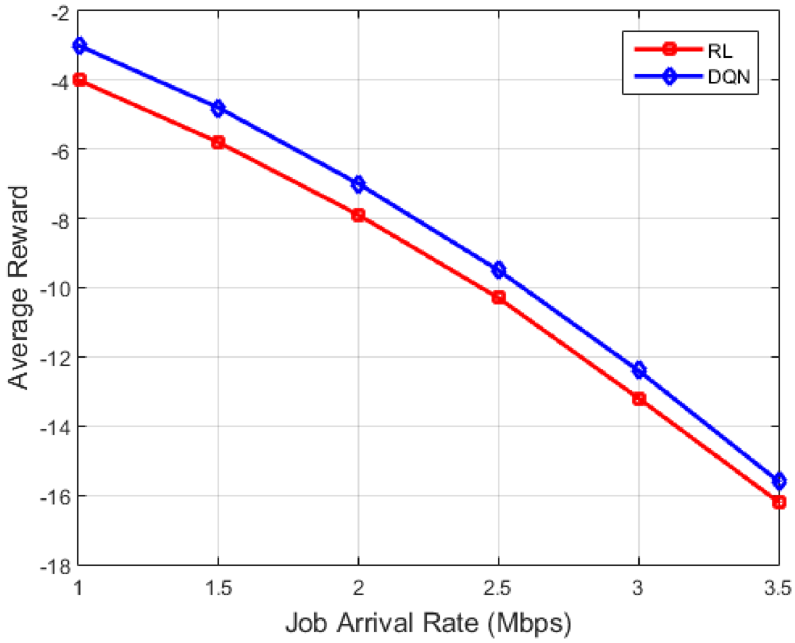

Figure 5.9. Average reward vs Job Arrival Rate.

Figure 5.9 shows the average reward against the job arrival rates, where based on Figure 5.7 and Figure 5.8, the average reward will decrease as the job arrival rates increase. The plummeting of the average reward (i.e., an increasing cost) indicates that the computation cost increases in response to the longer buffer delays shown in Figure 5.7. As the buffer delays increase with increasing job arrival rates, more power is used to push them out of the buffer as shown in Figure 5.8, which then results in increasing costs. However, the DQN algorithm still shows better performance in reducing the cost compared to the traditional RL algorithm. The reason for this is that the DQN combines both the greedy approach and the replay, as opposed to RL, which exploits only the greedy approach.

5.7 CONCLUSION

In this chapter, the problems of QoE and QoS in energy-constrained CRNs for SUs with heterogeneous traffic using a deep hierarchical framework of RL were considered. The first objective was to perform QoE-based RA for SUs in a distributed manner using the docitive strategy. The use of perfect docition to achieve better distributed dynamic RA ect docition to achieve better distributed dynamic RA through distributed QoE evaluation gave a better performance compared to the other learning techniques that use the traditional RL strategy. As a result, it was observed that the perfect docition technique with DRL achieved better spectral efficiency compared to the other docitive paradigms when the number of SUs increased. The simulation results for QoE-based RA indicate that docitive approaches have a 55% better convergence rate compared to the individual learning at 5 SUs and 70% better at 42 SUs, and also achieves a 6.67% superiority in spectral efficiency.

The second objective, was to reduce the cost of power consumption and transmission delays by learning energy-efficient dynamic RA using MDPs and improving the algorithm by using a DQN framework. Minimizing the power consumption was conducted using CMDPs, subject to acceptable transmission deadlines, such that the optimization problem was formulated as a constrained power minimization problem. The solution was obtained using MDPs through Lagrange approximation of the costs of power and queue lengths and solved using FMDPs. The FMDP solution was improved using a DQN algorithm, where the average buffer delays, average power and the average cost (i.e., negative reward in this context) were analyzed against the job arrival rates.

CHAPTER 6 CONCLUSION AND RECOMMENDATIONS FOR FUTURE WORK

6.1 CONCLUSION

CRNs have taken center stage as the de facto technology for solving the dual problem of spectrum scarcity and spectrum under-utilization through dynamic spectrum sharing. For the CR technology to indisputably realize its full potential and make the promised possibilities a plausible reality, energy-efficient RA is an integral component that needs to be addressed. Energy-efficient RA describes how CRNs will be able to optimize the utilization of the limited radio spectrum resources to meet the demands of diverse unlicensed users with heterogeneous services with minimal energy consumption. With the IoT paradigm forming the key strategy for deployment in the next generation of mobile and wireless communications, solving the energy problem through in-depth investigations into the essentials and intricacies of energy-efficient RA is a necessity. Having identified the knowledge gap and several limiting factors in CRN RA, the challenge to undertake research study with an extension of signal processing techniques using ML strategies motivated this study. The investigations conducted in this thesis form a cogent, concise and well-coordinated response to many open-ended problems in energy-efficient RA in CRNs. A detailed summary of the contributions to knowledge and important findings that have already been reported, together with recommendations for further study, are provided as follows:

Chapter 1 provided a succinct introduction and importantly, established the necessary premises that built a strong pivot around which the entire research work eventually revolved. The actual problem definition, which was to find the limiting challenges that hinder the deployment of CRNs in future mobile and wireless networking, was established. Then, a clear presentation of the research objectives and a list of contributions and publications realised during the course of this extensive work was presented.

Chapter 2 presented a comprehensive systematic review of ML strategies which delved into the body of knowledge of CRN RA. Current solution models brought forward to solve RA problems were assessed, classified and analysed, then DL strategies objectively advocated DL to enhance future RA solutions in CRNs. Furthermore, the critical limitations of DL strategies that have been neglected in the advocation of DL strategies to enhance the significance of CRNs were revealed. The chapter concluded by pinpointing future research directions worth pursuing, especially the cross-fertilization of DL strategies with wireless Big Data to improve network planning, spectrum allocation and resource management in future mobile and wireless networking.

Chapter 3 investigated one of the main problems commonly encountered in distributed co-operative spectrum sensing, that of incomplete spectrum sensing data. The problem of spectrum occupancy reconstruction when SSD contributed by SUs have entries with missing values has been overlooked in research because of model and algorithmic deficiencies. After painstakingly identifying the different interpretations of limitations of the different solution approaches published so far, a missing value imputation model that incorporated various missing value mechanisms into the design was developed. An optimization problem based on matrix factorization was formulated and through a careful study of the problem structure, it was solved using computational DL. This technique was an investigation of hybridization between DL and computational physics because it involved solution concepts borrowed from MRFs, ferro-magnetics and Gibbs sampling. The simulation results obtained indicate that spectrum occupancy reconstruction problems can be solved better using the SGD algorithm because it takes advantage of correlations in multiple dimensions better than SVD in matrix factorization.

Chapter 4 presented a solution model for one of the pervasive problems in OSA, the problem of energy-efficient QoS provisioning for multimedia services. One widely adopted energy-efficient strategy to solve this problem uses the power control technique, which has achieved interesting results for voice and data applications. However, the increasing demand for multimedia applications necessitates the establishment of other techniques suitable for multimedia sources. The approach adopted in this work seeks to solve this problem using the application of dynamic games on queues, which was motivated from a perspective of manifold learning and an information theory and is used to train AEs that are stacked to initialize a deep architecture. DL was applied to derive and solve a cost function weighted with power consumption and traffic load by treating the BS processor as a hybrid switching system. The cost function, which is associated with a search for an optimum operating state was optimized for different multimedia information sources, and control actions were used to drive a SAE to achieve better energy efficiency for multimedia services by selecting an appropriate packet-scheduling scheme. The simulation results obtained indicate that the processor sharing scheduling scheme achieves better energy-efficiency than FCFS at higher packet arrival rates.

Chapter 5 presents another solution model to opportunistic spectrum access for CRNs with constrained energy management involving QoS and QoE. In CRN spectrum management problems the primary optimization objective is RA, thus energy consumption is treated as one of the optimization constraints, which makes the realization of energy-efficient CRN operation very challenging. Thus, achieving energy-efficient CRN operation while providing better QoS for CRs requires good spectrum management schemes that are able to handle energy consumption as one of the primary optimization objectives along with RA. After careful consideration of existing solution approaches, the lack optimization techniques that are computationally efficient to handle the use of such a complex integrated model to design sustainable energy-efficient RA strategies was realized. A novel approach that combines both RA and energy consumption into a single formulation was proposed. In this approach, an explicit separation of the representation of state-values (i.e., RA) and state-dependence (i.e., energy consumption) was done. RA was optimized using DNNs, while energy consumption was optimized using FMDPs and the two solutions were combined into a single formulation using a novel DQN approach. The novelty of this this contribution was to demonstrate how DQNs can improve RL strategies if the known and unknown MDP dynamics have been realized and achieve better performance.

6.2 RECOMMENDATIONS FOR FUTURE RESEARCH WORK

Of all the techniques discussed in the systematic review in Chapter **2**, only a few have been explored in this thesis to address spectrum management problems. It would be interesting to investigate the application of deep architectures in solving the problems in distributed RA extensively. An equally important point to consider is that all the problem formulations investigated in this thesis have developed their RA problems in CRN using different solution techniques.

6.2.1 Recommendations Based on the Use of DL in Spectrum Occupancy Reconstruction

In Chapter **3** a spectrum occupancy reconstruction problem was solved without considering the presence of PU, which means that interference constraints were completely ignored. The spectrum occupancy reconstruction employed in chapter 3 to address the problem of missing spectrum sensing data combines techniques from different fields such as computational physics. This strategy indeed introduces an important aspect of viewing spectrum management problems in the context of future mobile and wireless networking. Spectrum occupancy reconstruction is an interesting research area; however, only one aspect of the CRN was adopted in this thesis.

It was assumed that the PUs are inactive during the SU message transfer; therefore the issue of interference was not considered. An extension of this argument to include PU dynamics is recommended, where the problem can be viewed from the perspective of chemical reaction networks using synthon models (an approach that can be borrowed from chemical kinetics). This would indeed be an exciting research focus that can also consider the thinning processes of SUs, both in the presence and in the absence of PU activities.

6.2.2 Recommendations Based on the Transition from Opportunistic Transmission into Opportunistic Computing Using DL

The solution approach in Chapter **4** considered a resource reservation technique that is a variation of the underlay paradigm which suggests that there should always be resources reserve for handling mission-critical applications. Extrapolating this technique to other CRN paradigms would be an interesting exploration. The transition between opportunistic transmissions and opportunistic computing investigated is an interesting technique to tackle the energy efficiency problem in multimedia traffic scenarios. Forecasting short-term changes in network traffic load is a promising way to attempt the RA of real-time traffic and DL offers interesting solutions when dealing with cost functions weighted with different objectives.

Since edge-processing devices will be an integral part of smart orchestration for the 5G vision in creating heterogeneous and multi-domain network environments, it would be interesting to extend this research to focus on consolidation algorithms that are largely effective in knowing the context of users. From the plethora of energy efficiency algorithms in literature, it can be argued that they operate blindly because they are unaware of the different applications that have different QoS requirements and hence different energy consumption rates. Thus, this lack of energy-efficient mechanisms motivates the development of extensions to power management and QoS that explicitly target the introduction of more context awareness for applications or services. In this way, the knowledge of which resources and how many of them are effectively used and which ones are provisioned for other purposes such as mission-critical applications can lead to better resource management solutions.

6.2.3 Recommendations Based on the Use of DRL in Constrained Energy CRN Environments

The problem of limited and strict energy constraints in CRNs which was studied in Chapter **5**, is a critical problem in the deployment of CRNs in future wireless networks. The solution provided in this section translated the problem into a learning problem where a constrained one which was able to handle an energy management problem and safely transport it into a DQN architecture with experience replay. However, with the IoT forming a key component of the next generation strategy, where networks with be distributed and autonomous, this solution becomes inefficient, since it does not take individual network users and their specific technologies into account. In future mobile and wireless networks, the balance of the relative importance among RA, energy efficiency and traffic models in a single QoE formulation, will be crucial. Even though double DQNs are a step towards achieving that, it is still not clear how their "value" and "advantage" dualization can be able to handle more than one optimization problem simultaneously.

Thus, one might recommend that future research on this aspect should focus on the integration of microscopic traffic models that represent individual users and account for user-specific technologies. This is an interesting direction to embed the most detailed user behaviour and network energy supply model into the CR system, by coupling microscopic traffic with an instantaneous power consumption model in order to yield detailed network-wide power consumption estimates for subsequent decisions can solve the energy problem. This means that a third objective, i.e., network traffic, has been added to the problem and may not be treated as a constraint. This is the point where ML techniques also have limitations. Because of this lack of ML techniques that are able to handle than two objectives, one would recommend that such problems be handled using genetic programming techniques. In genetic programming, the microscopic traffic model can capture the evolution of individual users, which requires strategies more advanced than AI that can capture the artificial evolution of active technologies within the network through adaptive computation techniques.

6.3 OTHER RESEARCH DIRECTIONS

In the pursuit of more advanced optimization strategies, which entail an emergent move further from AI, where optimization techniques can mirror RA, energy consumption and network traffic processes simultaneously, here are problems that will need serious attention and application of techniques similar to genetic programming.

