8-2018

# Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning

Snehal Sudhir Chitnavis
ssc3322@rit.edu

# Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning

Snehal Sudhir Chitnavis

# Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning

Snehal Sudhir Chitnavis

August 2018

A Thesis Submitted
in Partial Fulfillment
of the Requirements for the Degree of
Master of Science
in
Computer Engineering

R·I·T | Kate Gleason
College of ENGINEERING

*Department of Computer Engineering*

# Cross Layer Routing in Cognitive Radio Network Using Deep Reinforcement Learning

Snehal Sudhir Chitnavis

**Committee Approval:**

Dr. Andres Kwasinski *Advisor* Date
Associate Professor, Department of Computer Engineering

Dr. Raymond Ptucha Date
Assistant Professor, Department of Computer Engineering

Dr. Panos P. Markopoulos Date
Assistant Professor, Department of Electrical and Microelectronic Engineering

# Acknowledgments

I would like to thank Dr. Andres Kwasinski for his continuous guidance and support in my research. I would also like to thank Fatemeh Mohammadi for her help and support in validation of this work.

*This thesis is dedicated to my husband, Devavrat. His constant support and encouragement played a big role to help me finish. This work is also dedicated to my mother (Smt. Sushama Chitnavis), uncle (Mr. Vilas Pradhan) and my in-laws (Mr. Chandrashekhar Dighe and Mrs. Manasi Dighe) who have been my source of inspiration and have blessed me with unconditional love and support.*

# Abstract

Development of 5G technology and Internet of Things (IoT) devices has resulted in higher bandwidth requirements leading to increased scarcity of wireless spectrum. Cognitive Radio Networks (CRNs) provide an efficient solution to this problem. In CRNs, multiple secondary users share the spectrum band that is allocated to a primary network. This spectrum sharing of the primary spectrum band is achieved in this work by using an underlay scheme. In this scheme, the Signal to Interference plus Noise Ratio (SINR) caused to the primary due to communication between secondary users is kept below a threshold level.

In this work the CRNs perform cross-layer optimization by learning the parameters from the physical and the network layer so as to improve the end-to-end quality of experience for video traffic. The developed system meets the design goal by using a Deep Q-Network (DQN) to choose the next hop for transmitting based on the delay seen at each router, while maintaining SINR below threshold set by primary channel. A fully connected feed-forward Multilayer Perceptron (MLP) is used by secondary users to approximate the action value function. The action value comprises of SINR to the primary user (at the physical layer) and next hop to the routers for each packet (at the network layer). The reward to this neural network is Mean Opinion Score (MOS) for video traffic which depends on the packet loss rate and the bitrate used for transmission. As compared to the implementation of DQN learning at the physical layer only, this system provides 30% increase in the video quality for routers with small queue lengths and also achieves a balanced load on a network with routers with unequal service rates.

# Contents

# List of Figures

# List of Tables

# Chapter 1

## Introduction

### 1.1 Motivation

With the increasing use of multimedia devices to transfer data, the wireless spectrum assigned for different services is getting exhausted. As the 5G technology needs to support much higher data rates, it requires more bandwidth. Also the development of Internet of Things (IOT) applications have led to higher demand of spectrum. This has lead to increased scarcity in the available wireless spectrum, [4].

A survey on spectrum occupancy measurements done in [5] has shown that most of the licensed bands allocated for different services are underutilized by the primary network incumbent to the spectrum band. Cognitive Radio Networks (CRNs) can provide an efficient solution to this spectrum scarcity problem by utilizing unused spectrum band and enabling dynamic sharing of underutilized portions of the spectrum.

CRNs is a technology that enables multiple users to share the same licensed wireless spectrum. In the CRN paradigm, the primary network is the owner of a licensed spectrum band while the secondary users communicate among themselves by sharing this primary channel in such a way that the primary user's communication is not disrupted. A cognitive radio is an intelligent radio that can monitor, sense and detect the surrounding wireless network environment and dynamically alter its own parameters in order to adapt. A CRN is a network of cognitive radios (secondary users).

## 1.2 Background

The two primary functions of CRNs are efficient utilization of the spectrum and a reliable communication over the secondary network [6].

Various spectrum sharing techniques have been studied in [7] to achieve efficient utilization of the primary channel. The underlay approach is one of the Dynamic Spectrum Access (DSA) techniques which enables co-existence of primary and secondary users on the same spectrum band. In this approach, the secondary users transmit on the same channel that is being used by the primary network by maintaining the Signal to Interference plus Noise Ratio (SINR) caused to the primary users below a threshold limit. This limit on SINR is imposed by the primary users. This enables a more efficient utilization of spectrum by both primary and secondary users.

Traditionally, reliability of communication over the secondary network has been measured using objective metrics of Quality of Service (QoS). These metrics are system-centric where performance variables like Bit Error Rate (BER), delays, jitter are used to calculate the quality of a service at the receiver end. Another approach that is used for performance assessment is the subjective metric of Quality of Experience (QoE). These metrics keep the end user as the center of decision making process in the system design. Mean Opinion Score (MOS) is a QoE metric that is widely used for measuring the quality of video transmissions over a network. MOS indicates the end-user perception of the video quality on the scale of 1(bad) to 5 (very good). Traditionally, a group of end-users would rate the video quality based on their perception and an average of all the ratings would be the MOS for transmitted video. To avoid high cost and offline nature of such tests, objective quality models are developed to predict QoE based on objective QoS parameters [8]. This is an indirect way of predicting QoE while the video is being transmitted. Since the cognitive radios operate in a dynamic environment, such objective quality models to predict QoE are useful to learn the effect of actions taken by secondary users on the transmitted video quality.

### 1.2.1 Cognition in Secondary Users

Cognitive radios usually implement a cognition cycle for effectively sharing the primary spectrum for communication while maintaining the QoE for end user. This cognitive cycle is an observe-learn-decide loop implemented in the secondary users as shown in Fig. 1.1.

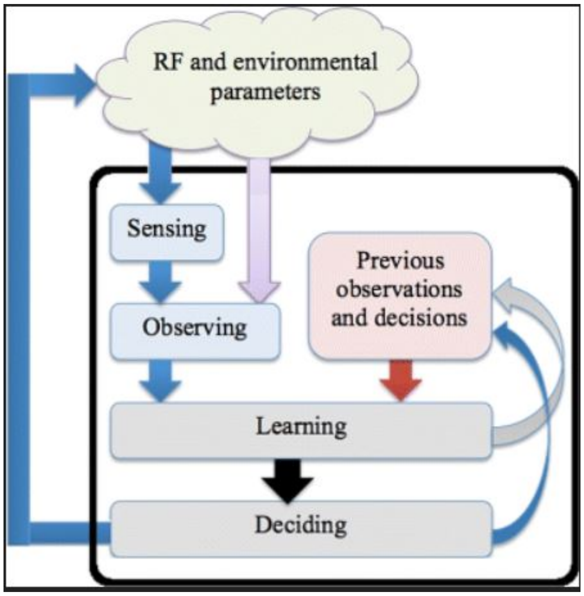

RF and environmental parameters
Sensing
Observing
Previous observations and decisions
Learning
Deciding

**Figure 1.1:** Cognitive cycle [1]

Secondary users sense the surrounding wireless environment. Parameters that are observed in the sensed environment depend upon the DSA technique being used for spectrum sharing. For underlay DSA, interference caused to the primary users is observed on the primary channel. This considers the environmental parameter from the physical layer of Open System Interconnection (OSI) model. Also in order to ensure QoE for transmission over secondary network, parameters like delays, BER, etc., affecting the communication for secondary users can be observed from other layers of the OSI.

The secondary user needs to take actions to adjust their own parameters in order to adapt. For underlay DSA, where interference to the primary users needs to be maintained below the threshold level, secondary users take decisions of changing their transmission rate (own parameter) as per the interference in surrounding environment (observed environmental variable). Updating the transmission rate affects transmission power, thereby changing the SINR. The secondary users learn from environmental observations and past decisions and observations. This then helps in taking decision of updating their own parameters.

### 1.2.2 Artificial Neural Network

The secondary users implement cognitive cycle to update their parameters in order to adapt as show in Fig. 1.1. This cycle can be implemented using a reinforcement learning approach. Artificial neural networks have been useful in implementing the reinforcement learning for environments having multiple learning agents [9].

An artificial neural network is characterized by input layer and the output layer. The output is generated by the Neural Network (NN) by processing the input through a number of hidden layers of neurons. A neuron is the basic building block of the NN which performs weighted addition of the input and applies an activation function. A typical NN is shown in Fig. 1.2. Each circle in neural network represents a single neuron. These neurons are arranged in layers. First layer is the input layer which receives input as the observations made in sensed environment and past observations and decisions. The output of a layer in neural network is provided as input to the following layers, thereby generating the output.

The implementation of a single neuron is shown in Fig. 1.3. This single neuron

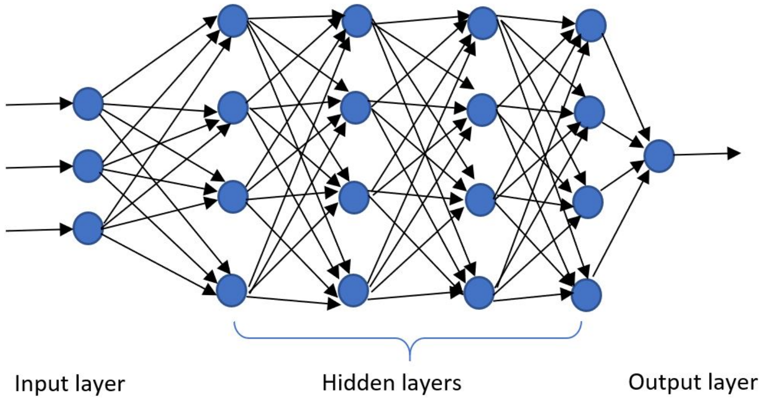


**Figure 1.2:** Deep neural network

is also called 'Perceptron'. As shown in Fig. 1.3, the neuron can have $m$ inputs $\{x_1, x_{2,\ldots}x_m\}$. A neuron is connected to the input layer with weights $\{w_1, w_{2,\ldots}w_m\}$. The output of neuron $\hat{y}$ is calculated by performing weighted addition of the inputs and applying activation function $\phi$ on the sum. This is shown in equation 1.1

$$\hat{y} = \phi\left(\sum_{i=1}^{m} w_i x_i\right) \tag{1.1}$$

The activation function used for Multi Layer Perceptron (MLP) is a sigmoid function given by equation 1.2

$$\phi\left(z\right) = \frac{1}{1+e^{-z}} \tag{1.2}$$

This optimal output value is obtained in the artificial neural network by back propagation of the error. The error is calculated as the difference between expected output and obtained output. The back propagation of error is done by gradient descent algorithm. The weights are updated after back propagation as shown in equation 1.3,

$$w_m^{(i)} = w_m^{(i)} - \alpha dw_m^{(i)} \tag{1.3}$$

where $\alpha$ is the learning rate and $dw_m^{(i)} = x_m^{(i)} * dz^{(i)}$ where $x_m^{(i)}$ is the $m^{th}$ input and $dz$

