ABSTRACT

Deep Learning for Energy-efficient Wireless Communications and Spectrum Management

Yuan Xing, Ph.D.

Mentor: Liang Dong, Ph.D.

In the past couple of decades, wireless communication has undergone rapid development. The current fourth generation and upcoming fifth generation wireless technologies promise us an ultra-fast data rate. However, a lot of energy is sacrificed in order to guarantee high quality communication. Therefore, energy-efficient wireless communication has been widely explored under the background of scarce energy resource and environmental-friendly data transmission.

In order to assure the fast communication speed and network reliability, the structures of wireless communication systems become more and more complicated. The rational resource allocation in the sophisticated communication systems is a tough problem that urgently needs to be solved. The conventional communication theories exhibit limitations in fulfilling the perfect resource allocation in the systems. Nevertheless, Deep Learning methods are expert at solving sophisticated optimization problems. The key advantages of Deep Learning are the efficient learning of an enormous amount of data and the precise analysis for the hidden distribution. Therefore, Deep

Learning methods can be used to solve complicated but useful energy efficiency optimization problems in wireless communication systems.

This dissertation first explores an energy-efficient wireless communication system: the Simultaneous Wireless Information and Power Transfer system. Specifically, the wireless transmitters stably communicate with the information receivers, while there are several energy harvesters. The harvesters can take the electromagnetic waves as the energy source in order to charge the low power Internet of things. Deep Learning algorithms are utilized to optimize the wireless information and power transfer strategies.

Second, this dissertation discusses another energy-efficient wireless communication system: a multiuser downlink Orthogonal Frequency Division Multiple Access data transmission system. In the system, the base station aims to achieve the highest communication quality with the least energy consumption. Deep Learning algorithms are applied to accomplish the energy-efficient wireless transmission.

In summary, this dissertation investigates the usefulness of Deep Learning algorithms to boost the performance of two energy-efficient wireless communication systems, the Simultaneous Wireless Information and Power Transfer system and multiuser downlink Orthogonal Frequency Division Multiple Access data transmission system. The numerical results prove the excellence of Deep Learning methods in solving the optimization problems in energy-efficient wireless communication systems.

Deep Learning for Energy-efficient Wireless Communications and Spectrum Management

by

Yuan Xing, B.Eng, M.S.E.E.

A Dissertation

Approved by the Department of Electrical and Computer Engineering

_____

Kwang Lee, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of

Doctor of Philosophy

Approved by the Dissertation Committee

_____

Liang Dong, Ph.D., Chairperson

_____

Robert Marks, Ph.D.

_____

Scott Koziol, Ph.D.

_____

Enrique Blair, Ph.D.

_____

Greg Hamerly, Ph.D.

Accepted by the Graduate School

May 2020

_____

J. Larry Lyon, Ph.D., Dean

Page bearing signatures is kept on file in the Graduate School.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I would like to express my great gratitude to my advisor Professor Liang Dong for his patience, encouragement, and immense knowledge. His guidance motivates and helps me do the research and write this dissertation.

I would also like to thank the other members in my dissertation committee: Professor Robert Marks, Professor Scott Koziol, Professor Enrique Blair and Professor Greg Hamerly for their insightful comments and constructive suggestions on my dissertation.

I would also like to express my deepest appreciation and love to my parents Mr. Zhixing Xing and Mrs. Lan Zhao and my wife Dr. Dongfang Hou. They are my strongest supporters.

My sincere thanks also goes to my good friend Mr. Yuchen Qian. I am particularly grateful to him for his timely help.

Finally, I would like to express my special thanks to my good friends: Professor Cristiano Tapparello, Mr. Tianchi Zhao, Mr. Haowen Pan, Mr. Peizhong Cong, Mr. Shengyang Zhou and my furry friend Ozzy. They made me who I am today.

CHAPTER ONE

Introduction

The fifth generation (5G) wireless communication system is designed to satisfy the huge demands for high speed and low latency communication. However, the traditional communication theories have difficulties in achieving satisfactory system performance for several reasons. First, in many complicated communication systems, it is difficult to conduct a real-time channel estimation. In all communication systems, the signals are transmitted through a medium, which is called the channel. The distortion or noise in the channel is added to the signals. Imprecise channel estimation dramatically degrades the system performance since the design of the traditional communication systems relies heavily on the channel conditions. Second, the existing communication systems show deficiency in achieving low latency communication. Since a large number of traditional communication methods can only operate iteratively, it takes a long time for the base stations to make the transmission decision [1]. In order to deal with these difficulties in the real communication systems, Machine Learning (ML), especially Deep Learning (DL) methods have been widely applied to solve practical communication problems [2, 3].

In the DL process, the Deep Neural Network (DNN) is proven to be a universal function approximator [4]. No matter how complicated the optimization problems are, the learned operating rules can be represented by the tuned weights in the neural network (NN). Besides, once the NN is well trained and utilized in the communication systems, NN can

immediately determine a transmission strategy for the transmitter, which largely increases the signal processing speed and realizes the low latency communication.

Moreover, some complicated communication problems are formulated as the long-term optimization problems, such as the multiuser Orthogonal Frequency Division Multiple Access (OFDMA) data packet transmission problem and the everlasting wireless charging problem. The problems are analyzed in a period of time. Whether the optimization goal is accomplished depends on each action that is taken within the time period. In order to solve such global optimization problems, the transmitter has to dynamically alter its transmission strategy in accommodating to the channel variations. However, the hardware limitations are the main obstacles to carry out the effective real-time channel estimation [5, 6]. Without the channel distribution information, the conventional communication methods cannot be implemented dynamically. Nevertheless, Deep Reinforcement Learning (DRL) is skilled at solving the global optimization problems with limited environmental information [7]. Reinforcement learning (RL) is a good method in achieving long-term benefits which are not afforded by traditional approaches. Nonetheless, RL shows bad performance in complex decision-making tasks. For this reason, DRL, which has the architecture of DNN with RL algorithms, was invented. Focusing on long-term reward, DRL can optimize the transmission strategies in a complicated system even without complete channel information. More recently, DRL has been applied to deal with complex communication problems and has shown to achieve good performance [3]. Multi-Armed Bandit (MAB) is another important model for studying the exploration and exploitation tradeoff in RL. MAB is good at maximizing the expected system performance without the

full knowledge of the environment. Recently, MAB has been used to deal with wireless communication issues and achieved excellent system performance [8].

The summary of the above-mentioned algorithms is presented in Fig. 1.1. Given all its advantages, DL is a good method in solving the optimization problems in energy-efficient wireless communication systems.

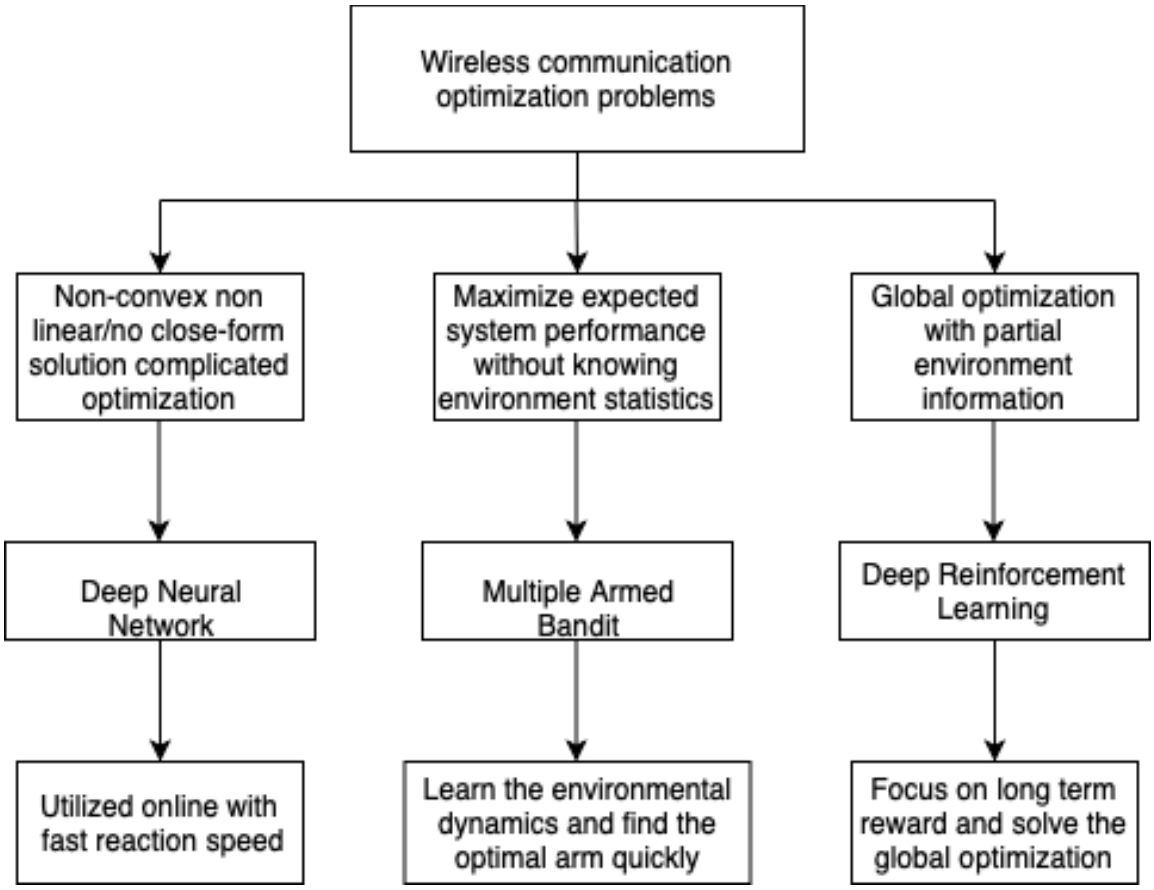

Figure 1.1: Artificial Intelligence algorithms in solving wireless communication optimization problems.

Energy-efficient communication attracts more and more attentions from both academia and industry [9, 10]. 5G communication systems are able to deliver up to one hundred times as much data as the fourth generation (4G) communication systems.

However, 5G communication systems consumes up to one hundred times as much energy as 4G communication systems at a cost. In order to reduce the energy consumption and greenhouse gas emission, energy saving communication infrastructures are urgently needed. The research in this dissertation focuses on improving the system performance in two energy-efficient wireless communication systems: Simultaneous Wireless Information and Power Transfer (SWIPT) system and multiuser downlink OFDMA data transmission system.

Simultaneous Wireless Information and Power Transfer

Radio-frequency (RF) energy harvesting is a promising and feasible technology in 5G communication systems [11]. The energy harvesters are able to harvest energy from RF signals. The structure of the SWIPT system is shown in Fig. 1.2. The multi-antenna wireless transmitter communicates with information receivers while delivering the electromagnetic waves to each energy harvester as the energy supply. The harvested energy can help the Internet of Things (IoT) devices prolong their battery life, which can effectively improve the system energy efficiency. With the knowledge of the channel information, the transmitter can adjust its transmission strategy to boost the energy harvesting rate at the harvesters. However, the estimation of channel at the harvesters is difficult due to the hardware limitations [6]. Moreover, the SWIPT system not only radiates the energy to the energy harvesters, but delivers the information to the information receiver. It is difficult to determine the optimal transmission strategy that coordinates both the wireless power transfer and the wireless information transfer. Given these obstacles, a DNN is used to solve the optimization problem in the SWIPT system. To avoid high computational complexity, the optimal power allocation is acquired by a DNN instead of

solving a convex optimization problem. The simplified channel vectors and the information rate requirement are the input to the DNN. The DNN is trained offline with a large amount of simulated data. When the channels experience block fading, the K-means clustering algorithm is applied to classify the channels into several classes. For each class, a DNN is trained. The transmitter determines what class the channel belongs to and uses the DNN to find the optimal transmission strategy.

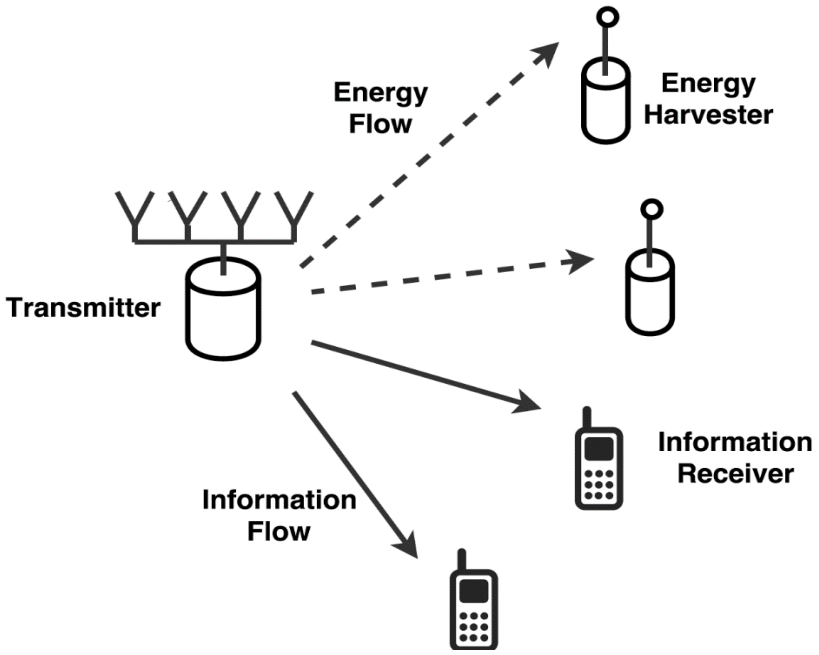

Figure 1.2: Simulataneous Wireless Information and Power Transfer system

Next, the wireless power transfer is formulated through a continuous charging process. By adapting its transmission strategy, the transmitter fully charges the energy buffers of all energy harvesters in the shortest amount of time while maintaining the target information rate toward the receiver. At the beginning of each time slot, the transmitter determines the particular transmission strategy to transmit with. Throughout the whole charging process, the transmitter doesn't estimate the channel condition from the

transmitter to any energy harvester. Due to the high complexity of the system, the goal of the work is to apply a Deep Q-Network (DQN) algorithm to dynamically determine the optimal transmission strategy at the transmitter.

After that, one transmitter system is upgraded to multiple transmitters case. Multiple wireless transmitters communicate with their intended information receivers while radiating RF energy to multiple nearby energy harvesters. The channels from the transmitters to the receivers and to the energy harvesters are time-varying and unknown. The transmitters jointly determine their multi-antenna transmissions to fairly charge all the energy harvesters while maintaining the signal-to-interference-and-noise ratio (SINR) at the receivers. This task is formulated as a Combinatorial Multi-Armed Bandit (CMAB) problem and is solved with the Upper Confidence Bound (UCB) algorithm. Numerical results show that the algorithm can quickly converge to the optimal strategy with moderate signaling overhead. The UCB algorithm has superior performance in fair energy harvesting while maintaining communication quality.

Multiuser OFDMA Energy-efficient Wireless Transmission

As a promising multi-access technique, OFDMA is applied to many broadband wireless communication systems. Multiple access is achieved in OFDMA by spectrum management. Once OFDMA is adopted by multiple users, a good spectrum management policy can sufficiently exploit multiple users' diversity in order to enhance the overall system performance [12]. An OFDMA operating procedure is shown in Fig. 1.3. In order to reduce the energy consumption in communication system and green gas emission, energy-efficient transmission attracts much attentions. Energy efficiency is defined as the number of bits that can be sent over a unit of energy consumption, which is an effective

metric to evaluate the efficiency of the energy consumption in wireless communication [13]. Energy-efficient transmission aims at achieving the highest communication quality with the least energy consumption. The energy efficiency problems in OFDMA wireless communication systems have recently been discussed. In [14], the authors maximized the energy efficiency of the worst-case communication link under the information rate, total transmit power and available subcarrier constraints.

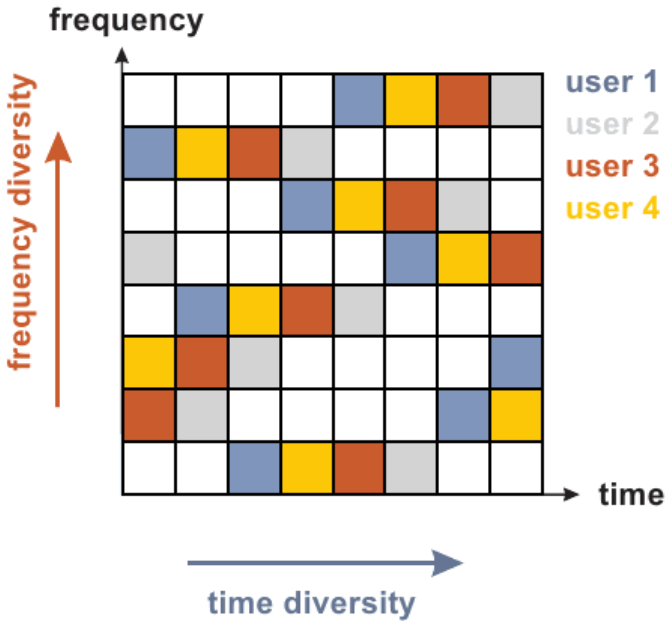

Figure 1.3: OFDMA operating mechanism

In this dissertation, a multiuser downlink OFDMA wireless transmission system is constructed. The base station communicates with multiple mobile users. The base station intends to consume the least energy to guarantee the highest Quality of Service (QoS). The spectrum management strategies are adjusted in order to maximize the total energy efficiency while maintaining the communication quality.

In OFDMA system, adaptively performing resource allocation (spectrum management) on each subcarrier can significantly improve the energy efficiency of the

system. However, it is not easy to acquire the optimal subcarrier assignment and power allocation strategies in solving the complex energy efficiency optimization problem. Iterative algorithm only leads to a sub-optimal solution [14]. Due to the non-convexity of the optimization, two individual DNNs are trained to solve the optimization problem. With the channel gain as the input, two DNNs can output the optimal power allocation and subchannel assignment, respectively. A Refined Exhaustive Search algorithm is invented in order to generate the training data. The DNNs are trained with simulated data offline but can be utilized online for an immediate reaction. The simulation results prove that the DNNs achieve good system performance with extremely fast reaction speed.

After that, the energy-efficient transmission is formulated as a continuous process. The dynamic subchannel assignment and power allocation strategies are optimized in order to maximize the total energy efficiency while delivering the information payload to each mobile user within the time budget. The optimization is formulated as a long-term optimization problem, in which the optimization target is related to the real-time resource allocation strategies. A Deep Deterministic Policy Gradient (DDPG) framework is utilized to solve the optimization problem. With proper design of the system state and the reward function, the resource allocation policy can be determined for multiple users' energy-efficient transmission. As the number of available subchannels increases, high dimensional action increases the difficulty of implementing DDPG algorithm. Therefore, a hybrid algorithm is invented: a DDPG is applied to determine the power allocation strategy, while a heuristic approach is utilized to assign the subchannels to multiple users. The simulation results prove the excellent performance of the invented hybrid approach.

Dissertation Overview

In Chapter One, the growing demand of energy-efficient wireless communication systems is analyzed. In order to achieve rational resource allocation in the energy-efficient wireless communication systems, the powerful DL methods are strongly recommended. The motivation of applying DL methods in solving the complicated optimization problems is discussed. Two practical energy-efficient wireless communication systems are established. They are SWIPT system and multiuser downlink OFDMA data transmission system, respectively. The background of designing two wireless systems are provided.

The specific system models are discussed from Chapter Two to Chapter Six. The relationship diagram of the system models are shown in Fig. 1.4.

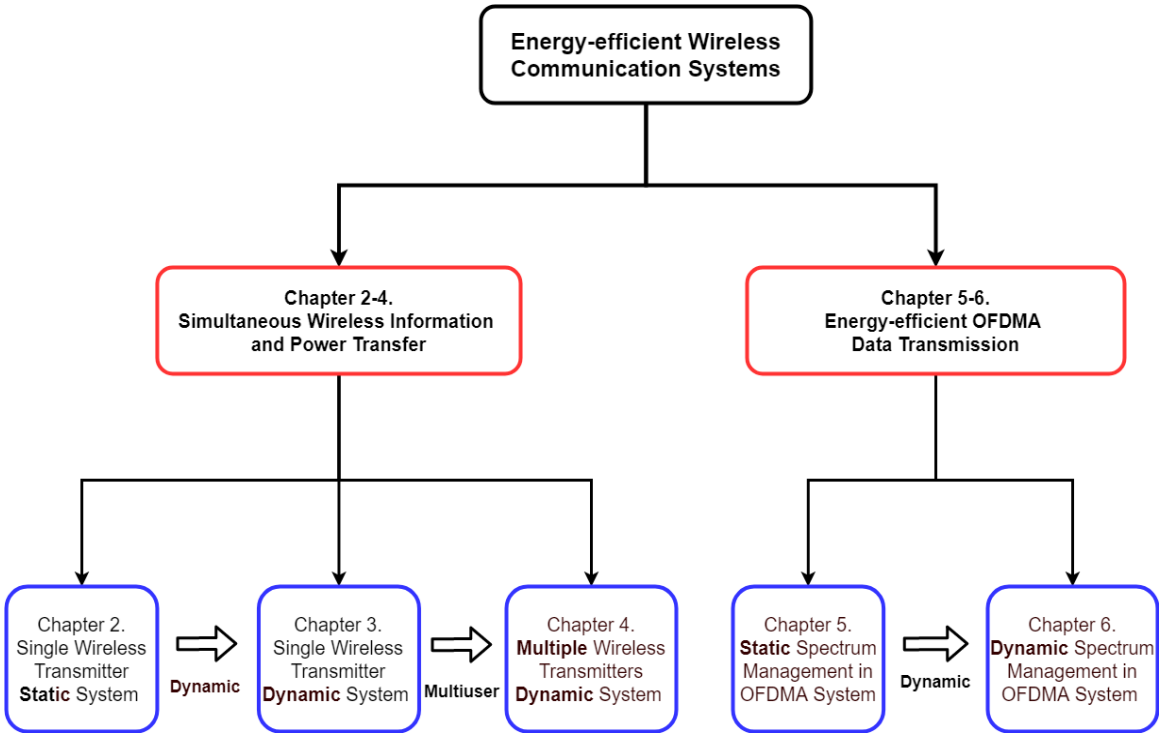

Figure 1.4: Relationship diagram of the system models

In Chapter Two, the mathematical model of SWIPT system is formulated. The single wireless transmitter static system model is considered. The base station aims at fairly charging multiple energy harvesters while maintaining the communication QoS. In order to reduce the computational complexity, the DNN is applied to solve the problem. Chapter Two is published as: [15] and [16]. In [15], my contributions included system modeling, mathematical derivation, and coding. Dr. Liang Dong contributed to simulation verification. In [16], my contributions included system modeling, mathematical derivation, and coding. Dr. Liang Dong and Mr. Yuchen Qian contributed to simulation verification.

In Chapter Three, the system model is extended from the single transmitter static system to dynamic system, which indicates the channel condition is time-variant. The long-term optimization problem is formulated in the SWIPT system. A DQN is trained to solve the dynamic power allocation problem in a continuous wireless charging process. DQN algorithm shows the preeminence in determining the real-time transmission strategy for complex systems.

In Chapter Four, the single communication pair in SWIPT system is extended to multiple communication transceivers. The transmitters jointly determine their multi-antenna transmissions to fairly charge all the energy harvesters while maintaining the required SINR at the receivers. This task is formulated as a CMAB problem and is solved with the UCB algorithm. Chapter Four is published as: [18]. My contributions included system modeling, mathematical derivation, and coding. Dr. Liang Dong and Mr. Yuchen Qian contributed to simulation verification.

In Chapter Five, the mathematical model of multiuser downlink OFDMA data transmission system is formulated. The spectrum management is implemented in OFDMA

system. In the system, the channel is invariant over the time. The subchannel assignment and power allocation strategies are optimized in order to maximize the total energy efficiency while maintaining the information rate from the base station to each mobile user. A Refined Exhaustive Search algorithm is invented to generate the training data and two DNNs are trained to determine the power allocation and subchannel assignment.

In Chapter Six, the channel environment is considered to be time-variant and the dynamic subchannel assignment and power allocation strategies are optimized in order to maximize the total energy efficiency while successfully delivering the information payload to each user within the time budget. A DDPG framework is utilized to determine the resource allocation policy for multiple users' energy-efficient transmission. As the number of available subchannels increases, high dimensional action increases the difficulty of training the NNs in DDPG algorithm. Therefore, a hybrid algorithm is invented in solving the optimization problem.

Chapter Seven summarizes the dissertation and proposes the future research directions.

All research in this dissertation has been published or submitted for publication. The modeling of SWIPT system is published in [15]. The application of DNN in solving SWIPT optimization problem is published in [16]. The DQN framework in solving continous charging problem is presented in [17]. The MAB approach in solving multiuser SWIPT problem is published in [18]. The DNN algorithm in solving multiuser downlink OFDMA optmization problem is presented in [19]. The DDPG algorithm in solving dynamic spectrum management problem in OFDMA system is presented in [20, 21].

CHAPTER TWO

Deep Learning for Optimized Wireless Information and Power Transfer

This chapter published as: Y. Xing and L. Dong, "Passive radio-frequency energy harvesting through wireless information transmission", in *Pro. of IEEE DCOSS*, Jun. 2017, pp. 73-80.

Y. Xing, Y.Qian and L. Dong, "Deep learning for optimized wireless transmission to multiple rf energy harvesters", in *Pro. of IEEE VTC Fall*, 2018.

Introduction

Energy harvesting is essential in green wireless communications and networks because it can alleviate the problem of limited battery life that restricts the massive deployment of small wireless devices. Passive RF energy harvesting collects the radiated energy from adjacent wireless information transmitters instead of using a dedicated wireless power source. For a wireless transmitter-receiver pair with multiple antennas, adjusting the transmit signal covariance matrix can provide high data-rate communication over the multiple-input multiple-output (MIMO) channel. Meanwhile, the radiated RF energy can be harvested by the surrounding RF energy harvesters.

In practice, it is difficult for the transmitter to acquire the knowledge of the channels to the RF energy harvesters. The random locations of the energy harvesters and the hardware restriction make the channel estimation challenging [6]. The analytic center cutting plane method (ACCPM) was proposed for the transmitter to iteratively approach the channel with a few bits of feedback from the RF energy receiver [6]. The method is implemented by solving a convex optimization problem, which incurs high computational

complexity. To reduce complexity, Kalman filtering was proposed to implement the channel estimation. However, the convergence is slow.

To manage the RF energy harvesting network, either time-splitting or weight-splitting beamforming strategies can be used. Time-splitting beamforming focuses on charging one energy harvester at a time. Weight-splitting beamforming splits the microwave beam toward multiple energy harvesters simultaneously [22]. The weight-splitting method transmits with a fixed beam pattern and outperforms the time-splitting method with less computational complexity. In this work, treating each RF energy harvester consistently and fairly, the weight-splitting beamforming method is adopted to transmit with a beam pattern that maximizes the minimum harvested energy among the multiple energy harvesters [23].

In the proposed model, the matrix channel between the communication pair is assumed to be known to the transmitter, whereas the vector channel to the RF energy harvester is unknown. Parameters of a simplified channel vector can be estimated through particular transmissions and very limited feedback. Once the transmitter obtains the simplified channel vector, it can find the optimal transmit covariance matrix using optimization methods, e.g., the interior point method.

To avoid high computational complexity, a novel method is proposed to find the optimal power allocation with a DNN instead of solving a convex optimization problem. The simplified channel vectors and the information rate requirement are the input to the DNN. The DNN is trained offline with a large number of simulated data.

When the channels experience block fading, the K-means clustering algorithm is applied to classify each of the channels into one of several classes. For each class, a DNN

is trained. The transmitter determines what class the channel belongs to and uses the DNN to find the optimal transmit covariance matrix.

Simultaneous Wireless Information and Power Transfer System

System Model

As a wireless communication transmitter transmits to its receiver, the radiation can be collected by adjacent RF energy harvesters. As shown in Fig. 2.1, an information transmitter is perceived by K surrounding energy harvesters. The information transmitter has M_t antennas and its corresponding receiver has M_r antennas. Each of the RF energy harvesters in the vicinity has one receive antenna.