6.3.1 Improving High-Speed Communications

High-speed communications are usually encountered in high-speed trains, where the issue of vertical handover between BSs creates numerous QoS problems for passengers. Consider, as an example, passengers desiring high communication in a high-speed train travelling in a typical urban environment where the communication signals are affected by all kinds of fading, giving rise to a multipath channel. In this case, accurate and effective channel models are critical in the design, assessment, and optimization of reliable communication. However, because of high velocity, performing the above-mentioned tasks requires accurate channel learning in which the traditional SP strategies may fall short, thus leading to model deficit problems. Channel learning, which entails the use of dictionaries (i.e, dictionary learning), is one option that can be exploited to solve problems related to channel estimation by exploring the possible underlying channel structures at high velocities. In this case, an AE can be applied for end-to-end communications. A distinctive feature of an AE is that it can handle rapidly time-varying channels using domain-specific regularizers [319]. In this case, DL can be applied in the PHY layer using ANNs in order to simplify the detection of operations such as information acquisition, coding, and localization.

6.3.2 Refining Resource Management Models Using Wireless Big Data

Due to the massive usage of mobile devices, huge amounts of data are produced everyday and this has a profound impact in both our society as well as our social interaction. This has created numerous challenges for mobile business operators, since the volume and heterogeneity of the data that is produced by both network users and the network itself have been exploding exponentially. This renders the wireless network as a Big system Data [320]. Since Big Data are already entrenched in everyday mobile life, it will surely form the core of the upcoming 5G cellular and wireless communications [321]. As the surge towards the era of massive IoT gains momentum, mobile and wireless networks with inaccurate or non-optimal power consumption models need to be refined. The incorporation of AI techniques in both core and access networks which include improving networks' operational efficiency to avoid unplanned network downtime and predict potential network catastrophes, is a requisite. One of the key directions in achieving this feat is the use of data-aided models to find new insights from network data. Big Data are hurtling toward the wireless communication enterprise and one cannot stop it, but can at least be prepared for its imminent arrival. In terms of wireless Big Data, wireless network infrastructure becomes a top priority among enterprise executives as it produces large amounts of network data every second.

With all technological fields (i.e., information technology, communications, physics, economics, behavioural psychology, etc.) converging into one body of knowledge, there is no shortage of advice on how resource management can be refined using wireless Big Data. However, network bandwidth and system throughput are not the aspects of the network that need to be addressed in order to make the most out of wireless Big Data, but better and flexible network connectivity for dynamic environments with rapid scalability are required in order to handle the intermittent nature of Big Data loads. This is because applications for Big Data are usually bursty, since these concern data that can easily stretch into the tera-bytes that need to be processed to improve future network operation.

6.3.2.1 Big Data in Mobile and Wireless Networks

The wide ontogenesis of heterogeneous wireless communication systems and wireless devices has resulted in the arrival of the wireless Big Data era, whose key features are of diverse, high-volume, real-time velocity and huge value, leading to unique research challenges that are different from existing computing systems. In the context of mobile and wireless networking, wireless Big Data data can be divided into two, i.e., there is user data and there is network operator data. A comprehensive analysis of both types of data can provide useful insights for mobile network operators to perform network optimization. These data can be analysed by mobile network operators to perform better network planning, RA and resource management. User and operator data are defined as follows:

- **User Data**: User data are the information collected about network users that are related to the profile and individual user behaviour. This information offers a great deal of insights into users' context, such location, mobility and their communication behaviour. Thus, with the enormous increase in the deployment of smart mobile systems and the rapid expansion of mobile and wireless networks, huge amounts of data are generated from the applications run in mobile systems. In this case, the application-level data have thus become one of the principal sources of wireless Big Data in today's wireless networks and can be used to provide the right amount of resources to users. In CRNs, the resources requested and the services run by SUs vary or may tend to vary and dynamically change from time to time, resulting in dynamically changing user behaviour. For example, the amount of resources consumed by applications with similar content may be requested by users distributed in one hotspot owing to a certain social pattern among the users. In this case, there are users with the same personal communication behaviour gathered in a small area. From a macro-cell point of view, the number of users with the same service streams in a certain time window may be large enough, which may probably bring some advantages by concentrating network resources within each service group. Such a user pattern is supposed to exhibit certain group dynamics that are associated with strong social phenomena that the CR system needs to understand. Therefore, in order to understand and model user behaviour based on traffic dynamics in CRNs better, reference can be made from the statistics and economics perspective based on the Gini coefficient.
- **Network Operator Data**: The main source of the data that are collected by mobile network operators is mainly the core network as well as the RAN. The core network possesses and abundance of service data regarding network performance information, successful calls and per application usage index. A large amount of data is also sourced from the RAN, including (i) cell information such as BS configuration information, resource status information, interference information, handover reports, mobility information, fault status, link utilization, call dropping ratio, (ii) signalling messages exchanged between the BS and users such as radio resource controller messages for connection establishment and handover, and (iii) radio signal

measurements such as reference signal received power, reference signal received quality, etc. This information can be used to develop learning-based cognition techniques such as PU traffic statistics and channel occupancy, channel estimation and energy consumption that may help in developing a fully learned CR system.

However, in as far as deep transfer-learning is concerned, it is still not clear how hyper-parameter setting (i.e., the amount of model-based data, number of NN layers) can be performed to avoid negative transfer, which leaves it as one of the interesting open issues. A possible research direction that could provide guidance in achieving this cross-fertilization is deriving a theoretical explanation of how NNs work and how to configure them to perform certain wireless networking tasks. Therefore, opening the black box of NNs to understand the information-theoretic principles that regulate their behaviour is surely a major topic for future investigation.

Once both types of data have been collected from these two sources, the next challenge becomes using it efficiently, since it needs to be processed in order to be converted into useful and actionable knowledge, which can then be utilized to improve network performance. Using advanced data collection techniques and data analytics, adaptive algorithms and optimization strategies can be designed to improve network operation and performance. Since the collection of raw wireless data is the first and important step in performing Big Data analytics in mobile and wireless networking, mobile network operators can collect data from mobile users using data mining techniques. The users may share/download information associated with their mobility and data mining techniques, through the use of filtering and extraction can be used to remove interference and useless data that might cause classification errors.

However, one of the major challenges is still the obtaining of useful information from incomplete, redundant and uncertain data. And with the unprecedented increase in collected data, network operators may require effective tools to perform both data and predictive analytics that will enable fast response and real-time classification and analysis. With current powerful ML techniques such as SVM and DL, Big Data applications offer predictive analytics, since these reflect the emerging advance of multivariate statistics, pattern recognition, and data mining. In such cases, DL plays the most important role if the deep and predictive insights are required to uncover hidden knowledge from large and fast-changing data sets. This is because the accuracy, the scale and the speed that are required are the main metrics in the evaluation of sequence classification.

REFERENCES

[1] R. K. Singh and N. Tiwari, "An Investigation on Wireless Mobile Network and Wireless LAN (WiFi) for Performance Evaluation," *International Journal of Computer Applications*, vol. 126, no. 6, pp. 1 - 8, September 2015.

[2] R. N. Clarke, "Expanding Mobile Wireless Capacity: The Challenges Presented by Technology and Economics," *Telecommunications Policy*, vol. 38, no. 8 - 9, pp. 693 - 708, 01 September 2014.

[3] R. Struzak, T. Tjelta, and J. P. Borrego, "On Radio-Frequency Spectrum Management," *The Radio Science Bulletin*, vol. 2015, Issue 354, pp. 11 - 35, September 2015.

[4] S. D. Barnes, P. A. Jansen van Vuuren, and B.T. Maharaj, "Spectrum Occupancy Investigation: Measurements in South Africa," *Measurement*, vol. 46, no. 9, pp. 3098 - 3112, 01 November 2013.

[5] J. Reed, M. Vassiliou, and S. Shah, "The Role of New Technologies in Solving the Spectrum Shortage [Point of View]," in *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1163 - 1168, June 2016.

[6] J. L. Kiddoo, E. Kwerel, S. Javid, M. Dunford, G. M. Epstein, C. E. Meisch Jr, K. L. Hoffman, B. B. Smith, A. B. Coudert, R. K. Sultana, and J. A. Costa, "Operations Research Enables Auction to Repurpose Television Spectrum for Next-Generation Wireless Technologies," *INFORMS Journal of Applied Analytics*, vol. 49, no. 1, pp. 7 - 22, January 2019.

[7] S. Yin, D. Chen, Q. Zhang, M. Liu, and S. Li, "Mining Spectrum Usage Data: A Large-scale Spectrum Measurement Study," *IEEE Transactions on Mobile Computing*, vol. 11, no. 6, pp. 1033 - 1046, June 2012.

[8] S. Haykin, "Cognitive Radio: Brain-empowered Wireless Communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201 - 220, 07 February 2005.

[9] G. Joshi, S. Nam, and S. Kim, "Cognitive Radio Wireless Sensor Networks: Applications, Challenges and Research Trends," *Sensors*, vol. 13, no. 9, pp. 11196 - 11228, September 2013.

[10] P. K. Sangdeh, H. Pirayesh, H. Zeng, and H. Li, "A Practical Underlay Spectrum Sharing Scheme for Cognitive Radio Networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, Paris, France, pp. 2521 - 2529, 29 April - 02 May 2019.

[11] B. S. Awoyemi, B. T. Maharaj, A. S. Alfa, "Solving Resource Allocation Problems in Cognitive Radio Networks: A Survey," *EURASIP Journal on Wireless Communications and Networking*, vol. 176, no. 1, pp. 1 - 14, 29 July 2016.

[12] M. T. Masonta, M. Mzyece, and N. Ntlatlapa, "Spectrum Decision in Cognitive Radio Networks: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1088 - 1107, 29 November 2012.

[13] J. Wu, T. Song, Y. Yu, C. Wang, and J. Hu, "Sequential Cooperative Spectrum Sensing in the Presence of Dynamic Byzantine Attack for Mobile Networks," *PloS one*, e0199546, vol. 13, no. 7, pp. 1 - 32, 05 July 2018.

[14] Ericsson Incorporated, "Ericsson Interim Mobility Report." February 2018. Accessed September 16, 2018, [Online]. Available: https://www.ericsson.com/assets/ local/mobility-report/documents/2018/emr-interim-feb-2018.pdf

[15] Index, Cisco Visual Networking. "Global Mobile Data Traffic Forecast Update, 2016-2021." White Paper, 07 February 2017. Accessed September 16, 2018, [Online]. Available: http://tinyurl.com/zzo6766.pdf

[16] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Mobile Edge Computing and Networking for Green and Low-latency Internet of Things," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 39 - 45, 17 May 2018.

[17] D. Erdogmus and J. C. Principe, "From Linear Adaptive Filtering to Nonlinear Information Processing - the Design and Analysis of Information Processing Systems," *IEEE Signal Processing Magazine*, vol. 23, no. 6, pp. 14 - 33, 20 November 2006.

[18] T. Schenk, *RF Imperfections in High Rate Wireless Systems: Impact and Digital Compensation.* New York, NY, USA: Springer Science & Business Media, 29 January 2008.

[19] K. Hornik, H. Stinchcombe, and H. White, "Multilayer Feedforward Networks and Universal Approximators," *Neural Networks*, vol. 2, no. 5, pp. 359 - 366, 01 July 1989.

[20] S. Reed and N. de Freitas, "Neural Programmer-Interpreters," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Caribe Hilton, San Juan, Puerto Rico, 02 - 04 May 2016.

[21] H. T. Siegelmann and E. D. Sontag, "On the Computational Power of Neural Networks," in *Proceedings of the 5th Annual Workshop on Computational Learning Theory*, ACM, Pittsburgh, Pennsylvania, USA, pp. 440 - 449, 27 - 29 July 1992.

[22] V. Vanhoucke, A. Senior, and M. Z. Mao, "Improving the Speed of Neural Networks on CPUs," in *Proceedings of Deep Learning and Unsupervised Feature Learning NIPS Workshop*, Telecabina, Sierra Nevada, Spain, pp. 1 - 8, 16 December 2011.

[23] T. O'Shea and J. Hoydis, "An Introduction to Deep Learning for the Physical Layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563 - 575, December 2017.

[24] A. Goldsmith, "Joint Source/Channel Coding for Wireless Channels," in *Proceedings of the IEEE Vehicular Technology Conference*, Chicago, Illinois, USA, pp. 614 - 618, 25 - 28 July 1995.

[25] E. Zehavi, "8-PSK Trellis Codes for Rayleigh Channel," *IEEE Transactions in Communications*, vol. 40, no. 5, pp. 873 - 884, May 1992.

[26] H. Wymeersch, *Iterative Receiver Design.* Cambridge: Cambridge University Press, 24 September 2007.

[27] M. Chen, U. Challita, W. Saad, C. Yin, and M. Debbah, "Machine Learning for Wireless Networks with Artificial Intelligence: A Tutorial on Neural Networks," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3039 - 3071, 09 October 2017.