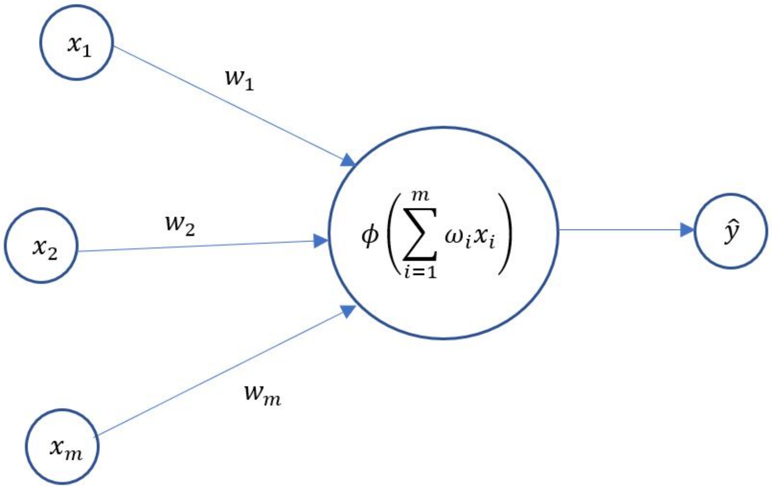


**Figure 1.3:** Single layer feed-forward neural network

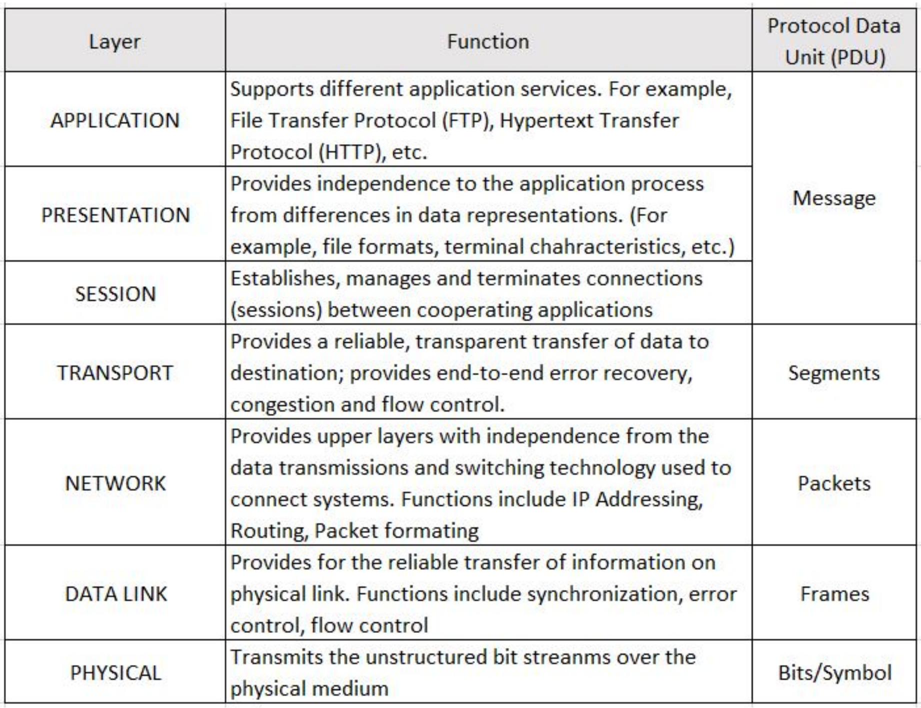
is the derivative of the activation function of the neuron. For the sigmoid function used in this system (equation 1.2), this derivative is,

$$dz = \phi(z)(1 - \phi(z))$$

The output value is the optimal parameter value of the secondary user for the observed environmental variables. For an underlay DSA technique for spectrum sharing, this optimal parameter value is the transmit rate for corresponding target SINR for secondary users.

### 1.2.3 Protocol Layers

Other layers that can be considered for communication over the secondary network are represented by the OSI protocol. The OSI model is a conceptual model that characterizes and standardizes the communication functions of a telecommunication system without regard to its underlying internal structure and technology. Its goal is the interoperability of diverse communication systems with standard protocols [10]. The model partitions a communication system into abstraction layers as shown in



| Layer | Function | Protocol Data Unit (PDU) |
|---|---|---|
| APPLICATION | Supports different application services. For example, File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), etc. | Message |
| PRESENTATION | Provides independence to the application process from differences in data representations. (For example, file formats, terminal chahracteristics, etc.) | |
| SESSION | Establishes, manages and terminates connections (sessions) between cooperating applications | |
| TRANSPORT | Provides a reliable, transparent transfer of data to destination; provides end-to-end error recovery, congestion and flow control. | Segments |
| NETWORK | Provides upper layers with independence from the data transmissions and switching technology used to connect systems. Functions include IP Addressing, Routing, Packet formating | Packets |
| DATA LINK | Provides for the reliable transfer of information on physical link. Functions include synchronization, error control, flow control | Frames |
| PHYSICAL | Transmits the unstructured bit streanms over the physical medium | Bits/Symbol |

**Figure 1.4:** OSI Layers

Fig. 1.4.

The data to be transmitted is composed as a message at the topmost layer (Application layer). This message is then passed through the underlying layers to the physical layer. This is done by encapsulation of information at each layer by adding header information specific to that layer. At the physical layer bits in the information are sent over the physical medium (wireless environment in the CRNs). The function of each layer is explained in the Fig. 1.4.

For a communication system over a secondary network sharing the spectrum band with primary user, spectrum sensing implemented over only the physical layer does not consider other network protocol layer magnitudes that can introduce error in the transmission and hence effect the video quality (QoE). A cross layer approach for resource allocation can prove to be efficient in these case. In a cross layer approach, secondary users observe environmental parameters from other layers of OSI model and take decisions to adapt its own parameters across these observed OSI layers.

### 1.2.4 Queuing Theory

In this work, cross-layer resource allocation is performed over physical layer and network layer. For underlay DSA scheme, SINR caused to the primary network is kept below threshold by allocating optimal transmit rate for secondary users at the physical layer. Optimal routing of packets is considered at the network layer. The routers store and forward the packets received from the secondary user to the destination node. Queues are used for storing the packets arriving at the router. Model of a queue is described by Kendall's notation as A/S/c/K where A is the arrival process, S is the service time distribution, c is the number of servers and K is the size of queue. We use M/M/1 queue model for the intermediate routers. In the M/M/1 queue, arrival of packets follows Poisson process, service time follows an exponential distribution and a single server serves packets in the queue. Fig. 1.5 shows a M/M/1 queue.

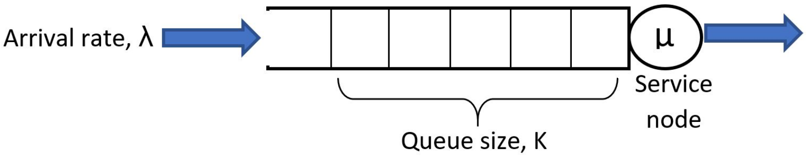


**Figure 1.5:** M/M/1/K queue

The queue utilization is given by $\rho = \frac{\lambda}{\mu}$ where $\lambda$ is the packet arrival rate and $\mu$ is the service rate. If $\rho\ > 1$, then the arrival rate of packets is greater than the service rate of queue indicating an over utilized queue. Such system is not stable as in this case the queue continues to grow until the congestion point. For a stable system, the queue utilization should satisfy $\rho \leq 1$.

## 1.3 Thesis Contributions

The main contribution of this work is to research a cross-layer (physical and network layers) scheme for resource allocation in underlay DSA with the goal to improve the end-to-end quality of experience QoE for video traffic.

In this thesis we develop a system to use the knowledge obtained from the network layer variables along with physical layer variables to perform resource allocation in the transmission of video traffic over the secondary network. Multi-agent Deep Q-networks (DQN) model uses the artificial neural network in secondary users to perform reinforcement learning. The DQN based learning framework is used to implement reinforcement learning in secondary users to consider routing delays at the network layer in order to determine next hop of packets along with maintaining SINR caused to the primary network below the threshold level. Our simulation results show that the developed system outperforms the previous system that uses DQN to find the optimal SINR by considering only physical layer variables.

# Chapter 2

## Related Work

Cognitive Radios being a promising technology, there is a significant volume of on-going research in this field. Related work on CRNs is discussed in the following sections.

## 2.1 Spectrum Sharing

The key aspect of CRNs is that the primary spectrum band needs to be shared dynamically between multiple secondary users. DSA utilizes unused spectrum bands in the spatial and/or temporal domain, called spectrum holes, for communication over the secondary network. There are three DSA models : interweave, overlay and underlay.

### 2.1.1 Interweave Dynamic Spectrum Access

In Interweave DSA, also called Opportunistic Spectrum Access (OSA), SUs are constrained to opportunistically utilize the spectrum holes or white spaces in the temporal, spatial, and/or frequency domain [11]. The key components of this scheme are spectrum sensing, spectrum access and spectrum handoff. The secondary user senses the primary channel to determine unused spectrum (spectrum holes). When an unused spectrum is sensed, the user dynamically accesses the primary channel. If a primary signal appears on the channel, secondary user needs to wait until the

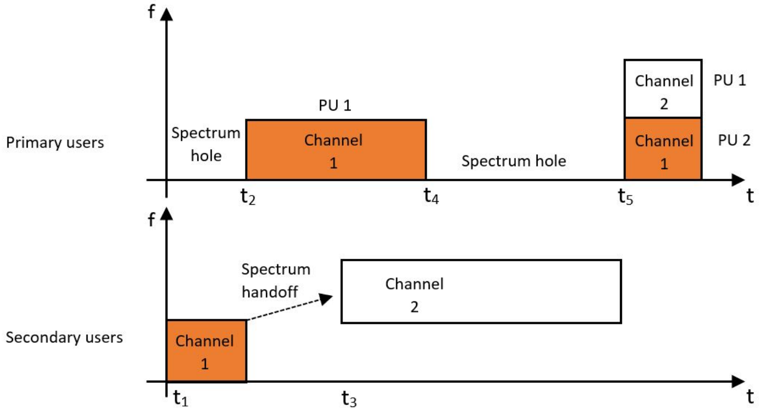


**Figure 2.1:** Interweave Dynamic Spectrum Access

channel is free. In this case, the user can decide to transmit to another channel if it is available. This is done by performing a spectrum handoff.

Fig. 2.1 shows an overlay scheme of spectrum access of the primary spectrum band that has two channels for transmission (Channel 1 and Channel 2). At time $t_1$, the secondary user (SU) senses that the primary spectrum is not being used by primary users (PU1 and PU2). This indicates a spectrum hole, so the SU starts transmission on Channel 1. At time $t_2$, PU1 starts using Channel 1, thus, forcing SU to stop the transmission. Subsequently, the SU waits for PU1 to complete the communication until time $t_3$ at which point it senses that channel 2 is available for transmission. The SU then performs spectrum handoff and starts communicating over Channel 2. At $t_5$, PU1 and PU2 are both communicating over both channels. So the SUs communication gets interrupted as it cannot use any of the primary channels at this time.

Spectrum sensing is an important function of secondary users for transmitting by sensing spectrum holes (white spaces) in the primary channel. The IEEE 802.22 standard for cognitive wireless regional area networks (WRANs) enables broadband wireless access using the cognitive radio technology and spectrum sharing in these spectrum holes [12]. Reinforcement learning is used in [13] for the secondary users to access primary channels based on probability that the channel is occupied by primary

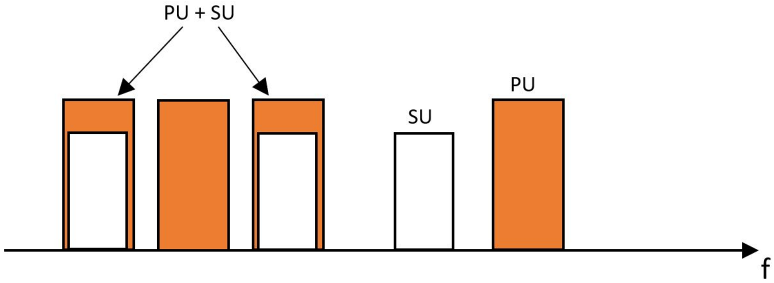


**Figure 2.2:** Overlay Dynamic Spectrum Access

user and mean vacant time on the channel. Work has been done in [14] to implement these opportunistic sharing schemes to minimze the response time to variations of network parameters when the primary user starts transmissions on the channel.

### 2.1.2 Overlay Dynamic Spectrum Access

In the interweave schemes of DSA, secondary users need to stop transmission as soon as the primary user reappears on the primary spectrum band. This disrupts the communication in the secondary network and also adds additional overhead for spectrum sensing and handoff [11]. Also, the interweave DSA suffer from high false alarm probability for spectrum hole detection. The false alarm probability is defined as the probability that the secondary user detects that a primary channel is busy even though it is idle [15]. A false alarm translates into a missed opportunity for spectrum use.

The DSA overlay scheme overcomes these overheads by permitting co-existence of primary and secondary users on the primary spectrum band. In the overlay approach the secondary users are allowed to transmit simultaneously with primary users as long as there is no performance degradation for the primary users. The overlay technique of spectrum sharing is shown in Fig. 2.2. There are two approaches by which the overlay approach can be implemented by the secondary users. One approach is to use channel coding. In this approach, if the primary user is transmitting a packet known to the secondary user then the secondary user splits its transmit power into two parts, one to transmit its own packet and the other to transmit the packet sent by the primary user. This enhances the total power received at the primary receiver, thus, improving the SINR at the primary user. The second approach is network coding. In this approach, secondary users behave as relay nodes to transmit the primary user's information over the network. While relaying packets sent by primary user, the secondary users may encode their own packets onto the primary user packet.

### 2.1.3 Underlay Dynamic Spectrum Access

In the overlay approach of DSA if the performance for primary users cannot be guaranteed not to degrade, then the secondary users need to yield to the primary users for spectrum access. This interrupts the communication for secondary users.