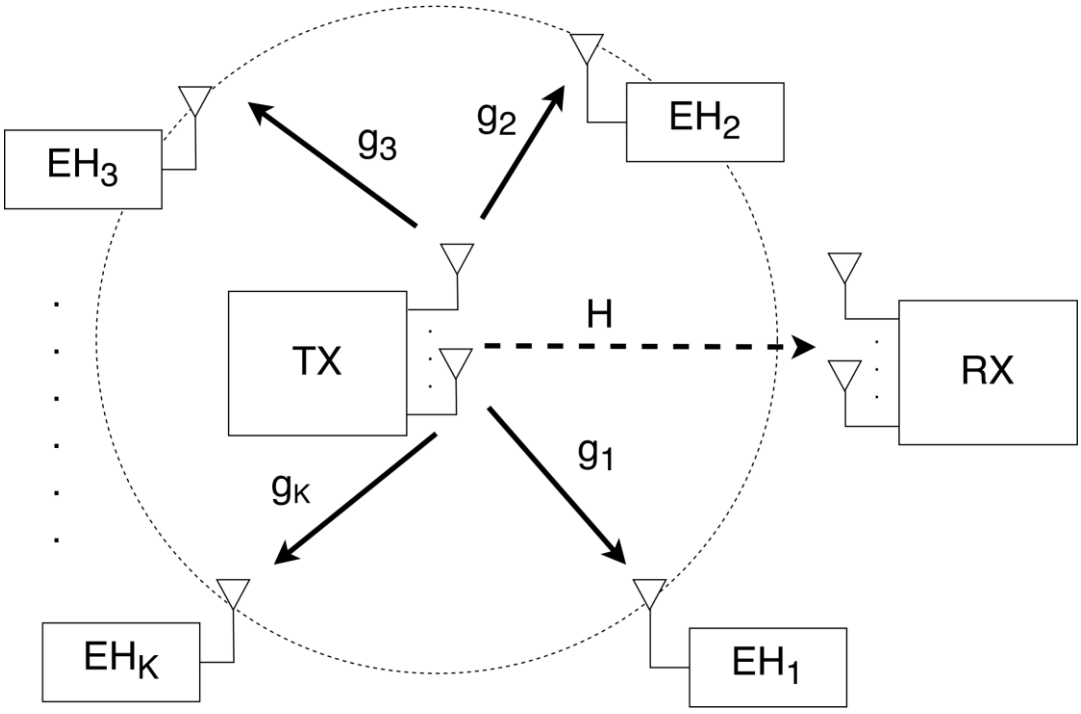

Figure 2.1: A pair of wireless communication transmitter and receiver as well as multiple energy harvesters

The narrowband signals are transmitted over quasi-static fading channels and the RF energy harvesters respond well to the carrier frequency. In the baseband equivalent model, the signal transmitted at the information transmitter is $\mathbf{x} \in \mathbb{C}^{M_t}$. The energy harvester does not need to convert the received signal from the RF band to the baseband. Nevertheless, the harvested RF power is proportional to the power of the baseband signal. The baseband received signal at the ith energy harvester, $i \in \mathcal{K} = \{1,2, \dots, K\}$, is given by

$$u_i = \mathbf{g}_i^H \mathbf{x} + n_i \tag{2.1}$$

where $\mathbf{g}_i \in \mathbb{C}^{M_t \times 1}$ is the conjugate channel vector from the information transmitter to the ith energy harvester, and n_i is the background noise. The signal received at the information receiver is $\mathbf{y} \in \mathbb{C}^{M_r}$, which is given by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z} \tag{2.2}$$

where $\mathbf{H} \in \mathbb{C}^{M_r \times M_t}$ is the normalized baseband equivalent channel from the information transmitter to its receiver, $\mathbf{z} \in \mathbb{C}^{M_r \times 1}$ is a zero-mean circularly symmetric complex Gaussian noise vector with $\mathbf{z} \sim \mathcal{CN}(\mathbf{0}, \sigma_n^2 \mathbf{I})$. It is assumed without loss of generality that $\sigma_n^2 = 1$ for clarity. The channels are known to the transmitter [24].

The received power at energy harvester i indicates the harvested energy normalized by the baseband symbol period. It can be written as

$$\zeta \mathrm{E}[|u_i|^2] = \zeta \mathbf{g}_i^H \mathbf{Q} \mathbf{g}_i \tag{2.3}$$

where $\mathbf{Q}$ denotes the transmit covariance matrix of signal $\mathbf{x}$, i.e.,

$$\mathbf{Q} = \mathrm{E}[\mathbf{x}\mathbf{x}^H] \tag{2.4}$$

ζ is a constant that indicates energy conversion efficiency. It is implied that the noise power is negligible compared with the received signal power.

For the information transmission, it is assumed that the Gaussian codebook with infinitely many codewords is used for the symbols and the expectation of the transmit covariance matrix is taken over the entire codebook. The covariance matrix is Hermitian positive semidefinite, i.e., $\mathbf{Q} \succeq 0$. The transmit power is limited by the transmitter's power constraint P, i.e.,

$$\mathrm{Tr}(\mathbf{Q}) \leq P \tag{2.5}$$

When channel matrix $\mathbf{H}$ is known to the transmitter and the receiver, from an information-theoretical perspective, the maximum achievable information rate is given by

$$r = \log|\mathbf{I} + \mathbf{H}\mathbf{Q}\mathbf{H}^H| \tag{2.6}$$

With transmitter precoding and receiver filtering, the capacity of the MIMO channel is the sum of the capacities of the parallel non-interfering single-input single-output channels (eigenmodes of channel $\mathbf{H}$) [25, 26].

The channel vector $\mathbf{g}_i$ is used to describe the wireless link from the transmitter to energy harvester i. Without loss of generality, suppose that the energy harvesting channel vector is normalized, i.e.,

$$\| \mathbf{g}_i \| = 1 \tag{2.7}$$

The average channel gain can be easily estimated.

Problem Formulation

For the wireless transmitter, the objective is to maximize the minimum energy harvesting rate of multiple RF energy harvesters while satisfying the transmit power constraint and the minimum achievable rate requirement of the information receiver. This is accomplished by designing the transmit covariance matrix $\mathbf{Q}$.

The optimization problem can be formulated as

$$\mathcal{P}_1: \begin{array}{ll} \underset{\mathbf{Q}}{\text{maximize}} & \min_{i\in\mathcal{K}}\{\mathbf{g}_i^H\mathbf{Q}\mathbf{g}_i\} \\ \text{subjectto} & \mathbf{Q}\succeq\mathbf{0} \\ & \text{Tr}(\mathbf{Q})\le P \\ & \log|\mathbf{I}+\mathbf{H}\mathbf{Q}\mathbf{H}^H|\ge R \end{array} \tag{2.8}$$

where R is the minimum achievable information rate. Problem $\mathcal{P}_1$ is a convex optimization problem for which efficient numerical optimization is possible [27] .

A singular value decomposition on **H** gives

$$\mathbf{H}=\mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \tag{2.9}$$

With

$$\widehat{\mathbf{Q}}=\mathbf{V}^H\mathbf{Q}\mathbf{V} \tag{2.10}$$

and

$$\hat{\mathbf{g}}_i=\mathbf{V}^H\mathbf{g}_i \tag{2.11}$$

Problem $\mathcal{P}_1$ can be written as

$$\mathcal{P}_1: \begin{array}{ll} \underset{\widehat{\mathbf{Q}}}{\text{maximize}} & \min_{i\in\mathcal{K}}\{\hat{\mathbf{g}}_i^H\widehat{\mathbf{Q}}\hat{\mathbf{g}}_i\} \\ \text{subjectto} & \widehat{\mathbf{Q}}\succeq\mathbf{0} \\ & \text{Tr}(\widehat{\mathbf{Q}})\le P \\ & \log\left|\mathbf{I}+\mathbf{\Sigma}\widehat{\mathbf{Q}}\mathbf{\Sigma}^H\right|\ge R. \end{array} \tag{2.12}$$

When the constraint on the achievable information rate in (8) and (12) takes effect, $\widehat{\mathbf{Q}}$ is a diagonal matrix. This is because the capacity formula (6) is derived as MIMO channel **H** can be decomposed in parallel eigenmode channels. The Hadamard's inequality states that the determinant of a positive definite matrix is less than or equal to the product of its diagonal entries.

Therefore, the above optimization problem is modified as

$$\mathcal{P}_1: \begin{matrix} \underset{\{\hat{q}_m\}}{\text{maximize}} & \min\limits_{i\in\mathcal{K}}\{\sum_{m=1}^{M} |\hat{g}_{im}|^2\hat{q}_m\} \\ \text{subjectto} & \hat{q}_m \geq 0, \ \ \forall m \\ & \sum_{m=1}^{M} \hat{q}_m \leq P \\ & \sum_{m=1}^{M} \log(1+|\sigma_m|^2\hat{q}_m) \geq R \end{matrix} \tag{2.13}$$

where $\{\hat{q}_m\}$ are the diagonal elements of $\widehat{\mathbf{Q}}$, $\{\sigma_m\}$ are the diagonal elements of $\mathbf{\Sigma}$, and $\{\hat{g}_{im}\}$ are the elements of $\hat{\mathbf{g}}_i$.

The power allocated on each eigen-channel is regulated as

$$\widehat{\mathbf{q}} = [\hat{q}_1, \hat{q}_2, \dots, \hat{q}_M]^T \tag{2.14}$$

The simplified channel vector from the transmitter to the ith RF energy harvester is defined as

$$\mathbf{a}_i = [|\hat{g}_{i1}|^2, |\hat{g}_{i2}|^2, \dots, |\hat{g}_{iM}|^2]^T, i \in \mathcal{K} = \{1,2, \dots, K\} \tag{2.15}$$

With the assumption of channel normalization,

$$\| \mathbf{a}_i \|_1 = 1 \tag{2.16}$$

The simplified channel vector contains no phase information. The K simplified channel vectors compose matrix $\mathbf{A} \in \mathbb{R}^{M\times K}$ that

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_K] \tag{2.17}$$

The channel vectors $\{\mathbf{g}_i\}$ are usually unknown to the transmitter. Nevertheless, in Problem $\mathcal{P}_1$, given the simplified channels $\mathbf{A}$, the optimal $\widehat{\mathbf{q}}$ can be calculated. The algorithm for the transmitter to acquire $\mathbf{A}$ is presented in Alg. 1. If the transmitter has M antennas, the simplified channel vector to each energy harvester can be estimated with M feedbacks of the received power levels. Comparatively, acquiring the channel vectors $\{\mathbf{g}_i\}$ incurs much larger feedback overhead.

Algorithm 1: Acquiring the simplified channels **A**

input: channel matrix **H**

output: simplified channels **A**

1. Perform SVD on **H**: $\mathbf{H} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H$.
2. **for** $l = 1,2,\dots M$ **do**
3. Construct power allocation vector $\hat{\mathbf{q}} \in \mathbb{R}^{M\times 1}$, where $\hat{q}_j = \begin{cases} P, & j = l \\ 0, & j \neq l \end{cases}$, $j = 1, \dots, M$.
4. Calculate the transmit covariance matrix as $\mathbf{Q} = \mathbf{V}\hat{\mathbf{Q}}\mathbf{V}^H$. Transmit with $\mathbf{Q}$.
5. The ith harvester measures the received power p_i and feeds it back to the transmitter, $\forall i \in \mathcal{K}$.
6. The element of the simplified channel vector to the ith RF energy harvester on the lth eigen-channel is obtained as $|\hat{g}_{il}|^2 = p_i/P$, $\forall i \in \mathcal{K}$.
7. The simplified channels are acquired $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_K]$.
8. **end for**

Optimal Transmission Strategy with Deep Neural Network

Deep Neural Network Structure

Instead of using any optimization solver, a DNN is implemented at the transmitter to find the optimal power allocation $\hat{\mathbf{q}}$ on the eigen-channels for the max-min problem $\mathcal{P}_1$. The DNN is trained offline with a large number of simulated data. It reduces the complexity of online execution and increases the response speed of the transmitter. In [28], an artificial NN was applied to transmit-power control in a MIMO system. The NN output was a selection from a group of fixed power allocation patterns. In this work, the output of the DNN is required to approach the exact optimal power allocation. The structure of the DNN is shown in Fig. 2.2.

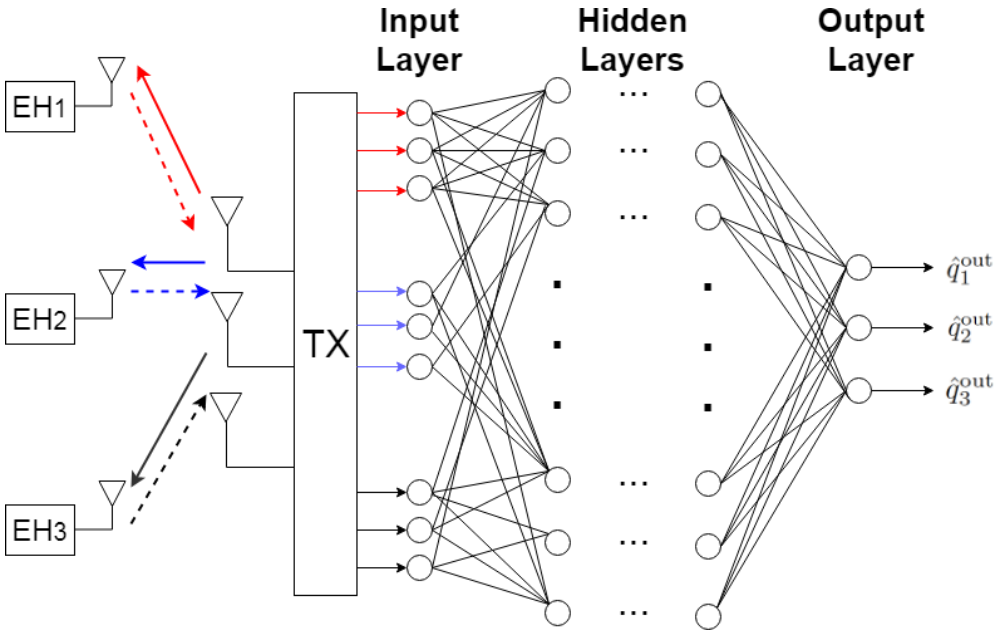

Figure 2.2: Structure of deep neural network for transmitter power allocation. ←: Pilot signals, ⇢: Feedbacks of received power levels. DNN output $\hat{\mathbf{q}}^{\text{out}}$.

The transmitter acquires the simplified channels $\mathbf{A}$ according to Alg. 1 and uses it as the input to the DNN. The input is

$$\mathbf{a}^{\text{in}} = \text{vec}(\mathbf{A}) \tag{2.18}$$

and $\mathbf{a}^{\text{in}} \in \mathbb{R}^{MK\times 1}$ The output of the DNN is

$$\hat{\mathbf{q}}^{\text{out}} = [\hat{q}_1^{\text{out}}, \hat{q}_2^{\text{out}}, \dots, \hat{q}_M^{\text{out}}]^T \tag{2.19}$$

The input-output relation of the DNN is defined as

$$\hat{\mathbf{q}}^{\text{out}} = \mathbf{F}(\mathbf{a}^{\text{in}}) \tag{2.20}$$

where function $\mathbf{F}(\cdot)$ derives $\hat{\mathbf{q}}^{\text{out}}$ based on $\mathbf{a}^{\text{in}}$ in order to maximize $\min_{i\in\mathcal{K}} \sum_{m=1}^{M} |\hat{g}_{im}|^2 \hat{q}_m^{\text{out}}$. The DNN is trained using the optimal $\hat{\mathbf{q}}^*$ solved in Problem $\mathcal{P}_1$.

The size of input to the DNN is set as MK. Therefore, the DNN can handle the case of at most K nearby RF energy harvesters. If the actual number of the RF energy harvesters

$\widetilde{K}$ is less than K, in order to regulate the size of the DNN input, the simplified channels $\mathbf{A}$ is constructed as

$$\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_{\widetilde{K}}, \underbrace{\mathbf{a}_1, \dots, \mathbf{a}_1}_{K-\widetilde{K}}] \tag{2.21}$$

and let

$$\mathbf{a}^{\text{in}} = \text{vec}(\mathbf{A}) \tag{2.22}$$

to maximize $\min\{p_1, \dots, p_{\widetilde{K}}\}$ is equivalent to maximize $\min\{\mathbf{A}^T\widehat{\mathbf{q}}\}$.

In Problem $\mathcal{P}_1$, the optimal $\widehat{\mathbf{q}}$ is also determined by the information rate requirement R. The DNN can be adaptable when R is treated as an additional input feature. That is,

$$\mathbf{a}^{\text{in}} = \begin{bmatrix} \text{vec}(\mathbf{A}) \\ R \end{bmatrix} \tag{2.23}$$

and $\mathbf{a}^{\text{in}} \in \mathbb{R}^{(MK+1)\times 1}$. With the extra input feature R, the DNN can be trained to produce $\widehat{\mathbf{q}}^{\text{out}}$ that maximizes the minimum received power and satisfies any preset information rate requirement R.

The input feature R should be in the range $R \in [0, R_{\max}]$ to make it feasible to train the DNN. If $R > R_{\max}$, Problem $\mathcal{P}_1$ does not have any feasible solution. Considering only the constraints of Problem $\mathcal{P}_1$, $R_{\max}$ can be acquired with the water-filling algorithm. The power allocation that maximizes the information rate is given by

$$\hat{q}_m = \max\left(0, \frac{1}{\nu} - \frac{1}{|\sigma_m|^2}\right), m = 1,2, \dots, M \tag{2.24}$$

where the water level is

$$1/\nu = (P + \textstyle\sum_{m\in\mathcal{W}} \frac{1}{|\sigma_m|^2})/|\mathcal{W}| \tag{2.25}$$

and $\mathcal{W}$ is the set that indicates the eigen-channels that are allocated with non-zero power.

The upper limit of the achievable information rate is given by

$$R_{\max} = \sum_{m\in\mathcal{W}} \log\left(1 + |\sigma_m|^2 \left(\frac{1}{\nu} - \frac{1}{|\sigma_m|^2}\right)\right). \tag{2.26}$$

$R_{\max}$ denotes the upper bound.

Deep Neural Network Algorithm

The transmitter aims to determine the optimal transmission strategy over vector channels $\mathbf{g}_i$. The simplified channel vector $\mathbf{a}^{\text{in}}$ are taken as the input to the NN. The desired outputs of the NN are $\widehat{\mathbf{q}}$, which are used to construct the optimal transmit covariance matrix.

N_L is defined as the total number of the NN's layers. $\mathbf{v}^l$ denotes the output vector of the lth layer

$$\mathbf{v}^l = [v_1^l, v_2^l, \dots, v_{n_l}^l]^T \tag{2.27}$$

where the size of the lth layer is regulated as n_l. The output of the lth layer's ith neuron is defined as v_i^l. $w_{i,j}^l$ denotes the weight between the ith node in the lth layer and the jth node in the $(l-1)$th layer. The bias of the ith node in the lth layer is denoted as b_i^l. The input to the ith node in the lth layer is denoted as

$$s_i^l = b_i^l + \sum_{j=1}^{n_{l-1}} w_{i,j}^l v_j^{l-1}. \tag{2.28}$$

The tanh function is used as the activation function to calculate the output of the ith node in the lth layer.

$$v_i^l = f(s_i^l) = \frac{e^{s_i^l} - e^{-s_i^l}}{e^{s_i^l} + e^{-s_i^l}} \tag{2.29}$$

The cost function C of the NN is acquired by

$$C = \frac{1}{2} \| \widehat{\mathbf{q}} - \mathbf{v}^{N_L} \|^2. \tag{2.30}$$

The gradient descent is utilized in order to minimize C

$$\widehat{w}_{i,j}^{l}(n+1) = \widehat{w}_{i,j}^{l}(n) - \mu \frac{\partial C}{\partial w_{i,j}^{l}} \tag{2.31}$$

$$\hat{b}_{i}^{l}(n+1) = \hat{b}_{i}^{l}(n) - \mu \frac{\partial C}{\partial b_{i}^{l}} \tag{2.32}$$

where μ is defined as a positive value, which denotes the step size. In the nth training epoch, $\hat{b}_{i}^{l}(n)$ and $\widehat{w}_{i,j}^{l}(n)$ denote the estimate value of b_{i}^{l} and $w_{i,j}^{l}$, respectively.

The output error of the ith node in the lth layer is defined as e_{i}^{l}

$$e_{i}^{l} \equiv \frac{\partial C}{\partial s_{i}^{l}}. \tag{2.33}$$

$e_{i}^{l}, l = 1,2, \dots, N_L - 1$ is defined as

$$e_{i}^{l} = \sum_{k=1}^{n_{l+1}} \frac{\partial C}{\partial s_{k}^{l+1}} \frac{\partial s_{k}^{l+1}}{\partial s_{i}^{l}} = \left(\sum_{k=1}^{n_{l+1}} \delta_{k}^{l+1} w_{k,i}^{l+1}\right) f'\ (s_{i}^{l}). \tag{2.34}$$

And $e_{i}^{N_L}$ is defined as

$$e_{i}^{N_L} = \frac{\partial C}{\partial s_{i}^{N_L}} = \frac{\partial}{\partial s_{i}^{N_L}} \frac{1}{2} \parallel \widehat{\mathbf{q}} - \mathbf{v}^{N_L} \parallel^2 = -(\theta_i - v_{i}^{N_L}) f'\ (s_{i}^{N_L}). \tag{2.35}$$

The partial derivatives of C with respect to b_{i}^{l} is denoted as

$$\frac{\partial C}{\partial b_{i}^{l}} = \delta_{i}^{l}. \tag{2.36}$$

The partial derivatives of C with respect to $w_{i,j}^{l}$ is denoted as

$$\frac{\partial C}{\partial w_{i,j}^{l}} = \delta_{i}^{l} v_{j}^{l-1}. \tag{2.37}$$

Both the weights and the biases of the neural network are arbitrarily assigned before the first training epoch. As the cost function C indicates, the output of the NN $\mathbf{v}^{N_L}$ is compared with the desired output $\widehat{\mathbf{q}}$. The weights and biases are updated as the calculated error is back-propagated to the previous layers. The training process is continued until the

cost function is lower than a threshold. A well trained NN is capable to produce the precise output with the input data [29].

Deep Neural Network for Block Fading Channel

Next, both channels $\mathbf{H}$ and $\mathbf{g}_i$ are assumed to experience block fading. Within the period of each block, the channel states are quasi-static. However, the channel states vary independently from block to block. Suppose that the channel matrix $\mathbf{H}(n)$ from the transmitter to its receiver in the nth fading block is measured. Suppose that the simplified channels $\mathbf{A}(n)$ from the transmitter to the RF energy harvesters in the nth fading block are acquired by Alg. 1. When the channel matrix $\mathbf{H}(n)$ is measured within the nth fading block, the SVD of $\mathbf{H}(n)$ gives the eigen-value vector

$$\mathbf{s}(n) = [|\sigma_1(n)|^2, |\sigma_2(n)|^2, \dots, |\sigma_M(n)|^2]^T \tag{2.38}$$

Due to channel variation, $\mathbf{s}(n)$ has to be reevaluated in each fading block. Consequently, a specific DNN has to be trained during each fading block. This is impractical.

The K-means clustering method is utilized to solve this problem for block fading channels. The K-means clustering method effectively assigns each vector in a multidimensional feature space with a class label [30]. For many channel matrices $\{\mathbf{H}(n)\}$, the possible eigen-value vectors are classified into N clusters. The cluster centers are denoted as

$$\mathbf{c}_i = [|\sigma_1^i|^2, |\sigma_2^i|^2, \dots, |\sigma_M^i|^2]^T, i = 1,2, \dots, N. \tag{2.39}$$

For each cluster of eigen-value vectors, the cluster center is used as $\{|\sigma_m|^2\}_{m=1}^M$ to train a DNN. In the nth fading block, the cluster center that has the shortest Euclidean distance to $\mathbf{s}(n)$ is selected and the corresponding DNN is called to produce $\widehat{\mathbf{q}}$.

The process to determine the N cluster centers is presented in Alg. 2. Suppose that the Frobenius norm of the channel matrix $\| \mathbf{H}(n) \|_F$ is constant. A large number T_c of eigen-value vectors are randomly generated to find the N cluster centers. The random samples of the eigen-value vectors are chosen as

$$\mathbf{d}_i = [|\sigma_1^i|^2, |\sigma_2^i|^2, \dots, |\sigma_M^i|^2]^T, i = 1,2, \dots, T_c \tag{2.40}$$

such that

$$|\sigma_1^i|^2 > |\sigma_2^i|^2 > \cdots > |\sigma_M^i|^2 \tag{2.41}$$

and

$$\sum_{m=1}^{M} |\sigma_m^i|^2 = \| \mathbf{H} \|_F^2 \tag{2.42}$$

For any given N, Alg. 2 converges to N cluster centers.

Algorithm 2: Determine the N cluster center

input: eigen-value vectors $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{T_c}$

output: cluster centers $\{\mathbf{c}_m\}_{m=1}^N$

1. Choose T_c random samples of the eigen-value vectors $\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_{T_c}$.
2. Initialize N cluster centers as $\mathbf{c}_u = \mathbf{d}_{T_c-u+1}, u = 1,2, \dots, N$.
3. Initialize sets $\mathcal{U}_k = \emptyset, k = 1,2, \dots, N$.
4. **while** $\{\mathbf{c}_m\}_{m=1}^N$ don't converge **do**
5. **for** $i = 1, \dots, T_c$ **do**
6. Calculate $l_{i,u} = \| \mathbf{d}_i - \mathbf{c}_u \|, u = 1,2, \dots, N$.
7. $v_i = \arg \min_{u \in \{1,2,\dots,N\}} l_{i,u}$.
8. $\mathcal{U}_{v_i} = \mathcal{U}_{v_i} + \{i\}$.
9. **end for**
10. **for** $m = 1, \dots, N$ **do**
11. $\mathbf{c}_m = \sum_{i \in \mathcal{U}_m} \mathbf{d}_i / |\mathcal{U}_m|$.
12. **end for**
13. **end while**

Simulation Results

A MIMO wireless communication system with nearby RF energy harvesters is simulated. The wireless transmitter has $M = 3$ antennas. The maximum transmitted power

is $P = 120$ mW. With Matlab NN toolbox, the DNN is established with 10 layers and each hidden layer has 50 nodes. All layers of the neural networks are fully connected. The total number of weights is 18100. The activation function for the hidden layers is tanh. The learning rate is 0.01. The NN minimum gradient threshold is set as 10^{-6}.

Random (uniformly distributed) simplified channel vectors $\{\mathbf{a}\}$ are used to train the DNN. There are 5027, 23436 and 37820 simplified channel vectors for the cases with $K = 1$, 2 and 3 RF energy harvesters, respectively. The simplified channel vectors are the inputs to the DNN. The corresponding optimal transmit power allocations $\{\hat{\mathbf{q}}\}$ of Problem $\mathcal{P}_1$ are generated by the CVX convex optimization solver [31] and used as the DNN outputs for training.

The minimum harvested power among all of the RF energy harvesters derived from the NN result is denoted as

$$p^{\mathrm{NN}} = \min\{\mathbf{A}^T\hat{\mathbf{q}}^{\mathrm{out}}\} \tag{2.43}$$

and the information rate

$$r^{\mathrm{NN}} = \sum_{m=1}^{M} \log(1 + |\sigma_m|^2\hat{q}_m^{\mathrm{out}}) \tag{2.44}$$

The minimum harvested power derived from the CVX solver result is

$$p^{\mathrm{CVX}} = \min\{\mathbf{A}^T\hat{\mathbf{q}}\} \tag{2.45}$$

and the information rate requirement is R.

We define the power loss ratio as

$$\lambda_p = (p^{\mathrm{CVX}} - p^{\mathrm{NN}})/p^{\mathrm{CVX}} \tag{2.46}$$

and the information rate loss ratio

$$\lambda_R = (R - r^{\mathrm{NN}})/R \tag{2.47}$$

If $\lambda_p < 0$ (or $\lambda_R < 0$), which means that the DNN has a better result than the CVX solver, then $\lambda_p = 0$ (or $\lambda_R = 0$). The power loss ratio threshold is defined as n_p and the information rate loss ratio threshold is defined as n_R. Of all of the N_T DNN testing outputs, transmissions with N_S particular transmit power allocations satisfy $\lambda_p \leq n_p$ (or $\lambda_R \leq n_R$). The precision

$$\eta = N_S/N_T \tag{2.48}$$

is used to evaluate the DNN performance.

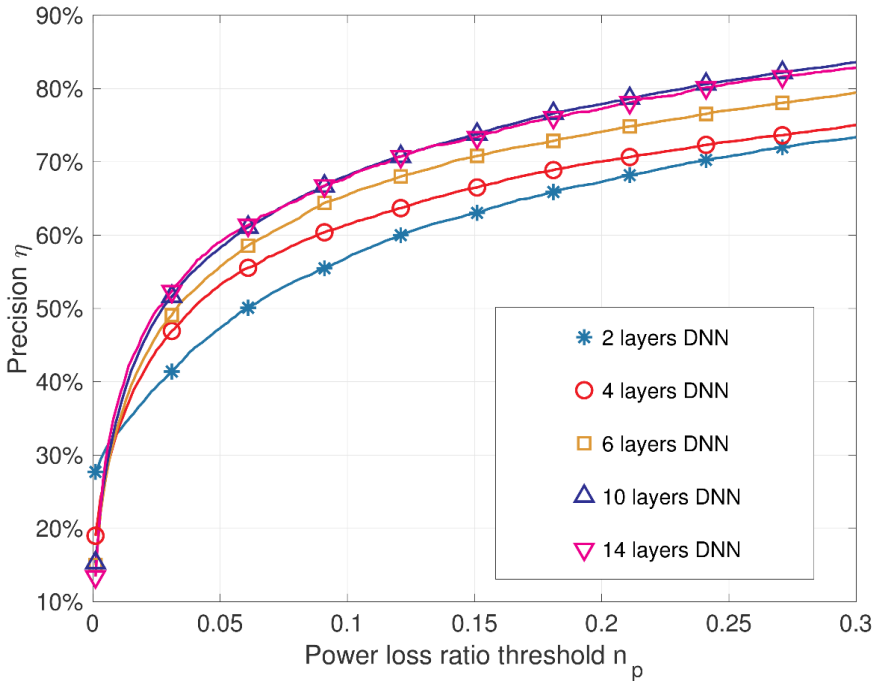

Figure 2.3: Precision η of the proposed DNN for power allocation versus the power loss ratio threshold n_p. The number of the RF energy harvesters is $K = 3$. The information rate requirement is $R = 5$ bps/Hz.