[28] R. Raina, A. Madhavan, and A. Y. Ng, "Large-scale Deep Unsupervised Learning Using Graphics Processors," in *Proceedings of 26th International Conference on Machine Learning (ICML)*, Montreal, Quebec, Canada, pp. 873 - 880, 14 - 18 June 2009.

[29] H. Iwata, "Report on 21st global standards collaboration (GSC-21): Global Standardization Activities," NTT DoCoMo, Chiyoda, Tokyo, Japan, NTT Technical Review, vol. 16, no. 3, March 2018. Accessed September 16, 2018, [Online]. Available: https://www.ntt-review.jp.pdf

[30] A. Sivakumaran, A. S. Alfa, and B. T. Maharaj, "An Empirical Analysis of the Effect of Malicious Users in Decentralised Cognitive Radio Networks," in *In Proceedings of IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, Kuala Lumpur, Malaysia, pp. 1 - 5, 28 April - 01 May 2019.

[31] A. Rao, J. Voyles, and P. Ramchandani, "Top 10 Artificial Intelligence Technology Trends for 2018," 2017. Accessed September 29, 2018, [Online]. Available: http://usblogs.pwc.com/emergingtechnology/top-10-ai-tech-trends-for-2018.html

[32] H. He, S. Jin, C. K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-driven Deep Learning for Physical Layer Communications," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 77 - 83, 15 may 2019.

[33] R. Yu, "Huawei Reveals the Future of Mobile AI at IFA 2017," 02 September, 2017. Accessed September 29, 2018, [Online]. Available:

http://www.businesswire.com/news/home/20170902005020/en/Huawei-Reveals-Future-Mobile-AI-IFA-2017.html

[34] S. Kovach, "Qualcomm CEO Steve Mollenkopf: What the Big Innovation House that Powered the Mobile Boom is Betting on Next," 02 August 2017. Accessed August 17, 2018, [Online]. Available: https://www.businessinsider.com.au/qualcomm-ceo-steve-mollenkopf-interview-2017-7.html

[35] H. Yu, L. Gao, Y. Li, X. Gan, X. Wang, Y. Xu, W. Chen, and A. V. Vasilakos, "Information Sharing in Spectrum Auction for Dynamic Spectrum Access," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, Hawaii, USA, pp. 1 - 5, 30 November - 04 December 2009.

[36] L. Zhang, L. Huang, and H. Xu, "Queuing-theoretical Multi-user Dynamic Spectrum Allocation & Sharing Mechanism in Multi-channel Cognitive Radio Networks," in *Proceedings of the IEEE Computers, Communications and IT Applications Conference*, Beihang University, Beijing, China, pp. 137 - 142, 20 - 22 October 2014.

[37] P. Steenkiste, D. Sicker, G. Minden, and D. Raychaudhuri, "Future Directions in Cognitive Radio Network Research," *NSF Workshop Report*, vol. 4, no. 1, pp. 1 - 2, 09 March 2009.

[38] L. Ding, T. Melodia, S. N. Batalama, and J. D. Matyjas, "Distributed Resource Allocation in Cognitive and Cooperative Ad-hoc Networks Through Joint Routing, Relay Selection and Spectrum Allocation," *Computer Networks*, vol. 83, pp. 315 - 331, 04 June 2015.

[39] S. Chitnavis and A. Kwasinski, "Cross Layer Routing in Cognitive Radio Networks using Deep Reinforcement Learning," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Marrakesh, Morocco, 2019, pp. 1-6.

[40] J. Cheng, W. Chen, F. Tao, and C. L. Lin, "Industrial IoT in 5G Environment Towards Smart Manufacturing," *Journal of Industrial Information Integration*, vol. 10, pp. 10 - 19, 01 June 2018.

[41] S. Chen, H. Wen, J. Wu, J. Chen, W. Liu, L. Hu, and Y. Chen, "Physical-layer Channel Authentication for 5G via Machine Learning Algorithm," *Wireless Communications and Mobile Computing, Hindawi*, Volume 2018, Article ID 6039878, pp. 1 - 10, 02 October 2018.

[42] E. J. Kitindi, S. Fu, Y. Jia, A. Kabir, and Y. Wang, "Wireless Network Virtualization with SDN and CRAN for 5G Networks: Requirements, Opportunities, and Challenges," *IEEE Access*, vol. 5, pp. 19099 - 19155, 04 September 2017.

[43] D. Bouneffouf and R. Feraud, "Multi-armed Bandit Problem with Known Trend, *Neurocomputing, Elsevier*, vol. 205, pp. 16 - 21, 12 September 2016.

[44] Union IT, "ITU-T recommendation G.114: one-way transmission time," Rep. TR-5, 2003. Accessed November 20, 2018, [Online]. Available: https://www.itu.int/rec/T-REC-G.114.pdf

[45] R. A. Suliman, K. H. Bilal, and I. Elemam, "Review Paper on Cognitive Radio Networks," *Journal of Electrical and Electronic Systems*, vol. 7, Issue 1, pp. 1 - 3, January 2018.

[46] G. Salami, O. Durowoju, A. Attar, O. Holland, R. Tafazolli, and H. Aghvami, "A Comparison Between the Centralized and Distributed Approaches for Spectrum Management," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 274 - 290, 27 May 2010.

[47] M. van Steen, and A. S. Tanenbaum, "A Brief Introduction to Distributed Systems," *Computing*, vol. 98, no. 10, pp. 967 - 1009, 01 October 2016.

[48] S. Filin, H. Harada, H. Murakami, and K. Ishizu, "International Standardization of Cognitive Radio Systems," *IEEE Communications Magazine*, vol. 49, no. 3, pp. 82 - 89, 07 March 2011.

[49] C. Moy, "High-level Design Approach for the Specification of Cognitive Radio Equipments Management APIs," *Journal of Network and Systems Management*, vol. 18, no. 1, pp. 64 - 96, 01 March 2010.

[50] Q. Zhang, A. B. Kokkeler, and G. J. Smit, "A Reconfigurable Radio Architecture for Cognitive Radio in Emergency Networks," in *Proceedings of the 9th IEEE European Conference on Wireless*

Technology, Manchester, UK, pp. 35 - 38, 10 - 12 September 2006.

[51] M. Hoyhtya, A. Mammela, X. Chen, A. Hulkkonen, J. Janhunen, J-C. Dunat, and J. Gardey, "Database-Assisted Spectrum Sharing in Satellite Communications: A Survey," *IEEE Access*, vol. 5, pp. 1 - 20, 09 November 2017.

[52] T. Sturman, A. Burr, J. Fitzpatrick, T. James, M. Rupp, and S. Weiss, "An Evaluation of Software Defined Radio - An Overview," QinetiQ, Farnborough, Hampshire, UK, QinetiQ Version 1.0, March 2006. Accessed March 06, 2017, [Online]. Available: https://www.ofcom.org.uk/data/assets/pdf/file/0018/40338/evaluation/sdr.pdf

[53] J. R. Machado-Fernandez, "Software Defined Radio: Basic Principles and Applications," *Revista Facultad de Ingenieria*, Universidad de Antioquia, Antioquia, Colombia, vol. 24, no. 38, pp. 79 - 96, January 2015.

[54] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. da Silva, "Learning and Reasoning in Cognitive Radio Networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1761 - 1777, 28 March 2013.

[55] I. Malanchini, M. Cesana, and N. Gatti, "On Spectrum Selection Games in Cognitive Radio Networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, Hawaii, USA, pp. 1 - 7, 30 November - 04 December 2009.

[56] F. Fu and M. van der Schaar, "Learning to Compete for Resources in Wireless Stochastic Games," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1904 - 1919, 29 June 2008.

[57] J. Huang and V. Krishnamurthy, "Transmission Control in Cognitive Radio as a Markovian Dynamic Game: Structural Result on Randomized Threshold Policies," *IEEE Transactions on Communications*, vol. 58, no. 1, pp. 301 - 310, 22 January 2010.

[58] D. Niyato and E. Hossain, "Dynamics of Network Selection in Heterogeneous Wireless Networks: An Evolutionary Game Approach," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 2008 - 2017, May 2009.

[59] T. Jiang, D. Grace, and P. Mitchell, "Efficient Exploration in Reinforcement Learning-based Cognitive Radio Spectrum Sharing," *IET Communications*, vol. 5, no. 10, pp. 1309 - 1317, 01 July 2011.

[60] C. Wu, K. Chowdhury, M. Di Felice, and W. Meleis, "Spectrum Management of Cognitive Radio Using Multi-agent Reinforcement Learning," in *Proceedings if the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'10)*, Toronto, Canada, pp. 1705 - 1712, 10 - 14 May 2010.

[61] B. Lo and I. Akyildiz, "Reinforcement Learning-based Cooperative Sensing in Cognitive Radio Ad hoc Networks," in *Proceedings of the 21st IEEE International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, Istanbul, Turkey, pp. 2244 - 2249, 26 - 29 September 2010.

[62] I. Macaluso, L. DaSilva, and L. Doyle, "Learning Nash Equilibria in Distributed Channel Selection for Frequency-agile Radios," in *Proceedings of the Workshop on Artificial Intelligence for Telecommunications and Sensor Networks*, Montpellier, France, pp. 07 - 10, 27 - 31 August 2012.

[63] T. E. Bogale, X. Wang, and L. B. Le, "Machine Intelligence Techniques for Next-generation Context-aware Wireless Networks," *ITU Journal on ICT Discoveries*, Special Issue no. 1, pp. 109 - 119, 02 February 2018.

[64] M. Matinmikko, M. Mustonen, T. Rauma, and J. Del Ser, "Decision-making System for Obtaining Spectrum Availability Information in Opportunistic Networks," in *Proceedings of the 4th International Conference on Cognitive Radio and Advanced Spectrum Management*, Barcelona, Spain, pp. 1 - 6, 26 - 29 October 2011.

[65] Y. Wu, F. Hu, S. Kumar, Y. Zhu, A. Talari, N. Rahnavard, and J. D. Matyjas, "A Learning-based QoE-driven Spectrum Handoff Scheme for Multimedia Transmissions over Cognitive Radio Networks," *IEEE Journal on Selected Areas Communications*, vol. 32, no. 11, pp. 2134 - 2148, 18 December 2014.

[66] S. Y. Lien, K. C. Chen, Y. C. Liang, and Y. Lin, "Cognitive Radio Resource Management for Future Cellular Networks," *IEEE Wireless Communications*, vol. 21, no. 1, pp. 70 - 79, 06 March 2014.

[67] T. O. Olwal, K. Djouani, and A. M. Kurien, "A Survey of Resource Management Toward 5G Radio Access Networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1656 - 1686, 05 April 2016.

[68] W. Tong and P. Zhu, Huawei, "5G: A Technology Vision, White Paper," Huawei, Shenzhen, China, White Paper, March 2014. Accessed June 12, 2018, [Online]. Available: https://www.huawei.com/5gwhitepaper.pdf

[69] M. Z. Asghar, M. Abbas, K. Zeeshan, P. Kotilainen, and T. Hamalainen, "Assessment of Deep Learning Methodology for Self-Organizing 5G Networks," *Applied Sciences Journal*, vol. 9, no. 15, pp. 2975, 25 January 2019.

[70] Z. Zhang, K. Long, and J. Wang, "Self-organization Paradigms and Optimization Approaches for Cognitive Radio Technologies, A Survey," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 36 - 42, 25 April 2013.

[71] F. Ahmed, "Self-Organization: A Perspective on Applications in the Internet of Things," in *Natural Computing for Unsupervised Learning*, New York, USA, Springer, Cham, pp. 51 - 64, January 2019.

[72] A. F. Molisch, H. Asplund, R. Heddergott, M. Steinbauer, and T. Zwick, "The COST259 Directional Channel Model-Part I: Overview and Methodology," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3421 - 3433, 11 December 2006.

[73] D. G. Costa and L. A. Guedes, "The Coverage Problem in Video-Based Wireless Sensor Networks: A Survey," *Sensors*, vol. 10, no. 9, pp. 8215 - 8247, 02 September 2010.

[74] S. S. Alam, L. Marcenaro, and C. Regazzoni, "Opportunistic Spectrum Sensing and Transmissions," in *Cognitive Radio and Interference Management: Technology and Strategy*, Hershey,

Pennsylvania, USA, IGI Global, pp. 1 - 28, 2013.

[75] F. Zhou, X. Zhang, R. Q. Hu, A. Papathanassiou, and W. Meng, "Resource Allocation Based on Deep Neural Networks for Cognitive Radio Networks," in *Proceedings of the IEEE/CIC International Conference on Communications in China (ICCC)*, Beijing, China, pp. 40 - 45, 16 - 18 August 2018.

[76] X. Xing, T. Jing, Y. Huo, H. Li, and X. Cheng, "Channel Quality Prediction Based on Bayesian Inference in Cognitive Radio Networks," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, Turin, Italy, pp. 1465 - 1473, 14 - 19 April 2013.