The underlay DSA scheme is similar to the overlay scheme since it also allows coexistence of primary and secondary users on the same spectrum bands. However, the transmission by secondary users is constrained by the condition that the accumulated interference from all secondary users is tolerable by the primary users. This is the approach taken by the ultra-wide band (UWB) technology. This approach is primarily for short range communications [11].

Fig. 2.3 shows the underlay scheme for sharing the primary channel with multiple users. In this scheme, the secondary users transmit on the same frequency channel and at the same time as the primary user by keeping the SINR level below a threshold set by primary. This threshold is set by the primary user according to its minimum acceptable interference.

Although limited in the transmit power, in underlay schemes, more spectrum bandwidth is available for secondary users since they transmit along with the primary user. For transmitting time sensitive data like live streaming video, the channel should be available for transmission for the entire duration of communication. In this study,

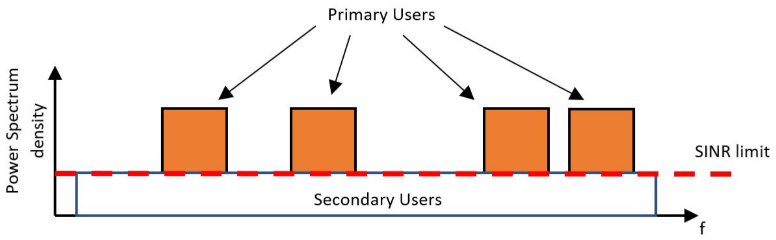


**Figure 2.3:** Underlay Dynamic Spectrum Access

underlay scheme of DSA is used for transmitting video over secondary network.

Underlay DSA allows the primary and secondary users on the same frequency bands but it requires strict SINR threshold limits to be imposed on secondary user communication. Resource allocation strategies in secondary users are therefore crucial to achieve efficient communication on the secondary network along with maintaining the SINR threshold limits.

## 2.2 Cognition in Secondary Users

Cognition in secondary users is to learn from the surrounding environment. This can be divided into two parts - learning status awareness from the environment (spectrum sensing and sharing) and learning to adapt for efficient communication over secondary network. The information gained from the environment is used to implement effective communication strategy. Two types of learning methods can be implemented for cognition in secondary users - supervised and unsupervised. Unsupervised learning algorithms, like model free reinforcement learning, have been effective in cognitive radio. Model-free approach enables secondary users to adapt their behaviors based on the reinforcement from their interaction with the environment and build their understanding of the system from scratch through trial-and-error [16].

The resource allocation problem for model-free reinforcement learning in CRNs is solved with the use of a discrete time Markovian Decision Process MDP shown in Fig. 2.4. A MDP consists of three elements:

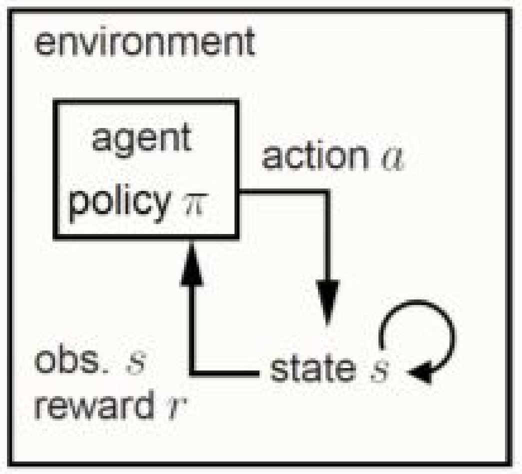


**Figure 2.4:** Markov Decision Process (MDP) structure [2]

1. States $S$ : The state space indicates the variables from the environment that will be sensed by an agent to derive its states. In this work, the secondary users are the decision-making agents.

2. Actions $A$ : The set of possible actions that could be taken in order to improve the performance of the agent in its environment is indicated by an action space. The secondary user selects an action $a$ from the action space by applying the policy $\pi$.

3. Transition Probabilities $T$ : Transition function specifies how likely it is to end up at any state, given the current state and a specific action performed by the agent. Transition probabilities are specified based on the Markovian assumption. We focus on homogeneous processes in which the system dynamics are independent of the time. Thus the transition function is stationary with respect to time [17]:

$$T\left(s, a, s'\right) \overset{\text{def}}{=} P_r\left(s_{t+1} = s' | a_t = a, s_t = s\right).$$

where $P_r$ is the transition probability at time $t$ for state $s_t$ and action $a_t$.

4. Rewards $R$ : Reward is a measure of success or failure of the action selected by the agent. This is measured by observing the environment for changes in the states after the agent takes the action.

5. Discount factor $\gamma$ : $\gamma \in [0, 1)$ is the discount rate used to calculate the long-term return.

The agent starts at state $s_0 \in S$. At each time step $t$, the agent takes an action $a_t$ from the action space $(A)$. The system then makes a transition to next state as per the transition function $T$ and the agent receives an immediate reward $R$. The goal of the agent is to maximize the discounted sum of rewards over a long duration.

The agent's action selection as per the changes in environment is defined as the agent policy $(\pi)$. The agent interacts with the environment and takes actions according to the policy. The value function of the policy is defined to be the expectation of the return given that the agent acts according to that policy. This value function defined over the state-action pair is the $Q$-function or the Q-value of that pair.

It has been shown (Bellman, 1957) that for any MDP, there exists an optimal deterministic policy that is no worse than any other policy for that MDP [17]. For secondary users in CRNs sharing the spectrum using underlay DSA, the condition of keeping SINR to the primary user needs to satisfied by all the secondary users. According to the Bellman's condition of optimality, this can be achieved by taking the optimal action if all the strategies thereafter are optimal.

### 2.2.1 Reinforcement Learning

Reinforcement learning is an artificial intelligence approach which can be used to determine the optimal policy for the MDP. The work in [18] provides a extensive review on wide range of traditional and enhanced reinforcement learning algorithms in the CRNs context. The review shows that performance enhancements have been achieved with the use of reinforcement learning algorithms.

Q-learning is one of the popular model-free reinforcement learning techniques. The Q-learning algorithm was used in [19] to implement cross layer CRNs. In this work, the cognitive cycle at the secondary users was used to adapt in an integrated manner transmit bit rate and joint source-channel coding rate. The end-to-end distortion was measured at the application layer in this work, thus, performing cross-layer resource allocation for physical and application layer in OSI model. This algorithm was also used in [20] for routing in a wireless sensor network. In this paper, a variant of Q-learning algorithm, $Q - RC$ was used to find the best routing strategy to compress and aggregate packets resulting in increasing the energy efficiency. The Q-learning algorithm works by taking intermediate reward from the environment which results in gradual optimization of the transmit parameters. Hence, Q-learning algorithms suffer from slow convergence for a large action space. This drawback can be compensated by using artificial neural networks to approximate the optimal solution for actions of secondary users.

DQN is a class of an emerging class of reinforcement learning algorithms combined with neural networks. Google's Deep Mind developed the artificial agent, DQN, using this type of neural network. This agent surpasses all previous neural network algorithms for single learning agent playing Go and video games [9]. The work in [21] extends this DQN agent to a multi agent environment where all the agents perform reinforcement learning through collaboration to play video games. This concept can be extended to the CRNs architecture under consideration. All the secondary users can implement DQN to collaborate and share information learnt about the environment. This can lead to faster convergence and better adaptability to the dynamic nature of wireless environment for CRNs. Also DQNs are used in [22] to develop power allocation method for secondary users in CRNs. This work focuses on optimizing the transmission from secondary users by considering physical layer parameter.

The neural network that is used in this work is a feed-forward MLP. A feed-forward neural network is a neural network that does not contain any signal cycles, i.e. the output is not connected as a feedback to the neural network [23]. Reinforcement learning is used in the MLP. In this type of learning, the neural network learns the network parameters by choosing actions from the action space and observing the effect on environment. The effect on environment is measured in terms of rewards - positive actions result in increasing the rewards and negative actions cause reduction of rewards. After performing iterations on the action value set, the neural network converges to an optimal value.

The selection of optimal values is done by the DQN agent in a state from the state space at time $t$ by selecting an action from the action space. The effect of this action on the environment is observed to get reward for the action.

## 2.3 Cross Layer Cognitive Radio Networks

The DQNs can work on a larger set of actions. Neural networks have been used for many applications where they need to find optimal solutions from larger action spaces. In [24], reinforcement learning is used to find solution over large discrete action spaces. The cognitive cycle in secondary users can be used for cross layer approach by increasing the set of actions and observations from the environment to consider parameters from other OSI layer along with the physical layer. Traditionally for underlay DSA techniques, CRNs have dealt with adjusting the power allocation at the secondary users to reduce interference to the primary. The power allocation is adjusted by adjusting the transmit bit rate of the secondary users. The function of the physical layer of OSI model is transmission of the unstructured bit stream as shown in Fig. 1.4. Thus, such systems target a single layer of OSI model which is the physical layer for resource allocation by optimal policy selection.

The Cross Layer CRNs observes several parameters from the environment and performs optimization across all the sensed and observed parameters. The learning from variables from different layers of the OSI model helps secondary users to decide reduction of interference to the primary network along with increasing the efficiency of communication on the secondary network.

In [25], the secondary users learn the best routing path along with the dynamic spectrum allocation. This being an overlay DSA scheme, the secondary users recognize the free spectrum sub-bands. The secondary users transmit data on these frequencies while determining the best routing path. Thus, the parameters from physical layer (unused frequency) and network layer (routing) are used together to adapt by the secondary. The work in [26], proposes a cross-layer opportunistic spectrum access and dynamic routing algorithm for CRNs, called ROSA (ROuting and Spectrum Allocation algorithm). This algorithm jointly considers routing, spectrum assignment, power allocation, and (potentially) congestion control in a distributed way. In [27], decentralized and localized algorithms for joint dynamic routing, relay assignment, and spectrum allocation under a distributed and dynamic environment are studied for co-operative CRNs. This work shows that these algorithms lead to increased throughput with respect to non-cooperative strategies. In [28], CRNs are studied to develop analytical framework model to perform congestion control over the transport layer. This framework is also designed for an overlay approach where the operation on cognitive radios will be stopped when the primary user starts using the channel for its communication.

All the previous work discussed above is done for CRNs implementing the overlay technique of DSA for spectrum sharing. In this work we study and develop a cross layer resource allocation for the underlay DSA approach. The novel contribution of this work is to develop a cross-layer resource allocation scheme using a multi-agent DQN learning framework for multiple secondary users in a CRNs.

# Chapter 3

## System Setup and Problem Description

## 3.1 Problem Setup

In this work we study the system implementing underlay scheme for cross-layer resource allocation in CRNs for reinforcement learning approach using a DQN [22]. The system comprises of a primary user that can transmit exclusively on the primary channel at any time instant. This primary channel is shared with N secondary network transmissions in a way as to keep the SINR to the primary network below a set threshold level. It is assumed that both primary and secondary users use Adaptive Modulation and Coding (AMC), where the modulation scheme and the coding rate can be adapted as per the SINR on the transmission link. This helps the secondary users to change the transmit rate in order to keep the SINR in the primary network below the threshold limit. The communication channel is assumed to be a quasi-static channel with Additive White Gaussian Noise (AWGN).

### 3.1.1 Primary and Secondary Networks at the Physical Layer

The underlay DSA technique used in this system requires knowledge of SINR. It is measured at the primary base station and at secondary base stations for primary link $SINR^{(P)}$ and secondary link $SINR^{(S)}$, respectively as shown in equations (3.1) and

(3.2)

$$SINR^{(p)} = \frac{G_0^{(p)} P_0}{\sigma^2 + \sum_{j=1}^{N} G_j^{(s)} P_j} \tag{3.1}$$

$$SINR_i^{(s)} = \frac{G_i^{(s)} P_i}{\sigma^2 + G_0^{(s)} P_0 + \sum_{j \neq i} G_j^{(s)} P_j} \tag{3.2}$$

where $P_0$ is the transmit power of primary user, $P_j$ is the transmit power of the $j^{th}$ secondary user ($SU_j$). $G_0^{(p)}$ and $G_j^{(p)}$ are the channel gains of primary user and secondary user to the primary base station respectively, $G_0^{(s)}$ is the channel gains between primary user and the $j^{th}$ link receiver (j), $G_j^{(s)}$ is channel gain on the $j^{th}$ secondary link and $\sigma^2$ is the background noise power.