First, the channel matrix **H** is supposed to be fixed. Each trained DNN is applied to $N_T = 3000$ test data, which are randomly generated with the case of $K = 3$ RF energy

harvesters. The information rate requirement is regulated as $R = 5$ bps/Hz. bps/Hz stands for bits per second per Hertz.

Fig. 2.3 shows the system performance in DNN precision η versus the power loss ratio threshold n_p. When $n_p = 0.3$, the outputs of DNN with 2 hidden layers have 74% precision, while the outputs of DNN with 10 hidden layers have 84% precision. The simulation shows that it is hard to train a DNN with more than 10 layers for better performance. Fig. 2.4 shows the system performance in DNN precision versus the information rate loss ratio threshold n_R. The DNN has a higher precision on λ_R with more hidden layers. In the following simulations, the number of DNN's hidden layers is regulated as 10.

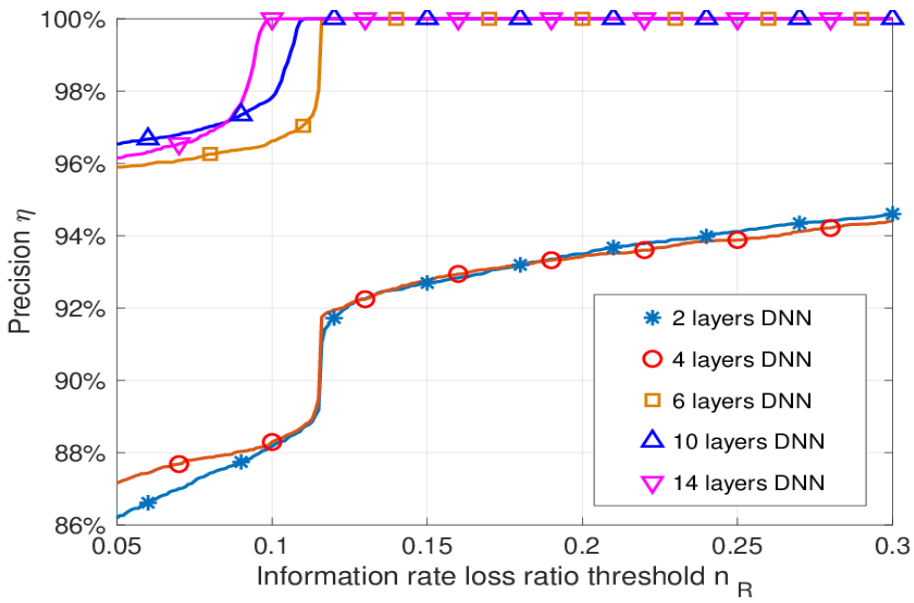

Figure 2.4: Precision η of the proposed DNN for power allocation versus the information rate loss ratio threshold n_R. The number of the RF energy harvesters is $K = 3$. The information rate requirement is $R = 5$ bps/Hz.

The DNN is enhanced with information rate requirement R as an additional input. The trained DNN is applied to $N_T = 9000$ test data and the particular information rate requirements R.

Fig. 2.5 shows the system performance in DNN precision versus the power loss ratio threshold n_p with different information rate requirements R. When $R = 5$ bps/Hz and $n_p = 0.05$, the DNN precision of one-harvester case can be as high as 100%, while the precision of three-harvester case is as low as 60%.

The precision gets lower with more RF energy harvesters in the system. This is because there isn't enough training data for the case of more RF energy harvesters in the system, which results in worse DNN performance. Fig. 2.5 also indicates that the variation of R does not affect the precision of the DNN.

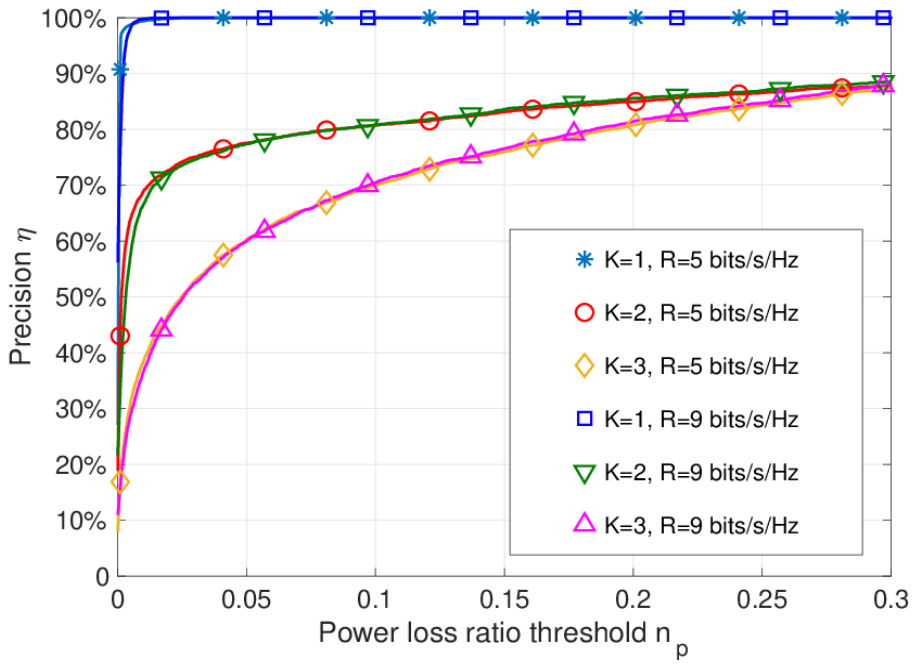

Figure 2.5: Precision η of the proposed DNN for power allocation versus the power loss ratio threshold n_p. The number of the RF energy harvesters is $K = 1,2,3$. The Information rate requirements $R = 5,9$ bps/Hz.

Second, the channel is assumed to experience block fading. Suppose $\sum_{m=1}^{M} |\sigma_m^i|^2 = \| \mathbf{H} \|_F^2 = 40$. The K-means clustering algorithm is applied to classify the eigen-value vector $\mathbf{s}(n)$ into $N = 1,3,5$ clusters.

For each cluster, a particular DNN is trained. $N_T = 9000$ test data are used for each of the $R = 5,7,9$ bps/Hz conditions with the cases of $K = 1,2,3$ RF energy harvesters. Each data point is with a random **A** and a random **H**.

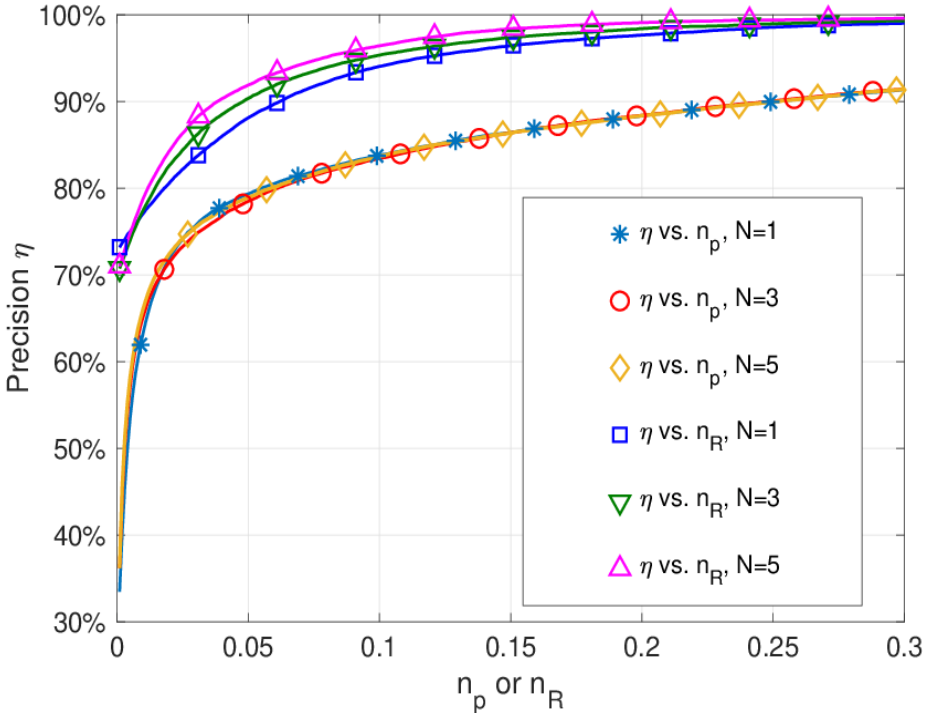

Figure 2.6: Precision η of the proposed DNN for power allocation versus the power loss ratio threshold n_p and the information rate loss ratio threshold n_R in block fading channels. The information rate requirements $R = 5,7,9$ bps/Hz.

Fig. 2.6 shows the system performance in the precision versus both the power and the information rate loss ratio thresholds with different cluster numbers. As the channel matrix **H** is normalized, all three clustering cases have similar good performances. Clustering with $N = 5$ achieves the best precision on λ_R. When clustering with $N = 1$, i.e.,

just taking the expected value of the eigen-value vectors as the cluster center, the DNN has relatively good performance in both λ_p and λ_R. Henceforth, $N = 1$ cluster is used in practice.

Table 2.1. Program running time of 100 fading channel blocks

M	R (bps/Hz)	Method	Running time (sec)
3	5	DNN	2.47
3	5	CVX Solver	96.46
3	7	DNN	2.38
3	7	CVX Solver	130.90
3	9	DNN	2.65
3	9	CVX Solver	135.31

With Matlab R2017a and the CVX package [31], the running time of different algorithms (excluding the offline-training of the DNN) is shown in Tab. 2.1. Tab. 2.1 shows the effectiveness of the proposed method that is based on DNN.

Conclusions

A design method is proposed for wireless power transfer to multiple RF energy harvesters with unknown channels. The simplified channel vectors and the information rate requirement are used as the input to a DNN, which outputs the optimal transmit power allocation that can maximize the minimum harvested energy of all the RF energy harvesters. At the same time, the information rate requirement at the communication receiver is satisfied. When the channels experience block fading, the transmitter applies the K-means clustering method to classify the eigen-value vectors into a few clusters. During one fading block, the transmitter estimates the communication channel and finds the cluster it belongs to. Then, a corresponding DNN is called to find the optimal transmit power allocation.

CHAPTER THREE

Deep Reinforcement Learning for Optimized Wireless Information and Power Transfer

Introduction

In Chapter Three, the constructed SWIPT system is same as the one in Chapter Two. In the system, a multi-antenna transmitter communicates with an information receiver while radiating the electromagnetic waves to multiple energy harvesters. Since energy charging is a continuous process, a practical dynamic energy charging scenario is discussed. Each energy harvester is equipped with a limited volume energy buffer, the energy collected from the transmitter can be accumulated in the buffer. By adapting to the channel variations, the transmitter can adjust its transmission strategy to take care of each energy harvester. The transmitter intends to fully charge all surrounded energy harvesters' energy buffers in the shortest time while maintaining a target information rate toward the receiver.

The communication link is established as a strong line of sight (LOS) transmission, thus the channel condition from the transmitter to the information receiver is assumed to be invariant. However, the channel conditions from the transmitter to the energy harvesters experience block fading. Due to the hardware limitations, the estimation of the energy harvesting channel vectors is not able to be implemented under the fast varying channel conditions. Therefore, the wireless charging problem can be modeled as a high complexity discrete time stochastic control process with unknown system dynamics [32].

To deal with this complicated optimization problem, DQN is applied to solve the energy charging problem and find the optimal transmission strategy. DQN was introduced to learn how to play complex games with very large number of system states, and unknown state transition probabilities [32]. More recently, DQN has been applied to deal with complex communication problems and shown to achieve good performance [3, 33, 34]. In this model, the accumulated energy at the energy harvesters is defined as the system states, while the transmit power allocation is regulated as the action. At the beginning of each time slot, each energy harvester feedbacks the accumulated energy level to the wireless transmitter. Then, the transmitter collects all the information in order to form it as the system state and inputs the system state into the well-trained DQN. The DQN outputs the Q values corresponding to all possible actions. The action with the maximum Q value is selected as the beam pattern to be used for the transmission during the current time slot.

Based on the traditional DQN, the Double DQN and Dueling DQN algorithms are applied in order to reduce the observed overestimations [35] and improve the learning efficiency [36]. Henceforth, Dueling Double DQN is utilized to solve the multiple energy harvesters' wireless charging problem in Chapter Three.

The contribution of Chapter Three is summarized as follows: the wireless charging problem is formulated as a Markov Decision Process (MDP) and the DQN algorithm is applied to find the optimal transmission policy without estimating the channel conditions. Using DQN, a centralized power allocation strategy is derived. The impact of the channel conditions, energy buffer size and the number of energy harvesters on the optimal transmission strategy is explored as well.

System Model

As shown in Fig. 2.1, an information transmitter communicates with its receiver while perceived by K nearby RF energy harvesters. Both the transmitter and the receiver are equipped with M antennas, while each RF energy harvester is equipped with one receive antenna. The baseband received signal at the receiver can be represented as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{z}, \tag{3.1}$$

where $\mathbf{H} \in \mathbb{C}^{M\times M}$ denotes the normalized baseband equivalent channel from the information transmitter to its receiver, $\mathbf{x} \in \mathbb{C}^{M\times 1}$ represents the transmitted signal, and $\mathbf{z} \in \mathbb{C}^{M\times 1}$ is the zero-mean circularly symmetric complex Gaussian noise with $\mathbf{z}\sim\mathcal{CN}(\mathbf{0}, \rho^2\mathbf{I})$.

The transmit covariance matrix is denoted with $\mathbf{Q}$, i.e.,

$$\mathbf{Q} = \mathrm{E}[\mathbf{x}\mathbf{x}^H] \tag{3.2}$$

The covariance matrix is Hermitian positive semidefinite, i.e., $\mathbf{Q} \succeq 0$. The transmit power is restricted by the transmitter's power constraint P, i.e., $\mathrm{Tr}(\mathbf{Q}) \leq P$. For the information transmission, it is assumed that a Gaussian codebook with infinitely many codewords is used for the symbols and the expectation of the transmit covariance matrix is taken over the entire codebook.

With transmitter precoding and receiver filtering, the capacity of the MIMO channel is the sum of the capacities of the parallel non-interfering single-input single-output (SISO) channels (eigenmodes of channel $\mathbf{H}$) [26]. The MIMO channel is converted to M eigen-channels for information and energy transfer [37, 38]. A singular value decomposition (SVD) on $\mathbf{H}$ gives

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \tag{3.3}$$

The $\mathbf{\Sigma}$ can be acquired as

$$\mathbf{\Sigma} = \text{diag}(\sigma_1, \sigma_2, \ldots \sigma_M) \tag{3.4}$$

contains the M singular values of $\mathbf{H}$. Since the MIMO channel is decomposed into M parallel SISO channels, the information rate can be given by

$$r = \sum_{m=1}^{M} \log(1 + \rho^{-2}|\sigma_m|^2 \hat{q}_m), \tag{3.5}$$

where $\{\hat{q}_m\}$ are the diagonal elements of $\widehat{\mathbf{Q}}$ with

$$\widehat{\mathbf{Q}} = \mathbf{V}^H \mathbf{Q} \mathbf{V} \tag{3.6}$$

The RF energy harvester received power specifies the harvested energy normalized by the baseband symbol period and scaled by the energy conversion efficiency. The received power at the ith energy harvester is

$$p_i = \mathbf{g}_i^H \mathbf{Q} \mathbf{g}_i \tag{3.7}$$

where $\mathbf{g}_i \in \mathbb{C}^{M \times 1}$ is the channel vector from the transmitter to the ith energy harvester. With MIMO channel decomposition, the received power at energy harvester i is denoted as

$$p_i = \sum_{m=1}^{M} |\hat{g}_{im}|^2 \hat{q}_m, \tag{3.8}$$

where $\{\hat{g}_{im}\}$ are the elements of vector $\hat{\mathbf{g}}_i$ with

$$\hat{\mathbf{g}}_i = \mathbf{V}^H \mathbf{g}_i \tag{3.9}$$

The simplified channel vector from the transmitter to the ith RF energy harvester is defined as

$$\mathbf{c}_i = [|\hat{g}_{i1}|^2, |\hat{g}_{i2}|^2, \ldots, |\hat{g}_{iM}|^2]^T, \tag{3.10}$$

for each $i \in \mathcal{K} = \{1,2, \ldots, K\}$. The simplified channel vector contains no phase information. The K simplified channel vectors compose matrix $\mathbf{C} \in \mathbb{R}^{M \times K}$ as

$$\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K]. \tag{3.11}$$

In what follows, it is assumed that time is slotted, each time slot has a duration T. Each energy harvester is equipped with an energy buffer of size $B_i \in [0, B_{\max}]$, $i \in \mathcal{K}$. Without loss of generality, it is assumed that at $t = 0$, all harvesters' buffers are empty, which corresponds to system state

$$s_0 = [0,0,\ldots,0] \tag{3.12}$$

At a generic time slot t, the transmitter transmits with one of the designed beam patterns. Each harvester i can harvest the specific amount of power p_i, and its energy buffer values increase to

$$B_i^{t+1} = B_i^t + p_i T \tag{3.13}$$

Therefore, each state of the system includes the accumulated harvested energy information of all K harvesters, i.e.,

$$s_t = [B_1^t, B_2^t, \ldots, B_K^t], \tag{3.14}$$

where B_i^t denotes the i-th energy harvester's accumulated energy up to time slot t.

Once all harvesters are fully charged, the system arrives at a final goal state, which is denoted as

$$s_G = [B_{\max}, B_{\max}, \ldots, B_{\max}] \tag{3.15}$$

The energy buffer level $B_{\max}$ also accounts for situations in which $B_i > B_{\max}$.

Problem Formulation

In this section, the communication link is characterized by strong LOS transmission, which indicates an invariant channel matrix $\mathbf{H}$. The energy harvesting channel vector $\mathbf{g}$ varies over time slots. The wireless charging problem is modeled as a MDP and the optimization problem is solved by RL. When the number of system states is

very large, DQN algorithm is applied to acquire the optimal strategy at each particular system state.

In order to model the optimization problem as a RL problem, the beam pattern chosen in a particular time slot t is defined as the action $\mathbf{a}^t$. The set $\mathcal{A}$, which contains all possible actions, is formulated by equally generating L different beam patterns with power allocation vector

$$\hat{\mathbf{q}} = [\hat{q}_1, \dots, \hat{q}_m] \tag{3.16}$$

that satisfies the power and information rate constraints, i.e.,

$$\sum_{i=1}^{M} \hat{q}_m = P \tag{3.17}$$

and

$$\sum_{i=1}^{M} \log(1 + \rho^{-2}|\sigma_m|^2 \hat{q}_m) \geq R \tag{3.18}$$

Each beam pattern corresponds to a particular power level p_i, which not only depends on the action $\mathbf{a}^t$ but also on the channel condition experienced by the harvester during time slot t.

Given the above, the SWIPT problem for a time-varying channel can be formulated as

$$\mathcal{P}_1: \begin{array}{ll} \underset{\{\mathbf{a}^t\}}{\text{minimize}} & n \\ subject to & a_m^t \geq 0 \\ & \sum_{m=1}^{M} a_m^t \leq P \\ & \sum_{m=1}^{M} \log(1 + \rho^{-2}|\sigma_m|^2 a_m^t) \geq R \\ & \sum_{t=1}^{n} \sum_{m=1}^{M} |\hat{g}_{im}^t|^2 a_m^t T \geq B_{\max}, \forall i \in \mathcal{K} \end{array} \tag{3.19}$$

By adapting to the current channel conditions and current energy buffer state of the harvesters, the action is selected at each time slot. Therefore, the evolution of the system can be described by a Markov chain.

The generic state s is identified by the current buffer levels of the harvester, i.e.,

$$s = \{B_1, B_2, \dots, B_K\} \tag{3.20}$$

The set of all states is denoted by $\mathcal{S}$. Among all states, the initial system state describes the situation that all harvesters' buffer are empty, namely

$$s_0 = \{0, \dots, 0\} \tag{3.21}$$

and the final system state s_G appears when all the harvesters are fully charge, i.e.,

$$s_G = \{B_{\max}, \dots, B_{\max}\} \tag{3.22}$$

Suppose that all the channel coefficients at each time slot are known, problem $\mathcal{P}_1$ can be seen as a stochastic shortest path (SSP) problem from state s_0 to state s_G. At each time slot the system is in a generic state s, the transmitter selects a beam pattern (action $\mathbf{a} \in \mathcal{A}$), and the system transits to a new state s'. The dynamics of the system is captured by transition probabilities $p_{s,s'}(\mathbf{a})$, $s, s' \in \mathcal{S}$ and $\mathbf{a} \in \mathcal{A}$, describing the probability that the harvesters' energy buffer reach the levels in s' after a transmission with beam pattern $\mathbf{a}$. It is noted that the final state s_G is absorbing, i.e.,

$$P_{s_G,s_G}(\mathbf{a}) = 1, \forall \mathbf{a} \in \mathcal{A} \tag{3.23}$$

Each system state transition is associated with a reward $w(s, \mathbf{a}, s')$. $w(s, \mathbf{a}, s')$ denotes the reward when the current state is $s \in \mathcal{S}$, action $\mathbf{a} \in \mathcal{A}$ is selected and the system moves to state $s' \in \mathcal{S}$. Since the optimization aims at reaching s_G in the fewest transmission time slots, the action is considered to entail a positive reward related to the difference between the current energy buffer level and the full energy buffer level of all harvesters. When the system reaches state s_G, the reward is set as 0. In this way, the system not only tries to fully charge all harvesters in the shortest time but also uniformly charges all the harvesters.

In detail, the reward function is defined as

$$w(s, \mathbf{a}, s') = -\lambda(KB_{\max} - \sum_{i=1}^{K} \min(s'_i, B_{\max})), \tag{3.24}$$

where

$$\lambda s'_i = \lambda s_i + \lambda \sum_{m=1}^{M} |\hat{g}_{im}|^2 a_m T, \tag{3.25}$$

and λ denotes the unit price of the harvested energy.

It can be noted that different reward functions can also be selected. As an example, it is also possible to set a constant negative reward (e.g., a unitary cost) for each transmission that the system doesn't reach the final state, and a big positive reward only for the states and actions that bring the system to the final state s_G. In formulas, this can be expressed as

$$w(s, \mathbf{a}, s') = \begin{cases} +\infty, & s' = s_G \\ -1, & otherwise. \end{cases} \tag{3.26}$$

It can be noted that the reward formulation (74) is actually equivalent to minimizing the number of time slots to reach state s_G starting from state s_0.

Using the above formulation the optimization problem $\mathcal{P} = (\mathcal{S}, \mathcal{A}, p, w, s_0, s_G)$ can then be seen as a stochastic shortest path search from state s_0 to state s_G on the Markov chain with states $\mathcal{S}$ and probabilities $\{p_{s,s'}(\mathbf{a})\}$, actions $\mathbf{a} \in \mathcal{A}$, and rewards $w(s, \mathbf{a}, s')$. The optimization objective is to find, for each possible state $s \in \mathcal{S}$, an optimal action $\mathbf{a}^*(s)$ so that the system reaches the final state following the path with maximum average reward. A generic policy can be written as $\pi = \{\mathbf{a}(s): s \in \mathcal{S}\}$.

Different techniques can be applied to solve problem $\mathcal{P}_1$, as it represents a particular class of MDPs. In Chapter Three, however, it is assumed that the channel conditions at each time slot are unknown, which corresponds to not knowing the transition probabilities

$\{p_{s,s'}(\mathbf{a})\}$. RL is a good method in solving this problem. Therefore, in the next section RL is applied to solve the proposed optimization problem.

Optimal Transmission Strategy with Deep Q-Network

RL is suitable to solve optimization problems in which the system dynamics follow a particular transition probability function, however, the probabilities $\{p_{s,s'}(\mathbf{a})\}$ are unknown. In what follows, the Q-learning algorithm [39] is utilized to solve the optimization problem. Then the RL approach is combined with a NN to approximate the system model in case of large states and actions sets [32].

Q learning Algorithm

If the number of the system states is small, the traditional Q-learning method can be used to find the optimal strategy at each system state.

To this end, the cost function of action $\mathbf{a}$ on system state s is regulated as $Q(s,\mathbf{a})$, with $s \in \mathcal{S}, \mathbf{a} \in \mathcal{A}$. The algorithm initializes with $Q(s,\mathbf{a}) = 0$, and then updates the Q values using the following equation.

$$Q(s,\mathbf{a}) = (1-\alpha(s,\mathbf{a}))Q(s,\mathbf{a}) + \alpha(s,\mathbf{a})[w(s,\mathbf{a},s') + \gamma f(s',\mathbf{a})] \quad (3.27)$$

where

$$f(s',\mathbf{a}) = \min_{\mathbf{a}\in\mathcal{A}} Q(s',\mathbf{a}) \quad (3.28)$$

and $\alpha(s',\mathbf{a})$ denotes the learning rate. In each time slot, only one Q value is updated, hence all the other Q values remain the same.

At the beginning of the learning iterations, since the Q-table does not have enough information to choose the best action at each system state, the algorithm randomly explores

new actions with a particular probability. The selection threshold is defined as $\varepsilon_c \in [0.5,1]$. A probability is randomly generated as $p \in [0,1]$, if $p \geq \varepsilon_c$, the action $\mathbf{a}$ is chosen as

$$\mathbf{a} = \max_{\mathbf{a} \in \mathcal{A}} Q(s, \mathbf{a}) \tag{3.29}$$

On the contrary, if $p < \varepsilon_c$, the action is randomly selected from the action set $\mathcal{A}$.

When Q^* converges, the optimal strategy at each state is determined as

$$\pi^*(s) = \arg\max_{\mathbf{a} \in \mathcal{A}} Q^*(s, \mathbf{a}) \tag{3.30}$$

which corresponds to the optimal beam pattern for each system state explored during the charging process.

Deep Q-Network Algorithm

When considering the complex system, in which there are multiple harvesters and channel conditions vary by time, the number of system states dramatically increases. In order to learn the optimal transmission strategy at each system state, the Q-learning algorithm requires a Q-table with a large number of elements, which makes it very difficult for all the values in the Q-table to converge. Therefore, in what follows the DQN approach is applied to find the optimal transmission policy.

The main idea of DQN is to train a NN to find the Q function of a particular system state and action combination. When the system is in state s, and action $\mathbf{a}$ is selected, the Q function is denoted as $Q(s, \mathbf{a}, \theta)$. θ denotes the parameters of the Q network. The purpose of training the NN is to make

$$Q(s, \mathbf{a}, \theta) \approx Q^*(s, \mathbf{a}) \tag{3.31}$$

According to the DQN algorithm [35], two NNs are used to solve the problem: the evaluation network and the target network, which are denoted as $eval\_net$ and

$target\_net$, respectively. The output of $eval\_net$ and $target\_net$ are denoted as $Q_e(s, \mathbf{a}, \theta)$ and $Q_t(s, \mathbf{a}, \theta')$, respectively. The evaluation network is continuously trained to update the value of θ, however, the target network only copies the weight parameters from the evaluation network intermittently (i.e., $\theta' = \theta$). The loss function is defined as

$$\mathrm{L}oss(\theta) = E[(y - Q_e(s, \mathbf{a}, \theta))^2]. \tag{3.32}$$

y represents the real Q value, and is is calculated as

$$y = w(s, \mathbf{a}, s') + \varepsilon \max_{\mathbf{a}' \in \mathcal{A}} Q_t(s', \mathbf{a}', \theta') \tag{3.33}$$

where ε is the learning rate.