[77] P. S. Yawada and M. T. Dong, "Intelligent Process of Spectrum Handoff/Mobility in Cognitive Radio Networks," *Journal of Electrical and Computer Engineering*, vol. 2019, Article ID 7692630, pp. 1 - 12, 2019.

[78] P. A. Iannucci, J. Perry, H. Balakrishnan, and D. Shah, "No Symbol Left Behind: A Link-layer Protocol for Rateless Codes," in *Proceedings of the 18th ACM Annual International Conference on Mobile Computing and Networking*, Istanbul, Turkey, pp. 17 - 28, 22 - 26 August 2012.

[79] Y. Wu, S. Kumar, F. Hu, Y. Zhu, and J. D. Matyjas, "Cross-layer Forward Error Correction Scheme Using Raptor and RCPC Codes for Prioritized Video Transmission over Wireless Channels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 6, pp. 1047 - 1060, 22 January 2014.

[80] F. Zhou, N. C. Beaulieu, Z. Li, J. Si, and P. Qi, "Energy-efficient Optimal Power Allocation for Fading Cognitive Radio Channels: Ergodic Capacity, Outage Capacity and Minimum-rate Capacity," *IEEE Transactions on Wireless Communications*, vol. 15, no. 4, pp. 2741 - 2755, 17 December 2015.

[81] F. Zhou, Y. Wu, R. Q. Hu, Y. Wang, and K. K. Wong, "Energy-efficient NOMA Enabled Heterogeneous Cloud Radio Access Networks," *IEEE Networks*, vol. 32, no. 2, pp. 152 - 160, 13 March 2018.

[82] X. Zhang, H. Li, Y. Lu, and B. Zhou, "Distribution Energy Efficiency Optimization for MIMO Cognitive Radio Network," *IEEE Communication Letters*, vol. 19, no. 5, pp. 847 - 850, 18 March 2015.

[83] B. Dickson, "Decentralizing IoT Networks Through Blockchain," TechCrunch, Bay Area, California, USA, 28 June 2016. Accessed January 07, 2019, [Online]. Available: https://techcrunch.com/2016/06/28/decentralizing-iot-networks-through-blockchain.html

[84] J. Kalliovaara, T. Jokela, H. Kokkinen, and J. Paavola, "Licensed Shared Access Evolution to Provide Exclusive and Dynamic Shared Spectrum Access for Novel 5G Use Cases," in *Cognitive Radio in 4G/5G Wireless Communication Systems*, London, UK, IntechOpen, pp. 55 - 72, 05 November 2018.

[85] D. Giusto, A. Iera, G. Morabito, and L. Atzori, *The Internet of Things: The 20th Tyrrhenian Workshop on Digital Communications.* New York, NY, USA: Springer Science & Business Media, 10 March 2010.

[86] Z. Qin, H. Ye, G. Li, and B-H. F. Juang, "Deep Learning in Physical Layer Communications," *IEEE Wireless Communications*, vol. 26, Issue 2, pp. 93 - 99, April 2019.

[87] P. H. Kuo, H. T. Kung, and P. A. Ting, "Compressive Sensing Based Channel Feedback Protocols for Spatially-correlated Massive Antenna Arrays," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Paris, France, pp. 492 - 497, 01 - 04 April 2012.

[88] O. Simeone, "A Brief Introduction to Machine Learning for Engineers," in *Foundations and Trends in Signal Processing*, vol. 12, no. 3 - 4, 14 August 2018, pp. 200 - 431.

[89] T. Nakano, "Molecular Communication: A 10 Year Retrospective," *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, vol. 3, Issue 2, pp. 71 - 78, June 2017.

[90] A. J. Majda, and M. Branicki, "Lessons in Uncertainty Quantification for Turbulent Dynamical Systems," *Discrete & Continuous Dynamical Systems-A*, vol. 32, no. 9, pp. 3133 - 3221, 01

April 2012.

[91] N. N. Srinidhi, S. D. Kumar, and K. R. Venugopal, "Network Optimizations in the Internet of Things: A Review," *International Journal on Engineering Science and Technology*, vol. 22, Issue 1, pp. 1 - 21, February 2019.

[92] P. Barry and P. Crowley, *Modern Embedded Computing: Designing Connected, Pervasive, Media-rich Systems.* San Francisco, USA: Elsevier Science & Tecbnology, Elsevier, 10 February 2012.

[93] M. Haenggi, J. G. Andrews, F. Baccelli, O. Dousse, and M. Franceschetti, "Stochastic Geometry and Random Graphs for the Analysis and Design of Wireless Networks," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 7, pp. 1029 - 1046, 28 August 2009.

[94] V. De Oliveira, "Poisson Kriging: A Closer Investigation," *Spatial Statistics*, vol. 7, pp. 1 - 20, 01 February 2014.

[95] J. Jalden and P. Elia, "Sphere Decoding Complexity Exponent for Decoding Full-rate Codes over the Quasi-static MIMO Channel," *IEEE Transactions on Information Theory*, vol. 58, no. 9, pp. 5785 - 5803, 12 June 2012.

[96] A. Stopczynski, V. Sekara, P. Sapiezynski, A. Cuttone, M. M. Madsen, J. E. Larsen, and S. Lehmann, "Measuring Large-scale Social Networks with High Resolution," *PloS One*, vol. 9, no. 4, e95978, pp. 1 - 24, 25 April 2014.

[97] S. Reed and N. de Freitas, "Neural Programmer-Interpreters," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Caribe Hilton, San Juan, Puerto Rico, 02 - 04 May 2016.

[98] C. Song, T. Ristenpart, and V. Shmatikov, "Machine Learning Models that Remember Too Much," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas, Texas, USA, pp. 587 - 601, 30 October - 03 November 2017.

[99] M. Lanctot, V. Zambaldi, A. Gruslys, A. Lazaridou, K. Tuyls, J. Perolat, D. Silver, and T. Graepel, "A Unified Game-theoretic Approach to Multiagent Reinforcement Learning," in *Proceedings of the 31st Conference in Neural Information Processing Systems (NIPS)*, Long Beach, California, USA, pp. 4190 - 4203, 04 - 09 December 2017.

[100] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, and H. Arshad, "State-of-the-art in Artificial Neural Network Applications: A Survey," *Heliyon*, vol. 4, no. 11, e00938, 01 November 2018.

[101] E. Guresen and G. Kayakutlu, "Definition of Artificial Neural Networks with Comparison to other Networks," *Procedia Computer Science*, vol. 3, pp. 426 - 433, 01 January 2011.

[102] B. O. Ayinde, T. Inanc, and J. M. Zurada, "On Correlation of Features Extracted by Deep Neural Networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, Budapest, Hungary, pp. 1 - 8.

[103] Y. Altun, M. Johnson, and T. Hofmann, "Investigating Loss Functions and Optimization Methods for Discriminative Learning of Label Sequences," in *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics*, Sapporo, Japan, pp. 145 - 152, 11 - 12 July 2003.

[104] J. Schmidt, "Artificial intelligence: An online exploration of the public's attitude towards the increased use of AI in today's society and the associated future consequences," M. S. Dissertation, School of Business & Economics, Catholic University of Portugal, Portugal. 2019.

[105] H. Trevor, T. Robert, J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* New York, NY, USA: Springer Science & Business Media, 26 August 2009.

[106] Z. Zhang and M. Sabuncu, "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels," in *Advances in Neural Information Processing Systems (NIPS)*, Montreal, Canada, 03 - 08 December 2018, pp. 8778 - 8788.

[107] P. McKeigue, "Sample Size Requirements for Learning to Classify with High-dimensional Biomarker Panels," *Statistical Methods in Medical Research*, vol. 28, no. 3, pp. 904 - 910, March 2019.

[108] A. Quemy, "Binary Classification in Unstructured Space with Hypergraph Case-based Reasoning," *Information Systems*, vol. 85, pp. 92 - 113, November 2019.

[109] C. M. Judd, G. H. McClelland, and C. S. Ryan, *Data Analysis: A Model Comparison Approach.* New York, NY, USA: Routledge, 15 March 2011.

[110] M. Buhmann, "Radial Basis Function," *Acta Numerica*, vol. 9, pp. 1 - 38, January 2000.

[111] P. Garg, D. Neider, P. Madhusudan, and D. Roth, "Learning Invariants Using Decision Trees and Implication Counterexamples," *ACM Sigplan Notices*, vol. 51, no. 1, pp. 499 - 512, 11 January 2016.

[112] E. Nachmani, E. Marciano, l. Lugosch, W. J. Gross, D. Burshtein, and Y. Be'ery, "Deep Learning Methods for Improved Decoding of Linear Codes," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 119 - 131, 01 January 2018.

[113] H. Agirman-Tosun, Y. Liu, A. M. Haimovich, O. Simeone, W. Su, J. Dabin, and E. Kanterakis, "Modulation Classification of MIMO-OFDM Signals by Independent Component Analysis and Support Vector Machines," in *Proceedings of the 45th IEEE Conference on Signals, Systems and Computers (ASILOMAR)*, Pacific Grove, California, USA, pp. 1903 - 1907, 06 - 09 November 2011.

[114] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to Optimize: Training Deep Neural Networks for Wireless Resource Management," in *Proceedings of the 18th IEEE International Workshop Signal Processing Advances in Wireless Communications (SPAWC)*, Sapporo, Japan, pp. 1 - 6, 03 - 06 July 2017.

[115] A. Balatsoukas-Stimming, "Non-linear Digital Self-interference Cancellation for in-Band Full-duplex Radios Using Neural Networks," in *Proceedings of the 19th IEEE International Workshop*

on Signal Processing Advances in Wireless Communications (SPAWC), Kalamata, Greece, pp. 1 - 5, 25 - 28 June 2018.

[116] N. Strodthoff, B. Goktepe, T. Schierl, C. Hellge, and W. Samek, "Enhanced Machine Learning Techniques for Early HARQ Feedback Prediction in 5G," *IEEE Journal on Selected Areas in Communications*, vol. 37, Issue 11, pp. 2573 - 2587, November 2019.

[117] V. K. Tumuluru, P. Wang, and D. Niyalo, "A Neural Network Based Spectrum Prediction Scheme for Cognitive Radio," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, pp. 1 - 7, 23 - 27 May 2010.

[118] M. Chen, W. Saad, C. Yin, and M. Debbah, "Echo State Network for Proactive Caching in Cloud-based Radio Access Networks with Mobile Users," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3520 - 3535, June 2017.

[119] A. Zappone, M. Di Renzo, and M. Debbah, "Wireless Networks Design in the Era of Deep Learning: Model-Based, AI-Based, or Both?," *IEEE Transactions on Communications*, vol. 67, Issue 10, pp. 7331 - 7376, 05 February 2019.

[120] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "A Brief Survey of Deep Reinforcement Learning," *IEEE Signal Processing Magazine*, vol. 34, Issue 6, pp. 26 - 38, November 2017.

[121] Z. Akbari and R. Unland, "A Novel Heterogeneous Swarm Reinforcement Learning Method for Sequential Decision Making Problems," *Machine Learning and Knowledge Extraction*, vol. 1, no. 2, pp. 590 - 610, June 2019.

[122] M. van Otterlo, and M. Wiering, "Reinforcement Learning and Markov Decision Processes," in *Reinforcement Learning*, Berlin, Heidelberg, Springer, pp. 3 - 42, 2012.

[123] E. Altman, "Constrained Markov Decision Processes with Total Cost Criteria: Occupation Measures and Primal LP," *Mathematical Methods of Operations Research*, vol. 43, Issue 1, pp. 45 - 72, February 1996.

[124] E. Altman, *Constrained Markov Decision Processes.* Boca Raton, Florida, USA: CRC Press, 30 March 1999.

[125] L. I. Sennott, "Constrained Average Cost Markov Decision Chains," *Probability in the Engineering and Informational Sciences*, vol. 7, no. 1, pp. 69 - 83, January 1993.

[126] D. V. Djonin and V. Krishnamurthy, V. "Q-Learning Algorithms for Constrained Markov Decision Processes With Randomized Monotone Policies: Application to MIMO Transmission Control," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2170 - 2181, May 2007.

[127] N. Mastronarde and M. van der Schaar, "Joint Physical-layer and System-level Power Management for Delay-sensitive Wireless Communications," *IEEE Transactions on Mobile Computing*, vol. 12, no. 4, pp. 694 - 709, 07 February 2012.

[128] T. Le-Ngoc and K. T. Phan, "Joint Data Admission Control and Power Allocation Over Fading Channel Under Average Delay Constraint," in *Radio Resource Allocation Over Fading Channels Under Statistical Delay Constraints*, New York, NY, USA: Springer, pp. 29 - 41, 2017.

[129] A. Pietrabissa, "A Reinforcement Learning Approach to Call Admission and Call Dropping Control in Links with Variable Capacity," *European Journal of Control*, vol. 17, no. 1, pp. 89 - 103, 01 January 2011.

[130] S. Chobsri, W. Sumalai, and W. Usaha, "A Parametric POMDP framework for Efficient Data Acquisition in Error Prone Wireless Sensor Networks," in *Proceedings of the 4th IEEE International Symposium on Wireless Pervasive Computing*, Melbourne, Australia, pp. 1 - 5, 11 - 13 February 2009.