We denote the SINR threshold requirements as $\beta_0$ for the primary link SINR and $\beta_i$ for the $i^{th}$ secondary link SINR. This is shown in equation (3.3)

$$\begin{gathered} SINR^{(p)} \geq \beta_0 \\ SINR_i^{(s)} \geq \beta_i, \qquad i = 1, ..., N \end{gathered} \tag{3.3}$$

The digital modulation scheme used is Orthogonal frequency-division multiple access (OFDMA). The power settings for transmission on secondary links using OFDMA is (3.4)

$$\begin{gathered} P_i = \frac{\Psi_i \left( \sigma^2 + G_0^{(s)} P_0 \right)}{G_i^{(s)} \left( 1 - \sum_{j=1}^{N} \Psi_i \right)}, \qquad i = 1, ..., N \\ \Psi_i = \left( 1 + \frac{1}{\beta_i} \right)^{-1} \end{gathered} \tag{3.4}$$

For the power allocation to be valid, the condition $1 - \sum_{i=1}^{N} \Psi_i > 0$ needs to be satisfied. After replacing secondary user powers in (3.1) with the results of (3.4), primary users' SINR constraints in (3.3) can be written as [22]:

$$\sum_{j=1}^{N} \alpha_j \Psi_j \leq 1, \tag{3.5}$$

where

$$\alpha_j = \frac{G_j^{(P)}\left(\sigma^2 + G_0^{(s)} P_0\right)}{G_j^{(s)}\left(\frac{G_0^{(p)} P_0}{\beta_0 - \sigma^2}\right)} + 1.$$

Secondary users control the interference they create on the primary link by adapting their SINR target, $\beta_i$, and through (3.4) their transmit power. The target SINR on the $i^{th}$ secondary link, once met, results in the transmit bitrate $r_i$, through [29]

$$\beta_i = \frac{2^{\left(\frac{r_i^{(s)}}{W}\right)} - 1}{k}, \tag{3.6}$$

where

$$k = \frac{1.5}{-\ln\left(5BER\right)}.$$

### 3.1.2 Network Layer Setup

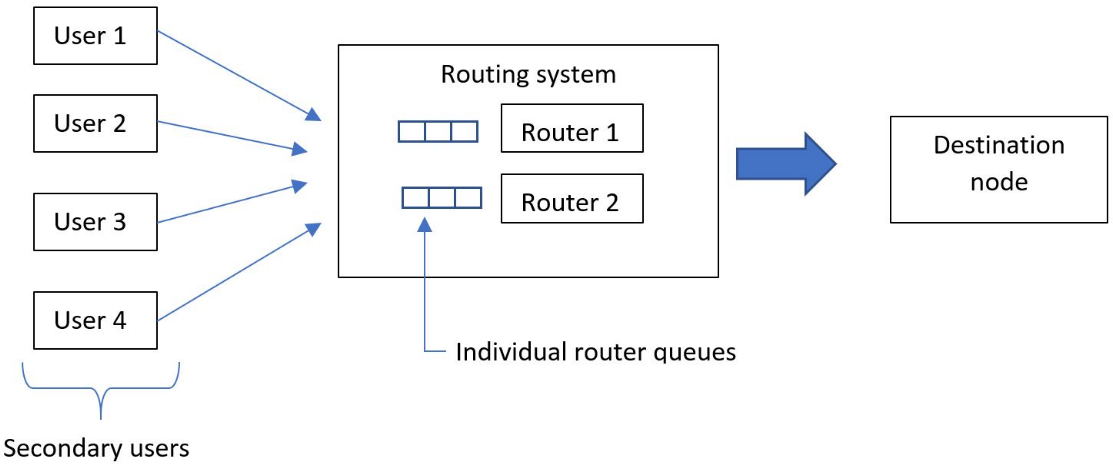


**Figure 3.1:** Routing system

The secondary users need to transmit the data to the destination nodes through routers. Fig. 3.1 shows four secondary users transmitting over such system. Routing path to the destination node through each router is compared by the secondary users and the packets are transmitted to the router with highest throughput. If the queue of the selected router is not full, packets arriving are placed in the queue, else the packets get dropped. These queues are modeled as M/M/1/K queues. They are finite queue with size of K packets. The arrival of packets is modeled as a Poisson process with arrival rate $\lambda$. The service times for packets follow an exponential distribution with mean $\mu$. When the arrival rate is greater than the service rate ($\lambda >\mu$), then the utilization of M/M/1/K queues is large $\left(\rho = \frac{\lambda}{\mu}\right) >1$. This indicates that the system is unstable resulting into increase in the queue lengths to the congestion point. When the queue length reaches its size K all packets that arrive will be dropped until the queue length decreases. This results in packet loss. In this work we consider that the traffic carried over the secondary links is video transmissions for Internet Protocol Television (IPTV) services. The data at network layer is measured in packets, where one packet size is 7*188 bytes (10528 bits).

### 3.1.3 Quality Metrics

QoE is used to measure the perception of video quality for the end-user after distortion due to source coding and packet loss at the routers. In this work, MOS is the metric used for measuring QoE. Model in [3] is used to estimate the MOS for video transmission over the secondary network. This model estimates QoE using the objective measurements of network performance. The network performance for the routing system is measured objectively by packet arrival rate (coded bit rate) and Packet Loss Frequency (PLF) (number of packet lost burst events in one video of 10 second duration). This estimation model is shown in Fig. 3.2

This model uses parameters, PLF and the coded bit rate, to estimate the video

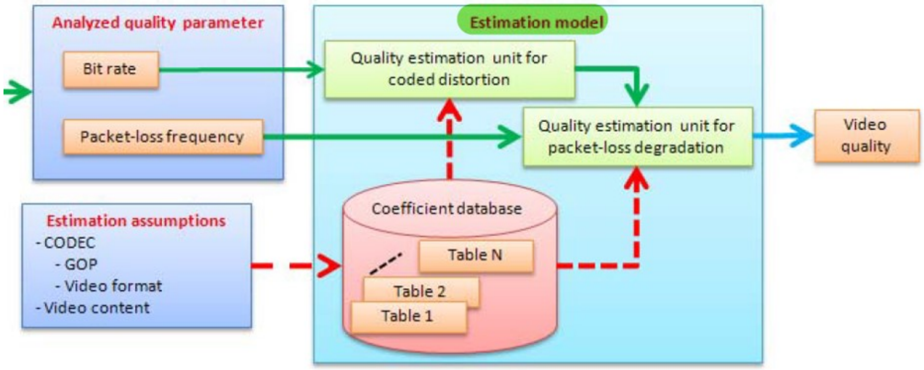


**Figure 3.2:** Estimation model for QoE [3]

quality. The estimated video quality by this model, $V_q$, is given by equation (3.7) [3]

$$V_q = 1 + I_c \text{exp}\left(-\frac{PLF}{v_4}\right), \tag{3.7}$$

where

$$I_c = v_1 - \frac{v_1}{1 + \left(\frac{B_r}{v_2}\right) \cdot v_3}. \tag{3.8}$$

where PLF is number of packet lost burst events for a video of 10sec duration, $B_r$ is the coded bit rate and $v_1$, $v_2$, $v_3$ and $v_4$ are the coefficients of CODEC used for video transmissions. In this work we used the H.264 CODEC for IPTV video transmissions. The coefficients of this CODEC ($v_1$, $v_2$, $v_3$ and $v_4$) are as indicated in Table 3.1 [3]. The coded bit rate for this system depends upon SINR as can be seen from equation 3.6.

| v1 | v2 | v3 | v4 |
|---|---|---|---|
| 3.8 | 4.9 | 3.6 | 3.5 |

**Table 3.1:** Coefficients for H.264 CODEC [3]

The value of MOS $V_q$ in (3.7) ranges from 1 to 5. The significance of the value of MOS from an end-user perspective is shown in Table 3.2

| MOS | Video Quality |
|---|---|
| 1 | Very poor |
| 2 | Poor |
| 3 | Average |
| 4 | Good |
| 5 | Very good |

**Table 3.2:** Perception of quality measured through MOS

## 3.2 DQN-Based Learning Framework

In this system, the secondary users need to transmit over the secondary network by keeping SINR to the primary users below a threshold level while maintaining the QoE. In order to maintain the QoE over the network layer, the environmental variables that need to be observed are the queuing delays at the router and packet loss in the network along with the SINR. The metric for QoE used in this work, MOS, depends upon the coded bit rate (Refer 3.6). Thus, changing the transmit rate affects QoE as well as SINR. This resource allocation of transmit rates for secondary users needs to be dynamic as per the changes observed in the environmental variables.

In this work, we use artificial neural network to solve this dynamic resource allocation problem. The dynamic resource allocation framework used in this work is the multi-agent DQN network. This framework maximizes the overall QoE while maintaining the threshold SINR level. This is done by reinforcement learning. In this technique, the secondary user (a reinforcement learning agent) performs near-optimal control actions by observing environmental state and receiving immediate rewards [22].

### 3.2.1 Framework

Let the set of state space for this framework be $S = \{s_1, s_2, ...s_n\}$. The state at time $t$ reflects the interference caused by the secondary users on both the primary and secondary networks. The states, $S_t = (I_t, L_t)$ are as shown in equations (3.9) and

(3.10)

$$I_t = \begin{cases} 0, & \text{if } \sum_{i=1}^{N} \Psi_i \left(\beta_t^{(i)}\right) < 1. \\ 1, & \text{otherwise.} \end{cases} \tag{3.9}$$

$$L_t = \begin{cases} 0, & \text{if } \sum_{i=1}^{N} \alpha_i \Psi_i \left(\beta_t^{(i)}\right) < 1. \\ 1, & \text{otherwise.} \end{cases} \tag{3.10}$$

The states $S_t$ reflect the interference caused by the secondary users. The value of $I_t$ depends upon the power allocation validity for secondary users and $L_t$ depends upon the SINR constraint for primary users.

At time $t$, an agent observes state $S\left(t\right)$ and selects an action from the action space, $A\left(t\right)$. The action space is a finite discrete space of candidate target SINRs, denoted by $A = \{\beta_1^{(i)}, \beta_2^{(i)}, ...\beta_n^{(i)}\}$. By selecting one action from the action space, the secondary user is modifying its transmit power and also the transmit code rate and modulation scheme.

In reinforcement learning, the optimal action selection is done by receiving immediate rewards. In this system, the reward is calculated as given in equation 3.11

$$r_t^{(i)}\left(a_t, s_t\right) = \begin{cases} M, & \text{if } I_{t+1} + L_{t+1} > 0. \\ V_q, & \text{otherwise.} \end{cases} \tag{3.11}$$

where M is a constant that is set to be smaller than the reward for any action selected by the system. This encourages the state (0,0). $V_q$ is the MOS for IPTV packets as given by (3.7). It is assumed that the secondary users don't have any information about other users and all other users are considered a part of environment to calculate the reward. The part of this design corresponds to the physical layer.

### 3.2.2 Optimal Action Selection

The optimal action is selected in reinforcement learning by selecting an action from the action space and receiving reward for the action. Each secondary user conducts a search into the action space to find an optimal policy through the DQN learning algorithm to maximize its own reward. The reward reflects experienced MOS for each secondary user. The overall MOS is maximized when each secondary user selects actions that maximize the cumulative future reward. The optimal action-value function is as follows:

$$Q_i^* \left(s,a\right) = \max_{\pi} \left\{ \sum_{t=0}^{\infty} \gamma^t E \left( r_i(s,a)_t^{(i)} |_{s_t=s,a_t=a,\pi} \right) \right\} \tag{3.12}$$

where $Q_i^* \left(s,a\right)$ is the maximum discounted sum of rewards $r\left(s,a\right)$ over a long time under an optimal policy $\pi$. According to Bellman's principle of optimality, the solution to (3.5) can be obtained by taking the optimal action if all the strategies thereafter are optimal:

$$Q_i^* \left(s,a\right) = \max_{a'} \left[ r_i\left(s,a\right) + \gamma Q_i^* \left(s',a'\right) \right]. \tag{3.13}$$

The equation (3.12) is a value iteration algorithm that converges to the optimal Q-value if $t \rightarrow \infty$. A function estimator is used to estimate the optimal action-value function. We use a neural network as nonlinear approximator to estimate the action-value function. A fully connected feed-forward Multilayer Perceptron (MLP) network is used to perform the maximization of reward and obtain the optimal action values.

DQN is a deep reinforcement learning algorithm that combines the process of reinforcement learning with deep neural networks to approximate the Q action-value function. DQN uses the technique of ”experience replay”. This technique uses a replay memory which stores the experience of each secondary user after taking an action from the action set. At each time step the value of this replay memory is updated by next state values and current values of action, state and reward $e_i(t) = (a_i(t), s_i(t), r_i(t), s_i(t+1))$.