Algorithm 3: Deep Q-Network algorithm training process

input: experience pool ep

output: well trained evaluation network

1. Randomly generate the weight parameter θ for the $eval\_net$. The $target\_net$ clones the weight parameters $\theta' = \theta$. $D = d = 1$.
2. **for** $u = 1, \ldots, U$ **do**
3. $t = 0$. $s = s_t$. $\mathbf{C} = \mathbf{C}_t$.
4. **while** $s \neq s_G$ **do**
5. Randomly generate a probability $p \in [0,1]$.
6. **if** $D > 200$ and $p \geq \varepsilon_{ch}$ **then**
7. The action $\mathbf{a}$ is chosen as $\mathbf{a} = \max_{\mathbf{a} \in \mathcal{A}} Q(s, \mathbf{a})$
8. **else**
9. Randomly choose the action from action set $\mathcal{A}$.
10. **end if**
11. The transmitter transmits with the selected beam pattern.
12. Throughout the whole time slot, the RF energy is accumulated in the harvesters' energy buffer, as $s'_i = s_i + \sum_{m=1}^{M} |\hat{g}_{im}^t|^2 a_m T$, $\forall i \in \mathcal{K}$. At the end of each time slot, each harvester feedbacks the energy level to the transmitter and the system state is updated to s'.
13. $ep(d, :) = \{s, \mathbf{a}, w(s, \mathbf{a}, s'), s'\}$. $d = d + 1$. If D reaches the maximum of experience pool, D remain constant, $d = 1$; otherwise, $D = d$. $s = s'$. $t = t + 1$. $\mathbf{C} = \mathbf{C}_t$.
14. After experience pool accumulates enough data, from D experiences, randomly select D_s experiences to train the NN $eval\_net$. Back-propagation method is applied to minimize the loss function $\mathrm{L}oss(\theta)$. Clone the weight parameters from $eval\_net$ to $target\_net$ after several time intervals.
15. **end while**
16. **end for**

In order to better train the NN, the experience reply method is utilized to remove the correlation between different training data. Each experience consists of the current system state s, the action $\mathbf{a}$, the next system state s', and the corresponding reward $w(s, \mathbf{a}, s')$. The experience is denoted by the set $ep = \{s, \mathbf{a}, w(s, \mathbf{a}, s'), s'\}$. The algorithm conserves D experiences and randomly select D_s (with $D_s < D$) experiences from D for training. After the training is finished, $target\_net$ clones all the weight parameters from the $eval\_net$ (i.e., $\theta^{'} = \theta$).

The algorithm used for the DQN training process is presented in Alg. 3.0.2. In each training iteration, D usable experiences ep are generated, and D_s experiences are selected for training the $eval\_net$. In total, there are U training iterations. For both the $eval\_net$ and the $target\_net$, the NNs have N_l hidden layers. In the learning process, $\mathbf{C}_t$ is used to denote all energy harvesters' channel conditions at a particular time slot t.

Dueling Double Deep Q-Network Algorithm

Dueling Double DQN has been proved to boost the performance of traditional DQN since it can effectively deal with the overestimating problem during the training process and improve the learning efficiency of the NN. Doubling DQN is a technique that strengthens the traditional DQN algorithm by preventing the overestimating to happen [35]. In traditional DQN, the $target\_net$ is utilized to predict the maximum Q value of the next state. However, the $target\_net$ is not updated at every training episode, which probably leads to an increase in the training error, and therefore complicate the training process. However, Doubling DQN shows superiority in solving that problem. In Doubling DQN, both the $target\_net$ and the $eval\_net$ are used to predict the Q value.

The $eval\_net$ is used to determine the optimal action to be taken for the system state s' as

$$y = w(s, \mathbf{a}, s') + \varepsilon \max_{a' \in \mathcal{A}} Q_e \left(s', \arg\max_{\mathbf{a} \in \mathcal{A}} Q(s', \mathbf{a}, \theta), \theta' \right). \quad (3.34)$$

It can be shown that following this approach, the training error considerably decreases [35].

In traditional DQN, the NN only has the Q value as the output. In order to speed up the convergence, Dueling DQN is applied by setting up two output streams from the NN. The first stream is the NN's output $V(\mathbf{s}, \theta, \beta)$. The second stream is called advantage output $A(s', \mathbf{a}, \theta, \alpha)$ and describes the advantage of applying each particular action to the current system state [36]. α and β are parameters that related to the two streams and the NN output. The Q value of the NN is denoted as

$$Q(s, \mathbf{a}, \theta, \alpha, \beta) = V(s, \theta, \beta) + (A(s', \mathbf{a}, \theta, \alpha) - \frac{\sum_{a'} A(s', \mathbf{a}, \theta, \alpha))}{|A|} \quad (3.35)$$

Dueling DQN can efficiently eliminate the extra training freedom, which speeds up the training [36].

Simulation Results

Simulated Channel Model

In order to evaluate the performance of the proposed algorithm, in this section the wireless channel from the transmitter to each harvester is modeled as a block fading channel. Both the Rician fading and Rayleigh fading model [40] are exploited. The established channel models are used to derive the simulation results.

It is supposed that within each time slot t, the channel is invariant and varies in different time slots [41]. At the end of each time slot, the energy harvester feedbacks the current energy level back to transmitter.

For Rician fading channel model, the signal often arrives at the receiver with a LOS components. The total gain of the signal is denoted as **g**.

$$\mathbf{g} = \mathbf{g}^s + \mathbf{g}^d \tag{3.36}$$

where $\mathbf{g}^s$ is the invariant LOS component and $\mathbf{g}^d$ denotes a zero mean Guassian diffuse component. In the system model, the transmitter has M antennas, and the channel from the transmitter antenna m to the energy harvester i can be denoted as

$$g_{im} = g_{im}^s + g_{im}^d \tag{3.37}$$

The magnitude of the faded envelope can be modeled using the Rice factor K^{r} such that

$$\mathrm{K}_{im}^{\mathrm{r}} = \frac{\rho_{im}^2}{2\sigma_{im}^2} \tag{3.38}$$

where ρ_{im}^2 denotes the average power of the main LOS component between the transmitter antenna m and energy harvester i, and σ_{im}^2 denotes the variance of the scatter component. The magnitude of the main LOS component can be derived as

$$|g_{im}^s| = \sqrt{2\mathrm{K}_{im}^{\mathrm{r}}}\sigma_{im} \tag{3.39}$$

since

$$\frac{1}{2}E[(|g_{im}^d|)^2] = \sigma_{im}^2 \tag{3.40}$$

The mean and the variance of g_{im} are denoted as $\mu_{g_{im}} = g_{im}^s$ and $\sigma_{g_{im}}^2 = \sigma_{im}^2$, respectively, or in polar coordinates,

$$g_{im} = r_{im}e^{j\theta_{im}} \tag{3.41}$$

Therefore, g_{im} is analyzed by both its amplitude and its phase. The probability density function of the amplitude r_{im} is given by

$$p_{r_{im}}(r_{im}, \mathrm{K}_{im}^{\mathrm{r}}, \sigma_{im}) = \frac{r_{im}}{\sigma_{im}^2} e^{-\frac{r_{im}^2}{2\sigma_{im}^2} - \mathrm{K}_{im}^{\mathrm{r}}} I_0\left(\frac{r_{im}\sqrt{2\mathrm{K}_{im}^{\mathrm{r}}}}{\sigma_{im}}\right), \tag{3.42}$$

where $I_0(\cdot)$ denotes the first kind zero order Bessel function. For zero mean phase angle, the probability density function can be denoted as

$$p_{\theta_{im}}(r_{im}, \mathrm{K}_{im}^{\mathrm{r}}, \sigma_{im}) = \frac{e^{-\mathrm{K}_{im}^{\mathrm{r}}}}{2\pi} \left[1 + \sqrt{4\pi \mathrm{K}_{im}^{\mathrm{r}}} cos\theta_{im} e^{\mathrm{K}_{im}^{\mathrm{r}} cos\theta_{im}^2} \left(1 - Q\left(\sqrt{2\mathrm{K}_{im}^{\mathrm{r}}} cos\theta_{im}\right)\right)\right] \quad (3.43)$$

where $\theta \in [-\pi, \pi]$, and $Q(\cdot)$ is the tail distribution function of the standard normal distribution. It is clear that the channel amplitude is not independent of the phase angle. When the Rice factor K^{r} is large enough, the rician distribution can be approximated by a Normal distribution [40]. Therefore, when $\mathrm{K}_{im}^{\mathrm{r}} \geq 10$, for the channel from the m-th antenna of the trasmitter to the i-th harvester, the fading channel main LOS component is denoted as

$$g_{im}^{s} = r_{im} e^{j\theta_{im}} \quad (3.44)$$

The probability density function of the amplitude and phase can be approximated by two Gaussian distribution

$$|g_{im}| \sim N\left(\sqrt{2\mathrm{K}_{ik}^{\mathrm{r}}}\sigma_{im}, \sigma_{im}^2\right) \quad (3.45)$$

and

$$\angle g_{im} \sim N\left(\arg(g_{im}^{s}), \frac{1}{2\mathrm{K}_{im}^{\mathrm{r}}}\right) \quad (3.46)$$

respectively, and

$$r_{im} = \sqrt{2\mathrm{K}_{im}^{\mathrm{r}}}\sigma_{im} \quad (3.47)$$

$\theta_{im} = \arg(g_{im}^{s})$.

In case of Non-line-Of-sight (NLOS) transmission from the transmitter to each energy harvester, the fading channel is characterized by a Rayleigh fading channel model.

As a result, both the real and imaginary components of the channel are Gaussian distributed with variance σ_{im}^2. The probability density function of g_{im} is denoted as

$$p_{g_{im}}(r_{im}, \theta_{im}) = \frac{r_{im}}{2\pi\sigma_{im}^2} e^{-\frac{r_{im}^2}{2\sigma_{im}^2}} \tag{3.48}$$

The amplitude and the phase are independent from each other, and their individual probability density function are given by

$$p_{r_{im}}(r_{im}, \theta_{im}) = \frac{r_{im}}{\sigma_{im}^2} e^{-\frac{r_{im}^2}{2\sigma_{im}^2}}, \tag{3.49}$$

$$p_{\theta_{im}}(r_{im}, \theta_{im}) = \frac{1}{2\pi}, \theta \in [-\pi, \pi]. \tag{3.50}$$

It can be noted that the Rician fading channel model with $\mathrm{K}_{im}^{\mathrm{r}} = 0$ corresponds to a Rayleigh fading channel, while as $\mathrm{K}_{im}^{\mathrm{r}} \to \infty$ the channel becomes invariant.

Simulation Results

A MIMO wireless communication system with nearby RF energy harvesters is simulated. The wireless transmitter has $M = 3$ antennas. The 3×3 communication MIMO channel matrix **H** is measured by two Wireless Open-Access Research Platform (WARP) v3 boards. Both WARP boards are mounted with the FMC-RF-2X245 dual-radio module, who is operated in 5.805-GHz frequency band. The Xilinx Virtex-6 FPGA operates as the central processing system and the WARPLab is used for rapid physical layer prototyping which is compiled by MATLAB [42].

Two transceivers are deployed as LOS transmission. Henceforth, the Eigen-value vector for communication channel($\mathbf{s}$) is $[13.52, 2.9, 0.6]^T$. The maximum transmitted power is $P = 12$W. $\rho^2 = -70$dBm. The information rate requirement R is 53bps/Hz. The

average channel gain from the transmitter to the energy harvester is -30dB. The energy conversion efficiency is 0.1. The duration of one time slot is defined as $T = 100$ms.

DQN is trained to solve for the optimal transmission strategies for each system state. The number of the hidden layers is 4. The number of the nodes of each hidden layer is 100. The network is fully connected. The total number of the weights is 30400. The activation function is ReLU. The learning rate is 0.1. The mini-batch is 10. The size of experience pool is 20000.

The exploration rate ε_c determines the probability whether the network selects an action randomly or follows the values of the Q-table. Initially, $\varepsilon_c = 1$ because the experience pool has to accumulate reasonable amount of data to train the NN. ε_c decreases with 0.001 at each training interval, and finally stops at $\varepsilon_{ch} = 0.1$, since the experience pool has collected enough training data. The structure of Dueling Double DQN is presented in Fig. 3.1. The software environment for simulation is TensorFlow 0.12.1 with Python 3.6 in Jupyter Notebook 5.6.0.

The action set $\mathcal{A}$ contains 13 actions: $[2,2,8]^T$, $[2,4,6]^T$, $[2,6,4]^T$, $[2,8,2]^T$, $[4,2,6]^T$, $[4,4,4]^T$, $[4,6,2]^T$, $[4,8,0]^T$, $[6,2,4]^T$, $[6,4,2]^T$, $[6,6,0]^T$, $[8,2,2]^T$, $[8,4,0]^T$. All these actions satisfy the information rate requirement. For the energy harvesting channel conditions, both the Rician and the Rayleigh channel fading models are exploited as described previously.

First, the optimal DQN structure is explored under fading channels. It is assumed that there are $K = 2$ harvesters. The channel from each antenna of the transmitter to each harvester is individually Rician distributed.

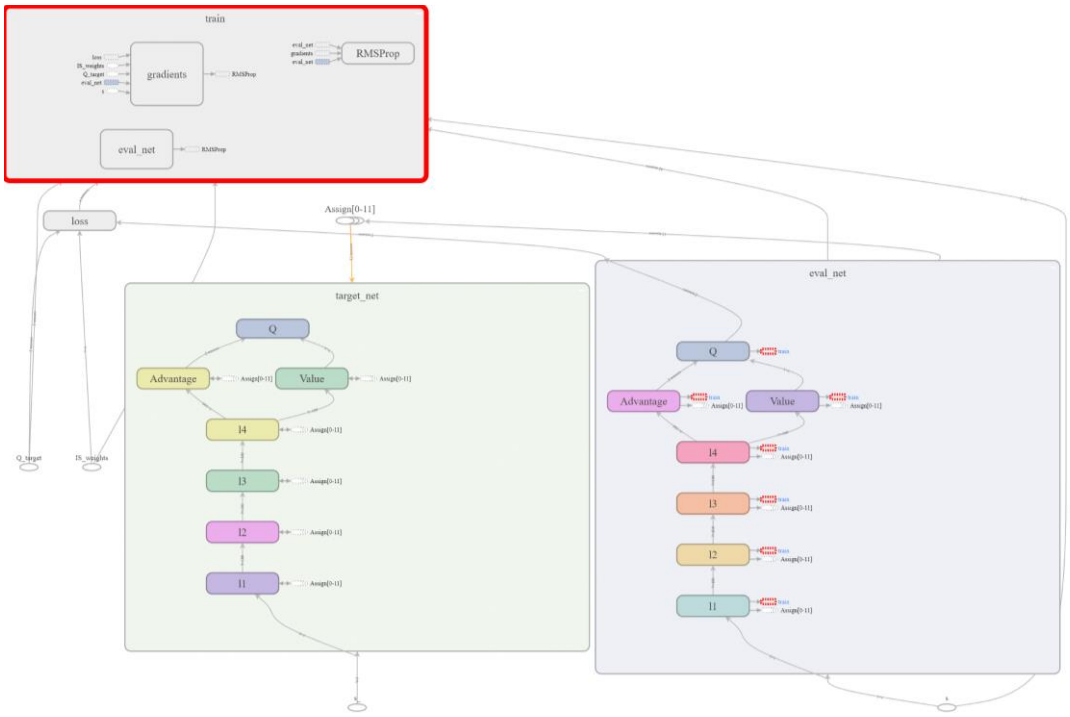

Figure 3.1: Dueling Double Deep Q-Network structure.

The LOS amplitude component of all channel links are defined as $r_{im} = 0.5$, with $i = 1,2$ and $m = 1,2,3$. For the LOS phase component of all channel links are defined as: $\theta_{11} = \frac{\pi}{4}$, $\theta_{12} = \frac{\pi}{2}$, $\theta_{13} = -\frac{\pi}{4}$, $\theta_{21} = -\frac{\pi}{2}$, $\theta_{22} = 0$, $\theta_{23} = \frac{3\pi}{4}$. According to Eqs. (3.45) and (3.46), the standard deviation of the g_{im} amplitude and phase are denoted as: σ_{im} and $\frac{1}{\sqrt{2K_{im}^{r}}}$, respectively. It is supposed that $\sigma_{im} = 0.05,\ \forall i, m$. Hence $\frac{1}{\sqrt{2K_{im}^{r}}} = (\frac{r_{im}}{\sigma_{im}})^{-1} = 0.1$, $\forall i, m$.

Using the fading channel model above, Fig. 3.2, shows how the structure of the NN, together with the learning rate can affect the performance of the DQN for a fixed number of training episodes (i.e., 40000). The performance of DQN is measured by the average number of time slots required to fully charge both harvesters. The average is obtained over 1000 testing data.

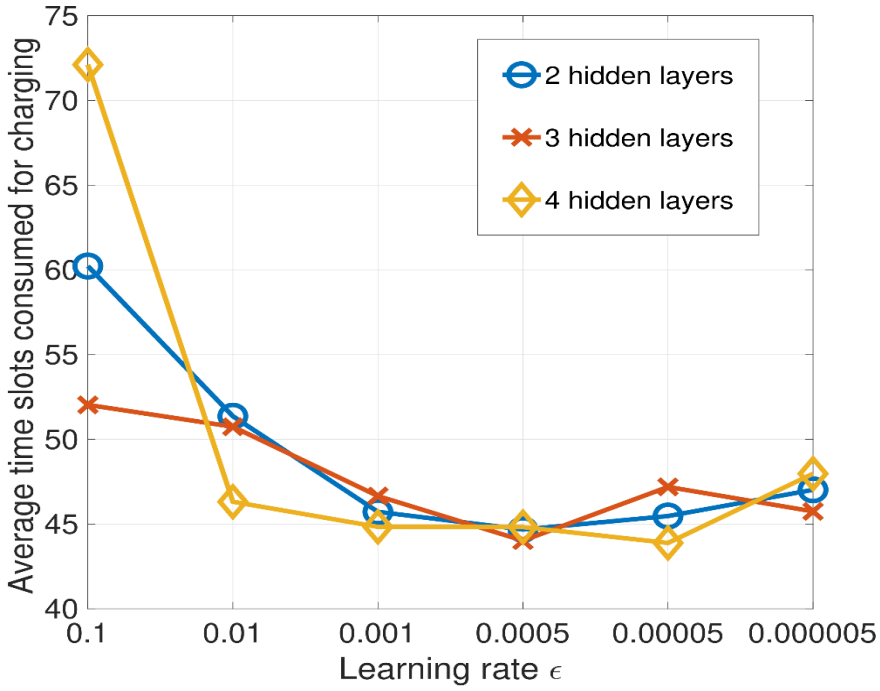

Figure 3.2: Deep Q-Network performance on different learning rate and number of hidden layers for the NN.

Fig. 3.2 shows that if the DNN has multiple hidden layers, a smaller learning rate is necessary to achieve better performance. When the learning rate is 0.1, the DQN with 4 hidden layers performs worse than a NN with 2 or 3 hidden layers. On the other side, when the learning rate decreases, it shows that the NN with 4 hidden layers and a learning rate of 0.00005 achieves the best overall performance. It is noted that there is not a monotonic decrease of the average number of time slots due to the stochastic nature of the channel that causes some fluctuations in the DQN optimization. After an initial improvement, decreasing the learning rate results in a slight increase in the average number of charging steps for all three NN structures. This is due to the fixed number of training episodes. Given longer training episodes, the DNN with smaller learning rate can achieve better performance. As a result, for all the simulations presented in this section, a DQN is

constructed with a 4 hidden layers DNN, with 100 nodes in each layers and a learning rate of 0.00005.

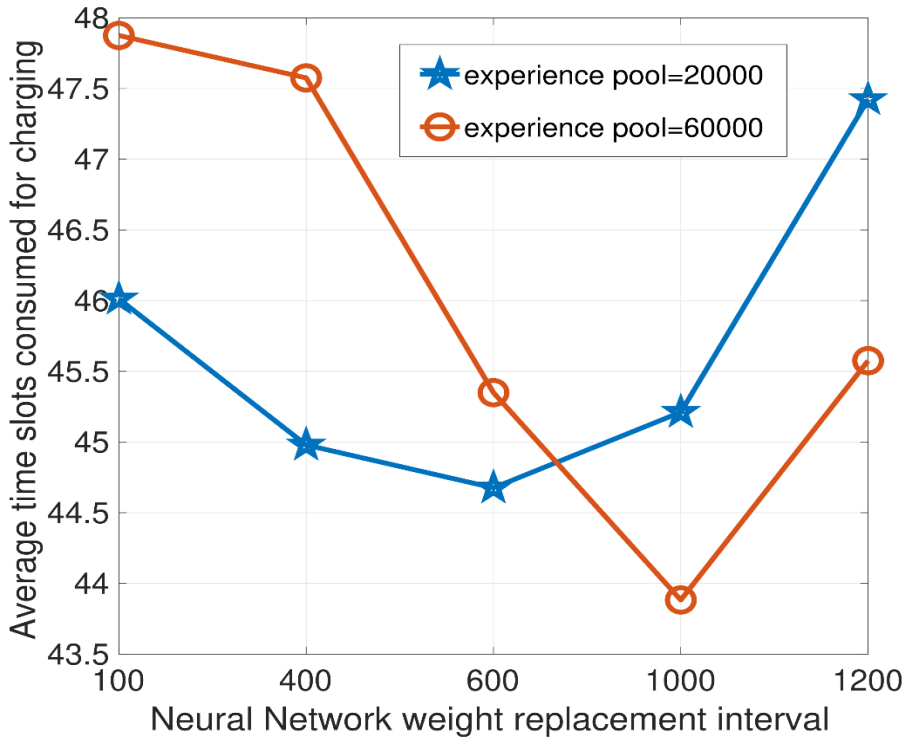

Figure 3.3: Deep Q-Network performance for different values of neural network replacement iteration interval and experience pool.

In Fig. 3.3, it can be observed that the size of the experience pool also affects the performance of DQN (40000 training episodes). To eliminate the correlation between the training data, only part of the experiences in the pool are selected for training. The number of extracted experiences from the experience pool are called mini batch, which is set to 10. Larger experience pool contains more training data. Selecting the mini batch data from the experience pool for training can eliminate the correlation between the training data. However, there is a need to balance the size of the experience pool and the $target\_net$ weight replacement interval. If the experience pool is large but the replacement iteration

interval is small, even if the correlation problem among the training data is addressed, the NN does not have enough training episodes to reduce the training error before the weight of the $target\_net$ is replaced.

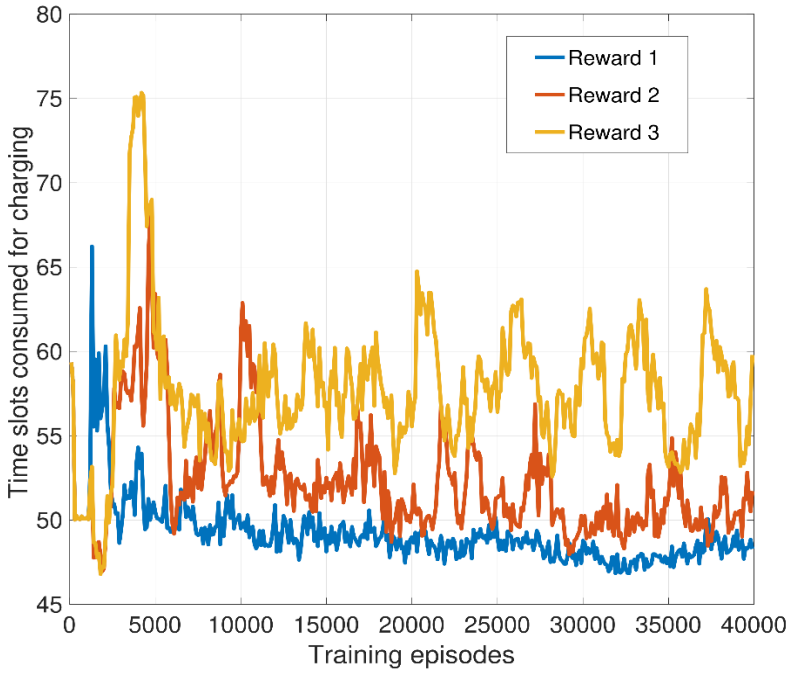

Figure 3.4: The Deep Q-Network performance for different reward functions.

From Fig. 3.3, it can be observed that a large replacement iteration interval doesn't achieve the optimal system performance as well. Fig. 3.3 shows that the optimal size of the experience pool and the optimal NN replacement iteration interval are 60000 and 1000, respectively.

Fig. 3.4 shows the impact of the reward function on the DQN performance. In this figure, the following three reward functions are considered. Reward 1 is defined as

$$w(s,\mathbf{a},s') = \begin{cases} 0, & s' = s_G \\ -\lambda(KB_{\max} - \sum_{i=1}^{K} \min(s'_i, B_{\max})), & otherwise. \end{cases}$$

Reward 2 is defined as

$$w(s, \mathbf{a}, s') = \begin{cases} 10, & s' = s_G \\ -1, & otherwise. \end{cases}$$

Reward 3 is defined as

$$w(s, \mathbf{a}, s') = \begin{cases} 1, & s' = s_G \\ -1, & otherwise. \end{cases}$$

So in total three different rewards are defined.

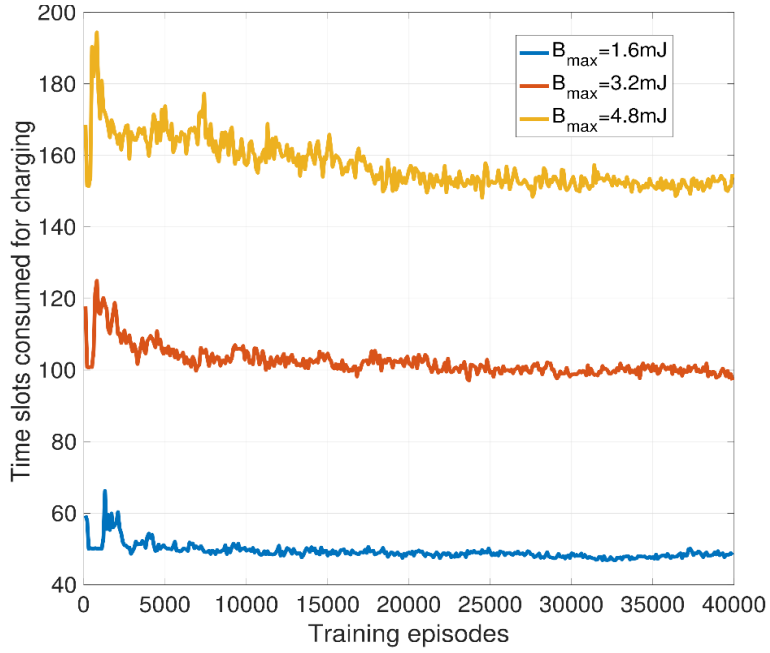

Figure 3.5: The Deep Q-Network performance for different energy buffer size $B_{\max}$.

The number of the users is $K = 2$ and $\lambda = 0.25$. All three reward functions are designed to minimize the number of time slots required to fully charge all the harvesters. However, from Fig. 3.4, it can be observed that the best performance can be obtained using Reward 1. In this case, the energy level accumulated by each harvester increases uniformly, which motivates the DQN to converge faster to the optimal policy. Both Reward 2 and Reward 3, instead, do not penalize states that unevenly charge the harvesters. Therefore,

both Reward 2 and Reward 3 require more iterations to converge to the optimal solution (not shown in the figure) due to the large number of system states to explore. Therefore, in the following simulations, the reward function Reward 1 is used. It is noted that, in both Fig. 3.4 and Fig. 3.5, 40000 training steps is averaged in every 100 steps in order to better show the convergence of the algorithm.

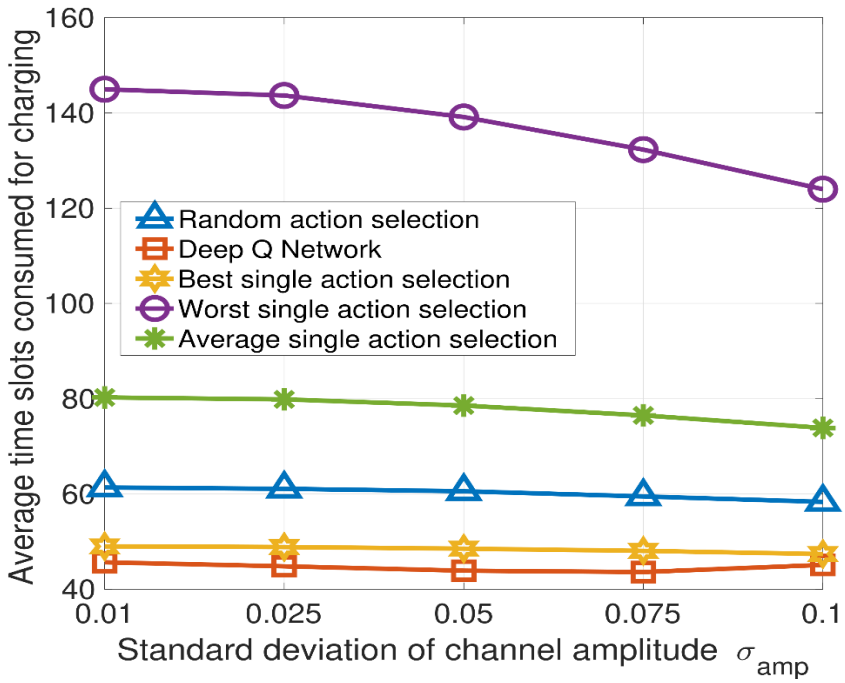

Figure 3.6: The comparison between Deep Q-Network and other action selection algorithms in Rician fading channel model.

In Fig. 3.5, as each energy harvester in the system is equipped with a larger energy buffer, the number of system states increases. Therefore, DQN requires more training episodes to converge to the steady transmit strategy for each system state. It can be observed that when $B_{\max} = 1.6$mJ, the system only needs less than 5000 training episodes to converge to the optimal strategy. When $B_{\max} = 3.2$mJ, the system needs around 12000 training episodes to converge to the optimal policy. While as $B_{\max} =$

4.8mJ, the system needs as many as 20000 training episodes to converge to the optimal strategy.