[131] C. Boutilier, "Planning, Learning and Coordination in Multiagent Decision Processes," in *Proceedings of the 6th Conference on Theoretical Aspects of Rationality and Knowledge*, San Francisco, California, USA, pp. 195 - 210, 17 March 1996.

[132] C. Amato, G. Chowdhary, A. Geramifard, N. K. Ure, and M. J. Kochenderfer, "Decentralized Control of Partially Observable Markov Decision Processes," in *Proceedings of the 52nd IEEE*

Conference on Decision and Control, Florence, Italy, pp. 2398 - 2405, 10 - 13 December 2013.

[133] E. Solan and B. Ziliotto, "Stochastic Games with Signals," in *Advances in Dynamic and Evolutionary Games*, F. Thuijsman, F. Wagener, Eds. Boston, MA, USA, Birkhauser, pp. 77 - 94, 2016.

[134] Y. E. Sagduyu and A. Ephremides, "Power Control and Rate Adaptation as Stochastic Games for Random Access," in *Proceedings of the 42nd IEEE Conference of Decision and Control*, Maui, Hawaii, USA, pp. 4202 - 4207, 09 - 12 December 2003.

[135] F. Afghah, A. Razi, and A. Abedi, "Stochastic Game Theoretical Model for Packet Forwarding in Relay Networks," *Telecommunication Systems*, vol. 52, no. 4, pp. 1877 - 1893, 01 April 2013.

[136] E. Altman, "Applications of Dynamic Games in Queues," in *Advances in Dynamic Games*, vol. 7, A. S. Nowak, K. Szajowski, Eds. Boston, MA, USA, Birkhauser, pp. 309 - 342, 2005.

[137] V. Francois-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An Introduction to Deep Reinforcement Learning," *Foundations and Trends in Machine Learning*, vol. 11, no. 3 - 4, pp. 219 - 354, 2018.

[138] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with Deep Reinforcement Learning," in *Proceedings of Neural Information Processing Systems (NIPS) Conference*, Lake Tahoe, Nevada, USA, 05 - 10 December 2013.

[139] J. Y. Choi and B. Lee, "Combining LSTM Network Ensemble via Adaptive Weighting for Improved Time Series Forecasting," *Mathematical Problems in Engineering, Hindawi*, vol. 2018, pp. 1 - 8, 05 August 2018.

[140] T. T. Anh, N. C. Luong, D. Niyato, Y. C. Liang, and D. I. Kim, "Deep Reinforcement Learning for Time Scheduling in RF-Powered Backscatter Cognitive Radio Networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, Marrakech, Morocco, 15 - 18 April 2019, pp. 1 - 7.

[141] R. Liu and J. Zou, "The Effects of Memory Replay in Reinforcement Learning," in *Proceedings of the 56th Annual Allerton Conference on Communication, Control, and Computing*, Allerton, Illinois, pp. 478 - 485, 02 - 05 October 2018.

[142] H. Mao, M. Alizadeh, I. Menache, and S. Kandula, "Resource Management with Deep Reinforcement Learning," in *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*, Atlanta, Georgia, USA, pp. 50 - 56, 09 - 10 November 2016.

[143] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy Gradient Methods for Reinforcement Learning with Function Approximation," in *Proceedings of the 13th Conference on Advances in Neural Information Processing Systems (NIPS)*, Denver, Colorado, USA, pp. 1057 - 1063, 2000.

[144] R. Li, Z. Zhao, Q. Sun, I. Chih-Lin, C. Yang, X. Chen, M. Zhao, and H. Zhang, "Deep Reinforcement Learning for Resource Management in Network Slicing," *IEEE Access*, vol. 6, pp. 74429 - 74441, 19 November 2018.

[145] F. Aoudia and J. Hoydis. "End-to-End Learning of Communications Systems Without a Channel Model," in *Proceedings of the 52nd IEEE Conference on Signals, Systems, and Computers (ASILOMAR)*, Pacific Grove, California, USA, pp. 298 - 303, 28 - 31 October 2018.

[146] S. Lee, S. Park, G. Noh, Y. Park, and D. Hong, "Energy-efficient Spectrum Access for Ultra Low Power Sensor Networks," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, Orlando, Florida, pp. 1 - 6, 29 October - 01 November 2012.

[147] Y. Bengio, "Learning Deep Architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1 - 127, 15 November 2009.

[148] J. Ma, M. K. Yu, S. Fong, K. Ono, E. Sage, B. Demchak, R. Sharan, and T. Ideker, "Using Deep Learning to Model the Hierarchical Structure and Function of a Cell," *Nature Methods*, vol. 15, no. 4, pp. 290 - 298, April 2018.

[149] R. C. Gonzalez, "Deep Convolutional Neural Networks," *IEEE Signal Processing Magazine*, vol. 35, no. 6, pp. 79 - 87, November 2018.

[150] H. Karimi, J. Nutini, and M. Schmidt, "Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Lojasiewicz Condition," in *Machine Learning and Knowledge Discovery in Databases*, ECML PKDD 2016. Lecture Notes in Computer Science, vol. 9851, Springer, Cham.

[151] T. Parr and J. Howard, "The Matrix Calculus You Need For Deep Learning," 05 February 2018. Accessed April 25, 2018, [Online]. Available: https://explained.ai/matrix-calculus/

[152] L. Palagi, A. Pesyridis, E. Sciubba, and L. Tocci, "Machine Learning for the Prediction of the Dynamic Behaviour of a Small Scale ORC System," *Energy*, vol. 166, pp. 72 - 82, 01 January 2019.

[153] M. M. Mia, S. K. Biswas, M. C. Urmi, and A. Siddique, "An Algorithm for Training Multilayer Perceptron (MLP) for Image Reconstruction using Neural Network without Overfitting," *International Journal of Scientific & Technology Research*, vol. 4, no. 2, pp. 271 - 275, February 2015.

[154] A. Sarkar and J. K. Mandal, "Key Generation and Certification using Multilayer Perceptron in Wireless Communication," *International Journal of Security, Privacy and Trust Management (IJSPTM)*, vol. 1, no 5, pp. 27 - 43, October 2012.

[155] K. P. Bagadi and S. Das, "Multiuser Detection in SDMA-OFDM Wireless Communication System using Complex Multilayer Perceptron Neural Network," *Wireless Personal Communications*, vol. 77, no. 1, pp. 21 - 39, 01 July 2014.

[156] L. S. Ezema and C. I. Ani, "Artificial Neural Network Approach to Mobile Location Estimation in GSM Network," *International Journal of Electronics and Telecommunications*, vol. 63, no. 1, pp. 39 - 44, 01 March 2017.

[157] I. U. Rehman, M. M. Nasralla, and N. Y. Philip, "Multilayer Perceptron Neural Network-based

QoS-aware, Content-aware and Device-aware QoE Prediction Model: A Proposed Prediction Model for Medical Ultrasound Streaming over Small Cell Networks," *Electronics*, vol. 8, no. 2, pp. 1 - 27, February 2019.

[158] A. Oppermann, "Deep Learning meets Physics: Restricted Boltzmann Machines Part I: Theory behind Restricted Boltzmann Machines, A powerful Tool for Recomender Systems," Towards Data Science, 27 April 2018. Accessed May 16, 2018, [Online]. Available: https://towardsdatascience.com/deep-learning-meets-physics-restricted-boltzmann-machines-part-i-6df5c4918c15.html

[159] M. C. Hlophe and B. T. Maharaj, "Spectrum Occupancy Reconstruction in Distributed Cognitive Radio Networks Using Deep Learning," *IEEE Access*, vol. 7, pp. 14294 - 14307, 23 January 2019.

[160] T. Kuremoto, S. Kimura, K. Kobayashi, and M. Obayashi, "Time Series Forecasting Using Restricted Boltzmann Machine," in *Proceedings of the 8th International Conference on Intelligent Computing (ICIC)*, Huangshan, China, pp. 17 - 22, 25 - 29 July 2012.

[161] S. Goudarzi, M. Kama, M. Anisi, S. Soleymani, and F. Doctor, "Self-organizing Traffic Flow Prediction with an Optimized Deep Belief Network for Internet of Vehicles," *Sensors*, vol. 18, no. 10, pp. 1 - 18, October 2018.

[162] Y. H. Lee, D. J. Moss, J. Faraone, P. Blackmore, D. Salmond, D. Boland, and P. H. Leong, "Long Short-Term Memory for Radio Frequency Spectral Prediction and its Real-Time FPGA Implementation," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, Los Angeles, California, pp. 1 - 9, 29 - 31 October 2018.

[163] H. Sak, A. Senior, and F. Beaufays, "Long Short-term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling," in *Proceedings of the 15th Annual Conference of the International Speech Communication Association*, Singapore, 14 - 18 September 2014.

[164] J. Brownlee, "Gentle Introduction to the Adam Optimization Algorithm for Deep Learning," Machine Learning Mastery, 03 July 2017. Accessed July 17, 2017, [Online]. Available:

https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning.html

[165] L. Ale, N. Zhang, H. Wu, D. Chen, and T. Han, "Online Proactive Caching in Mobile Edge Computing Using Bidirectional Deep Recurrent Neural Network," *IEEE Internet of Things Journal*, vol. 6, Issue 3, pp. 5520 - 5530, June 2019.

[166] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep Learning with Long Short-Term Memory for Time Series Prediction," *IEEE Communications Magazine*, vol. 57, Issue 6, pp. 114 - 119, June 2019.

[167] T. Dlamini, A. F. Gambin, D. Munaretto, and M. Rossi, "Online Supervisory Control and Resource Management for Energy Harvesting BS Sites Empowered with Computation Capabilities," *Wireless Communications and Mobile Computing, Hindawi*, Volume 2019, Article ID 8593808, pp. 1 - 17, 2019.

[168] R. Abdulhammed, H. Musafer, A. Alessa, M. Faezipour, and A. Abuzneid, "Features Dimensionality Reduction Approaches for Machine Learning Based Network Intrusion Detection," *Electronics*, vol. 8, no. 3, pp. 1 - 27, 14 March 2019.

[169] D. S. Berman, A. L. BuczakL, J. S. Chavis, and C. L. Corbett, "A Survey of Deep Learning Methods for Cyber Security," *Information*, vol. 10, no. 4, pp. 1 - 35, 02 April 2019.

[170] H. He, S. Jin, C.-K. Wen, F. Gao, G. Y. Li, and Z. Xu, "Model-Driven Deep Learning for Physical Layer Communications," *IEEE Wireless Communications*, vol. 26, no. 5, pp. 77 - 83, 15 May 2019.

[171] A. Felix, S. Cammerer, S. Dorner, J. Hoydis, and S. T. Brink, "OFDM Autoencoder for End-to-end Learning of Communications Systems," in *Proceedings of the 19th IEEE International Workshop Signal Processing Advances in Wireless Communications (SPAWC)*, Kalamata, Greece, pp. 1 - 5, 25 - 28 June 2018.

[172] H. Ye, G. Y. Li, B.-H. F. Juang, and K. Sivanesan, "Channel Agnostic End-to-end Learning Based Communication Systems with Conditional GAN," in *Proceedings in IEEE Global*

Communication Workshops (GC Wkshps), Abu Dhabi, UAE, pp. 1 - 5, 09 - 13 December 2018.

[173] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and Composing Robust Features with Denoising Autoencoders," in *Proceedings of the 25th ACM International Conference on Machine Learning (ICML)*, Helsinki, Finland, pp. 1096 - 1103, 05 - 09 July 2008.

[174] K. I. Ahmed, H. Tabassum, and E. Hossain, "Deep Learning for Radio Resource Allocation in Multi-cell Networks," *IEEE Network*, vol. 33, no. 6, pp. 188 - 195, 22 April 2019.

[175] M. Zhang, L. Wang, Y. Feng, and H. Yin, "A Spectrum Sensing Algorithm for OFDM Signal Based on Deep Learning and Covariance Matrix Graph," *IEICE Transactions on Communications*, vol. E101.B, pp. 2435 - 2444, May 2018.

[176] G. Pan, J. Li, and F. Lin, "A Cognitive Radio Spectrum Sensing Method for an OFDM Signal Based on Deep Learning and Cycle Spectrum," *International Journal of Digital Multimedia Broadcasting*, vol. 2020, pp. 1 - 10, 06 March 2020.

[177] K. Yang, Z. Huang, X. Wang, and X. Li, "A Blind Spectrum Sensing Method Based on Deep Learning," *Sensors*, vol. 19, no. 2270, pp. 1 - 17, 16 May 2019.

[178] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep Learning for Computer Vision: A Brief Review," *Computational Intelligence and Neuroscience*, Volume 2018, Article ID 7068349, pp. 1 - 13, 01 February 2018.

[179] D. Wang, B. Bai, W. Zhao, and Z. Han, "A Survey of Optimization Approaches for Wireless Physical Layer Security," *IEEE Communications Surveys & Tutorials*, vol. 21, Issue 2, pp. 1878 - 1911, Secondquarter 2019.