Each secondary user utilizes two neural networks - one to approximate the action-value function $Q_i(s, a; \theta_i)$ and other to approximate the target action-value function $Q'_i(s, a; \theta'_i)$ where $\theta_i$ and $\theta'_i$ represent current and old parameters respectively. The current parameters are updated through mini-batch of random samples from replay memory $e_i(t)$. The parameters $\theta_i$ of action-value function are updated utilizing gradient descent algorithm based on following cost function:

$$L(\theta_i) = E\left[\left(r_i(s,a) + \gamma \max_{a' \in A}(Q'_i(s', a'; \theta'_i)) - Q_i(s, a; \theta_i)\right)^2\right] \tag{3.14}$$

### 3.2.3 Algorithm

Algorithm 1 explains the DQN-based learning framework. This algorithm optimizes the action value based upon SINR caused by the secondary user. Upon convergence, this algorithm finds the optimal value of the average transmission rate for secondary users. The secondary users transmit over the network with this transmission rate. The implementation of routing system for the network layer is explained in algorithm 2. Average packet loss frequency is calculated for the videos transmitted as,

$$Average\ PLF = (Number\ of\ packets\ lost)/(Duration\ of\ video).$$

**Algorithm 1** Multi-agent DQN-based learning framework

```
for all $SU_i$, i = 1, ..., N do
    Initialize replay memory
    Initialization of the neural network for action-value function $Q_i$ with random weights $\theta_i$
    Initialization of the neural network for target action-value function $Q'_i$ with $\theta'_i{=}\theta_i$
end for
for Monte-Carlo simulations = 1:K do
    for $t < T$ do
        for all $SU_i$, i = 1, ..., N do
            Select a random action with probability $\epsilon$
            Otherwise select the action
            $a_t^{(i)} = \arg\max_{a_t^{(i)}} \; Q_i\left(s_t^{(i)}, a_t^{(i)}; \theta_i\right)$
            Update the state $s_{t+1}^{(i)}$ (3.9) and (3.10) and the rewards $R_t^{(i)}$ (3.11)
            Store $e_i\left(t\right) = \left(a_i\left(t\right), s_i\left(t\right), r_i\left(t\right), s_i\left(t+1\right)\right)$ in experience replay memory of secondary user $i$, $D_i$
            Update parameters ($\theta$) of action-value function $Q\left(s_t^{(i)}, a_t^{(i)}; \theta_i\right)$, by sampling random mini-batch of transitions from $D_i$ (3.14)
            Every C step update parameters of target action-value function $\theta'_i{=}\theta_i$
        end for
    end for
end for
```

**Algorithm 2** Queueing System

```
for $tt < (VideoDuration)/\delta$ do
    Send a packet in the video to destination
    if equal output rate of routers then
        Probabilistic routing strategy for next hop
    else
        minimum cost = maximum path bandwidth
        Next hop = minimum cost
    end if
    if queue full at router then
        Packet is dropped. Increment the packet loss counter
    else
        Place packet in queue and calculate time to reach destination
    end if
end for
Calculate PLF, average delay for all packets sent and queue utilization
```

# Chapter 4

## Cross Layer Routing using Deep Reinforcement Learning

The quality of video for secondary users transmitting over a network gets affected due to packet loss at the routers. The resource allocation considering only physical layer does not account for packet loss observed at the network layer. Also, the static routing algorithms do not consider the dynamic nature of environment of CRNs. This affects QoE for the end-users. In this work we develop a cross layer routing scheme using DQN-based framework for reinforcement learning. In this technique, the secondary users perform dynamic resource allocation over both the physical as well as the network layer. In order to consider network layer, action space for each secondary user performing reinforcement learning is expanded. The expanded action space comprises of SINR on the transmission link and next hop for packets based on delays observed at the intermediate routers. The neural network uses DQN-based framework to perform optimization over the action space, $A = [\beta_1, \beta_2 \cdots \beta_n, h_1, h_2 \cdots h_n]$ where $\beta_1, \beta_2 \cdots \beta_n$ represents SINR and $h_1, h_2, ..., h_n$ represent next hop for each secondary user.

## 4.1 Cross layer Routing in CRN

The DQN framework for cross layer routing has the state space as $S_t = (I_t, L_t)$ given by equation (3.9) and (3.10). The state space reflects interference caused by the secondary users under the constraints of threshold set by primary user and validity of power allocation. The cross layer routing considers optimization over the network layer for IPTV video transmissions. As such, the action space is expanded to consider the network layer parameter, next hop for the packet routing. Routing delays observed by the secondary users are given as an input to the neural network. This routing delay is the total delay of service time for the packet leaving the queue ($\mu$) and the time required to service packets already present in the queue. Next hop is estimated by neural network for each packet generated based on this routing delay. The time spent by a packet in the routing system ($T_p$) is given by equation (4.1)

$$Tp\,(t) = (1 + qlen\,(t)) * pktlen * \mu \tag{4.1}$$

where $qlen\,(t)$ is length of queue at at time $t$, $pktlen$ is the length of IPTV packets and $\mu$ is the service rate.

Algorithm 3 depicts the steps for cross layer routing. Secondary users utilize the $\epsilon$-greedy approach for selecting the first action parameter, transmit rate. The video transmissions over network are performed using this selected transmit rate. For each video packet that is being sent over the network, the second action parameter (next hop) is selected based on delay seen at each router in the system. Action is updated with next hop to the router with minimum delay.

Five videos, each of 10 second duration, are sent over the network. When a packet in current video transmission arrives at the router with a full queue, then the packet gets dropped causing a packet loss event. After all the videos are transmitted over the system, average PLF is calculated for each video. PLF is the number of packet lost burst events over a 10 second duration. Reward, MOS, is calculated for these observed parameters, average PLF and transmit rate using equation (3.7).

**Algorithm 3** Multi-agent DQN-based learning framework

```
for all $SU_i$, i = 1, ..., N do
    Initialize replay memory
    Initialization of the neural network for action-value function $Q_i$ with random weights $\theta_i$
    Initialization of the neural network for target action-value function $Q'_i$ with $\theta'_i$=$\theta_i$
end for
for Monte-Carlo simulations = 1:K do
    for $t < T$ do
        Initialize the parameters of routing system to 0
        Select a random action with probability $\epsilon$
        Otherwise select the action
        $a_t^{(i)} = \arg\max_{a_t^{(i)}} \quad Q_i \left( s_t^{(i)}, a_t^{(i)}; \theta_i \right)$
        Update $\lambda$ with the action selected.
        for $tt < VideoDuration$ do
            if Probability of packet arrival $< \lambda * \delta$ then
                for all $SU_i$, i = 1, ..., N do
                    Compare router delays and update next hop for the packet
                end for
            end if
            if queue full at router then
                Packet is dropped. Increment the packet loss counter
            else
                Place packet in queue and calculate time to reach destination
            end if
        end for
        Update the state $s_{t+1}^{(i)}$ (3.9) and (3.10) and the rewards $R_t^{(i)}$ (3.11)
        Store $e_i\left(t\right) = \left(a_i\left(t\right), s_i\left(t\right), r_i\left(t\right), s_i\left(t+1\right)\right)$ in experience replay memory of secondary user $i$, $D_i$
        Update parameters $(\theta)$ of action-value function $Q\left( s_t^{(i)}, a_t^{(i)}; \theta_i \right)$, by sampling random mini-batch of transitions from $D_i$ (3.14)
        Calculate average PLF over the number of videos sent.
        Every C step update parameters of target action-value function $\theta'_i$=$\theta_i$
    end for
end for
```

## 4.2 Simulation Setup

Monte-Carlo simulations were used to study the action value selection. For primary user, threshold SINR is set at 1dB, transmission power is 10mW and power of AWGN is 1nW. The primary user and all secondary users are distributed within 300m radius around the primary base station and secondary base station respectively. Distance between primary and secondary base stations is 2km. The path loss model for channel gains is log-distance model with loss exponent of 2.8 transmission rate for a single secondary user could be chosen between range of 0.1 to 0.5 Mbps.

In the IPTV communication model, each elementary stream is converted into an interleaved stream of time stamped Packetized Elementary Stream (PES) packets. The transport stream (TS) is formed by breaking up the PES packets into fixed-sized TS packets of 188 bytes that are referenced to independent time bases [30]. One IP packet consists of 7 TS packets [3]. Thus number of bits in 1 packet = 7*188*8 = 10528 bits. 10 videos of 10 seconds each are transmitted over the routing system. The routing system consists of 2 routers with M/M/1/K queues. The transmission rate is adjusted by the secondary users in order to adjust the SINR. This is possible due to the AMC scheme used in secondary users. The arrival rate of packets is determined by this transmit rate range for the secondary users which is 0.1 to 0.5 Mbps. This rate is the result of action taken by the neural network in secondary users.

DQN has learning rate set to $\alpha = 0.01$ and discounting factor set to $\gamma = 0.9$ For $\epsilon$-greedy approach, $\epsilon$ is set to 0.8 initially, after convergence, it reduces to 0. Two separate feed-forward MLP networks are used to approximate action-value and target action-value functions. Each of these neural networks have three fully connected hidden layers with four, three and two neurons respectively. The input layer consists of four nodes representing state ($I_t$ and $L_t$) and selected action to be taken (SINR and next hop for the packets). The output layer is one neuron corresponding to the selected action SINR.

The neural networks were also tried with different configurations of number of hidden layers and number of neurons in each hidden layer to study performance changes. The alternate configuration tried was of 2 hidden layers with 10 neurons each. A little performance improvement is seen. The convergence of DQN algorithm is a little faster with this configuration of the neural network and the results for MOS are consistent with those with previous configuration.

The capacity of replay memory and mini-batch is 100 and 10 respectively. The number of Monte-Carlo simulation iterations is set to 500.

## 4.3 Simulation Results

### 4.3.1 Benchmark Algorithm

The algorithm 1 was selected as a benchmark. This algorithm performs resource allocation for only the physical layer. The DQN framework provides optimum bitrate for each secondary user for the observed environmental conditions. This average transmission rate is as shown in table 4.1. These transmit rates were selected from a set of 0.1 to 0.5 Mbps rates by the DQN algorithm.

| Number of Secondary Users | 2 | 4 | 6 |
|---|---|---|---|
| Average bitrate (Mbps) | 0.1646 | 0.1737 | 0.2439 |

**Table 4.1:** Average bitrates for secondary users

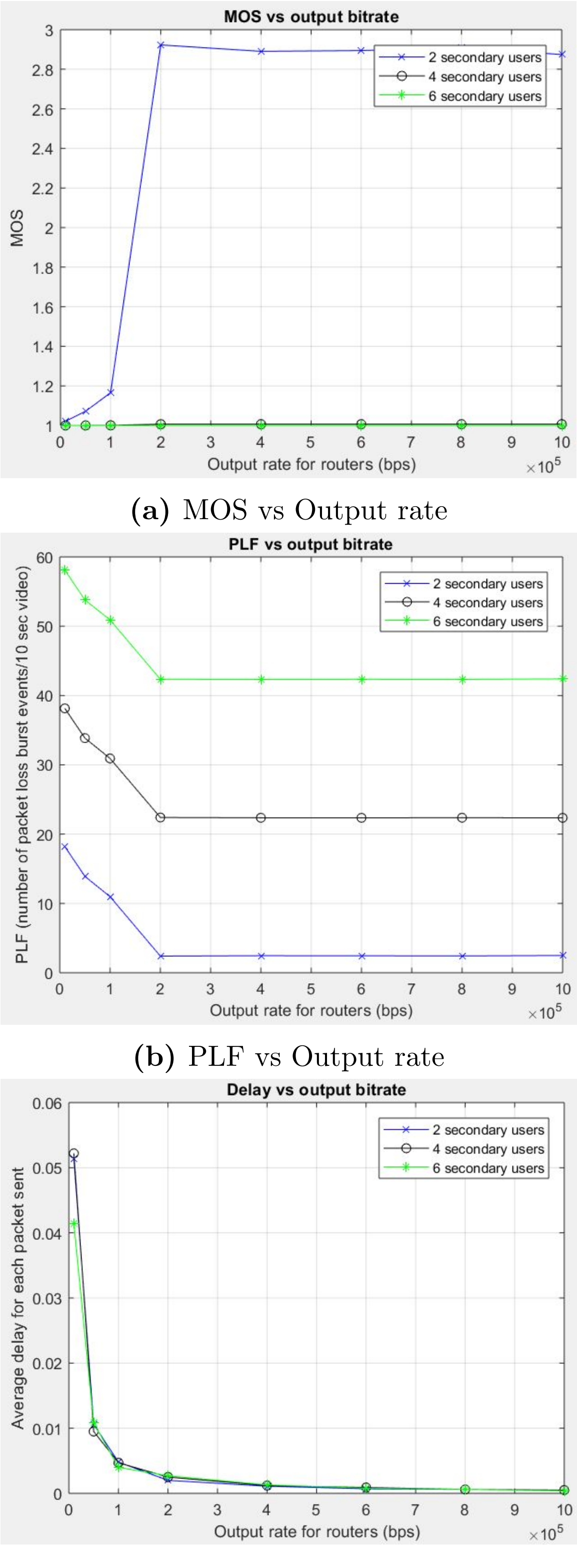
For CRNs performing resource allocation over physical layer only, video quality (QoE) is 4.8 which is a very good quality for the end user. However, this assumes that the end user receives the video directly from the secondary users. In practical cases, there are a number of routers in the network that route the video packets to the appropriate destination. To model this scenario, the output of secondary users is transmitted over a network.