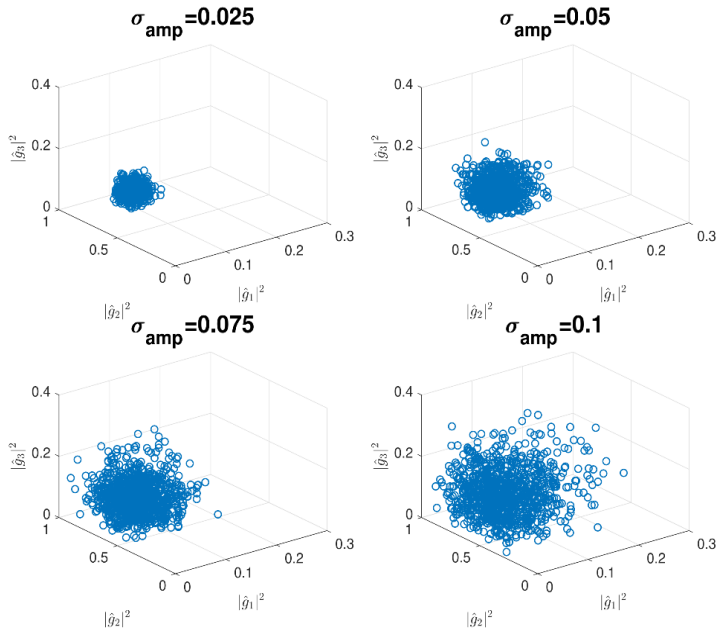

Figure 3.7: Simplified channel distribution with Rician fading channel model for different values of σ_{amp}.

In the following simulations, the impact of the channel model on optimization problem $\mathcal{P}_1$ is explored. For the Rician fading channel model, $\mathrm{K}_{im}^{\mathrm{r}} \geq 10$ and assumed to be same for all i, m. In this way, the Rician distribution is approximated as a Gaussian distribution. $r_{im} = 0.5,\ \forall i, m$, but it is allowed the standard deviation of both the amplitude and the phase of the channel to change to evaluate the performance on the system under different channel conditions. Since $r_{im} = 0.5$ and $\sqrt{2\mathrm{K}_{im}^{\mathrm{r}}} = \frac{r_{im}}{\sigma_{im}}, \frac{1}{\sqrt{2\mathrm{K}_{im}^{\mathrm{r}}}} = 2\sigma_{im}$. It is defined that $\sigma_{im} \leq 0.1$ to guarantee $\mathrm{K}_{im}^{\mathrm{r}} \geq 10$.

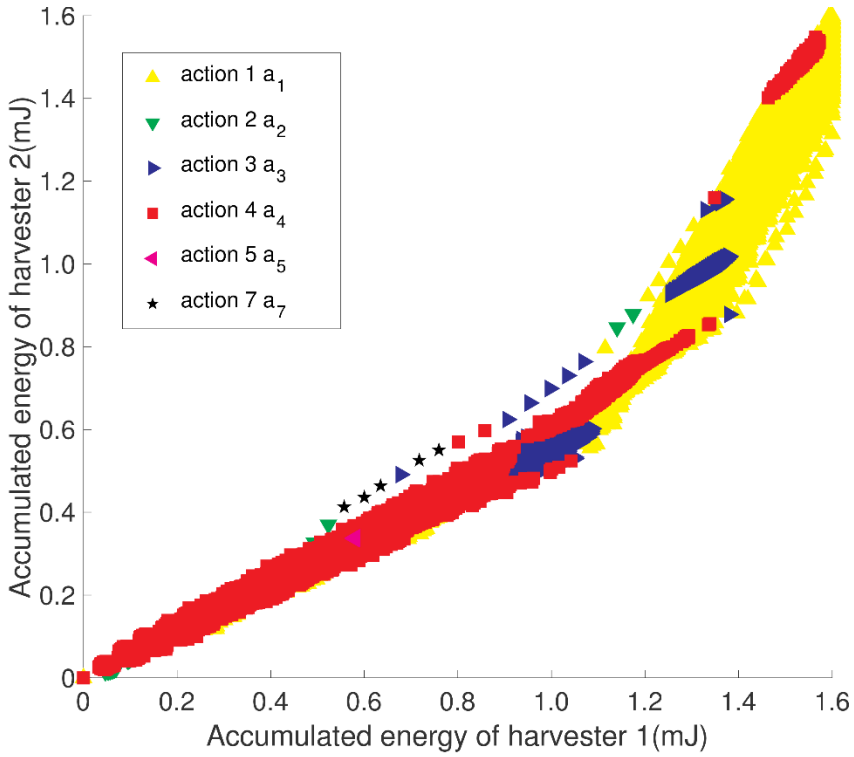

Figure 3.8: The action selection process of two harvesters scenario when $\sigma_{amp} = 0.05$.

In Fig. 3.6, the standard deviation $\sigma_{amp} = \sigma_{im},\ \forall i, m$ of the phase and amplitude of the channel are varied, and the performance of different algorithms are compared. Different heuristic approaches are implemented to compare with DQN. Single action selection selects a fixed single beam pattern for all transmissions. Random action selection selects an action at random at each transmission time slot. Fig. 3.6 is analyzed together with Fig. 3.7. It can be observed that increasing σ_{amp} results in a larger distribution of the values of $\hat{\mathbf{g}}$. No matter what the channel distributions are, DQN can always achieve the optimal system performance compared with all the other algorithms. In particular, DQN has a huge improvement compared to the random action selection policy. Comparatively, DQN requires 20% fewer time slots to fully charge all harvesters in the system.

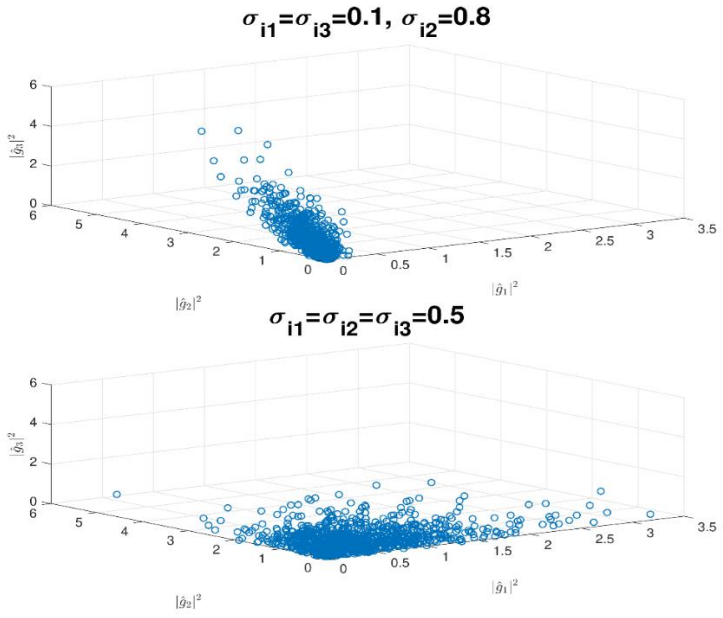

Figure 3.9: Simplified channel distribution with Rayleigh fading channel model with different values of σ_{im}.

Fig. 3.6 shows that the performance of the optimal policy achieved by DQN is around 10% better than the performance of a policy that selects the best single action for transmission. It seems the improvement is not large. However, it is noted that in order to determine this best single action, the transmitter needs to be able to estimate the channel conditions at the harvesters. The channel estimation at the energy harvesters is difficult to be implemented. Therefore, DQN obviously outperforms best single action selection on implementation. Even the action is randomly selected for all transmissions, there is only $\frac{1}{|\mathcal{A}|}$ probability that the optimal single action is selected. If the standard deviation of channel amplitude increases, the worst single action has a better performance. Due to the severe variations of the channel, there is a higher chance that the worst action selected at a time is the best action at another particular time slot.

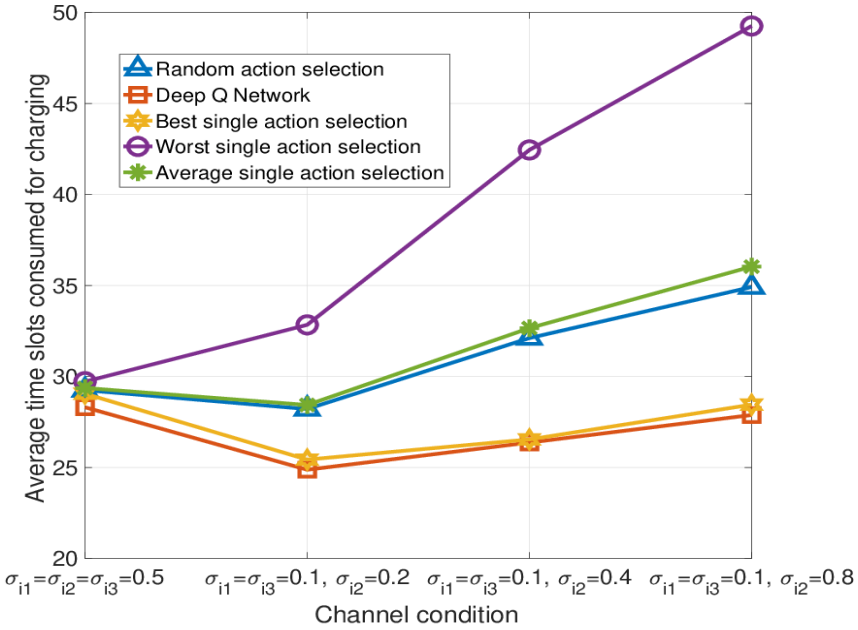

Figure 3.10: Comparison between Deep Q-Network and heuristic action selection algorithms in Rayleigh fading channel model.

To better explain the performance of the optimal policy, in Fig. 3.8, the action selected by DQN at a particular system state when $\sigma_{amp} = 0.05$ is plotted. From Fig. 3.6, it can be observed that when $\sigma_{amp} = 0.05$, the best single action selection can accomplish charging both harvesters in around 48 time slots. The best single action is the third action $\mathbf{a}_3 = [2,6,4]^T$. Correspondingly, the optimal policy determined by DQN can finish charging in around 43 time slots. To this end, Fig. 3.8 shows that the optimal charging strategy can be taken as two individual procedures: before harvester 1 accumulates 1.2mJ energy and harvester 2 accumulates 0.8mJ energy, action 4 $\mathbf{a}_4 = [2,8,2]^T$ is selected. After that, mostly, action 1 $\mathbf{a}_1 = [2,2,8]^T$ is selected. As defined above, if both the amplitude and the phase of the channel is Gaussian distributed with zero standard deviation, $\hat{\mathbf{g}}_1^0 = [0.05,0.59,0.11]^T$, and $\hat{\mathbf{g}}_2^0 = [0.04,0.19,0.51]^T$.

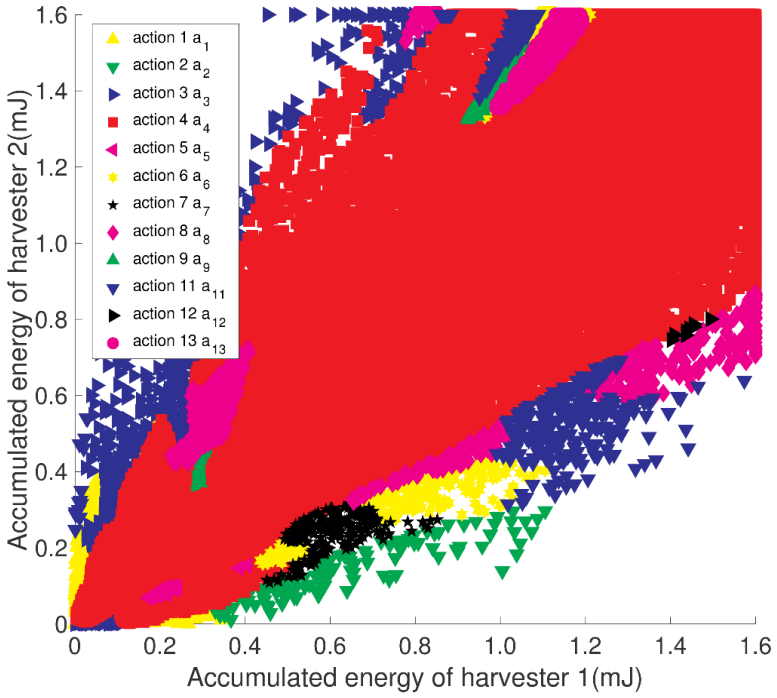

Figure 3.11: The action selection distribution of two harvesters scenario when $\sigma_{i1} = \sigma_{i3} = 0.1$, $\sigma_{i2} = 0.8$.

So when both the amplitude and the phase of the channel change, the simplified channel state information will be distributed around $\hat{\mathbf{g}}_1^0$ and $\hat{\mathbf{g}}_2^0$. The simulation result validates that a policy that selects either action 1 or action 4 with different probabilities can have better performance than the policy that only selects action 3. This explains the reason DQN outperforms the best single action selection, which shows the ability of DQN to determine the optimal policy.

When considering the channel model as Rayleigh fading channel model, two different conditions are considered: $\sigma_{i1} = \sigma_{i2} = \sigma_{i3}$ and $\sigma_{i1} = \sigma_{i3} \neq \sigma_{i2}$. The second condition considers that one of the antennas of the MISO (Multiple-Input Single-Output) communication system has a higher space diversity gain compared to the others. From Fig. 3.9, it can be observed that when $\sigma_{i1} = \sigma_{i3} = 0.1$ and $\sigma_{i2} = 0.8$, the variance of the

simplified channel vector $\hat{\mathbf{g}}$ is very large. When $\sigma_{i1} = \sigma_{i2} = \sigma_{i3} = 0.5$, the variance of the channel is even larger. The Rayleigh fading is featured as a NLOS transmission and the phase is uniformly distributed, thus there is large uncertainty in channel conditions.

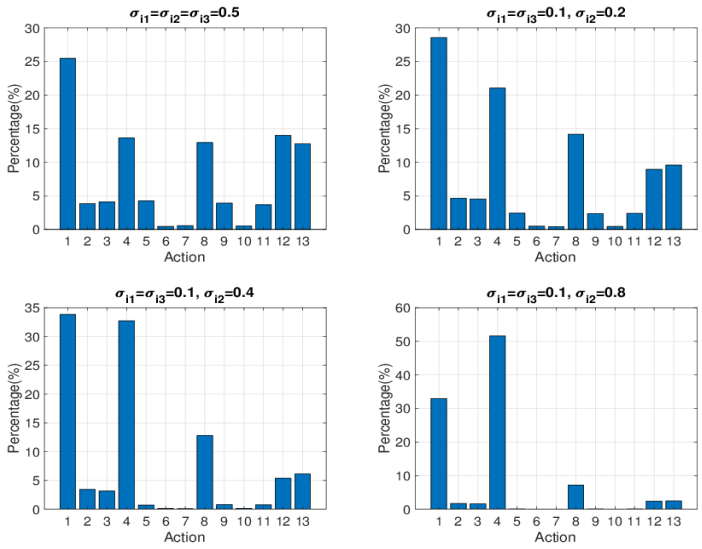

Figure 3.12: The possibility that the best single action is the action selected by the channel estimation of the first time slot.

In this case, DQN is still able to learn the channel conditions, and it determines the optimal strategy at each particular system state. Nonetheless, the performance of the optimal strategy is close to the performance of a random action selection or single action selection policy. In particular, Fig. 3.10 shows that when $\sigma_{i1} = \sigma_{i3} = 0.1, \sigma_{i2} = 0.8$, DQN doesn't have much advantages compared with a single action selection policy. When $\sigma_{i1} = \sigma_{i3} = 0.1, \sigma_{i2} = 0.8$, the best single action is to select action 4.

Fig. 3.11 validates that the action selected by DQN in the majority of the system states is action 4, which is the best single action. Fig. 3.12 shows that if the optimal action is chosen based on channel estimation of the first training time slot channel, it is hard to determine whether the chosen action is the best single action for this particular fading

channel environment. This proves the superiority of DQN, since DQN can make the optimal transmission decision without channel estimation.

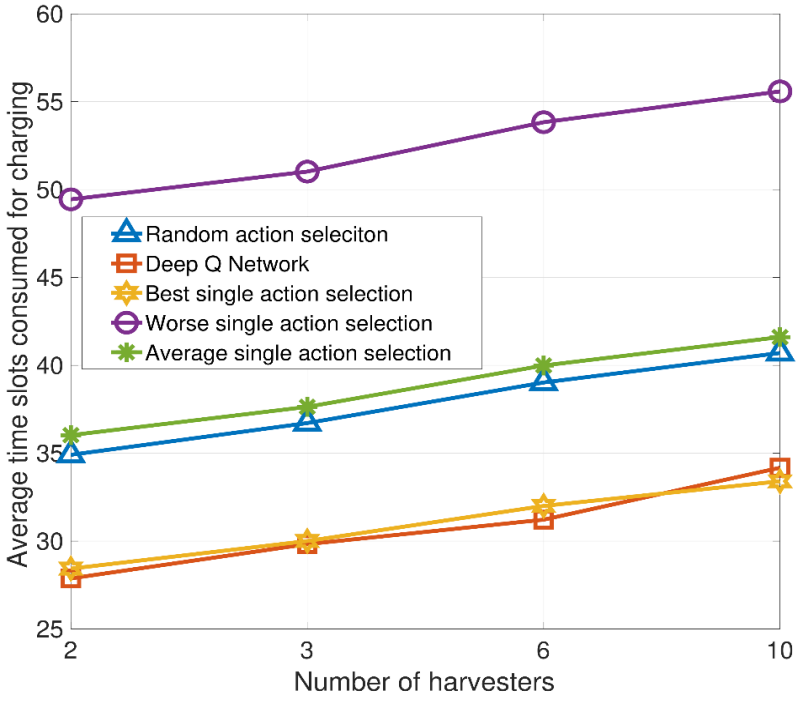

Figure 3.13: Deep Q-Network performance compared to other action selection algorithms when $\sigma_{i1} = \sigma_{i3} = 0.1$, $\sigma_{i2} = 0.8$.

Finally, Fig. 3.13 shows the performance of DQN together with the other heuristic action selection algorithms when the number of energy harvesters in the system is increased. Compared with the other algorithms, DQN achieves the optimal system performance.

Conclusions

In Chapter Three, the optimal wireless power transfer strategy for multiple RF energy harvesters is designed. DQN is used to fully charge the energy buffers of all energy harvesters in the shortest time while satisfying the information rate requirement of the communication system. The strong LOS communication channel conditions and time-

varying energy harvesting channel conditions are considered. The optimization problem is formulated as a MDP and solved with RL. Due to the large number of system states and the environment uncertainty, a DQN approach is adopted to find the optimal transmission strategy corresponding to each system state. Simulation results show the behavior of the optimal policy under different settings, as well as the performance of heuristic policies.

CHAPTER FOUR

A Multi-Armed Bandit Approach to Wireless Power Transfer

This chapter published as:Y. Xing, Y.Qian and L. Dong, "A multi-armed bandit approach to wireless information and power transfer", *IEEE Communication Letters*, 2020.

Introduction

SWIPT technology can support wireless devices that have RF energy harvesting capability and reduce over-reliance on batteries. To increase the harvested power, multi-transmitter SWIPT is configured. However, the wireless channels are usually assumed invariant and perfectly estimated [43, 44]. The SWIPT is also discussed with fading channels. However, either only a single transmitter is considered or the instantaneous channel gain or channel statistics are known [45].

In Chapter Four, the multi-transmitter SWIPT problem is considered over unknown block fading channels. Each wireless transmitter communicates with its corresponding information receiver, and all the transmitters collectively transfer power to multiple RF energy harvesters. The transmitters can be wireless base stations in an outdoor use case or WiFi access points in an indoor use case. The transmitters communicate with their receivers while wirelessly powering nearby micro devices [46]. Each transmitter has multiple antennas for beamforming. At the end of each time slot of a channel block, the receivers feed the SINR and the energy harvesters feed the levels of the harvested power back to a network coordinator. The coordinator determines the transmit beamforming weights and informs all the transmitters for the next time slot. Without any channel knowledge, the coordinator aims at fixing to the optimal beamforming weights for fair

energy harvesting while satisfying the SINR requirements of the receivers. Fair energy harvesting means that the minimum average received power among all the energy harvesters is maximized.

This task is formulated as a CMAB problem [47]. The combinatorial feature is necessary because a decision is made on meeting the SINR requirements of all the receivers and maximizing the minimum average received power among all the harvesters. Recently, CMAB has been used to deal with wireless communication issues [8]. In Chapter Four, the type-one UCB (UCB_1) algorithm [48] is proposed to solve the CMAB problem. Compared with the ε-greedy algorithm, the UCB_1 algorithm can converge to the optimal transmission strategy. Moreover, the convergence rate of the UCB_1 algorithm is improved with hierarchical arm selection. It can be implemented with less signaling overhead in terms of feedback from the information receivers and the RF energy harvesters. It has superior performance of satisfying the SINR requirements and maximizing the minimum average harvested power. When compared with the ideal case where optimization is done at every time slot with known channels, the proposed method can maximize the minimum average harvested power to about 80% of the ideal value while maintaining the communication quality.

Multiple Transmitters Wireless Information and Power Transfer System

System Model

As shown in Fig. 4.1, there are K information transmitters communicating with their intended receivers while transferring power to L nearby RF energy harvesters. Each

transmitter is equipped with D antennas. Each information receiver has one antenna and each harvester has one antenna.

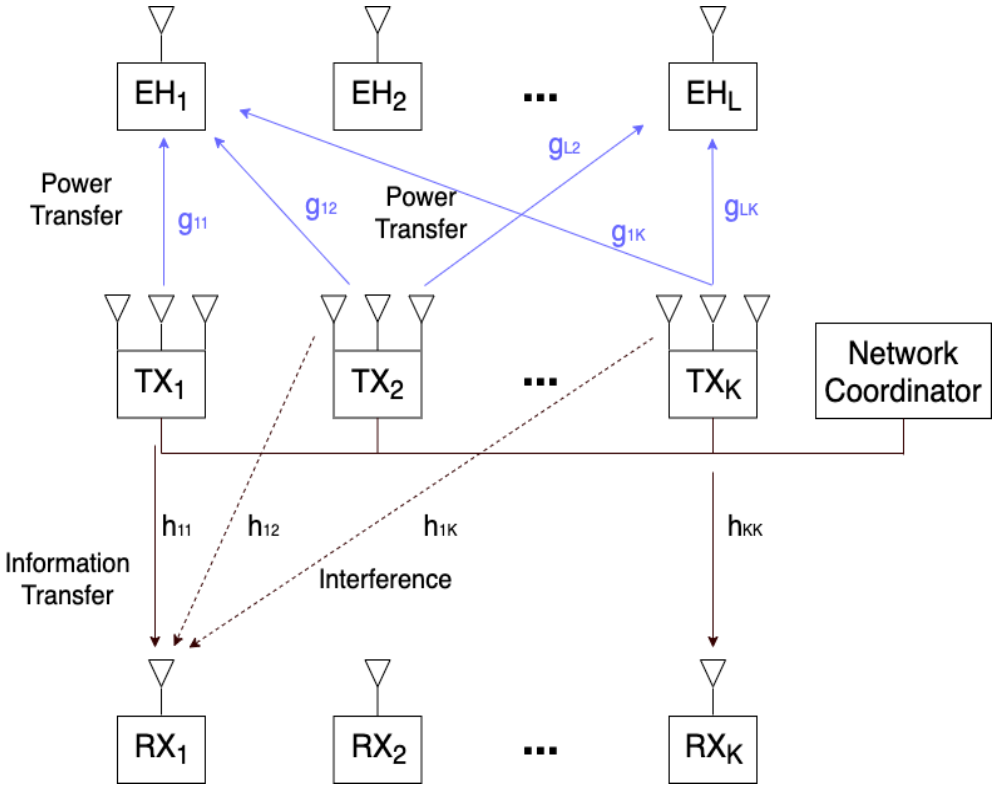

Figure 4.1: Multi-transmitter Simultaneous Wireless Information and Power Transfer with a network coordinator.

The baseband received signal at receiver i, $i \in \mathcal{K} = \{1,2,\dots,K\}$, is

$$y_i = \mathbf{h}_{ii}^H \mathbf{x}_i + \sum_{j \in \mathcal{K} \setminus \{i\}} \mathbf{h}_{ij}^H \mathbf{x}_j + z_i \qquad (4.1)$$

where $\mathbf{x}_i \in \mathbb{C}^{D \times 1}$ is the transmitted signal, $\mathbf{h}_{ii} \in \mathbb{C}^{D \times 1}$ denotes the baseband vector channel from transmitter i to its intended receiver, $\mathbf{h}_{ij} \in \mathbb{C}^{D \times 1}$ is the interference channel from transmitter j to receiver i, and $z_i \sim \mathcal{CN}(0, \sigma_n^2)$ is the circularly symmetric complex Gaussian noise with mean zero and variance σ_n^2.

The ith transmitter uses beamforming weight $\mathbf{w}_i$ to transmit symbol s_i,

$$\mathbf{x}_i = \mathbf{w}_i s_i \tag{4.2}$$

Suppose that

$$\mathrm{E}[|s_i|^2] = 1 \tag{4.3}$$

$$\| \mathbf{w}_i \|^2 = P_i \tag{4.4}$$

The SINR at the ith information receiver is

$$\mathrm{SINR}_i = |\mathbf{h}_{ii}^H \mathbf{w}_i|^2 / \left(\sum_{j \in \mathcal{K} \setminus \{i\}} |\mathbf{h}_{ij}^H \mathbf{w}_j|^2 + \sigma_n^2 \right) \tag{4.5}$$

The received power at the RF energy harvester specifies the harvested energy normalized by the symbol period. The received power at the lth harvester is

$$p_l = \eta \sum_{i \in \mathcal{K}} |\mathbf{g}_{li}^H \mathbf{w}_i|^2, l \in \mathcal{L} = \{1,2,\dots,L\} \tag{4.6}$$

where $\mathbf{g}_{li} \in \mathbb{C}^{D \times 1}$ is the vector channel from transmitter i to RF energy harvester l and η is the energy conversion efficiency. Assume that the noise power is negligible compared to the received signal power.

Suppose that the wireless channels $\{\mathbf{h}_{ii}\}, \{\mathbf{h}_{ij}\}$, and $\{\mathbf{g}_{li}\}$ experience Rician block fading. Each vector channel is invariant within a block but varies independent from block to block according to a Rician distribution.

Problem Formulation

There is a network coordinator that determines the beamforming weights $\{\mathbf{w}_i\}$ for the transmitters. Neither the network coordinator nor the transmitter knows the channels $\{\mathbf{h}\}$ or $\{\mathbf{g}\}$. The beamforming weights are designed to maximize the minimum harvested energy among the RF energy harvesters over time while maintaining the SINR for each information receiver. This optimization problem is formulated as a CMAB problem. The coordinator is the agent who can choose super arms. Each super arm consists of several

simple arms. A simple arm is the beamforming weight $\mathbf{w}_i$ for one transmitter. Therefore, the super arm is the combination of the beamforming weights of the transmitters, i.e.,

$$\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_K] \tag{4.7}$$

$\mathbf{W}$ contains all users beam weights. For a CMAB formulation, the agent aims to find the best super arm to achieve the best energy harvesting effect.

The optimization problem is

$$\mathcal{P}_1: \begin{matrix} \underset{\mathbf{W}}{\text{maximize}} & \min_{l \in \mathcal{L}} \mathrm{E}[p_l] \\ \text{subjectto} & ||\mathbf{w}_i||^2 = P_i \text{and} \mathrm{SINR}_i \geq \gamma_i, \forall i \in \mathcal{K} \end{matrix} \tag{4.8}$$

where γ_i is the SINR requirement of the ith receiver. The expectation indicates that it is to minimize the harvested energy over a long period of time of many channel blocks. The optimal beamforming weights will be used over channels that vary from block to block. However, the constraints are to be met within each channel block.

To limit the number of choices of super arms, the quantized beamforming weights are used as simple arms. Let $w_{ij}, j = 1,2, \dots, D$, be the jth element in the beamforming weight $\mathbf{w}_i$ of the ith transmitter. For each w_{ij}, we set δv_1 and δv_2 as its real and imaginary parts, where δ is the quantization step size and $v_1, v_2 \in \mathbb{Z}$. Of all the quantized beamforming weights, the specific weights are collected who satisfy

$$P_i - \varepsilon \leq \| \mathbf{w}_i^q \|^2 \leq P_i, \forall i \in \mathcal{K} \tag{4.9}$$

to form set

$$\mathcal{W} = \{[\mathbf{w}_1^q, \mathbf{w}_2^q, \dots, \mathbf{w}_K^q]\} \tag{4.10}$$

ε is a small constant that takes into account the quantization effect. The agent chooses the super arm from this set, i.e., $\mathbf{W} \in \mathcal{W}$.