[180] F. Alberge, "Deep Learning Constellation Design for the AWGN Channel With Additive Radar Interference," *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1413 - 1423, February 2019.

[181] D. M. Nguyen, R. Calderbank, and N. Deligiannis, "Geometric Matrix Completion with Deep Conditional Random Fields," *arXiv preprint arXiv:1901.10429*, 29 January 2019.

[182] M. Mozaffari, W. Saad, M. Bennis, Y. H. Nam, and M. Debbah, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *IEEE Communications Surveys & Tutorials*, vol. 21, Issue 3, pp. 2334 - 2360, Thirdquarter 2019.

[183] E. Munch, "A User's Guide to Topological Data Analysis," *Journal of Learning Analytics*, vol. 4, no. 2, pp. 47 - 61, 2017.

[184] S. Dara and P. Tumma, "Feature Extraction By Using Deep Learning: A Survey," in *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, Tamilnadu, India, pp. 1795 - 1801, 29 - 31 March 2018.

[185] V. Thakur and H. Sikarwar, "Deep Learning Feature Extraction for Handwritten Keyword Spotting in Historical Documents," in *Proceedings of the 2nd International Conference on Emerging Trends in Engineering & Applied Science (ICETEAS'19)*, Jaipur, Rajasthan, India, pp. 11 - 15, 15 - 16 January 2019.

[186] H. B. Ahmad, "Ensemble Classifier Based Spectrum Sensing in Cognitive Radio Networks," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 9250562, pp. 1 - 16, 01 January 2019.

[187] M. A. Alsheikh, D. Niyato, S. Lin, H. P. Tan, and Z. Han, "Mobile Big Data Analytics using Deep Learning and Apache Spark," *IEEE Network*, vol. 30, no. 3, pp. 22 - 29, May 2016.

[188] N. V. Chawla, G. Karakoulas, "Learning from Labeled and Unlabeled Data: An Empirical Study Across Techniques and Domains," *Journal of Artificial Intelligence Research*, vol. 23, pp. 331 - 366, 01 March 2005.

[189] M. Grimaldi, V. Tenace, and Calimera, "Layer-Wise Compressive Training for Convolutional Neural Networks," *Future Internet*, vol. 11, no. 1, pp. 7 - 21, January 2019.

[190] A. E. Eshratifar, and M. Pedram, "Energy and Performance Efficient Computation Offloading for Deep Neural Networks in a Mobile Cloud Computing Environment," in *Proceedings of the 28th ACM Great Lakes Symposium on VLSI (GLVLSI)*, Chicago, Illinois, USA, pp. 111 - 116, 23 - 25 May 2018.

[191] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big Data-driven Optimization for Mobile Networks Toward 5G," *IEEE Network*, vol. 30, no. 1, pp. 44 - 51, January 2016.

[192] Y. Huang, J. Tan, and Y. C. Liang, "Wireless Big Data: Transforming Heterogeneous Networks to Smart Networks," *Journal of Communications and Information Networks*, vol. 2, no. 1, pp. 19 - 32, 01 March 2017.

[193] A. Zaeemzadeh, M. Joneidi, B. Shahrasbi, and N. Rahnavard, "Missing Spectrum-Data Recovery in Cognitive Radio Networks Using Piecewise Constant Non-negative Matrix Factorization," in *Proceedings of the IEEE Military Communications Conference (MILCOM)*, Tampa, Florida, USA, 26 - 28 August 2015, pp. 1 - 6.

[194] J. Ma, G. Zhao, Y. Li, "Soft Combination and Detection for Cooperative Spectrum Sensing in Cognitive Radio Networks," *IEEE Transactions in Wireless Communications*, vol. 7, no. 11, pp. 4502 - 4507, 02 December 2008.

[195] J. W. Yufan "Cooperative Spectrum Sensing for Cognitive Radio Networks under Nakagami-M Fading Channels," in *Proceedings of the IEEE International Conference on Industrial Control and Electronics Engineering (ICICEE)*, Xi'an, China, pp. 1770 - 1772, 23 - 25 August 2012.

[196] S. Maleki, A. Pandharipande, and G. Leus, "Energy-efficient Distributed Spectrum Sensing for Cognitive Sensor Networks," *IEEE Journal on Sensor Networks*, vol. 11, Issue 3, pp. 565 - 573, March 2011.

[197] J. Yao, J. Cao, Q. Zheng, J. Ma, "Pre-processing of Incomplete Spectrum Sensing Data in Spectrum Sensing Data Falsification Attacks Detection: A Missing Data Imputation Approach," *IET Communications*, vol. 10, no. 11, pp. 1340 - 1347, 21 July 2016.

[198] J. Gillard and K. Usevich, "Structured Low-rank Matrix Completion for Forecasting in Time Series Analysis," *International Journal of Forecasting*, vol. 34, Issue 4, pp. 582 - 597, 01 October 2018.

[199] S. C. Shinde and A. N. Jadhav, "Performance Comparison of Centralized Cooperative Spectrum Sensing Based on Voting Rules in Cognitive Radio," *International Journal of Advanced Engineering Research and Science*, vol. 3, no. 5, pp. 72 - 76, May 2016.

[200] J. Meng, W. Yin, H. Li, E. Houssain and Z. Han, "Collaborative Spectrum Sensing from Sparce Observations Using Matrix Completion for Cognitive Radio Networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, Issue 2, pp. 327 - 337, 24 January 2011.

[201] E. J. Candes and Y. Plan, "Matrix Completion with Noise," in *Proceedings of the IEEE*, vol. 98, Issue 6, pp. 925 - 936, 26 April 2010.

[202] A. Ghasemi, E. Sousa, "Collaborative Spectrum Sensing for Opportunistic Access in Fading Environments," in *Proceedings of the IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, (DYSPAN)*, Baltimore, Maryland, USA, pp. 131 - 136, 08 - 11 November 2005.

[203] Z. Tian, "Compressed Wideband Sensing in Cooperative Cognitive Radio Networks," in *Proceedings of IEEE Global Communication Conference (GLOBECOM)*, New Orleans, Louisiana, USA, pp. 1 - 5, 30 November - 04 December 2008.

[204] D. Xue, E. Ekici, and M. C. Vuran, "Cooperative Spectrum Sensing in Cognitive Radio Networks Using Multidimensional Correlations," *IEEE Transactions on Wireless Communications*, vol. 13, no. 4, pp. 1832 - 1843, 03 March 2014.

[205] N. Ezzati and H. Taheri, "Distributed Spectrum Sensing Using Radio Environment Maps in Cognitive Radio Networks," *Wireless Personal Communications*, vol. 101, Issue 4, pp. 2241 - 2254, 01 August 2018.

[206] Z. Song, H. Jijun, L. Peiguo, and H. Jianguo, "Distributed Compressed Spectrum Sensing via

Cooperative Support Fusion," *International Journal of Distributed Sensor Networks*, vol. 9, no. 12, 21 December 2013.

[207] J. Riihijarvi, P. Mahonen, M. Wellens, and M. Gordziel, "Characterization and Modelling of Spectrum for Dynamic Spectrum Access with Spatial Statistics and Random Fields," in *Proceedings of the 19th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Cannes, France, pp. 1 - 6, 15 - 18 September 2008.

[208] M. Wellens, J. Riihijarvi, M. Gordziel, and P. Mahonen, "Spatial Statistics of Spectrum Usage: From Measurements to Spectrum Models," in *Proceedings of IEEE International Conference on Communications (ICC)*, Dresden, Germany, pp. 1 - 6, 14 - 18 June 2009.

[209] H. Li, Z. Han, and Z. Zhang, "Communication over Random Fields: A Statistical Framework for Cognitive Radio Networks," in *Proceedings of the IEEE Global Communication Conference (GLOBECOM)*, Houston, Texas, USA, pp. 1 - 5, 05 - 09 December 2011.

[210] P. Potier, C. Sorrells, Y. Wang, L. Qian, and H. Li, "Network-wide Spectrum Situation Reconstruction using Total Variation Inpainting in Cognitive Radio Ad-hoc Network," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Houston, Texas, USA, pp. 1 - 5, 05 - 09 December 2011.

[211] D. Willkomm, S. Machiraju, J. Bolot, and A. Wolisz, "Primary Users in Cellular Networks: A Large-Scale Measurement Study," in *Proceedings of the 3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Chicago, Illinois, USA, pp. 1 - 11, 14 - 17 October 2008.

[212] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, p. 436, May 2015.

[213] Y. Cui, X. jun Jing, S. Sun, X. Wang, D. Cheng, and H. Huang, "Deep Learning Based Primary User Classification in Cognitive Radios," in *Proceedings of the 15th IEEE International Symposium on Communications and Information Technologies (ISCIT)*, Nara, Japan, pp. 165 - 168, 07 - 09 October 2015.

[214] G. J. Mendis, J. Wei, and A. Madanayake, "Deep Learning-based Automated Modulation Classification for Cognitive Radio," in *Proceedings of the IEEE International Conference on Communication Systems (ICCS)*, Shenzhen, China, pp. 1 - 6, 14 - 16 December 2016.

[215] F. Paisana, A. Selim, M. Kist, P. Alvarez, J. Tallon, C. Bluemm, A. Puschmann, and L. DaSilva, "Context-aware Cognitive Radio Using Deep Learning," in *Proceedings of the IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, Baltimore, Maryland, USA, pp. 1 - 2, 06 - 09 March 2017.

[216] E. Candes and B. Recht, "Exact Matrix Completion via Convex Optimization," *Foundations of Computational Mathematics*, vol. 9, pp. 717 - 772, May 2008.

[217] R. Keshavan, S. Oh, and A. Montanari, "Matrix Completion from a Few Entries," *IEEE Transactions on Information Theory*, vol. 56, no. 6, pp. 2980 - 2998, June 2010.

[218] N. Agarwal, Z. Allen-Zhu, B. Bullins, E. Hazan, and T. Ma, "Finding Approximate Local Minima Faster Than Gradient Descent," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, Montreal, Canada, pp. 1195 - 1199, 19 - 23 June 2017.

[219] M. Hardt, B. Recht, and Y. Singer, "Train Faster, Generalize Better: Stability of Stochastic Gradient Descent," in *Proceedings of the 33rd International Conference on Machine Learning*, New York, USA, 19 - 24 June 2016.

[220] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," in *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, San Diego, California, USA, pp. 192 - 204, 09 - 12 May 2015.

[221] J. Rochner, L. Balents, K. P. Schmidt, "Spin Liquid and Quantum Phase Transition Without Symmetry Breaking in a Frustrated Three-dimensional Ising Model," *Physical Review B.*, vol. 94, no. 20, pp. 201111- 1 - 201111-4, 17 November 2016.

[222] A. Hamlili, "Intelligibility of Erdos-Renyi Random Graphs and Time Varying Social Network

Modeling," in *Proceedings of the ACM International Conference on Smart Digital Environment*, Rabat, Morocco, pp. 201 - 206, 21 - 24 July 2017.

[223] H. Chen, H. Zhang, and C. Shen, "Double Phase Transition of the Ising Model in Core-Periphery Networks," *Journal of Statistical Mechanics Theory and Experiment*, vol. 2018, no. 6, p. 063402, 07 June 2018.

[224] R. W. Krause, M. Huisman, C. Steglich, and T. A. Sniiders, "Missing Network Data: A Comparison of Different Imputation Methods," in *Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Barcelona, Spain, pp. 159 - 163, 28 - 31 August 2018.

[225] B. Andrew and A. Selamat, "Systematic Literature Review of Missing Data Imputation Techniques for Effort Prediction," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, Maui, Hawaii, USA, pp. 222 - 226, 29 October - 02 November 2012.

[226] S. Nie, M. Zheng, and Q. Jing, "The Deep Regression Bayesian Network and Its Applications: Probabilistic Deep Learning for Computer Vision," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 101 - 111, January 2018.

[227] T. Hazan, S. Maji, and T. Jaakkola, "On Sampling from the Gibbs Distribution with Random Maximum a-Posteriori Perturbations," in *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, Lake Tahoe, Nevada, USA, pp. 1268 - 1276, 05 - 09 December 2013.

[228] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings Algorithm," *The American Statistician*, vol. 49, no. 4, pp. 327 - 335, 01 November 1995.

[229] I. Yildirim, "Bayesian inference: Metropolis-Hastings sampling," Department of Brain and Cognitive Sciences, University of Rochester, Rochester, New York, NY, USA, August 2012.

[230] A. Ma, D. Needell, "Adapted Stochastic Gradient Descent for Linear Systems with Missing Data," *Numerical Mathematics Theory Methods and Applications*, vol. 12, no. 1, pp. 1 - 20, 23

February 2017.

[231] S. Tombaz, A. Vastberg, and J. Zander, "Energy- and Cost-efficient Ultra High Capacity Wireless Access," *IEEE Wireless Communication Magazine*, vol. 18, no. 5, pp. 18 - 24, October 2011.