The network comprises of two routers with a fixed queue length and equal output bitrate. The packets are routed to the destination using probabilistic routing strategy. Since the throughput for both the router is same, the probability of each router being selected is 0.5. When IPTV packets are sent over this network, the average MOS of 2.5 was observed which is a poor quality of video from the end user perspective. This can be seen in Fig. 4.1a. These results were obtained by increasing the output rates for the routers in the network. The packet arrival rate for IPTV packets sent over the network is (Average bitrate)/(IPTV packet length). Even when the output rate of router is very high (1Mbps) as compared to the range of packet arrival rates, it is seen that packet loss is still observed for this setup as seen in Fig. 4.1b. Since IPTV data consisting of video is sent over the network, packet loss frequency greater than 4 packets per 10 second video duration also results in degradation of video quality. The low MOS results is due to the static selection of the routers at physical layer for transmission. To overcome these drawbacks, we developed a model to use the routing system environment as a part of DQN learning algorithm.

It can also be seen from these results that as the number of secondary users increases, the packet arrival rate multiplies by the number of users in the system. The network consisting of 2 routers cannot handle this very high packet arrival rate. This results in high packet loss and the MOS drops to around 0. If a larger network with more routers is used, then this scenario of high number of secondary users can be supported.

### 4.3.2 Cross Layer Routing Algorithm

Algorithm 3 was used for resource allocation in the cross layer routing. Fig. 4.2(a) shows changes in MOS for varying the number of secondary users in the secondary network. The network parameters set for these simulation are : output rate of routers = 0.3 Mbps and queue length = 5 packets.



**(a)** MOS vs Output rate

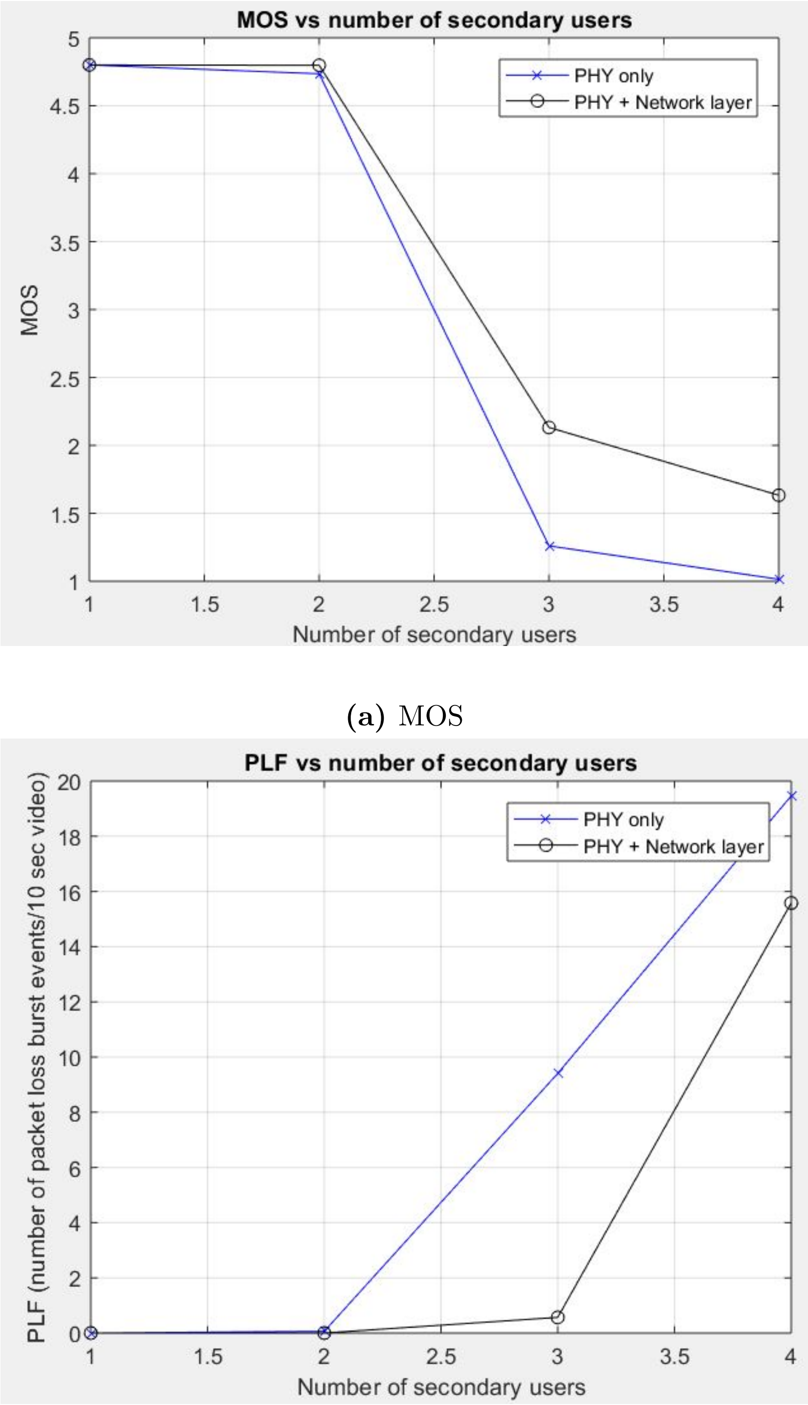
**(b)** PLF vs Output rate

**(c)** Average Delay vs Output rate

**Figure 4.1:** DQN implementation for only PHY layer

As it can be seen in Fig. 4.2 the addition of more than two secondary users result in significant degradation of the quality of video transmitted. This is because of the same reason explained before, two routers in the system cannot handle the high traffic load of multiple video sources (secondary users). With increase in secondary users, the transmission rate multiplies by the number of users actively sending data in the system. This results in $\rho > 1$ for higher number of users, causing an unstable system and congestion. This results in increase in queue lengths causing high packet loss as can be seen in Fig. 4.2(b).

However, even with decrease in MOS due to high influx of video traffic, the cross layer system achieves a better quality than that of single layer CRNs. This is because the cross layer network always tries to route to the node with minimum delay. So, even with queue lengths increasing exponentially, the cross layer network tries to balance the load on both the routers in an attempt to increase MOS.



**(a)** MOS

**(b)** PLF

**Figure 4.2:** Effects of changing number of secondary users

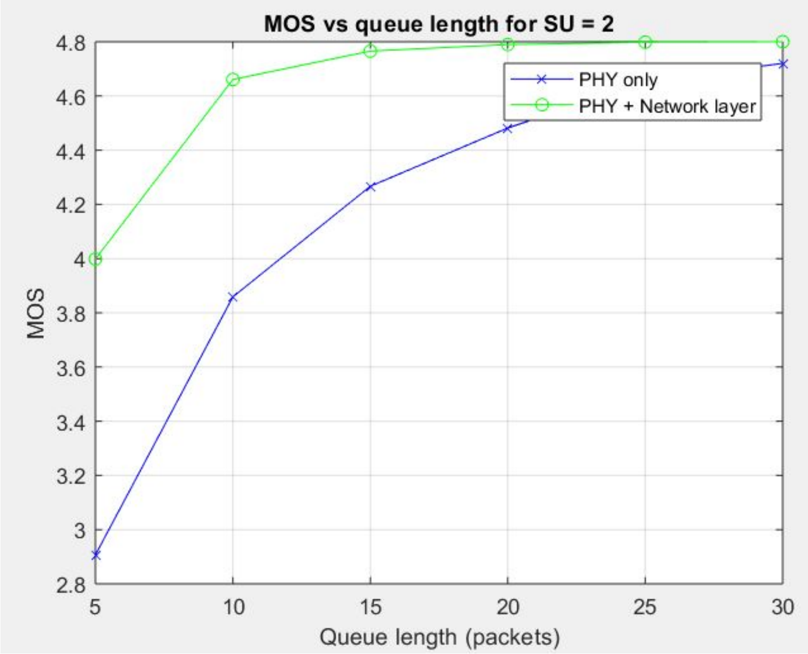
### 4.3.3 Increasing Queue Length

The transmission of packets over the network was compared by varying the queue sizes at the routers. The range of queue size is calculated based on packet arrival rates. Packet arrival rates are given by:

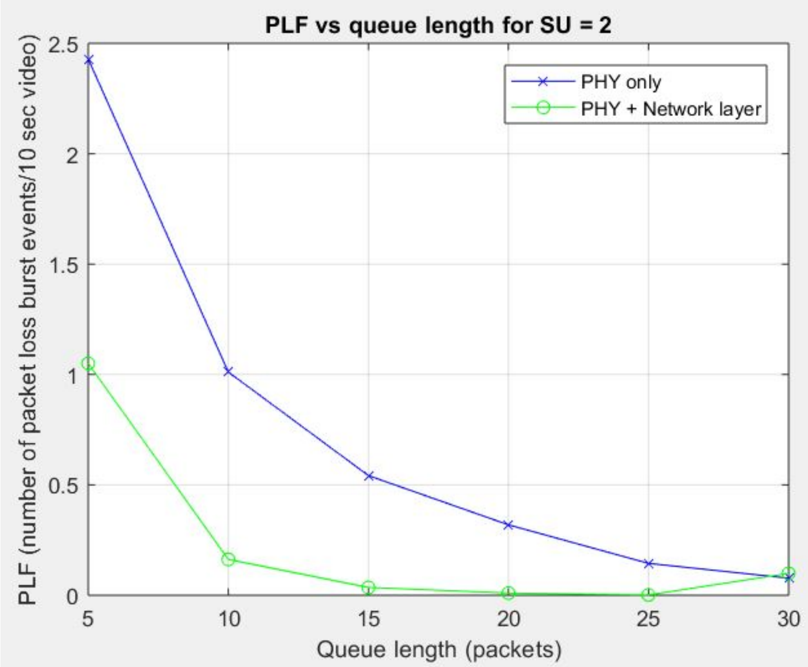
$$P_a = \frac{B_r}{P_l}$$

where $B_r$ is the transmit bitrate and $P_l$ is the IPTV packet length = 10528bits. Minimum transmission rate of the secondary users is 0.1 Mbps. The packet arrival rate for a secondary user transmitting with this bit rate is 9.49 packets/sec. Average transmission rate of secondary users is 0.3 Mbps. The packet rate for this bit rate is 28.49 packets/second. Hence the range of queue lengths, 5 to 30 packets, is used to observe the effect on system parameters.

Fig. 4.3(a) shows 30% more MOS in cross layer routing system for queue size of 5 packets. For higher values of queue size, single layer CRNs and cross layer CRNs have comparable video quality. This can be seen by decrease in packet loss (Fig. 4.3(b)). PLF decreases as more space is available in routers to store incoming packets due to higher queue lengths. This in turn results in better quality.