When the agent uses super arm $\mathbf{W}_m$, $m = 1,2,\ldots,|\mathcal{W}|$, at time slot n, it gets a reward as

$$r_m(n) = \beta[\alpha_1(n)\alpha_2(n)\cdots\alpha_K(n) - 0.5] \cdot \min p_l(n) \tag{4.11}$$

$p_l(n)$ is related to channels $\{\mathbf{g}_{li}\}$ at time slot n. And $\alpha_i(n)$ is used to indicate whether the SINR requiremend is fulfilled.

$$\alpha_i(n) = \begin{cases} 1, & \mathrm{SINR}_i(n) \geq \gamma_i \\ 0, & \mathrm{SINR}_i(n) < \gamma_i \end{cases}, \forall i \in \mathcal{K} \tag{4.12}$$

where $\mathrm{SINR}_i(n)$ is related to channels $\{\mathbf{h}_{ii}\}$ and $\{\mathbf{h}_{ij}\}$ at time slot n. β is a normalization factor. The reward function is designed to maximize the minimum harvesting rate among the RF energy harvesters as well as satisfying the SINR requirement of each receiver.

At the beginning of time slot n, the network coordinator determines a specific arm and notifies the transmitters. Each transmitter transmits with the corresponding beamforming weights. By the end of time slot n, each information receiver feeds back one bit to the coordinator indicating whether the SINR requirement is met, and each RF energy harvester feeds back the level of the harvested power $p_l(n)$. Then, the network coordinator calculates its reward $r_m(n)$. The feedback overhead in the network is moderate.

Optimal Transmission Strategy with Combinatorial Multi-Armed Bandit

To guarantee convergence, the UCB_1 algorithm is utilized to solve $\mathcal{P}_1$. The algorithm achieves a balance between exploiting the best known arm and exploring unused arms for the CMAB problem [48].

Let $\mu_m(n)$ be the expected reward of selecting the mth super arm at time slot n, i.e.,

$$\mu_m(n) = \mathrm{E}[r_m(n)] \tag{4.13}$$

As the channels vary at different time slots, $r_m(n)$ is random and its expectation is difficult to find. Instead, $\mu_m(n)$ is calculated with an empirical average. Define $v_m(n)$ as the number of times the mth super arm is used from the first time slot to time slot $n-1$. The agent acquires the average reward $\mu_m(n)$ by using the mth super arm up to time slot $n-1$, that is

$$\mu_m(n) = \sum_{n':I_{n'}=m} r_m(n')/v_m(n) \tag{4.14}$$

where I_n indicates which super arm is selected at time n. An upper bound of the expected reward is

$$l_m(n) = \mu_m(n) + \sqrt{2\log(n-1)/v_m(n)} \tag{4.15}$$

The UCB_1 algorithm first selects each super arm in turn from the super-arm set $\mathcal{W}$. Each super arm is used in one time slot. In the next time slot, the channels vary and the next super arm in the set is used. Next, at time slot $n \geq |\mathcal{W}| + 1$, the UCB_1 algorithm will select a super arm with the maximum upper bound of its expected reward. That is

$$m(n) = \arg\max_{m\in[1,M]} l_m(n) \tag{4.16}$$

where $M = |\mathcal{W}|$.

Define the cumulative regret of selecting arms I_n, $n = 1,2,\dots,T$, over a period of T time slots as

$$R_T = \mu^* T - \sum_{n=1}^{T} r_{I_n}(n) \tag{4.17}$$

where μ^* is the expected reward of using the optimal super arm to the CMAB problem. The regret is a non-negative number because $\mu^* T$ is the best long-term reward the agent can get. With the UCB_1 algorithm.

$\mathrm{E}[R_T]$, i.e., the expected regret after T actions, is bounded [48].

$$\lim_{T\to\infty} \mathrm{E}[R_T]/T = 0 \tag{4.18}$$

Therefore, the UCB_1 algorithm converges in time.

For the CMAB problem, the large number of bandit super arms results in time-consuming initial exploration and slow algorithm convergence. The UCB_1 algorithm is improved with hierarchical arm selection to deal with these issues. In the improved algorithm, the arm selection is performed in multiple stages. The first stage starts with a large quantization step size δ_1. A small set of super arms $\mathcal{W}_1$ is generated with beamforming weights that satisfy the transmit power constraints. The UCB_1 algorithm converges and finds the optimal beamforming weights $\{\mathbf{w}_i^{\mathrm{opt}}\}_{i=1}^{K}$. In the second stage, a quantization step size $\delta_2 = \delta_1/2$ is utilized to generate candidate beamforming weights. For the ith transmitter, only the candidate beamforming weights that have a small $\| \mathbf{w}_i^q \mathbf{w}_i^{qH} - \mathbf{w}_i^{\mathrm{opt}} \mathbf{w}_i^{\mathrm{opt}H} \|_F$ are selected to form the super arm set $\mathcal{W}_2$. That is

$$W_2 = \{[\mathbf{w}_1^q, \mathbf{w}_2^q, \dots, \mathbf{w}_K^q] | \| \mathbf{w}_i^q \mathbf{w}_i^{qH} - \mathbf{w}_i^{\mathrm{opt}} \mathbf{w}_i^{\mathrm{opt}H} \|_F \leq d, i = 1,2,\dots,K\} \tag{4.19}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and d is the limit. The UCB_1 algorithm is executed again to update the optimal beamforming weights $\{\mathbf{w}_i^{\mathrm{opt}}\}_{i=1}^{K}$. This process can be continued in multiple stages. In each stage, the quantized step size is halved to generate the candidate beamforming weights. Only the candidate beamforming weights that are close to the optimal beam weights of the previous stage are selected to form a new super-arm set.

Algorithm 4: UCB_1 with Hierarchical Arm Selection

input: quantization step δ, quantization deviation ε, beam weight deviation d

output: optimal beam weight $\{\mathbf{w}_i^{\mathrm{opt}}\}_{i=1}^K$

for $s = 1, \ldots,$ max_num_stages **do**
  $\delta_s = \delta/2^{s-1}$.
  With δ_s and ε, generate quantified $\{\mathbf{w}_i^q\}_{i=1}^K$ that satisfy the power constraints.
  if $s = 1$ **then**
    $\mathcal{W}_s = \{[\mathbf{w}_1^q, \mathbf{w}_2^q, \ldots, \mathbf{w}_K^q]\}$.
  else
    $\mathcal{W}_s = \left\{[\mathbf{w}_1^q, \mathbf{w}_2^q, \ldots, \mathbf{w}_K^q] \middle|\, \| \mathbf{w}_i^q \mathbf{w}_i^{qH} - \mathbf{w}_i^{\mathrm{opt}} \mathbf{w}_i^{\mathrm{opt}H} \|_F \leq d, i = 1,2, \ldots, K\right\}$.
    $\mathcal{U}_{v_i} = \mathcal{U}_{v_i} + \{i\}$.
  end if
  $M = |\mathcal{W}_s|$
  for $n = 1, \ldots, M$ **do**
    Select super arm $m = n$ at time slot n.
    $\mu_m(n) = r_m(n)$, $v_m(n) = 1$.
  end for
  for $n = M + 1, M + 2, \cdots$ **do**
    With $v_m(n)$ and $\mu_m(n)$, select $m(n)$ according to Eq. (4.16)
    The coordinate receives feedback $\{\alpha_i(n)\}_{i=1}^K$ and $\{p_l(n)\}_{l=1}^L$ and updates $r_m(n)$, $v_m(n)$, and $\mu_m(n)$.
    if $m(n)$ converges (unchanged over a number of slots)
      $m^{\mathrm{opt}} = m(n)$, update $\{\mathbf{w}_i^{\mathrm{opt}}\}_{i=1}^K$; **break**
    end if
  end for
end for

The UCB_1 algorithm with hierarchical arm selection is presented in Alg. 4. As the process deepens in the hierarchy, the solution to the optimization problem becomes more accurate.

As shown in Eq. (4.16), the computational complexity is largely due to the selection of the super arm that gives the maximum upper bound among a pool of M candidate super arms. The limited memory space for the huge set of candidate super arms is another practical concern. The improved UCB_1 algorithm reduces M and therefore solves these problems.

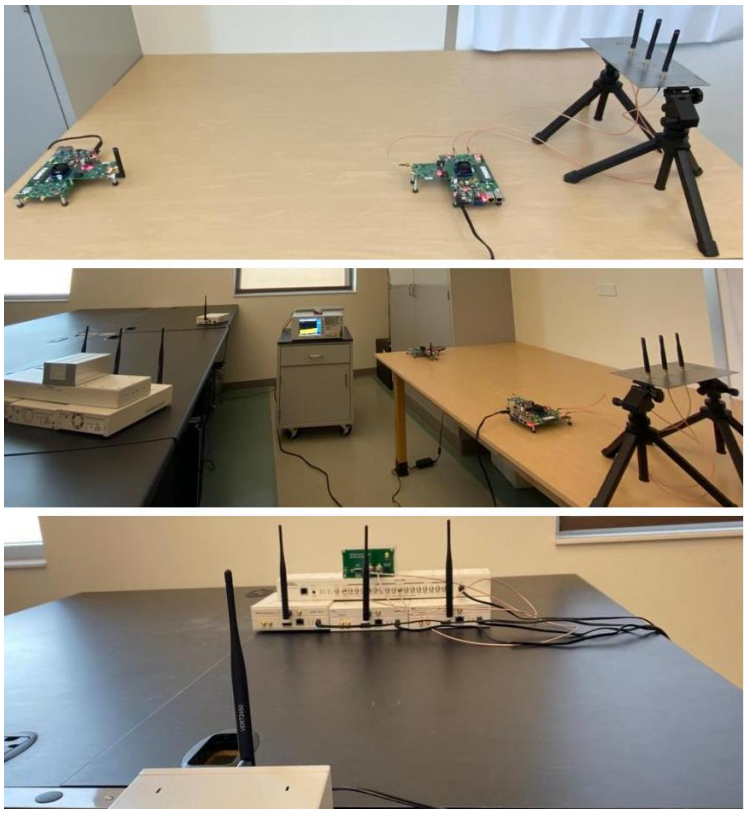

Figure 4.2: Channel measurement with the WARP and USRP boards.

Simulation Results

Indoor wireless channels are measured and used in the evaluation of the proposed method. The transceivers consist of the WARP v3 boards with FMC-RF-2X245 modules and the Universal Software Radio Peripheral (USRP) X310 and N210 boards with CBX daughterboards. An N9030A PXA Signal Analyzer is used to measure the received power at the locations of RF energy harvesters. Each transmitter has three antennas ($D = 3$). (Figure 4.2) The transmission is centered at 2.4 GHz with maximum transmit power $P = 16$ mW. The measured channels are used as the dominant components of the Rician fading channels and simulate the block fading effects with a K-factor of 10. The average channel gain from the transmitters to the receivers or the harvesters is -50 dB.

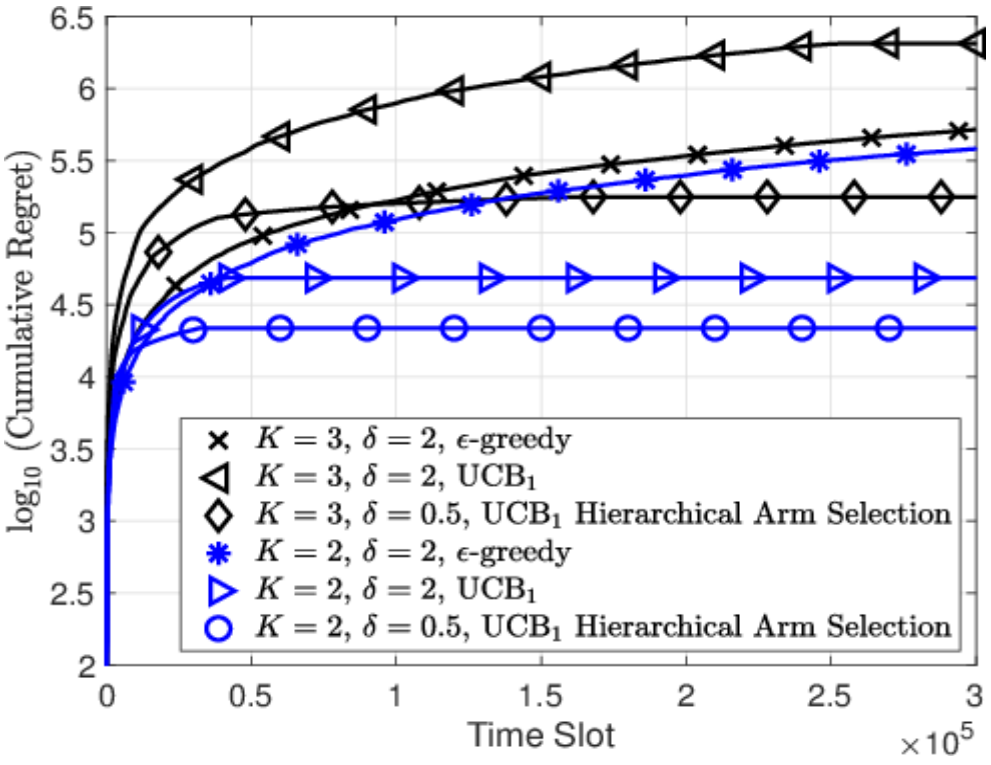

Figure 4.3: Comparison of cumulative regrets of the three algorithms. $L = 2$.

The scenarios where there are $K = 2$ or $K = 3$ transmitters are simulated. The SINR requirement is $\gamma = 0.5$ when $K = 2$ or $\gamma = 0.25$ when $K = 3$. There are $L = 2$ nearby RF energy harvesters.

The UCB_1 algorithm is compared with the ε-greedy algorithm. For the original UCB_1 algorithm, there are an excessive amount of bandit arms. Due to hardware limitations, it is defined that $\delta = 2$. The same $\delta = 2$ is used for the ε-greedy algorithm. The ε is defined as 0.2 with 20% exploration and 80% exploitation. The ε-greedy algorithm re-initiates after every 5000 time slots. Exploration is necessary because the ε-greedy algorithm tends to converge to a suboptimal strategy. With the hierarchical arm selection, four stages are considered. The quantization step starts with $\delta = 4$ in the first stage and goes down to $\delta = 2,1,0.5$ in the following stages. The algorithm is

computationally efficient, so δ can be pushed to 0.5. The arm selecting limits at the first three states are set as $d = 18,9,4.5$, respectively. The quantization constant is $\varepsilon = 0.25$. Fig. 4.3 compares the cumulative regrets of these methods. It shows that the improved UCB_1 algorithm converges the quickest.

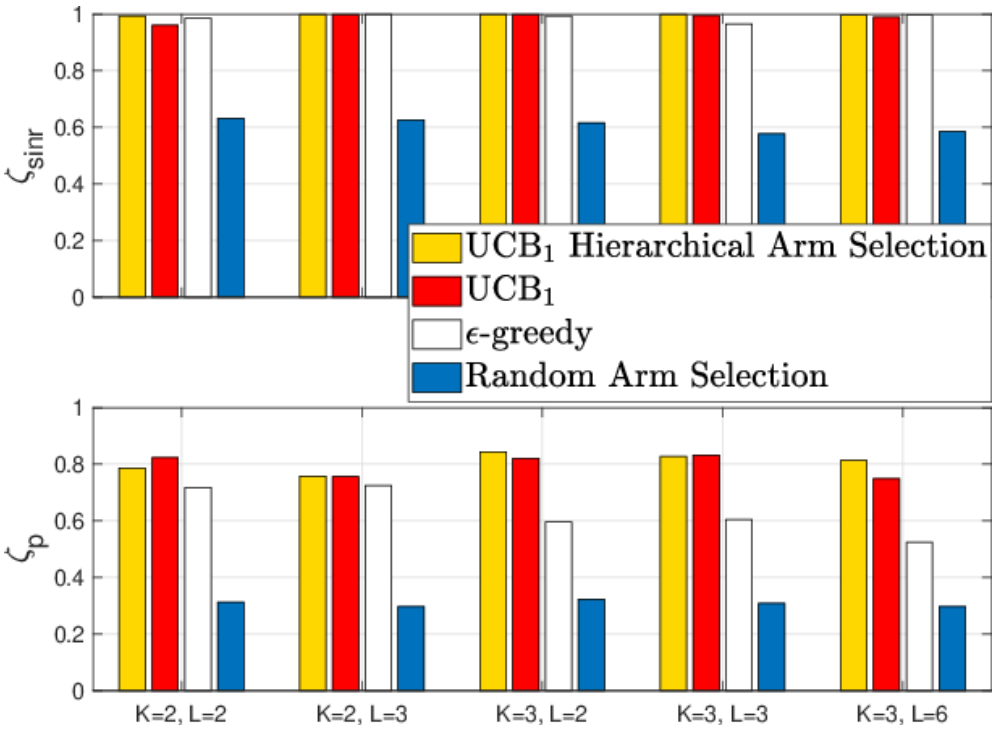

Figure 4.4: Comparison of UCB $_1$ with Hierarchical Arm Selection UCB $_1$, ε-greedy algorithms, and random arm selection against the benchmark CVX solver on receiver SINR satisfaction and harvested power.

Before any algorithm converges, there is signaling overhead that is due to feedback of SINR indicators $\alpha_1, \alpha_2, \ldots, \alpha_K$ from the K receivers and received power $p_l, p_2, \ldots, p_L$ from the L RF energy harvesters to the network coordinator. The amount of signaling overhead is linearly proportional to the number of time slots before algorithm convergence. After the algorithm converges, the transmitters' beamforming weights are fixed. ζ_{sinr} is defined as the proportion of time that the SINR requirements are satisfied over future time

slots. With these fixed weights, the minimum average harvested power of all the harvesters is $\hat{P} = \min_{l \in \mathcal{L}} \mathrm{E}[p_l]$.

Tab. 4.1 lists the signaling overhead (in terms of the number of time slots before algorithm convergence), ζ_{sinr}, and $\hat{P}$ of the algorithms. The improved UCB_1 algorithm effectively reduces the number of super arms. As a result, it has a smaller signaling overhead even with a smaller final-stage δ compared with the original UCB_1 algorithm. The UCB_1 algorithm has superior performance taking into account both receiver SINR requirements and minimum average harvested power. The ε-greedy algorithm has not converged during the test period.

Table 4.1. Performance comparison of the UCB_1 and ε-greedy algorithms on signaling overhead, receiver SINR satisfaction, and harvested power.

Algorithm	K	δ	Overhead	ζ_{SINR}	$\hat{P}(\mu W)$
UCB_1 hier. arm selec.	2	0.5	3.2×10^4	0.994	5.866
UCB_1	2	2	4.2×10^4	0.960	6.151
ε-greedy	2	2	$>3 \times 10^5$	0.985	5.360
UCB_1 hier. arm selec.	3	0.5	1.6×10^5	0.999	10.964
UCB_1	3	2	2.6×10^5	0.994	10.662
ε-greedy	3	2	$>3 \times 10^5$	0.999	7.749

In practice, the network coordinator has no knowledge of the channels. Given prefect channel knowledge at each time slot, the optimal solution of the optimization problem $\mathcal{P}_1$ can be obtained with the CVX solver [31]. This ideal solution is used as the benchmark against which the performance of the algorithms is evaluated. ζ_p is defined as the ratio of the minimum average harvested power with the super arm learned by the algorithm to the one from the ideal solution. The case where a random super arm is selected at each time slot is also compared with the proposed algorithm on ζ_{sinr} and ζ_p

calculated accordingly. Fig. 4.4 shows the performance of multiple algorithms against the ideal solution by the CVX solver. $K = 2,3$ and $L = 2,3,6$. When super arms are selected randomly, ζ_{sinr} is about 0.6 and ζ_p is merely about 0.3. The UCB $_1$ algorithms greatly improve the performance with ζ_{sinr} close to one and ζ_p increased to about 0.8.

Conclusion

The multi-transmitter SWIPT over unknown fading channels is treated as a CMAB problem. It is solved by the UCB algorithm with hierarchical arm selection. With moderate feedback from the information receivers and the RF energy harvesters, the algorithm can quickly converge to an optimal multi-antenna transmission strategy. It maximizes the minimum average received power among all the energy harvesters to about 80% of the ideal value, i.e., through optimization with known channels, while maintaining the communication quality at each information receiver.

CHAPTER FIVE

Deep Neural Networks for Optimized OFDMA Energy-efficient Transmission

Introduction

Applying multiuser transmission over parallel frequency channels is a promising technique, since multiple parallel subchannels can resist the transmission inference [49, 50]. Adapting to different channel conditions, the transmission strategy on each subchannel can be adjusted to maximize the communication quality. In [51], the authors dealt with a wireless communication system, which consisted of multiple parallel Gaussian broadcast channels. The authors determined a resource allocation strategy in order to maximize a weighted sum of rates under the sum power and additional receiver-specific rate constraints in the system.

To achieve certain amount of information transmission with minimum energy consumption, the energy efficiency is an important criterion to evaluated a wireless system performance [13, 52, 53, 54]. The energy efficiency can also be measured by the ratio between the information rate to the consumed power, such as in [52]. The authors applied a concave-convex fractional-programming framework to iteratively solve a energy efficiency optimization problem. The proposed Dinkelbach approach can determine the best power allocation strategy for a fixed energy efficiency in each iteration.

Recently, the energy-efficient wireless transmission in OFDMA system has been widely discussed [55, 56, 12, 57, 14, 58, 59]. In [56], the overall transmit power was minimized by assigning the Orthogonal Frequency Division Multiplexing (OFDM)

subcarriers to different users, determining the number of bits and allocating the transmit power on each subcarrier. The authors proposed an iterative algorithm for multiuser subcarrier assignment. The bit and power allocation algorithm was applied to each user on its allocated subcarrier after the subcarrier allocation was determined. In [12], the energy-efficient resource allocation was investigated in downlink together with uplink cellular networks with OFDMA. The authors aimed at maximizing the generalized energy efficiency for the downlink transmission, at the same time maximizing the minimum energy efficiency for each individual uplink case. The two optimizations were both established under certain prescribed QoS requirements. The authors provided both the optimal solution and a low-complexity suboptimal solution by exploring the inherent structure and property of the energy-efficient design. In [14], a single cell uplink communication system was discussed. The authors maximized the energy efficiency of the worst-case link subject to the information rate, transmit power, and subcarrier assignment constraints. An iterative algorithm was invented to solve the optimization problem with a generalized fractional programming theory and the Lagrangian dual decomposition. In order to further decrease the computational complexity, the authors devised algorithms to separate the subcarrier assignment and power allocation, which result in a suboptimal solution.

In Chapter Five, a practical energy efficiency optimization problem is formulated. The base station aims at maximizing the total energy efficiency while maintaining the achievable information rate requirement from the base station to each mobile user. The power allocation and channel assignment are conducted in order to solve the optimization problem.

To avoid high computational complexity, a novel method is proposed to find the optimal power allocation and subchannel assignment with DNNs instead of solving the optimization problem. DNN is a powerful tool in solving the complex optimization problems, especially non-convex optimization problems [60]. Compared with the traditional optimization algorithms, the DNN can achieve high precision in solving the optimization problems with extremely fast speed. DNN has been widely applied to solve the optimization problems in complicated communication systems [28, 61, 62]. In [61], the authors proposed a DNN based scheme for the real-time interference management over interference-limited channels. The DNN achieved orders of magnitude speedup in computational time compared to the state of the art interference management algorithm. In Chapter Five, two DNNs are trained individually, which determine the optimal power allocation and subchannel assignment, respectively. It consumes very long time to train the DNNs with a large number of training data offline. However, the well trained DNNs can be utilized online in a quick response. The simulation results prove the superiority of DNNs in solving the proposed optimization problem.

Multiuser Downlink OFDMA Data Transmission System

System Model

Considering a downlink OFDMA system: the base station serves K users. There are N subchannels. $\mathcal{N} = \{1,2,\ldots,N\}$. Any subchannel can be allocated to any user k. Each subchannel has same bandwidth W. The system model is shown in Fig. 5.1. $j_{k,n}$ is defined as the subchannel indicator. It indicates whether user k occupies the nth subchannel. $j_{k,n} =$

1 if user k occupies the nth subchannel; $j_{k,n} = 0$ if user k doesn't occupy the nth subchannel.

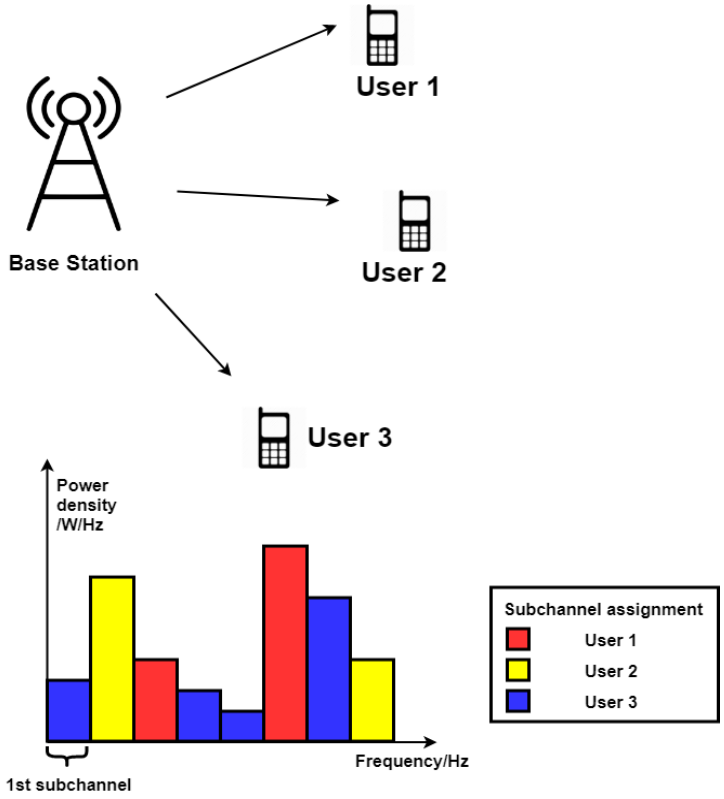

Figure 5.1: Multiuser downlink OFDMA wireless transmission system

The achievable information rate of the kth user is denoted as

$$r_k = \sum_{n=1}^{N} W \log_2 \left(1 + \frac{P_n j_{k,n} |h_{kn}|^2}{W N_0}\right) \tag{5.1}$$

where P_n is the power allocated on the $n$the subchannel. N_0 denotes the single-sided spectral density of additive Gaussian noise. h_{kn} denotes the frequency response of the nth suchannel of user k.

The total energy efficiency $\Gamma_{\mathrm{EE}}^{total}$ is denoted as

$$\Gamma_{\mathrm{EE}}^{total} = \frac{\sum_{k=1}^{K} r_k}{\sum_{k=1}^{K} \sum_{n=1}^{N} P_n j_{k,n}} \tag{5.2}$$

Problem Formulation

The optimization aims at maximizing the sum energy efficiency Γ_{EE}^{total}, at the same time, the achievable rate requirement R_k has to be satisfied. The maximum total transmit power is P. The problem is formulated as

$$\mathcal{P}_1: \begin{array}{ll} \underset{\{j_{k,n}\},\{P_n\}}{\text{maximize}} & \frac{\sum_{k=1}^{K} r_k}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n j_{k,n}} \\ subject\ to & r_k \geq R_k, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K} j_{k,n} \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} j_{k,n} \geq 1, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n j_{k,n} \leq P \end{array} \tag{5.3}$$

The rayleigh block fading channel model is used to characterize each subchannel. The channel variations of N subchannels can be seen as a frequency selective rayleigh block fading.

The channel gain vector from the base station to the kth mobile user is denoted as

$$\mathbf{a}_k = [|h_{k1}|^2, |h_{k2}|^2, \dots, |h_{kN}|^2]^T, k = \{1,2, \dots, K\} \tag{5.4}$$

All K channel gain vectors compose the channel gain matrix $\mathbf{A} \in \mathbb{R}^{K \times N}$ that

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_K] \tag{5.5}$$

The subchannel vector is regulated as

$$\mathbf{c}_l = [c_1^l, c_2^l, \dots, c_N^l], c_n^l = 1,2, \dots, K, l = 1,2, \dots, L \tag{5.6}$$

If $c_n^l = k$, then $j_{k,n}^l = 1$ and $j_{k',n}^l = 0$, $k' \in \mathcal{K} \backslash k$. $\mathcal{C} = \{\mathbf{c}_l\}$. $L = |\mathcal{C}|$. $j_{k,n}^l$ is defined as the subchannel occupation indicator corresponding to $\mathbf{c}_l$ in $\mathcal{C}$. The optimal subchannel assignment strategy is contained in $\mathcal{C}$. Each vector $\mathbf{c}_l$ in set $\mathcal{C} = \{\mathbf{c}_1, \dots, \mathbf{c}_L\}$ satisfies

$$\sum_{k=1}^{K} j_{k,n}^l \leq 1 \tag{5.7}$$

$$\sum_{n=1}^{N} j_{k,n}^l \geq 1 \tag{5.8}$$

Since the proposed non-convex optimization problem cannot be solved by any optimization tool, DNNs are implemented at the base station to find the optimal power allocation P_n and subchannel assignment $j_{k,n}$ for $\mathcal{P}_1$. The DNN is trained offline with a large number of simulated data. It reduces the complexity of online execution and increases the response speed of the base station. There are two NNs, one is called Power DNN, another one is called Subchannel DNN. Each DNN independently outputs in order to approach the optimal power allocation and subchannel assignment.