[232] L. Sanchez, O. Blume, M. Gonzalez, G. Biczok, D. Ferling, and I. Godor, "EARTH: Paving the Way for Future Energy Efficient Broadband Wireless Networks." in *Proceedings of the international Conference on Mobile Networks and Management*, Santander, Spain, pp. 326 - 339, 22 - 24 September 2010.

[233] R. Deng, J. Chen, C. Yuen, P. Cheng, and Y. Sun, "Energy-Efficient Cooperative Spectrum Sensing by Optimal Scheduling in Sensor-Aided Cognitive Radio Networks," *IEEE Transactions in Vehicular Technology*, vol. 61, no. 2, pp. 716 - 725, 01 February 2012.

[234] G. Vallero, M. Deruyck, M. Meo, and W. Joseph, "Accounting for Energy Cost When Designing Energy-Efficient Wireless Access Networks," *Energies*, vol. 11, no. 3, pp. 1 - 21, 09 March 2018.

[235] Z. Liu, B. Liu, C. Chen, and C. W. Chen, "Energy-efficient Resource Allocation with QoS Support in Wireless Body Area Networks," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, San Diego, California, USA, pp. 1 - 6, 06 - 10 December 2015.

[236] S. Ahmed, Y. D. Lee, S. H. Hyun, and I. Koo, "A Cognitive Radio-Based Energy-Efficient System for Power Transmission Line Monitoring in Smart Grids," *Journal of Sensors*, Volume 2017, Article ID 386275, pp. 1 - 12, 2017.

[237] P. Gandotra, R. K. Jha, and S. Jain, "Green Communication in Next Generation Cellular Networks: A Survey," *IEEE Access*, vol. 5, pp. 11727 - 11758, 06 June 2017.

[238] M. K. Hanawal, F. Malik, and Y. Hayel, "Differential Pricing of Traffic in the Internet," in *Proceedings of the 16th IEEE International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, Shanghai, China, pp. 1 - 8, 07 - 11 May 2018.

[239] M. S. Mushtaq, S. Fowler, and A. Mellouk, "Power Saving Model for Mobile Device and Virtual Base Station in the 5G Era," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Paris, France, pp. 1 - 6, 21 - 25 May 2017.

[240] E. Agrell, "Conditions for a Monotonic Channel Capacity," *IEEE Transactions on Communications*, vol. 63, no. 3, pp. 738 - 748, 18 December 2014.

[241] S. Kim, "Game Theory for Wireless Ad Hoc Networks" in *Game Theory: Breakthroughs in Research and Practice*, Hershey, Pennsylvania, USA, IGI Global, pp. 353 - 368, 2018.

[242] B. Nleya and A. Mutsvangwa, "Enhanced Congestion Management for Minimizing Network Performance Degradation in OBS Networks," *SAIEE Africa Research Journal*, vol. 109, no. 1, pp. 48 - 57, March 2018.

[243] E. Bjornson, E. A. Jorswieck, M. Debbah, and B. Ottersten, "Multiobjective Signal Processing Optimization: The Way to Balance Conflicting Metrics in 5G Systems," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 14 - 23, November 2014.

[244] U. Muhammad, S. K. Muhammad, V. Hiep, and I. Koo, "Energy-efficient Channel Hand-off for Sensor Network-Assisted Cognitive Radio Network," *Sensors*, vol. 15, no. 8, pp. 18012 - 18039, August 2015.

[245] Y. Zhao, S. Jin, and W. Yue, "A Novel Spectrum Access Strategy with α-Retry Policy in Cognitive Radio Networks: A Queuing-based Analysis," *Journal of Communications and Networks*, vol. 16, no. 2, pp. 193 - 201, 08 May 2014.

[246] R. Doost, M. Y. Naderi, and K. R. Chowdhury, "Spectrum Allocation and QoS Provisioning Framework for Cognitive Radio with Heterogeneous Service Classes," *IEEE Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3938 - 3950, July 2014.

[247] B. Awoyemi, B. Maharaj, and A. Alfa, "Optimal Resource Allocation Solutions for Heterogeneous Cognitive Radio Networks," *Digital Communications and Networks*, vol. 3, no. 2, pp. 129 - 139, 01 May 2017.

[248] M. J. Kaur, M. Uddin, and H. K. Verma, "Optimization of QoS Parameters in Cognitive Radio Using Adaptive Genetic Algorithm," *International Journal of Next Generation Networks (IJNGN)*, vol. 4, no. 2, pp. 1 - 15, June 2012.

[249] L. Zhai, H. Wang, and C. Gao, "A Spectrum Access Based on Quality-of-Service (QoS) in Cognitive Radio Networks," *PLoS one*, vol. 11, no. 5, e0155074, 12 May 2016.

[250] M. Alsharif, J. Kim, and J. Kim, "Green and Sustainable Cellular Base Stations: An Overview and Future Research Directions," *Energies*, vol. 10, no. 5, pp. 1 - 27, 25 April 2017.

[251] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. A. Imran, D. Sabella, M. J. Gonzalez, O. Blume, and A. Fehske, "How Much Energy is Needed to Run a Wireless Network?," *IEEE Wireless Communications Magazine*, vol. 18, no. 5, pp. 40 - 49, October 2011.

[252] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big Data-driven Optimization for Mobile Networks Toward 5G," *IEEE Network*, vol. 30, no. 1, pp. 44 - 51, 25 January 2016.

[253] M. W. Kang and Y. W. Chung, "An Efficient Energy Saving Scheme for Base Stations in 5G Networks with Separated Data and Control Planes Using Particle Swarm Optimization," *Energies*, vol. 10, pp. 1 - 28, 15 September 2017.

[254] D. Renga, H. A. H. Hassan, M. Meo, and L. Nuaymi, "Energy Management and Base Station On/Off Switching in Green Mobile Networks for Offering Ancilliary Services," *IEEE Transactions on Green Communications and Networks*, vol. 2, Issue 3, pp. 868 - 880, September 2018.

[255] C. K. Das, O. Bass, G. Kothapalli, T. S. Mahmoud, and D. Habibi, "Overview of Energy Storage Systems in Distribution Networks: Placement, Sizing, Operation, and Power Quality," *Renewable and Sustainable Energy Reviews*, vol. 91, pp. 1205 - 1230, 01 August 2018.

[256] L. Sboui, H. Ghazzai, Z. Rekzi, and M.-S. Alouini, "On Green Cognitive Radio Cellular Networks: Dynamic Spectrum and Operation Management," *Special Section on Green Commu-*

nications and Networking for 5G Wireless Networks, IEEE Access, vol. 4, pp. 4046 - 4057, 18 July 2016.

[257] J. Yang, B. Payne, M. Hitz, Z. Fei, L. Li, and T. Wei, "Location Aided Energy balancing Strategy in Green Cellular Networks," in *Proceedings of the 23rd IEEE International Conference on Computer Communication and Networks (ICCCN)*, Shanghai, China, pp. 1 - 6, 04 - 07 August 2014.

[258] G. Wang, C. Guo, S. Wang, and C. Feng, "A Traffic Prediction Based Sleeping Mechanism with Low Complexity in Femtocell Network," in *Proceedings of IEEE International Communication Conference (ICC)*, Budapest, Hungary, pp. 560 - 565, 09 - 13 June 2013.

[259] X. Tan, H. Huang, and L. Ma, "Frequency Allocation with Artificial Neural Networks in Cognitive Radio System," in *Proceedings of the IEEE TENCON Spring Conference*, Sydney, Australia, pp. 366 - 370, 17 - 19 April 2013.

[260] Y. H. Wang and S. L. Liao, "Applying a Fuzzy-Based Dynamic Channel Allocation Mechanism to Cognitive Radio Networks," in *Proceedings of the 31st IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Taipei, Taiwan, pp. 564 - 569, 27 - 29 March 2017.

[261] Y. El Morabit, F. Mrabti, and E. H. Abarkan, "Spectrum Allocation using Genetic Algorithm in Cognitive Radio Networks," in *Proceedings of the 3rd IEEE International Workshop on RFID And Adaptive Wireless Sensor Networks (RAWSN)*, Agadir, Morocco, pp. 90 - 93, 13 - 15 May 2015.

[262] B. Liu, J. Cheng, Q. Liu, X. Tang, "A Long Short-term Traffic Flow Prediction Method Optimized by Cluster Computing," Preprints 2018, 2018080163, DOI: 10.20944/preprints201808.0163.v1

[263] T. Zhao, J. Wu, S. Zhou, and Z. Niu, "Energy-Delay Trade-offs of Virtual Base Stations with A Computational Resource-Aware Energy Consumption Model," in *Proceedings of the IEEE*

Conference on Communication Systems (ICCS), Macau, China, pp. 26 - 30, 19 - 21 November 2014.

[264] P. S. Aravind, J. Shah, and D. G. Kurup, "Bit Error Rate (BER) Performance Analysis of DASH7 Protocol in Rayleigh Fading Channel," in *Proceedings of the 7th IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Bangalore, India, pp. 695 - 698, 19 - 22 September 2018.

[265] S. Ping, A. Aijaz, O. Holland, and A. H. Aghvami, "Green Cellular Access Network Operation Through Dynamic Spectrum and Traffic Load Management," in *Proceeding of IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, London, UK, pp. 2791 - 2796, 08 - 11 September 2013.

[266] M. C. Hlophe and B. T. maharaj, "QoS provisioning and Energy Saving Scheme for Distributed Cognitive Radio Networks using Deep Learning," *Journal of Communications and Networks*, vol. 22, no. 3, pp. 185 - 204, June 2020.

[267] M. Leconte, M. Lelarge, and L. Massoulie, "Bipartite Graph Structures for Efficient Balancing of Heterogeneous Loads," *ACM Sigmetrics Performance Evaluation Review*, vol. 40, no. 1, pp. 41 - 52, 07 June 2012.

[268] K. Hashimoto, S. Adachi, and D. V. Dimarogonas, "Self-triggered Model Predictive Control for Nonlinear Input-affine Dynamical Systems via Adaptive Control Samples Selection," *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 177 - 189, January 2017.

[269] H. Chen and F. Allgower, "A Quasi-infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205 - 1217, 01 October 1998.

[270] JORG Database. Accessed September 10, 2018. [Online]. Available: https://www.comm.utoronto.ca/ jorg/teaching/ece466/labs/lab1/?C=N;O=D.

[271] T. H. Nguyen, M. Forshaw, and N. Thomas, "Operating Policies for Energy Efficient Dynamic

Server Allocation," *Electronic Notes in Theoretical Computer Science*, vol. 318, pp. 159 - 177, 25 November 2015.

[272] W. C. Chan, T. C. Lu, and R. J. Chen, "Pollaczek-Khinchin Formula for the M/G/1 Queue in Discrete Time with Vacations," *IEEE Proceedings in Computers and Digital Techniques*, vol. 144, no. 4, pp. 222 - 226, July 1997.

[273] M. Draoli, C. Gaibisso, M. Lancia, and E. A. Mastromartino, "Satisfying High Quality Requirements of Videoconferencing on a Packet Switched Network," in *Proceedings of INET*, Kuala Lumpur, Malaysia, 25 - 27 June 1997.

[274] J. Dilley and R. L. Armstrong, "How Much Speed You Need for Online Gaming," Accessed September 18, 2018, [Online]. Available: https://www.highspeedinternet.com/resources/how-much-speed-do-i-need-for-online-gaming/.

[275] M. Hosseini and V. Swaminathan, "Adaptive 360 VR Video Streaming: Divide and Conquer," in *Proceedings of the IEEE International Symposium Multimedia (ISM)*, San Jose, California, USA, pp. 107 - 110, 11 - 13 December 2016.

[276] C. She and C. Yang, "Energy Efficient Predictive Resource Allocation for VoD and Real-time Services," *IEEE Transactions on Wireless Communications*, vol. 4, No. 1, 06 July 2017.

[277] T. P. Oliveira, J. S. Barbar, and A. S. Soares, "Multilayer Perceptron and Stacked Autoencoder for Internet Traffic Prediction," in *Proceedings of the IFIP International Conference on Network and Parallel Computing*, Ilan, Taiwan, pp. 61 - 71, 18 - 20 September 2014.

[278] T. Braud, F. H. Bijarbooneh, D. Chatzopoulos, and P. Hui, Future Networking Challenges: The Case of Mobile Augmented Reality," in *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, Atlanta, Georgia, USA, pp. 1 - 12, 05 - 08 June 2017.

[279] N. Khan and M. G. Martini, "QoE-driven Multi-user Scheduling and Rate Adaptation with Reduced Cross-layer Signaling for Scalable Video Streaming over LTE Wireless Systems,"

EURASIP Journal on Wireless Communications and Networking, vol. 2016, no. 93, pp. 1 - 23, 01 April 2016.

[280] N. Javaid, A. Sher, H. Nasir, and N. Guizani, "Intelligence in IoT-Based 5G Networks: Opportunities and Challenges," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 94 - 100, October 2018.

[281] R. Hassan, R. Razali, S. Mohseni, O. Mohamad, and Z. Ismail, "Architecture of Network Management Tools for Heterogeneous System," *International Journal of Computer Science and Information Security (IJCSIS),* vol. 6, no. 3, pp. 31 - 40, December 2009.