**(a)** MOS



**(b)** PLF

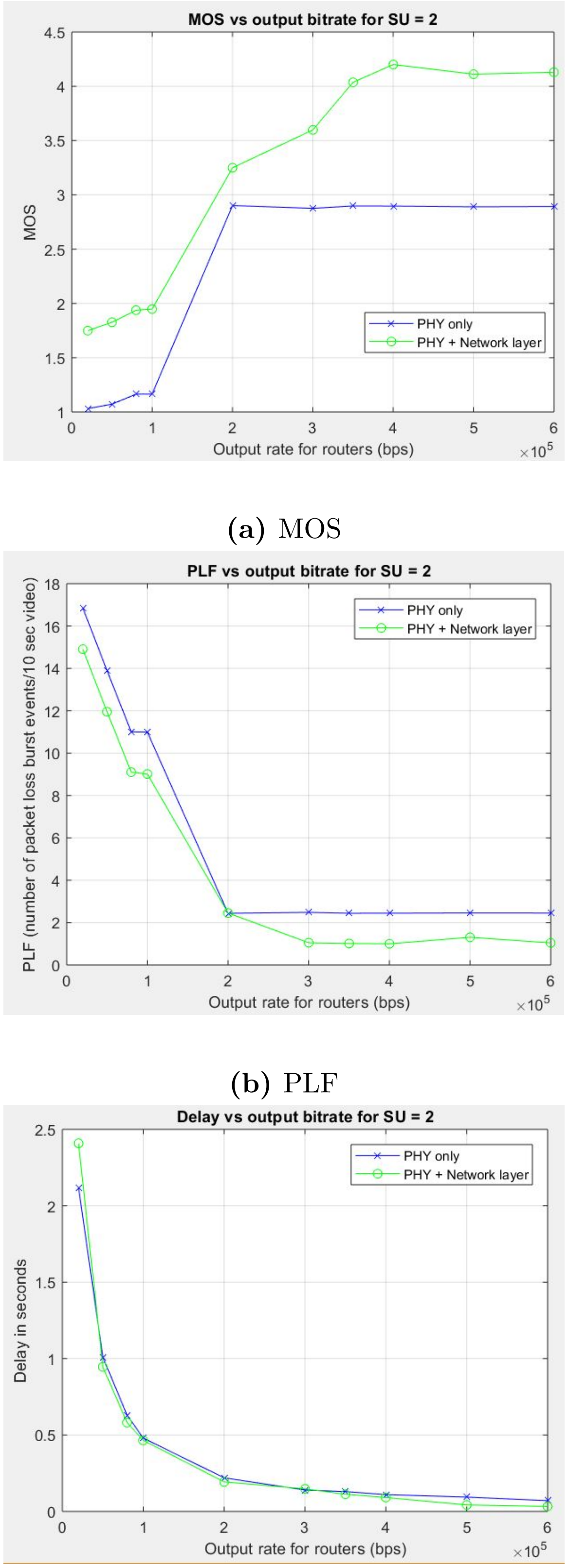
**Figure 4.3:** Effects of changing queue lengths

### 4.3.4 Varying Output Rates of Routers

The transmit rate that can be selected by secondary user ranges between 0.1 to 0.5 Mbps. Hence, to analyze the effect of service rates, values for output bit rate for the routers are selected between 0.02 to 0.6 Mbps. The queue size is set to minimum of 5 packets. Fig. 4.4(a) shows that the average video quality for DQN learning over only the physical layer converges to MOS = 2.9 when output rate of the queues is 0.2 Mbps. This is due to probabilistic determination of next hop for packets and also the transmission rate is optimized for the physical layer parameters before sending the packets on network. Even when the output rate of the queues is increased, these factors give rise to some packet loss as seen in Fig. 4.4(b).

However, for cross layer CRNs considering the network layer parameter (queue delay), MOS is greater than 4. This is because the best route is selected for each packet based on the delays observed in network which minimizes the probability of packet loss. Also the DQN framework for cross layer approach works on optimizing the transmission rate (physical layer parameter) as well as the packet delay (network layer parameter). The result is 30% increase in MOS.

The delays observed in the router are the same for both the systems. Delays are calculated for packets that are sent through the router queue. Since the output rate for routers in both the system are the same, the delay observed are similar as can be seen in Fig. 4.4(c). The average delay for packets in the network system will be different in the two cases compared if the network has different paths to destination and more number of routers.



**(a)** MOS

**(b)** PLF

**(c)** Delay

**Figure 4.4:** Effects of changing output rate for queues with minimum queue length

### 4.3.5 Router Loading

Utilization of a router in this system is given by the fraction of generated packets sent to the router. The utilization for ith router in the network is:

$$U_i = \frac{P_i}{\sum_{i=1}^{N_r} P_i}$$

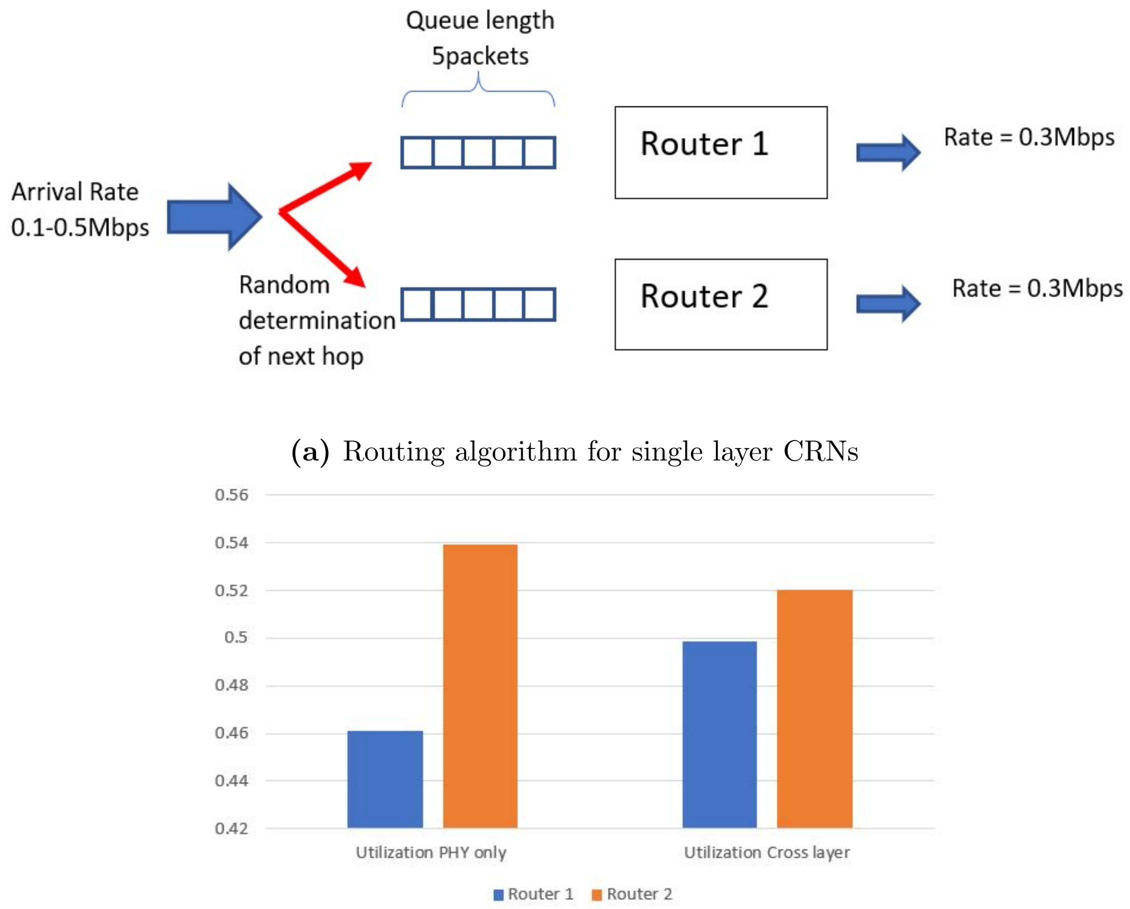
where $P_i$ is the number of packets sent to the $i^{th}$ router and $N_r$ is the total number of routers. Thus, utilization 50% of router 1 indicates that 50% of the generated packets were sent to this router. For calculating the utilization of queue, queue length is set to minimum, i.e. 5 packets.

#### 4.3.5.1 Equal Output Rates at Routers

The output rates at both the routers is set to 0.3 Mbps to calculate utilization of queue. When both the routers have equal service rate or throughput, the routing cost on both the paths to destination node are equal. For CRNs performing resource allocation over only physical layer, we assume that the environmental variables at only the physical layer are observed by the secondary users for routing packets to the destination node. Hence, the probabilistic routing approach is implemented for single layer CRNs. In this approach, the next hop for the packet is selected randomly as the cost to destination is same via both paths with probability of 0.5 for each router. However, cross layer routing system uses DQN to determine next hop for queue based on observed packet delays.

As expected, for random router selection and DQN based router selection, the router utilization is around 50%. This is because both the schemes utilize the routers equally to transmit the data. Fig. 4.5 shows average utilization of the routers for both systems. In system performing physical layer only optimization, utilization of queue varies due to the random selection of next hop for a packet for video transmissions.

However the probability of router being selected as the next hop is around 0.45 to 0.55.



**(a)** Routing algorithm for single layer CRNs

**(b)** Router utilization

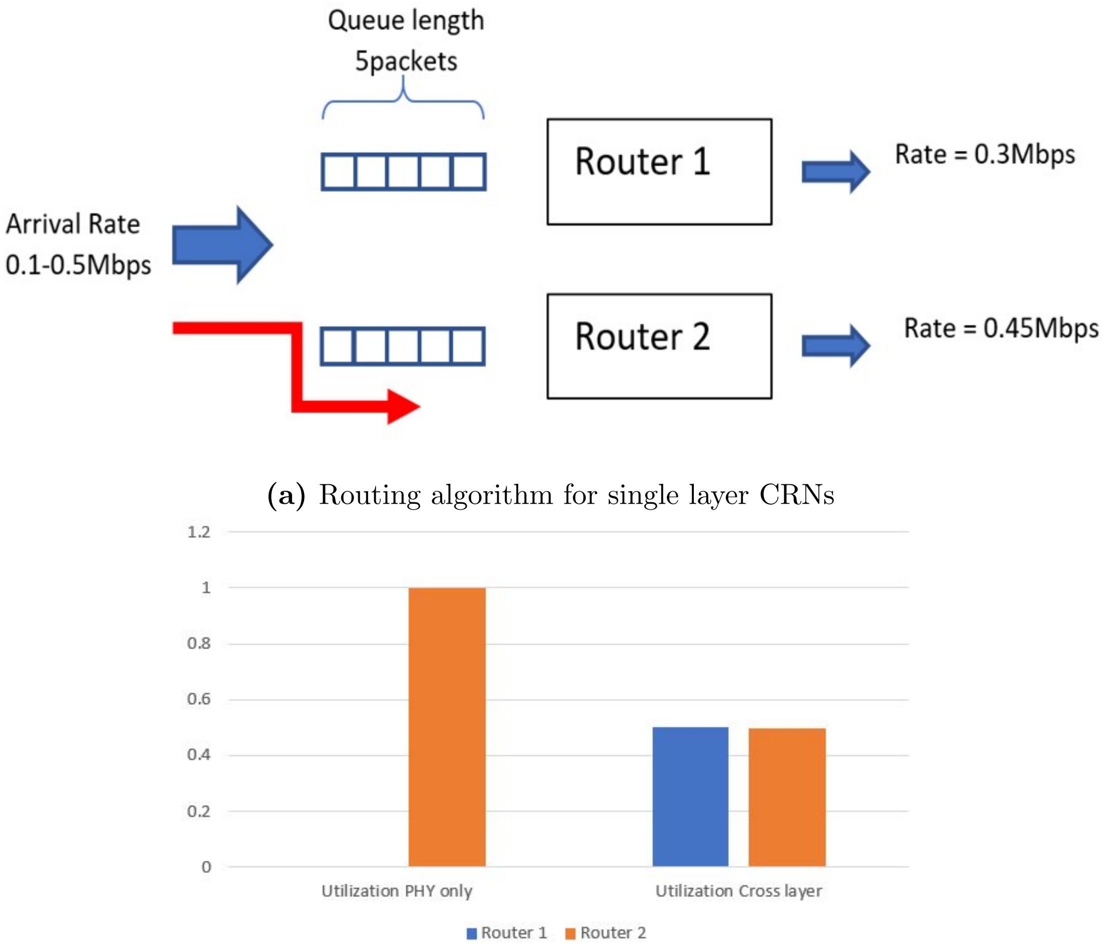
**Figure 4.5:** Equal output rates at the routers

#### 4.3.5.2 Different Output Rates at Routers

The output rate for router 1 is set to 0.3 Mbps while the output rate of router 2 is set to 1.5 times that at router 1 (0.45 Mbps). Queue sizes of both the routers are 5 packets Router selection at physical layer is done by routing through the path with maximum throughput to the destination node. In this case, for the physical layer only approach all the packets get routed to Router 2 giving its utilization of 100%.

Since the physical layer only DQN optimization does not consider delay or queue length, the secondary user routes packets to this router even if they get dropped at the routing node.

The cross-layer system observes delay at the next hop and accordingly decides the action. When queue length of the router with more throughput increases, delay at this router also increases. In this situation the cross layer system routes packets to Router 1. This can be seen from Fig. 4.6 where the utilization for the cross layer system is equal for both the routers, thus, showing that the packets are equally routed to both routers to prevent congestion at any one of them.