The base station acquires the channel gain matrix **A** and uses it as the input to the DNN. The input is

$$\mathbf{a}^{\text{in}} = \text{vec}(\mathbf{A}) \tag{5.9}$$

and $\mathbf{a}^{\text{in}} \in \mathbb{R}^{KN\times 1}$. For the Power DNN, it outputs

$$\mathbf{P}^{\text{out}} = [P_1^{\text{out}}, P_2^{\text{out}}, \dots, P_N^{\text{out}}]^T \tag{5.10}$$

The input-output relation of the Power DNN is defined as

$$\mathbf{P}^{\text{out}} = \mathbf{F}_1(\mathbf{a}^{\text{in}}) \tag{5.11}$$

For the Subchannel DNN, it outputs

$$\mathbf{I}^{\text{out}} = [I_1^{\text{out}}, I_2^{\text{out}}, \dots, I_L^{\text{out}}]^T \tag{5.12}$$

The output is normalized as $I_l^{\text{out}} \in [0,1]$. Each I_l^{out} corresponds to a particular $\mathbf{c}_l$. The optimal subchannel assignment index is selected as

$$l^* = \arg \max_{l\in\mathcal{L}} I_l^{\text{out}} \tag{5.13}$$

The input-output relation of the Power DNN is defined as

$$\mathbf{I}^{\text{out}} = \mathbf{F}_2(\mathbf{a}^{\text{in}}) \tag{5.14}$$

Both function $\mathbf{F}_1(\cdot)$ and function $\mathbf{F}_2(\cdot)$ derive the outputs based on $\mathbf{a}^{\text{in}}$ in order to maximize $\frac{\sum_{k=1}^{K} r_k}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{\text{out}} j_{k,n}^{l^*}}$. The DNN is trained using the optimal power allocation and subchannel assignment solved by Refined Exhaustive Search algorithm, which is described in the next section.

Since the channel experience rayleigh block fading, h_{kn} is generated with $\delta^2 = 1$, and the channel gains $|h_{kn}|^2$ follow the exponential distribution. $\mathbf{a}^{\text{in}}$ is taken as the input to two DNNs. In order to generate the outputs of the traning data, $\mathcal{P}_1$ has to be solved. However, problem $\mathcal{P}_1$ is non-convex and cannot be solved by CVX solver directly. Hence, a Refined Exhaustive Search algorithm is invented to approximate the optimal solution.

First, optimization $\mathcal{P}_2$ and $\mathcal{P}_3$ are formulated to acquire the upper and lower bound of the sum power and sum information rate, respectively.

Optimization $\mathcal{P}_2$ is formed as

$$\mathcal{P}_2: \begin{array}{ll} \underset{\{j_{k,n}\},\{P_n\}}{\text{maximize}} & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n j_{k,n} \\ subject\ to & r_k \geq R_k, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n j_{k,n} \leq P \\ & \sum_{k=1}^{K} j_{k,n} \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} j_{k,n} \geq 1, \quad \forall k \in \mathcal{K} \end{array} \tag{5.15}$$

The power allocation and channel assignment are solved as $\{P_n^*\}$ and $\{j_{k,n}^*\}$. The sum power lower bound is denoted as

$$P_s^L = \sum_{k=1}^{K}\sum_{n=1}^{N} P_n^* j_{k,n}^* \tag{5.16}$$

The sum rate lower bound is denoted as

$$R_s^L = \sum_{k=1}^{K} r_k \tag{5.17}$$

Optimization $\mathcal{P}_3$ is formulated as

$$\mathcal{P}_3: \begin{array}{ll} \underset{\{j_{k,n}\},\{P_n\}}{\text{maximize}} & \sum_{k=1}^{K} r_k \\ subject\ to & r_k \geq R_k, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K} \sum_{n=1}^{N} P_n j_{k,n} \leq P \\ & \sum_{k=1}^{K} j_{k,n} \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} j_{k,n} \geq 1, \quad \forall k \in \mathcal{K} \end{array} \tag{5.18}$$

The power allocation and channel assignment are solved as $\{P_n^*\}$ and $\{j_{k,n}^*\}$. The sum power upper bound is denoted as

$$P_s^U = \sum_{k=1}^{K} \sum_{n=1}^{N} P_n^* j_{k,n}^* \tag{5.19}$$

The sum rate upper bound is denoted as

$$R_s^U = \sum_{k=1}^{K} r_k \tag{5.20}$$

The sum power and the achievable rate are equally spaced between the upper and lower bounds. The spacing intervals are defined as

$$\Delta P = \frac{P_s^U - P_s^L}{T_1} \tag{5.21}$$

and

$$\Delta R = \frac{R_s^U - R_s^L}{T_2} \tag{5.22}$$

T_1 and T_2 are the spacing indexes. Multiple constraint thresholds are defined for power

$$P_{t_1} = P_s^U - \Delta P(t_1 - 1), t_1 = 1,2,\ldots,T_1 \tag{5.23}$$

and sum rate

$$R_{t_2} = R_s^U - \Delta R(t_2 - 1), t_2 = 1,2,\ldots,T_2 \tag{5.24}$$

For each $\mathbf{c}_l$ with particular power constraint P_{t_1}, convex optimization $\mathcal{P}_4$ is formulated in order to maximize the sum rate with the power constraint

$$\mathcal{P}_4: \begin{array}{ll} \underset{\{j_{k,n}^{l,t_1}\},\{P_n^{l,t_1}\}}{\text{maximize}} & \sum_{k=1}^{K} r_k^{l,t_1} \\ subject\ to & r_k^{l,t_1} \geq R_k, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{l,t_1} j_{k,n}^{l,t_1} \leq P \\ & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{l,t_1} j_{k,n}^{l,t_1} \leq P_{t_1} \\ & \sum_{k=1}^{K} j_{k,n}^{l,t_1} \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} j_{k,n}^{l,t_1} \geq 1, \quad \forall k \in \mathcal{K} \end{array} \tag{5.25}$$

where $j_{k,n}^{l,t_1}$ denotes the subchannel indicator of user k on subchannel n when the subchannel assignment strategy is $\mathbf{c}_l$ and the power constraint is P_{t_1}. P_n^{l,t_1} denotes the power allocation of user k on subchannel n when the subchannel assignment strategy is $\mathbf{c}_l$ and the power constraint is P_{t_1}. The optimal power allocation and subchannel assignment are solved as $\{P_n^{*,l,t_1}\}$, $\{j_{k,n}^{*,l,t_1}\}$, respectively.

$$l^* = \arg \max_{t_1 \in \mathcal{T}_1, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_1}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_1} j_{k,n}^{*,l,t_1}} \tag{5.26}$$

$$t_1^* = \arg \max_{t_1 \in \mathcal{T}_1, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_1}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_1} j_{k,n}^{*,l,t_1}} \tag{5.27}$$

The optimal power allocation and subchannel assignment are solved as $\{P_n^{l^*,t_1^*}\}$, $\{j_{k,n}^{l^*,t_1^*}\}$, respectively. The optimal achievable energy efficiency of all $\mathbf{c}_l$ and P_{t_1} combination is denoted as

$$\Gamma_{\mathrm{EE}}^{I\ *} = \max_{t_1 \in \mathcal{T}_1, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_1}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_1} j_{k,n}^{*,l,t_1}} \tag{5.28}$$

For each $\mathbf{c}_l$ with particular sum rate constraint R_{t_2}, convex optimization $\mathcal{P}_5$ is solved in order to minimize the sum power with the sum information rate constraint

$$\mathcal{P}_5: \begin{array}{ll} \underset{\{j_{k,n}^{l,t_2}\},\{P_n^{l,t_2}\}}{\text{minimize}} & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{l,t_2} j_{k,n}^{l,t_2} \\ subject\ to & r_k^{l,t_2} \geq R_k, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{l,t_2} j_{k,n}^{l,t_2} \leq P \\ & \sum_{k=1}^{K} r_k^{l,t_2} \geq R_{t_2} \\ & \sum_{k=1}^{K} j_{k,n}^{l,t_2} \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} j_{k,n}^{l,t_2} \geq 1, \quad \forall k \in \mathcal{K} \end{array} \tag{5.29}$$

where $j_{k,n}^{l,t_2}$ denotes the subchannel indicator of user k on subchannel n when the subchannel assignment strategy is $\mathbf{c}_l$ and the sum rate constraint is R_{t_2}. P_n^{l,t_2} denotes the power allocation of user k on subchannel n when the subchannel assignment strategy is $\mathbf{c}_l$ and the sum rate constraint is R_{t_2}.

The optimal power allocation and subchannel assignment are solved as $\{P_n^{*,l,t_2}\}$, $\{j_{k,n}^{*,l,t_2}\}$, respectively.

$$l^* = \arg\max_{t_2 \in \mathcal{T}_2, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_2}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_2} j_{k,n}^{*,l,t_2}} \tag{5.30}$$

$$t_2^* = \arg\max_{t_2 \in \mathcal{T}_2, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_2}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_2} j_{k,n}^{*,l,t_2}} \tag{5.31}$$

The optimal power allocation and subchannel assignment are solved as $\{P_n^{l^*,t_2^*}\}$, $\{j_{k,n}^{l^*,t_2^*}\}$, respectively.

The optimal achievable energy efficiency of all $\mathbf{c}_l$ and R_{t_2} combination is denoted as

$$\Gamma_{\mathrm{EE}}^{II\ *} = \max_{t_2 \in \mathcal{T}_2, l \in \mathcal{L}} \frac{\sum_{k=1}^{K} r_k^{*,l,t_2}}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{*,l,t_2} j_{k,n}^{*,l,t_2}} \tag{5.32}$$

The optimal total energy efficiency $\Gamma_{\mathrm{EE}}^{total^*} = \max(\Gamma_{\mathrm{EE}}^{I^*}, \Gamma_{\mathrm{EE}}^{II^*})$.

The corresponding optimal power $\{P_n^{l^*,t^*}\}$ and subchannel assignment $\{j_{k,n}^{l^*,t^*}\}$ can be determined. The Refined Exhaustive Search algorithm is shown in Alg. 5. Alg. 5 is used to generate the training data for the DNNs.

Algorithm 5: Refined Exhaustive Search Algorithm

input: channel gain $|h_{kn}|^2, \forall k \in \mathcal{K}, \forall n \in \mathcal{N}$

output: optimal power allocation $\{P_n^{l^*,t^*}\}$, optimal subchannel assignment $\{j_{k,n}^{l^*,t^*}\}$

1. The upper and lower bounds of the sum power and sum information rate P_s^U, P_s^L, R_s^U, R_s^L are calculated based on $\mathcal{P}_2$ and $\mathcal{P}_3$.
2. Equally space between the upper and lower bound of sum power and sum information rate. Generate multiple constraint thresholds for power P_{t_1} and sum information rate R_{t_2}.
3. For each $\mathbf{c}_l$ and P_{t_1} combination, solve $\mathcal{P}_4$. Acquire $\Gamma_{\mathrm{EE}}^{I^*}$ with Eq. (5.28).
4. For each $\mathbf{c}_l$ and R_{t_2} combination, solve $\mathcal{P}_5$. Acquire $\Gamma_{\mathrm{EE}}^{II^*}$ with Eq. (5.32).
5. The maximum total energy efficiency is calculated as $\Gamma_{\mathrm{EE}}^{total^*} = \max(\Gamma_{\mathrm{EE}}^{I^*}, \Gamma_{\mathrm{EE}}^{II^*})$. The optimal power allocation $\{P_n^{l^*,t^*}\}$ and subchannel assignment $\{j_{k,n}^{l^*,t^*}\}$ strategy are determined.

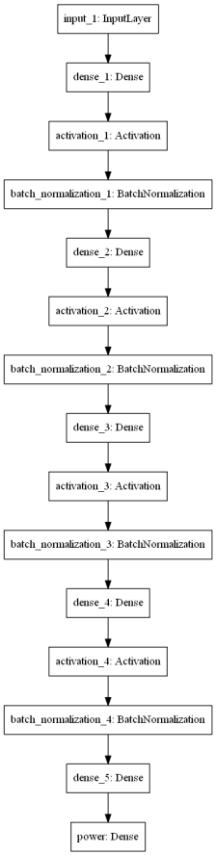

Figure 5.2: The structure of Power Deep Neural Network

Simulation Result

A downlink OFDMA system is simulated. The base station serves $K = 2$ mobile users with $N = 6$ subchannels. The maximum total transmitted power is $P = 2$ mW. The noise power spectrum density is $N_0 = -170$ dBm/Hz. The channel gain is -80 dBm. The bandwidth of each subchannel is defined as $W = 1$MHz.

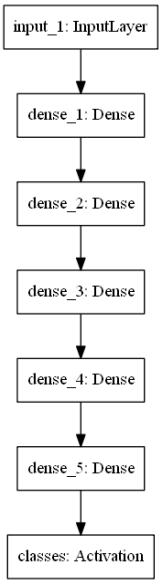

Figure 5.3: The structure of Subchannel Deep Neural Network

With Python Keras Toolkit [63], both the Power DNN and the Subchannel DNN are established with multiple layers. The Power DNN has 5 hidden layers and each layer contains 70 nodes. The Subchannel DNN also has 5 hidden layers and each layer has 50 nodes. The connectivity of each neural network is fully connected. The total number of the weights for the Power DNN is 20860 and the total number of the weights for the Subchannel DNN is 10900. The activation function for the hidden layers are ReLU function. For the Subchannel DNN, the activation function for the output layer is softmax function. Adam Optimizer is selected to train the DNNs. The learning rate for Power DNN and Subchannel DNN are 0.0001 and 0.001, respectively. The batch normalization is applied in training process in order to leverage the tranining effects. The structure of Power DNN and Subchannel DNN are shown in Fig. 5.2 and 5.3.

For training both Power DNN and Subchannel DNN, early stopping is used to avoid overtraining the neural network. The training process of the Power DNN is shown in in Fig. 5.4. It can be observed that the mean square error of the power decreases with the

training epochs increase. The training process of the Subchannel DNN is shown in Fig. 5.5. In Fig. 5.5, the categorical crossentropy loss decreases with the training epochs increase.

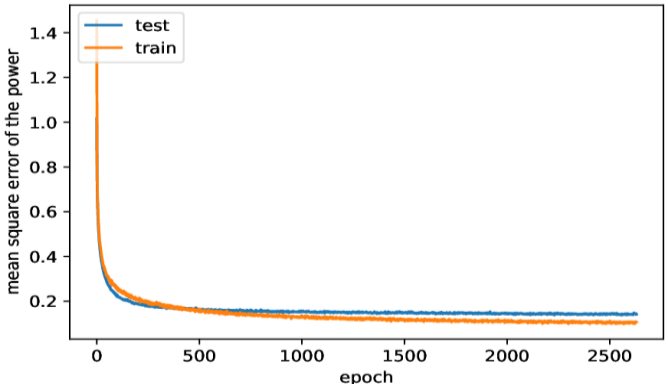

Figure 5.4: Mean square error of the power versus training epochs in Power Deep Neural Network

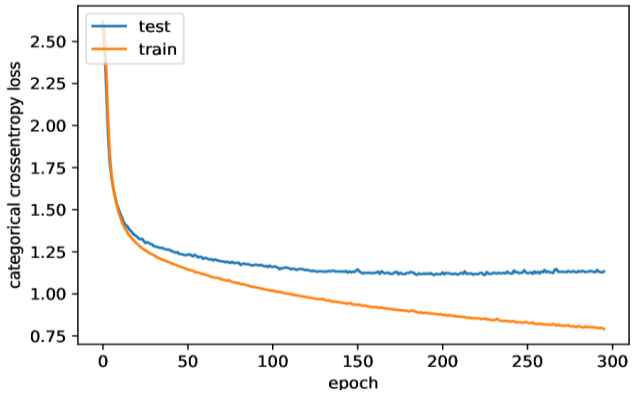

Figure 5.5: Categorical crossentropy loss versus training epochs in Subchannel Deep Neural Network

Random (exponential distributed) channel gain vectors $\{\mathbf{a}_k\}$ are utilized to train both Power and Subchannel DNNs. There are 8500 randomly generated channel gain vectors for $N = 4,5,6$ available subchannels conditions, respectively. The channel gains are the inputs to the DNNs. The corresponding optimal transmit power allocations $\{\hat{\mathbf{q}}\}$ and the optimal subchannel assignment of Problem $\mathcal{P}_1$ are generated by the invented Refined

Exhaustive Search algorithm. The Refined Exhuastive Search algorithm is solved in multiple steps with MATLAB CVX solver [31]. 80% of the generated data are used for training and 20% of the data are used for testing.

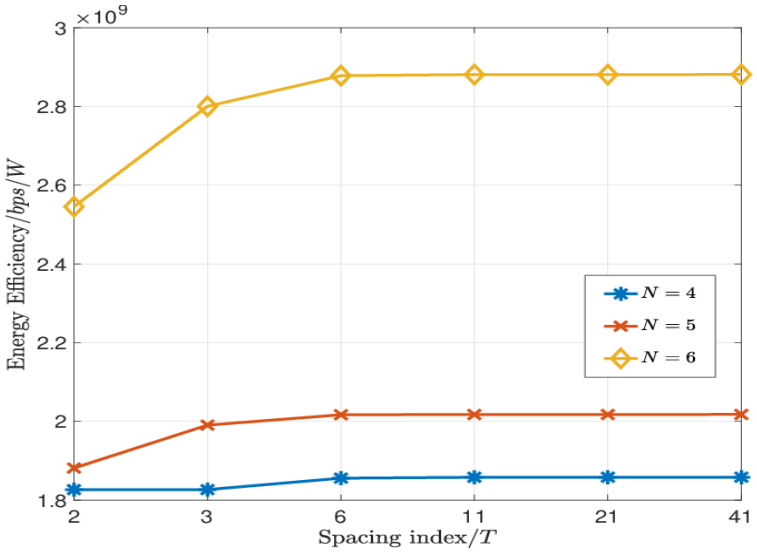

Figure 5.6: The energy efficiency of the RES algorithm versus spacing index.

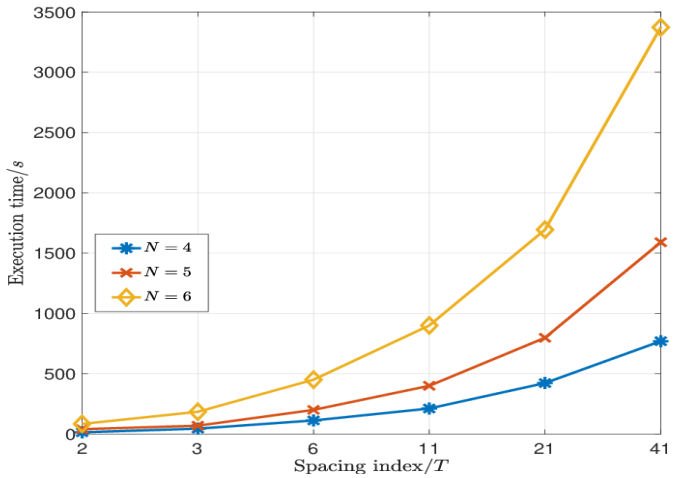

Figure 5.7: The execution time of the RES algorithm versus spacing index.

The maximum energy efficiency derived by two DNNs results is defined as

$$\Gamma_{\mathrm{EE}}^{\mathrm{NN}} = \frac{\sum_{k=1}^{K} r_k}{\sum_{k=1}^{K}\sum_{n=1}^{N} P_n^{\mathrm{out}} j_{k,n}^{l^*}} \tag{5.33}$$

$$l^* = \arg \max_{l \in \mathcal{L}} I_l^{\mathrm{out}} \tag{5.34}$$

l^* is the optimial subchannel assignment.

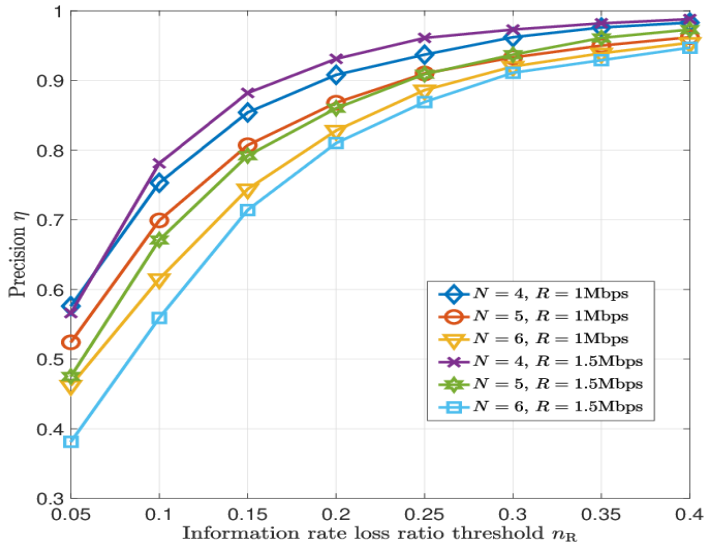

Figure 5.8: Precision η of the proposed DNNs versus the information rate loss ratio threshold n_{R}. The number of the mobile users is $K = 2$. The number of the available subchannels is $N = 4,5,6$. The information rate requirement is $R = 1,1.5$Mbps.

The information rate is denoted as

$$r_k^{\mathrm{NN}} = \sum_{n=1}^{N} W \log_2\left(1 + \frac{P_n^{\mathrm{out}} j_{k,n}^{l^*} |h_{kn}|^2}{W N_0}\right) \tag{5.35}$$

The maximum energy efficiency derived by the Refined Exhaustive Search algorithm is

$$\Gamma_{\mathrm{EE}}^{\mathrm{RES}} = \Gamma_{\mathrm{EE}}^{total*} \tag{5.36}$$

and the information rate requirement is R. The energy efficiency loss ratio is regulated as

$$\lambda_{\mathrm{EE}} = (\Gamma_{\mathrm{EE}}^{\mathrm{RES}} - \Gamma_{\mathrm{EE}}^{\mathrm{NN}})/\Gamma_{\mathrm{EE}}^{\mathrm{RES}} \tag{5.37}$$

The information rate loss ratio

$$\lambda_R^k = (R - r_k^{\mathrm{NN}})/R \tag{5.38}$$

If $\lambda_{\mathrm{EE}} < 0$ (or $\lambda_R^k < 0$), which means that the DNN has a better result than the CVX solver. Therefore, $\lambda_{\mathrm{EE}} = 0$ (or $\lambda_R^k = 0$). The energy efficiency loss ratio threshold is defined as n_{EE} and the information rate loss ratio threshold is defined as n_R. Of all of the N_T DNN testing outputs, transmissions with N_S particular transmit power allocations satisfy $\lambda_p \leq n_p$ (or $\forall k \in \mathcal{K}, \lambda_R^k \leq n_R$). The precision

$$\eta = N_S/N_T \tag{5.39}$$

is used to evaluate the DNN performance.

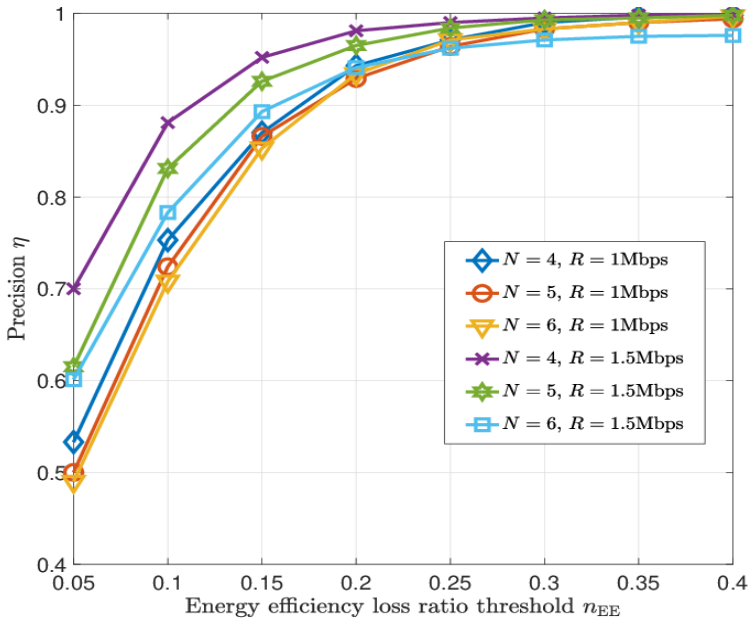

Figure 5.9: Precision η of the proposed DNNs versus the energy efficiency loss ratio threshold n_{EE}. The number of the mobile users is $K = 2$. The number of the available subchannels is $N = 4,5,6$. The information rate requirement is $R = 1,1.5\mathrm{Mbps}$.

Fig. 5.6 and 5.7 show the performance of the proposed Refined Exhaustive Search(RES) algorithm. The number of the mobile users is $K = 2$. The number of the

available subchannels is $N = 4,5,6$. The information rate requirement is $R = 1$Mbps. The spacing index is defined as $T = T_1 = T_2$. It can be observed that with a lager spacing index, the achieved energy efficiency gets higher, which however results in a longer execution time.

If the spacing index is large enough, the energy efficiency can converge. For the later simulation, the spacing index T is selected as 6 in generating the training data. Since in this way, a satisfactory energy efficiency performance can be ensured and the algorithm execution time is moderate.

Fig. 5.8 and 5.9 show the performance of DNNs in both information rate satisfaction and achieved energy efficiency. It can be observed that with more available subchannels, the information rate satisfaction decreases. Since same amount of training data are generated for $N = 4,5,6$ conditions. When the number of the subchannels increases, more training data are needed to guarantee a better performance. When the information rate requirement increases, the trained DNNs achieve better performance in the energy efficiency. At last, the execution time of DNNs and Refined Exhaustive Search algorithm are compared. The average execution time of DNNs is 6.46×10^{-7} second, however, the average execution time of Refined Exhuastive Search algorithm is 157.7 second. The DNNs obviously outperform the Refined Exhaustive Search algorithm in terms of algorithm running time.

Conclusions

A novel approach is proposed for downlink multiuser OFDMA data transmission. The base station uses the measured channel gains as the input to two different DNNs, which output the optimal transmit power allocation and subchannel assignment that can maximize

the system energy efficiency. At the same time, the information rate requirement at each communication receiver is satisfied. A Refined Exhaustive Search algorithm is invented to generate the training data. With larger spacing index, the Refined Exhaustive Search algorithm is shown to converge to the optimal solution. The simulation results show that the DNNs can dramatically reduce the execution time in solving the optimization problem while assuring an excellent system performance. The DNNs are trained offline with a large amount of simulated data, but it has very effective online performance.

CHAPTER SIX

Deep Reinforcement Learning for Optimized OFDMA Energy-efficient Transmission

Introduction

The energy-efficient wireless transmission in OFDMA system has been widely discussed [55, 56, 12, 57, 14, 58, 59]. In Chapter Five, a novel approach is proposed for downlink multiuser OFDMA data transmission. The base station uses the measured channel gains as the input to two different DNNs, which output the optimal transmit power allocation and subchannel assignment that can maximize the system energy efficiency. At the same time, the information rate requirement at each communication receiver can be satisfied.

Thus far, most research assumed an invariant channel environment and solved the resource allocation problems with complete knowledge about the channel. Only few papers considered the environmental dynamics and solved the long-term optimization problems [7, 64, 65, 66]. In [7], the authors proposed an energy efficiency optimization problem in energy harvesting Ultra Dense Network. Focusing on acquiring the optimal power control strategy, the power allocation strategy was determined without prior knowledge about energy arrival, user arrival and channel state information. The authors applied DDPG algorithm to solve the proposed problem and simulation results proved that the DDPG algorithm can enhance the energy efficiency performance significantly. In [66], the spectrum sharing problem in vehicular networks was investigated where multiple vehicle-to-vehicle (V2V) links reused the frequency spectrum pre-occupied by vehicle-to-

infrastructure (V2I) links. Collecting accurate instantaneous channel state information at the base station was impractical due to the fast variations of the channels in high mobile vehicular environments. The optimization problem was formulated as a multi-agent optimization problem and solved by DQNs. The multiple V2V agents can successfully learn to cooperate in a distributed way to simultaneously improve the sum capacity of V2I links and payload delivery rate of V2V links.

In order to solve the long-term optimization problems, DRL shows its superiority in decision making, thereby avoiding short-sighted result and achieving the long-term optimization goal. DQN was introduced to learn how to play complex games with very large number of system states, and unknown state transition probabilities [32]. In [67], the authors adapted the ideas underlying the success of DQN to the continuous action domain. Therefore, a DDPG algorithm was invented. DDPG can be taken as an actor-critic, model-free algorithm based on the deterministic policy gradient that operates over continuous action spaces. The proposed algorithm solved more than 20 simulated physics tasks and the performance was competitive with those found by a planning algorithm. However, the planning algorithm had full access to the dynamics of the domain.