[282] M. M. Marwangi, N. Fisal, S. K. Yusof, R. A. Rashid, A. S. Ghafar, F. A. Saparudin, and N. Katiran, "Challenges and Practical Implementation of Self-organizing Networks in LTE/LTE-Advanced Systems," in *Proceedings of the 5th IEEE International Conference on Information Technology & Multimedia (ICIMU)*, Kuala Lumpur, Malaysia, pp. 1 - 5, 14 - 16 November 2011.

[283] K. U. R. Laghari, N. Crespi, and K Connelly, "Towards Total Quality of Experience: A QoE for Multimedia Services in Communication Ecosystem," *IEEE Communication Magazine*, vol. 50, no. 4, pp. 58 - 65, April 2012.

[284] P. Du, "Adaptive Communication Techniques for the Internet of Things," *Journal of Sensor and Actuator Networks*, vol. 1, no. 2, pp. 122 - 155, March 2013.

[285] B. Chihani, E. Bertin, D. Collange, N. Crespi, and T. H. Falk, "User-centric Quality of Experience Measurement," in *Proceedings of the 5th International Conference on Mobile Computing, Applications, and Services*, Paris, France, pp. 33 - 46, 07 - 08 November 2013.

[286] Z. He, S. Mao, and S. Kompella, "QoE Driven Video Streaming in Cognitive Radio Networks: The Case of Single Channel Access," in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Austin, Texas, USA, pp. 1388 - 1393, 08 - 12 December 2014.

[287] J. Dai and S. Wang "QoE-driven Resource Allocation Method for Cognitive Radio Networks," in *Proceedings of the IEEE International Conference on Communications (ICC)*, Kuala Lumpur,

Malaysia, pp. 1 - 6, 23 - 27 May 2016.

[288] W. Yang, and X. Zhao, "Resource Allocation in Two-way OFDM-based Cognitive Radio Networks with QoE and Power Consumption Guarantees," *EURASIP Journal on Wireless Communications and Networking*, vol. 2017, no. 1, pp. 216 - 229, 01 December 2017.

[289] S. Thakolsri, W. Kellerer, S. Khan, and E. Steinbach, "QoE-driven Cross-layer Optimization for High Speed Downlink Packet Access," *Journal of Communications*, vol. 4, no. 9, pp. 669 - 680, October 2009.

[290] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A Generic Quantitative Relationship Between Quality of Experience and Quality of Service," *IEEE Network*, vol. 24, Issue 2, pp. 36 - 41, March - April 2010.

[291] L. Giupponi, A. Galindo-Serrano, P. Blasco, and M. Dohler, "Docitive Networks: An Emerging Paradigm for Dynamic Spectrum Management," *IEEE Wireless Communications*, vol. 17, Issue 4, pp. 47 - 54, August 2010.

[292] O. Berger-Tal, J. Nathan, E. Meron, and D. Saltz, "The Exploration-Exploitation Dilemma: A Multidisciplinary Framework," *PLoS one*, vol. 9, no. 4: pp. e95693, 22 April 2014.

[293] M. C. Machado, M. G. Bellemare, E. Talvitie, J. Veness, M. Hausknecht, and M. Bowling, "Revisiting the Arcade Learning Environment: Evaluation Protocols and Open Problems for General Agents," *Journal of Artificial Intelligence Research*, vol. 61, pp. 523 - 562, 19 March 2018.

[294] A. Galindo-Serrano, L. Giupponi, P. Blasco, and M. Dohler, "Learning from Experts in Cognitive Radio Networks: The Docitive Paradigm," in *Proceedings of the 5th IEEE International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, Cannes, France, pp. 1 - 6, 09 - 11 June 2010.

[295] P. Blasco, L. Giupponi, A. Galindo-Serrano, and M. Dohler, "Energy Benefits of Cooperative Docitive over Cognitive Networks," in *Proceedings of the European Wireless Technology*

Conference (EuWIT), Paris, France, pp. 109 - 112, 27 - 28 September 2010.

[296] M. C. Hlophe and B. T. Maharaj, "QoE-Driven Resource Allocation for SUs with Heterogeneous Traffic using Deep Reinforcement Learning," in *Proceedings of the 2nd IEEE Wireless Africa Conference*, Pretoria, South Africa, pp. 1 - 5, 19 - 20 August 2019.

[297] M. C. Hlophe and B. T. Maharaj, "Optimization and Learning in Energy Efficient Resource Allocation for Cognitive Radio Networks," in *Proceedings of 89th IEEE Vehicular Technology Conference*, Kuala Lumpur, Malaysia, pp. 1 - 5, 28 April - 01 May 2019.

[298] *Policy Language and Architecture for Managing Cognitive Radio for Dynamic Spectrum Access Applications,* IEEE 1900.5 Working Group, 2013. Accessed August 25, 2018, [Online]. Available: http://grouper.ieee.org/groups/dyspan/5

[299] X. Kang, Y-C. Liang, A. Nallanathan, H. K. Garg, and R. Zhang, "Optimal Power Allocation for Fading Channels in Cognitive Radio Networks: Ergodic Capacity and Outage Capacity," *IEEE Transactions on Wireless Communications*, vol. 8, Issue 2, pp. 940 - 950, February 2009.

[300] Z. Wang, T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot, and N. de Freitas, "Dueling Network Architectures for Deep Reinforcement Learning," in *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, New York, USA, 19 - 24 June 2016.

[301] A. Vakili and J. C. Gregoire, "Estimation of Packet Loss Probability from Traffic Parameters for Multimedia over IP," in *Proceedings of the 7th International Conference on Networking and Services (ICNS)*, Venice, Italy, pp. 44 - 48, 22 - 27 May 2011.

[302] T. Fujihashi, T. Koike-Akino, T. Watanabe, and P. V. Orlik, "High-quality Soft Video Delivery with GMRF-based Overhead Reduction," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 473 - 483, 24 August 2017.

[303] F. Nawaz, O. K. Hussain, N. Janjua, and E. Chang, "A Proactive Event-driven Approach for Dynamic QoS Compliance in Cloud of Things," in *Proceedings of the International Conference on Web Intelligence*, Leipzig, Germany, pp. 971 - 975, 23 - 26 August 2017.

[304] S. Shen, Y. Han, X. Wang, and Y. Wang, "Computation Offloading with Multiple Agents in Edge-Computing-Supported IoT," *ACM Transactions on Sensor Networks*, vol. 16, no. 1, pp. 8:1 - 8:27, December 2019.

[305] K. Lee, S-A. Kim, J. Choi, and S-W. Lee, "Deep Reinforcement Learning in Continuous Action Spaces: a Case Study in the Game of Simulated Curling," in *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden, pp. 2937 - 2946, 10 - 15 July 2018.

[306] P. Orbanz, "Projective Limit Random Probabilities on Polish Spaces," *Electronic Journal of Statistics*, vol. 5, pp. 1354 - 1373, 19 October 2011.

[307] W. Sun, J. A. Bagnell, and B. Boots, "Truncated Horizon Policy Search: Combining Reinforcement Learning & Imitation Learning," in *Proceedings of the International Conference on Learning Representations (ICLR)*, Vancouver, Canada, pp. 1 - 14, 30 April - 03 May 2018.

[308] A. M. Al-Jubari, M. Othman, B. M. Ali, and N. A. W. A. Hamid, "TCP Performance in Multi-hop Wireless Ad hoc Networks: Challenges and Solution," *EURASIP Journal on Wireless Communications and Networking*, volume 2011, Article number 198, pp. 1 - 25, 07 December 2011.

[309] M. Y. Mir, C-L. Hu, and S-Z. Huang, "Data Forwarding with Finite Buffer Capacity in Opportunistic Networks," in *Proceedings of the 27th Wireless and Optical Communications Conference (WOCC2018)*, Hualien, Taiwan, pp. 1 - 5, 30 April - 01 May 2018.

[310] J. N. Daigle, *Queueing Theory with Applications to Packet Telecommunication.* New York, NY, USA: Springer Science & Business Media, 16 January 2005.

[311] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained Policy Optimization," in *Proceedings of the 34th International Conference on Machine Learning*, Sydney, Australia, pp. 1 - 10, 07 - 09 August 2017.

[312] R. Cavazos-Cadena, and D. Hernandez-Hernandez, "Vanishing Discount Approximations

in Controlled Markov Chains with Risk-sensitive Average Criterion," *Advances in Applied Probability*, vol. 50, no. 1, pp. 204 - 230, 20 March 2018.

[313] S. Iwamoto, "Golden Optimal Policy in Calculus of Variation and Dynamic Programming," in *Advances in Mathematical Economics*, vol. 10, S. Kusuoka, A. Yamazaki, Eds. Tokyo, Japan, Springer, pp. 65 - 89, 2007.

[314] C. Guestrin, M. Hauskrecht, and B. Kveton, "Solving Factored MDPs with Continuous and Discrete Variables," in *Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence*, Banff, Canada, pp. 235 - 242, 07 - 11 July 2004.

[315] K. V. Delgado, S. Sanner, and L. N. de Barros, "Efficient Solutions to Factored MDPs with Imprecise Transition Probabilities," *Journal of Artificial Intelligence*, vol. 175, pp. 1498 - 1527, 04 January 2011.

[316] N. Salodkar, A. Bhorkar, A. Karandikar, and V. S. Borkar, "An On-Line Learning Algorithm for Energy-Efficient Delay Constrained Scheduling over a Fading Channel," *IEEE Journal On Selected Areas in Communications*, vol. 26, no. 4, pp. 732 - 742, May 2008.

[317] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y. C. Liang, and D. I. Kim, "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 21, Issue 4, pp. 3133 - 3174, 14 May 2019.

[318] M. Wu, Y. Gao, A. Yung, Q. Zhang, and S. Du, "The Actor-Dueling-Critic Method for Reinforcement Learning," *Sensors*, vol. 19, no. 7, pp. 1547-1 - 1547-20, March 2019.

[319] T. J. O'Shea, K. Karra, and T. C. Clancy, "Learning to Communicate: Channel Auto-encoders, Domain Specific Regularizers, and Attention," in *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Limassol, Cyprus, pp. 223 - 228, 12 - 14 December 2016.

[320] M. Musolesi, "Big Mobile Data Mining: Good or Evil?," *IEEE Internet Computing*, vol. 18, Issue 1, pp. 78 - 81, January - February 2014.

[321] F. Boccardi, R. W. Heath, A. Lozano, T.L. Marzetta, and P. Popovski, "Five Disruptive Technology Directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74 - 80, February 2014.

ADDENDUM A DERIVATION OF THE COST FUNCTION

Let $x(t+\Delta t)$ be the actual state when the sample-and-hold controllers $\{u(t),\cdots,u(t+\Delta t)\}$ are applied. Moreover, let $J(x(t+\Delta T))$ be the optimal cost obtained by solving (4.12) based on the new current state $x(t+\Delta T)$, provided the current cost function has decreased. Then, the condition that determines the next transmission time $t+1$ is obtained by checking if the optimal cost (i.e., the current cost function) regarded as a Lyapunov candidate is guaranteed to decrease, i.e.,

$$J(x(t+\Delta T)) - J(x(t)) < 0. \tag{A.1}$$

For more details in deriving this condition, considering **Lemma 3** in [269], the following result holds:

$$J^*(x^*(t+\Delta T)) - J^*(x(t)) \leq -\int_t^{t+\Delta T} \phi(x^*(s),u^*(s))ds, \tag{A.2}$$

where $J^*(x^*(t+\Delta T))$ is the optimal cost obtained by solving (4.12) if the current state at $t+\Delta T$ is $x^*(t+\Delta T)$. This means that the optimal cost would be guaranteed to decrease if the actual state followed the optimal state trajectory $x(s) = x^*(s)$ for $s \in [t,t+\Delta T]$. From (A.2), we obtain

$$J^*(x^*(t+\Delta T)) - J^*(x(t)) \leq J^*(x(t+\Delta T)) - J^*(x^*(t+\Delta T)) - \int_t^{t+\Delta T} \phi(x^*(s),u^*(s))ds, \tag{A.3}$$

where $\phi(x^*(s),u^*(s))$ (as is in (4.5)) is known at t when the solution is obtained. Once solved, the control action that needs to be applied in the next time-slot, $t+1$, is $\varsigma(t) \triangleq (\upsilon(t),\rho(t))$. Because the next time-slot is taken as the next transmission time, it is also the next decision time. The decision on the trade-off between QoS and energy saving is taken based on the value of the optimal cost function $J^*(x^*(t+\Delta T))$ computed for for the current state $x^*(t+\Delta T)$ at time $t+\Delta T$. Then the system state vector is denoted $x(t) = (I(t),E(t))$, which contains the number of available VMs and the energy levels. Therefore, $\varsigma(t) \triangleq (\upsilon(t),\rho(t))$ is the vector that determines the system behavior at time slot t such that the system evolution is described using the discrete-time state-space equation in (4.22).