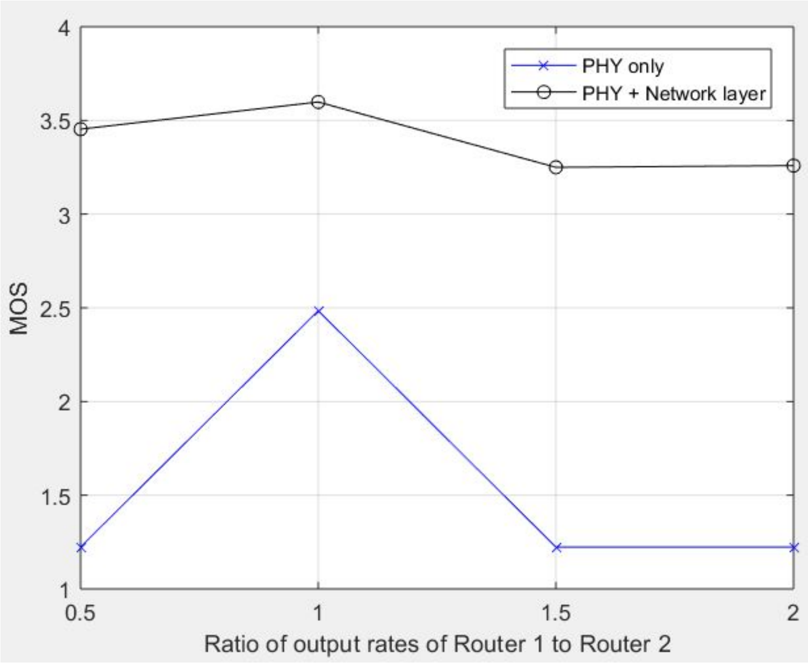
**(a)** Routing algorithm for single layer CRNs

**(b)** Router utilization

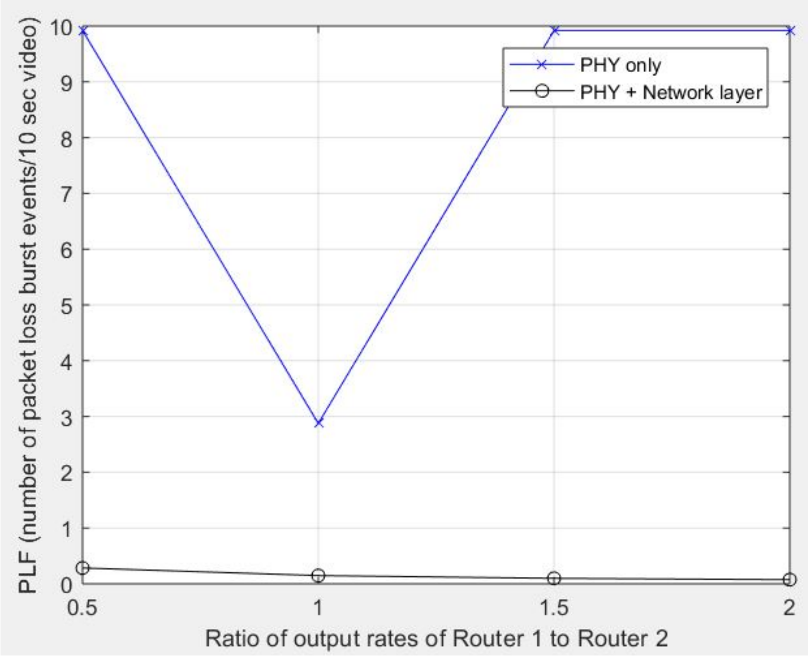
**Figure 4.6:** Unequal output rates at the routers

The effect of selecting different multiplication factor for output bit rate for router 2 is as shown in Fig. 4.7. Router 1 maintains an output rate of 0.3Mbps. The ratio of output rates between the two routers is set to [0.5, 1, 1.5, 2]. As can be seen, the performance metric for cross layer system are approximately invariant for different values of output rate of Router 2. This confirms the equal utilization of routers even when they have different output rates.

For optimizations on physical layer only, the system has better MOS and less packet loss only when both routers have equal output rates, i.e. multiplication factor for Router 2 is 1. Due to equal throughput on both routers, next hop is randomly selected which results in less packet loss than when only one router with more throughput is selected.



**(a)** MOS



**(b)** PLF

**Figure 4.7:** Changing ratio of output rates between routers

# Chapter 5

## Conclusion

A cross layer routing system developed in this work performs resource allocation by observing the environmental variables over physical and network layer and takes actions to update its own parameters across these layers in order to maximize QoE for video transmissions. For smaller queue lengths at the intermediate routers, the cross-layer system proves to be efficient by achieving MOS 1.4 times that of CRNs over physical layer. Thus, time sensitive information such as video or live streaming can be performed without much impairment over the secondary network with a cross-layer DQN resource allocation.

Also this system performs efficient utilization of the routers by performing load balancing based on observed packet delays. For a system of routers with different throughput, this system performs better by maintaining a constant level of video quality. For this configuration, the system outperformed the CRNs for physical layer only. The increase in MOS for cross layer routing is around 62%. This proves that the system can be extended to a system of multiple routers with varying queue lengths and service rates and the performace will not be effected.

# Chapter 6

## Future Work

Transfer learning for secondary users provides improved learning process. This was implemented for Q-learning algorithm in [22]. The idea can be further extended to transfer information through sharing the routing tables for a new secondary user joining the network to speed up the learning process. With current simulation setup, each user sends the video packets over the environment during the learning phase to update action values for next hop. This adds up in simulation time. Transfer learning approach will be useful to reduce this overhead.

The cross layer routing system can be used for more number of routers to be more closer to practical routing systems. Such systems might also handle more number of secondary user transmissions more effectively by finding routing paths with minimum delays. Graph theory has been proven to be useful in some CRNs implementations. In [31] routing is implemented for overlay spectrum sharing scheme using game theory.

The cross layer approach implemented in this routing scheme for CRNs can be extended for multiple layers of OSI communication model. For additional action set values, more hidden layers need to be connected for the MLP neural network.

Different aspects of cognitive radio design can be studied with the ease of increasing action space in DQN. Energy efficient communication (Green communication) is one such feature that can be optimized using DQN for resource allocation considering spectral as well as energy efficiency metrics. In [32], an optimal power allocation scheme that maximizes the instantaneous energy efficiency metrics is proposed. Lagrangian method is used for obtaining this power allocation. This work showed that the effect of the interference constraint in underlay CRNs is minimal on the energy efficiency metrics. The DQN-based reinforcement learning can be used by the secondary users to optimize power to achieve energy-efficient communication over secondary network.

# Bibliography

[1] N. Abbas, Y. Nasser, and K. E. Ahmad, "Recent advances on artificial intelligence and learning techniques in cognitive radio networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, p. 174, 2015.

[2] E. Zanini. Markov decision processes. [Online]. Available: http://www.lancaster.ac.uk/postgrad/zaninie/MDP.pdf

[3] K. Yamagishi and T. Hayashi, "Parametric packet-layer model for monitoring video quality of iptv services," in *2008 IEEE International Conference on Communications*, 2008.

[4] B. Moon, "Dynamic spectrum access for internet of things service in cognitive radio-enabled lpwans," *Sensors*, vol. 17, no. 12, 2017.

[5] D. Das and S. Das, "A survey on spectrum occupancy measurement for cognitive radio," *Wireless Personal Communications*, vol. 85, no. 4, pp. 2581–2598, 2015.

[6] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[7] A. Garhwal and P. P. Bhattacharya, "A survey on dynamic spectrum access techniques for cognitive radio," *CoRR*, vol. abs/1201.1964, 2012.

[8] Y. Chen, K. Wu, and Q. Zhang, "From qos to qoe: A tutorial on video quality assessment," *IEEE Communications Surveys Tutorials*, vol. 17, no. 2, pp. 1126–1165, 2015.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533H, 2015.

[10] J. D. Day and H. Zimmermann, "The osi reference model," *Proceedings of the IEEE*, vol. 71, no. 12, 1983.

[11] M. Song, C. Xin, Y. Zhao, and X. Cheng, "Dynamic spectrum access: from cognitive radio to network radio," *IEEE Wireless Communications*, vol. 19, no. 1, 2012.

[12] C. R. Stevenson, G. Chouinard, Z. Lei, W. Hu, S. J. Shellhammer, and W. Caldwell, "Ieee 802.22: The first cognitive radio wireless regional area network standard," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 130–138, 2009.

[13] A. V. Kordali and P. G. Cottis, “A reinforcement-learning based cognitive scheme for opportunistic spectrum access,” *Wireless Personal Communications*, vol. 86, no. 2, pp. 751–769, 2016.

[14] H. Mohammed, A. Ahmad, A. Ala, and A. Nidal, “Distributed opportunistic spectrum sharing in cognitive radio networks,” *International Journal of Communication Systems*, vol. 30, no. 7, p. e3147.

[15] K. Bhoopendra, K. D. Sanjay, and W. Isaac, “A survey of overlay and underlay paradigms in cognitive radio networks,” *International Journal of Communication Systems*, vol. 31, no. 2, 2018.

[16] W. Wang, A. Kwasinski, D. Niyato, and Z. Han, “A survey on applications of model-free strategy learning in cognitive wireless networks,” *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 1717–1757, 2016.

[17] J. Milani Fard, M; Pineau, “Non-deterministic policies in markovian decision processes,” vol. 40, pp. 1–24, 2011.

[18] K.-L. A. Yau, G.-S. Poh, S. F. Chien, and H. A. A. Al-Rawi, “Application of reinforcement learning in cognitive radio networks: Models and algorithms,” *The Scientific World Journal*, vol. 2014, pp. 1–23, 2014.

[19] W. Wang and A. Kwasinski, “Experience cooperative sharing in cross-layer cognitive radio for real-time multimedia communication,” in *Proceedings of the 4th International Conference on Cognitive Radio and Advanced Spectrum Management*, ser. CogART ’11. Barcelona, Spain: ACM, 2011.

[20] P. Beyens, M. Peeters, K. Steenhaut, and A. Nowe, “Routing with Compression in Wireless Sensor Networks: a Q-learning Appoach,” in *Proceedings of the 5th European Workshop on Adaptive Agents and Multi-Agent Systems (AAMAS)*, 2005.

[21] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, “Multiagent cooperation and competition with deep reinforcement learning,” *PLoS One*, vol. 12, no. 4, 2017.

[22] F. Shah-Mohammadi and A. Kwasinski.

[23] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT press Cambridge, 2016.

[24] G. Dulac-Arnold, R. Evans, P. Sunehag, and B. Coppin, “Deep reinforcement learning in large discrete action spaces,” *arXiv preprint arXiv:1512.07679*, 2015.

[25] L. Ding, T. Melodia, S. N. Batalama, J. D. Matyjas, and M. J. Medley, “Cross-layer routing and dynamic spectrum allocation in cognitive radio ad hoc networks,” *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 1969–1979, 2010.

[26] L. Ding, T. Melodia, S. Batalama, and M. J. Medley, “Rosa: Distributed joint routing and dynamic spectrum allocation in cognitive radio ad hoc networks,” ser. MSWiM ’09. Proceedings of the 12th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2009.

[27] L. Ding, T. Melodia, S. N. Batalama, and J. D. Matyjas, “Distributed routing, relay selection, and spectrum allocation in cognitive and cooperative ad hoc networks,” in *2010 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2010.

[28] V. Esmaeelzadeh, E. S. Hosseini, R. Berangi, and O. B. Akan, “Modeling of rate-based congestion control schemes in cognitive radio sensor networks,” *Ad Hoc Networks*, vol. 36, pp. 177–188, 2016.

[29] X. Qiu and K. Chawla, “On the performance of adaptive modulation in cellular systems,” *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 884–895, 1999.

[30] G. O’Driscoll, *Next Generation IPTV Services and Technologies.* John Wiley & Sons, Incorporated, 2008.

[31] Z. Yuan, J. B. Song, and Z. Han, “Interference aware routing using network formation game in cognitive radio mesh networks,” *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 11, pp. 2494–2503, 2013.

[32] L. Sboui, Z. Rezki, and M. Alouini, “Energy-efficient power allocation for underlay cognitive radio systems,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 1, no. 3, pp. 273–283, 2015.

# Acronyms

**AMC** Adaptive Modulation and Coding

**AWGN** Additive White Gaussian Noise

**BER** Bit Error Rate

**CRNs** Cognitive Radio Networks

**DQN** Deep Q-networks

**DSA** Dynamic Spectrum Access

**IOT** Internet of Things

**IPTV** Internet Protocol Television

**MDP** Markov Decision Process

**MLP** Multi Layer Perceptron

**MOS** Mean Opinion Score

**NN** Neural Network

**OFDMA** Orthogonal frequency-division multiple access

**OSA** Opportunistic Spectrum Access

**OSI** Open System Interconnection

**PLF** Packet Loss Frequency

**QoE** Quality of Experience

**QoS** Quality of Service

**SINR** Signal to Interference plus Noise Ratio