In 5G wireless network, the volume of edge resources is limited, while the number and complexity of tasks in the network are increasing sharply [68, 69, 70, 71]. Therefore, providing effective services to network users with limited resources is an urgent issue. In order to improve the communication utility with limited resource, the joint resource allocation problems are urgently to be solved. However, these problems are difficult to be solved by traditional approaches. Thanks to the development of Artificial Intelligence, the Artificial Intelligence algorithms, especially DRL algorithms have been applied to solve

complex decision-making optimization problems. More recently, DRL has been applied to deal with complex communication problems and has shown to achieve superior performance [3, 33, 72, 34, 7, 66, 73, 74]. In [72], an Artificial Intelligent-assisted wireless network architecture was proposed. Based on the proposed architecture, the authors utilized DQN to solve the complex and high-dimensional joint resource allocation problem and achieved better performance compared to other resource allocation schemes. In [73], the author proposed an energy management algorithm based on the DDPG algorithm. With only one day's real solar data and the simulated channel data for training, the proposed algorithm showed excellent performance in the validation with about 800 days length of real solar data and achieved the optimal performance in terms of long-term average net bit rate. In [74], a resource allocation problem in vehicular communications was proposed, in which each V2V communication acted as agent and adopted Non-Orthogonal Multiple Access (NOMA) technology to share the frequency spectrum that pre-allocated to V2I communications. A multi-agent DDPG was applied which was capable of handling continuous high dimensional action spaces to find the optimal allocation strategy.

In Chapter Six, a real-time optimization problem is formulated. Within a time budget, the base station aims at maximizing the total energy efficiency while the information payload has to be delivered to each user. The dynamic power allocation and subchannel assignment are conducted in order to maximize the global benefits. DDPG algorithm is applied to solve the optimization problem. The DDPG algorithm can learn a dynamic operating strategy even with limit knowledge about the environment. In the proposed DDPG framework, the system state contains the partial channel information together with system performance information. Considering both the objective and the

constraints of the optimization problem, the reward function is properly designed. The simulation results prove the superiority of the DDPG algorithm in solving the proposed optimization problem. As the number of available subchannels increases, a hybrid approach is invented: a DDPG is utilized to determine the power allocation and a heuristic approach is used to determine the subchannel assignment.

The contributions of Chapter Six are as follows. First, a practical long-term optimization problem is formulated in a multiuser downlink OFDMA system. The real-time power allocation and subchannel assignment have to be determined in order to maximize the total energy efficiency while delivering the information payloads to multiple users within the time budget. Second, a DDPG algorithm is applied to solve the proposed long-term optimization problem. With limited channel information, the DDPG algorithm can optimize the spectrum management strategy for each time slot in order to meet the long-term optimization goal and constraints. Third, in order to solve the optimization problem, both the continuous power control and discretized subchannel assignment strategies have to be determined by the DDPG algorithm. Based on [75], the output of the action is redesigned due to the combinatorial action spaces and constraints on resource. At last, as the number of available subchannels increases, traditional DDPG algorithm cannot solve the proposed problem well because the high dimensional action space results in bad training effect. A hybrid algorithm is invented to solve the problem. Specifically, a DDPG algorithm is utilized to determine the power allocation, while a heuristic approach is used to determine the subchannel assignment strategy in a timely manner.

System Model

In a downlink OFDMA system, the base station disseminates data to K mobile users in N parallel subchannels. All subchannels have identical bandwidth W. It is assumed that independent channel fading across different subchannels but same within one subchannel. The time is assumed to be slotted by channel coherence time. In each time slot, the channel gain remains constant [7, 66]. The channel gain between the base station and the kth user on the nth subchannel at time t is defined as

$$h_{kn}(t) = \alpha_k g_{kn}(t) \tag{6.1}$$

where $g_{kn}(t)$ is the small-scale fading power component, which is assumed to be exponentially distributed due to the Rayleigh fading channel feature. $\alpha_k = l_k^{-\beta}$ is the path loss between the base station and user k. l_k is the geographical distance between the base station and user k. β denotes the path-loss exponent [76].

The achievable information rate of the kth user at time t is denoted as

$$r_k(t) = \sum_{n=1}^{N} W \log_2\left(1 + \frac{P_n(t)\rho_{k,n}(t)h_{kn}(t)}{WN_0(t)}\right) \tag{6.2}$$

where $P_n(t)$ is the power allocated on the $n$the subchannel at time t. $\rho_{k,n}(t) = 1$ indicates the nth subchannel is assigned to user k at time t, otherwise $\rho_{k,n}(t) = 0$. $N_0(t)$ denotes the single-sided spectral density of additive Gaussian noise.

$\Gamma_{\mathrm{EE}}^{(t)}$ indicates the total energy efficiency by time t, which is denoted as

$$\Gamma_{\mathrm{EE}}^{(t)} = \frac{\sum_{u=1}^{t}\sum_{k=1}^{K} r_k(u)\Delta t}{\sum_{u=1}^{t}\sum_{k=1}^{K}\sum_{n=1}^{N} P_n(u)\rho_{k,n}(u)\Delta t} \tag{6.3}$$

where the coherent time duration is Δt.

Problem Formulation

The time budget is defined as T_b time slots. Within the time budget T_b, the optimization aims at maximizing the total energy efficiency Γ_{tot} while delivering information payload B to each user. The power is capable to be allocated on all available subchannels. The subchannel assignment $\{\rho_{k,n}(t)\}$ and the power allocation $\{P_n(t)\}$ are required to solve the optimization. The optimization is shown in $\mathcal{P}_1$.

$$\mathcal{P}_1: \begin{array}{ll} \underset{\{\rho_{k,n}(t)\},\{P_n(t)\}}{\text{maximize}} & \Gamma_{tot} = \Gamma_{\text{EE}}^{(T)} \\ subject\ to & \sum_{t=1}^{T} r_k(t)\Delta t = B, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K} \rho_{k,n}(t) \leq 1, \quad \forall n \in \mathcal{N} \\ & \sum_{n=1}^{N} \rho_{k,n}(t) \geq 1, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K} \sum_{n=1}^{N} P_n(t)\rho_{k,n}(t) \leq P \\ & T \leq T_b \end{array} \tag{6.4}$$

where the constraint regulates no subchannel can be assigned to more than one user and one user can be assigned with more than one subchannel. The total transmit power in each time slot is less than P. $\mathcal{P}_1$ is a complicated long-term optimization problem. Both the energy efficiency and the success of payload delivery depends on the resource allocation strategy in each time slot. Both the DDPG algorithm and DQN algorithm are applied to solve the proposed long-term optimization problem.

In order to model the optimization problem as a MDP, the system state $\mathbf{s}_t$ is defined as

$$\begin{aligned} &\mathbf{s}_t = [h_{11}(t), \ldots, h_{KN}(t), B_1^{acc}(t), \ldots, B_K^{acc}(t), \\ &E^{acc}(t), B^{acc}(t), t] \in \mathbf{R}^{1\times(KN+K+3)} \end{aligned} \tag{6.5}$$

where the accumulated delivered payload of user k by time t is denoted as

$$B_k^{acc}(t) = \sum_{u=1}^{t} r_k(u)\Delta t \tag{6.6}$$

The accumulated energy consumption by time t is denoted as

$$E^{acc}(t) = \sum_{u=1}^{t} \sum_{k=1}^{K} \sum_{n=1}^{N} P_n(u)\rho_{k,n}(u)\Delta t \tag{6.7}$$

the accumulated sum delivered payload of all the users by time t is defined as

$$B^{acc}(t) = \sum_{k=1}^{K} B_k^{acc}(t) \tag{6.8}$$

The set that contains all system states is denoted by $\mathcal{S}$.

For the DQN framework, the action $\mathbf{a}_t$ includes both the subchannel assignment $\{v_n(t)\}$ and power allocation strategy $\{P_n(t)\}$.

The subchannel assignment is indicated by $v_n(t)$. If the nth subchannel is assigned to user k at time slot t, then $\rho_{k,n}(t) = 1$, $v_n(t) = k$.

$$v_n(t) = \underset{k \in \mathcal{K}}{\arg} \quad \rho_{k,n}(t) = 1 \tag{6.9}$$

The power allocation on the nth subchannel at time t is denoted as $P_n(t)$, which is discretized between $[0, P]$ that satisfies $\sum_{n=1}^{N} P_n(t) \leq P$.

The action at time t is defined as $\mathbf{a}_t$, which is denoted as

$$\mathbf{a}_t = [v_1(t), \ldots, v_N(t), P_1(t), \ldots, P_N(t)] \in \mathbf{R}^{1 \times 2N} \tag{6.10}$$

where $v_n(t) \in \mathcal{K}$. $P_n(t) \in [0, P]$. The action set is denoted as $\mathcal{A}$.

In order to implement the DDPG algorithm, the action has to be reformed into a continuous value format in order to achieve better training effect [75]. Hence, the action is defined as

$$\mathbf{a}_t = [\alpha_1(t), \alpha_2(t), \ldots \alpha_N(t), \alpha_{N+1}(t), \alpha_{N+2}(t), \ldots \alpha_{2N}(t)] \tag{6.11}$$

In DDPG algorithm, each output of the NN is normalized between 0 and 1. Therefore, $\alpha_n(t) \in [0,1]$. Here defines there are 2 users, $[0,1]$ are evenly divided into 2 ranges: $[0,0.5]$, $(0.5,1]$. If $\alpha_n(t) \in (0.5,1]$, $v_n(t) = 2$.

The subchannal assignment at time t is determined as

$$v_n(t) = \left\lfloor \frac{\alpha_n(t)}{\frac{1}{2}} \right\rfloor, n = 1,2,\ldots,N \tag{6.12}$$

The power allocation on the nth subchannel at time t is calculated as

$$P_n(t) = (P - \sum_{j=1}^{n-1} P_j(t))\alpha_{n+N}(t), n = 1,2,\ldots,N \tag{6.13}$$

where $P_1(t) = P\alpha_{N+1}(t)$, $P_2(t) = (P - P_1(t))\alpha_{N+2}(t)$. In this way, $\sum_{n=1}^{N} P_n(t) \leq P$.

In general, the selection of the action at each time slot depends on the current channel conditions and payload delivery conditions. The first system state is defined for $t = 0$ as

$$\mathbf{s}_0 = [h_{11}(0),\ldots,h_{KN}(0),0,\ldots,0] \tag{6.14}$$

and the final state $\mathbf{s}_T$ is defined for $t = T$ as

$$\mathbf{s}_T = [h_{11}(T),\ldots,h_{KN}(T),B,\ldots,B,E^{acc}(T),KB,T] \tag{6.15}$$

The accumulated delivered information payload $B_k^{acc}(t) = B$ also accounts for situations in which $B_k^{acc}(t) > B$.

Since the optimization aims at maximizing the sum energy efficiency at the same time delivering information payload to each user within the time budget, the optimization target together with the constraint are both related to the reward function [66]. In detail, the reward function is defined as

$$w(\mathbf{s}_t, \mathbf{a}_t) = \Gamma_{\mathrm{EE}}(t)\eta_1(t)\ldots\eta_K(t) \tag{6.16}$$

where

$$\eta_k(t) = \begin{cases} 1 & B_k^{acc}(t) \geq t\frac{B}{T_b} \\ 0 & B_k^{acc}(t) < t\frac{B}{T_b} \end{cases} \tag{6.17}$$

where $\eta_k(t)$ can motivate DDPG to learn the strategy to satisfy the payload requirement within the time budget T_b. $t\frac{B}{T_b}$ denotes the specific amount of information payload, which is required to successfully delivered to each mobile user by time t.

$\mathcal{P}_1 = (\mathcal{S}, \mathcal{A}, p, w)$ can be seen as a MDP from state $\mathbf{s}_0$ to state $\mathbf{s}_T$ on the Markov chain. $\{p_{\mathbf{s}_t,\mathbf{s}_{t+1}}(\mathbf{a}_t)\}$ denotes the state transition probabilities. Without knowledge about $\{p_{\mathbf{s}_t,\mathbf{s}_{t+1}}(\mathbf{a}_t)\}$, the algorithm aims to find, for each possible state $\mathbf{s}_t \in \mathcal{S}$, an optimal action $\mathbf{a}_t^*(\mathbf{s}_t)$ so that the system maximizes the sum energy efficiency Γ_{tot} while delivering information payload to each user. A generic policy can be written as $\pi = \{\mathbf{a}_t(\mathbf{s}_t): \mathbf{s}_t \in \mathcal{S}\}$.

Optimal Spectrum Management with Deep Deterministic Policy Gradient

Deep Q-Network

In this section, the RL approach is combined with a NN to approximate the system model in case of large states and actions sets [38]. In DQN, the cost function(Q function) is acquired by a well trained DNN. The Q function is denoted as $Q(\mathbf{s}_t, \mathbf{a}_t, \theta)$. θ denotes the parameters of the Q network. The purpose of training the NN is to make

$$Q(\mathbf{s}_t, \mathbf{a}_t, \theta) \approx Q^*(\mathbf{s}_t, \mathbf{a}_t) \tag{6.18}$$

There are two NNs in the structure of DQN: the evaluation network and the target network. Both of them have N_l hidden layers. The current system state $\mathbf{s}_t$ is taken as the input to the evaluation network, while the next system state $\mathbf{s}_{t+1}$ is the input to target network. $Q_e(\mathbf{s}_t, \mathbf{a}_t, \theta)$ and $Q_t(\mathbf{s}_t, \mathbf{a}_t, \theta')$ are the outputs of evaluation network and target network, respectively. The evaluation network is trained in each training epoch by updating θ. The target network periodically clones θ from the evaluation network $\theta' = \theta$.

The loss function is denoted as

$$\mathrm{L}oss(\theta) = E[(y - Q_e(\mathbf{s}_t, \mathbf{a}_t, \theta))^2]. \tag{6.19}$$

The real cost value y is denoted as

$$y = w(\mathbf{s}_t, \mathbf{a}_t, \mathbf{s}_{t+1}) + \gamma \max_{\mathbf{a}_{t+1} \in \mathcal{A}}, Q_t(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}, \theta') \tag{6.20}$$

where γ denotes the reward discount. The loss function is renewed in each learning epoch and θ is updated as well.

Algorithm 6: Deep Q-Network algorithm training process

input: experience pool ep

output: well trained evaluation network

1. Randomly generate the weight parameter θ for the $eval\_net$. The $target\_net$ clones the weight parameters $\theta' = \theta$. $D = d = 1$.
2. **for** $u = 1, \ldots, U$ **do**
3. $t = 0$. System state is $\mathbf{s}_t$.
4. **while** $\mathbf{s}_t \neq \mathbf{s}_T$ **do**
5. Randomly generate a probability $p \in [0,1]$.
6. **if** $D > 10000$ and $p \geq \varepsilon_{ch}$ **then**
7. The action $\mathbf{a}$ is chosen as $\mathbf{a}_t = \max_{\mathbf{a}_t \in \mathcal{A}} Q(\mathbf{s}_t, \mathbf{a}_t)$
8. **else**
9. Randomly choose the action from action set $\mathcal{A}$.
10. **end if**
11. $B_k^{acc}(t), E^{acc}(t)$ renewed and feedbacked to base station. At the end of each time slot, the channel updates. Base station estimates the channel and the system state changes into $\mathbf{s}_{t+1}$.
12. $ep(d,:) = \{\mathbf{s}_t, \mathbf{a}_t, w(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1}\}$. $d = d + 1$. If $D = D_{\max}$, $d = 1$; otherwise, $D = d$. $\mathbf{s}_t = \mathbf{s}_{t+1}$. $t = t + 1$.
13. After experience pool accumulates enough data, from D experiences, randomly select D_s experiences to train the NN $eval\_net$. Back-propagation method is applied to minimize the loss function $\mathrm{L}oss(\theta)$. Clone the weight parameters from $eval\_net$ to $target\_net$ after several time intervals.
15. **end while**
16. **end for**

The algorithm used for the DQN training process is presented in Alg. 6. The experience reply method is applied in DQN. The experience is store in a buffer ep. The

experience buffer size is $D_{\max}$ and in each learning epoch, D_s (with $D_s < D_{\max}$) experiences are selected from ep for training. The training lasts U time slots.

Deep Deterministic Policy Gradient

In DDPG, the optimal Q-value at time t is denoted as $Q^*(\mathbf{s}_t, \mathbf{a}_t)$, which is approximated by a critic network. The approximated Q value can be calculated as

$$\hat{Q}_t(\mathbf{s}_t, \mathbf{a}_t, \theta^Q) = w(\mathbf{s}_t, \mathbf{a}_t) + \gamma \max_{\mathbf{a}' \in \mathcal{A}'} Q_{t+1}(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}, \theta^Q) \quad (6.21)$$

where γ is the reward discount. θ^Q denotes the weight parameters of the critic network. θ^Q is updated in order to minimize the Temporal-difference error (TD-error)

$$L = \frac{1}{T}\sum_{t=0}^{T} (Q_t(\mathbf{s}_t, \mathbf{a}_t, \theta^Q) - \hat{Q}_t(\mathbf{s}_t, \mathbf{a}_t, \theta^Q))^2 \quad (6.22)$$

Besides two critic networks, there are two actor networks in the structure. In order update the policy π, a actor network is trained by sampled policy gradient [29]

$$\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_t \nabla_{\mathbf{a}_t} Q(\mathbf{s}_t, \mathbf{a}_t, \theta^Q) \nabla_{\theta^\mu} \pi(\mathbf{s}_t, \theta^\mu) \quad (6.23)$$

where θ^μ is denoted as the weight parameters of an actor network. The weight parameters of two target networks are denoted as θ' and μ', respectively.

θ' and μ' are updated as

$$\theta^{Q'} = \tau\theta^Q + (1-\tau)\theta^{Q'} \quad (6.24)$$

$$\theta^{\mu'} = \tau\theta^\mu + (1-\tau)\theta^{\mu'} \quad (6.25)$$

where τ denotes the updating parameter. The alogorithm is shown in Alg. 7. $ep \in \mathbb{R}^{D_{\max} \times (5K+7)}$ is regulated as the experience pool. $D_{\max}$ is the maximum capacity of the experience pool.

Algorithm 7: Deep Deterministic Policy Gradient algorithm training process

input: experience pool ep

output: well trained evaluation critic and actor networks

1. Randomly generate the weight parameter θ^Q and θ^μ for the evaluation critic network and evaluation actor network, respectively. The target critic network and target actor network clone the weight parameters from evaluation critic network and evaluation actor network respectively: $\theta^{Q'} = \theta^Q$. $\theta^{\mu'} = \theta^\mu$. $D = d = 1$.
2. **for** $u = 1, \ldots, U$ **do**
3. $t = 0$. System state is $\mathbf{s}_t$.
4. **while** $\mathbf{s}_t \neq \mathbf{s}_T$ **do**
5. The action $\mathbf{a}_t$ is chosen as $\mathbf{a}_t = \pi(\mathbf{s}_t, \theta^\mu) + noise$, where $noise$ is the exploration noise.
6. $B_k^{acc}(t), E^{acc}(t)$ renewed and feedbacked to base station. At the end of each time slot, the channel updates. Base station estimates the channel and the system state changes into $\mathbf{s}_{t+1}$.
7. $ep(d,:) = \{\mathbf{s}_t, \mathbf{a}_t, w(\mathbf{s}_t, \mathbf{a}_t), \mathbf{s}_{t+1}\}$. $d = d + 1$. If $D = D_{\max}$, $d = 1$; otherwise, $D = d$. $\mathbf{s}_t = \mathbf{s}_{t+1}$. $t = t + 1$.
8. Random sample D_s data from D experiences. $\hat{Q}_t(\mathbf{s}_t, \mathbf{a}_t, \theta^Q) = w(\mathbf{s}_t, \mathbf{a}_t) + \gamma \max_{\mathbf{a}' \in \mathcal{A}} Q_{t+1}(\mathbf{s}_{t+1}, \mathbf{a}_{t+1}, \theta^Q)$.
9. The weights of evaluation critic network and actor networks are updated.
10. The weights parameters of target critic network and target actor network are updated as $\theta^{Q'} = \tau\theta^Q + (1-\tau)\theta^{Q'}$; $\theta^{\mu'} = \tau\theta^\mu + (1-\tau)\theta^{\mu'}$.
11. **end while**
12. **end for**

Optimal Spectrum Management with Hybrid Approach

In OFDMA system, the channel gains on a large number of subchannels have to be precisely estimated in a timely manner, which leads to an unaffordably high sampling rate or implementation complexity [5]. Henceforth, in the practical system, the mobile users are only required to evaluate the channel condition on each subchannel as good or bad and feedback them to the base station. $\hat{p}_s$ is defined as subchannel selection threshold. $\hat{p}_s$ is same for each subchannel of each mobile user. Each user categorizes all N subchannels into two sets at time t. Good subchannel set is denoted as

$$\mathcal{N}_k^G(t) = \{n | \arg_{n \in \mathcal{N}} h_{kn}(t) \geq \hat{p}_s\} \tag{6.26}$$

The bad subchannel set is denoted as

$$\mathcal{N}_k^B(t) = \{n|\arg_{n\in\mathcal{N}} h_{kn}(t) < \hat{p}_s\} \tag{6.27}$$

Each mobile user only informs the base station of its good and bad subchannel sets. The exclusively good subchannel set for user k is defined as

$$\mathcal{N}_k^U(t) = \{n|n \in \mathcal{N}_k^G(t), n \notin \mathcal{N}_j^G(t), j \in \mathcal{K}/k\} \tag{6.28}$$

in which the subchannels are only evaluated as good ones by user k. However, it is normal that multiple users take same subchannels as the good ones. Those subchannels are defined as the mutual good subchannels. To the greatest extend, the mutual good subchannels are evenly assigned to each user. Among all mutual good subchannels, the ones assigned to user k uniquely formulate set $\mathcal{N}_k^S(t)$. $\mathcal{N}_k^{tot}(t)$ is defined as the set includes all the subchannels assigned to user k at time t.

$$\mathcal{N}_k^{tot}(t) = \mathcal{N}_k^U(t) \cup \mathcal{N}_k^S(t) \tag{6.29}$$

The total power allocated for user k at time t is denoted as $P_k(t)$, which is equally allocated to each subchannel in $\mathcal{N}_k^{tot}(t)$. The method to acquire $\mathcal{N}_k^{tot}(t)$ for each user is called heuristic approach.

The achievable information rate of the kth user at time t is denoted as

$$r_k(t) = \sum_{n\in\mathcal{N}_k^{tot}(t)} W\log_2\left(1 + \frac{\frac{P_k(t)}{|\mathcal{N}_k^{tot}(t)|}h_{kn}(t)}{WN_0(t)}\right) \tag{6.30}$$

where $\rho_{k,n}(t) = 1$ indicates the nth subchannel is occupied by user k at time t, otherwise $\rho_{k,n}(t) = 0$. $N_0(t)$ denotes the single-sided spectral density of additive Gaussian noise.

The energy efficiency is used to observe the efficiency of energy in data transmission. The total energy efficiency at time instant t is defined as $\Gamma_{\mathrm{EE}}^{(t)}$.

$$\Gamma_{\mathrm{EE}}^{(t)} = \frac{\sum_{u=1}^{t}\sum_{k=1}^{K} r_k(u)\Delta t}{\sum_{u=1}^{t}\sum_{k=1}^{K} P_k(u)\Delta t} \tag{6.31}$$

where the channel coherence time duration is Δt.

Within the time budget T_b, the optimization aims at maximizing the total energy efficiency Γ_{tot} to deliver payload B to each user. The power allocation $\{P_k(t)\}$ is required to solve the optimization problem. The optimization is shown in $\mathcal{P}_2$.

$$\mathcal{P}_2: \begin{array}{ll} \underset{\{P_k(t)\}}{\text{maximize}} & \Gamma_{tot} = \Gamma_{\mathrm{EE}}^{(T)} \\ subject\ to & \sum_{t=1}^{T} r_k(t)\Delta t = B, \quad \forall k \in \mathcal{K} \\ & \sum_{k=1}^{K} P_k(t) \le P \\ & T \le T_b \end{array} \tag{6.32}$$

The total transmit power in each time slot is no greater than P. $\mathcal{P}_2$ is a complicated long-term optimization problem. Both the energy efficiency and the success of payload delivery depend on the real-time resource allocation strategy.

In order to solve $\mathcal{P}_2$, only the power allocation have to be determined in a timely manner since the subchannel assignment is calculated by the heuristic approach. Henceforth, it is appropriate to apply DDPG to solve such long-term optimization.

The system state at time t is defined as

$$\begin{aligned} &\mathbf{s}_t = [|\mathcal{N}_1^{tot}(t)|, \ldots, |\mathcal{N}_K^{tot}(t)|, B_1^{acc}(t), \ldots, B_K^{acc}(t), \\ &E^{acc}(t), B^{acc}(t), t] \in \mathbf{R}^{1\times(2K+3)} \end{aligned} \tag{6.33}$$

where the accumulated delivered payload of user k by time t is

$$B_k^{acc}(t) = \sum_{u=1}^{t} r_k(u)\Delta t \tag{6.34}$$

The accumulated energy consumption is

$$E^{acc}(t) = \sum_{u=1}^{t}\sum_{k=1}^{K} P_k(u)\Delta t \tag{6.35}$$

The accumulated sum delivered payload is

$$B^{acc}(t) = \sum_{k=1}^{K} B_k^{acc}(t) \tag{6.36}$$

The set contains all states is denoted by $\mathcal{S}$.

The action is defined as $\mathbf{a}_t$.

$$\mathbf{a}_t = [P_1(t), P_2(t), \ldots, P_K(t)] \in \mathbf{R}^{1\times K} \tag{6.37}$$

where $P_k(t) \in [0, P]$, $\sum_{k=1}^{K} P_k(t) \leq P$.

The action need to be reformed into the values between [0,1] in order to achieve better training performance. $\alpha_k(t) \in [0,1]$. Hence the action is defined as

$$\mathbf{a}_t = [\alpha_1(t), \alpha_2(t), \ldots \alpha_K(t)] \tag{6.38}$$

where

$$P_k(t) = \begin{cases} P\alpha_1(t), & k = 1 \\ (P - \sum_{j=1}^{k-1} P_j(t))\alpha_k(t), & k = 2,3,\ldots,K \end{cases} \tag{6.39}$$

The action set is denoted as $\mathcal{A}$.

In general, the action selected at each time slot depends on the current channel conditions and information payload delivery conditions. The first system state is defined for $t = 0$ as

$$\mathbf{s}_0 = [|\mathcal{N}_1^{tot}(0)|, \ldots, |\mathcal{N}_K^{tot}(0)|, 0, \ldots, 0] \tag{6.40}$$

and the final state $\mathbf{s}_T$ is defined for $t = T$ as

$$\mathbf{s}_T = [|\mathcal{N}_1^{tot}(T)|, \ldots, |\mathcal{N}_K^{tot}(T)|, B, \ldots, B, E^{acc}(T), KB, T] \tag{6.41}$$

The final state $\mathbf{s}_T$ is absorbing.

The reward function is same as the one in the previous section since the optimization goal and constraints of two optimization problems are the same.

$$w(\mathbf{s}_t, \mathbf{a}_t) = \Gamma_{EE}^{(t)} \eta_1(t) \ldots \eta_K(t) \tag{6.42}$$

$\eta_k(t)$ is denoted as

$$\eta_k(t) = \begin{cases} 1 & B_k^{acc}(t) \geq t\frac{B}{T_b} \\ 0 & B_k^{acc}(t) < t\frac{B}{T_b} \end{cases} \tag{6.43}$$

$\eta_k(t)$ is utilized to motivate DDPG to learn the strategy to satisfy the payload requirement within the time budget T_b.

Simulation Result

Indoor channels are measured with the USRP N210 with the CBX daughterboard. The bandwidth of each subchannel is $W = 1$MHz. The users are randomly distributed around the base station and remain stationary within time budget T_b. $T_b = 30$ms. The path loss exponential is $\beta = 2$. The channel gains are approximately in range $[-80, -60]$dB. The noise power spectrum density is $N_0 = -170$ dBm/Hz. The channel coherence time is $\Delta t = 1$ms. The total transmit power $P \leq 3$mW.

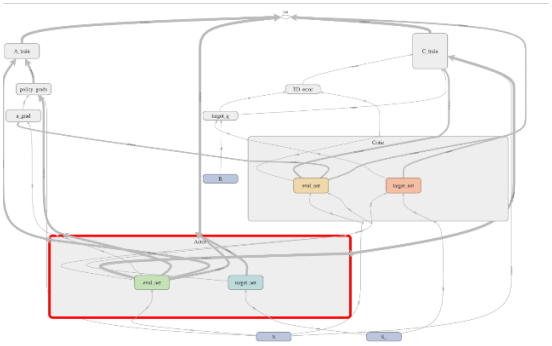

Figure 6.1: Deep Deterministic Policy Gradient framework.

The framework of DDPG is shown in Fig. 4. The evaluation actor network has 2 hidden layers. The first hidden layer has 300 nodes and the second hidden layer has 200 nodes. The second hidden layer is connected to two output layers. One output layer is used

to output the power and another one is used to determine the subchannel assignment. The total number of the weighs for the evaluation actor network is 64500. For the evaluation critic network, there are 2 hidden layers as well. The first hidden layer has 300 nodes and the second hidden layer has 20 nodes. The total number of the weighs for the evaluation actor network is 11120.

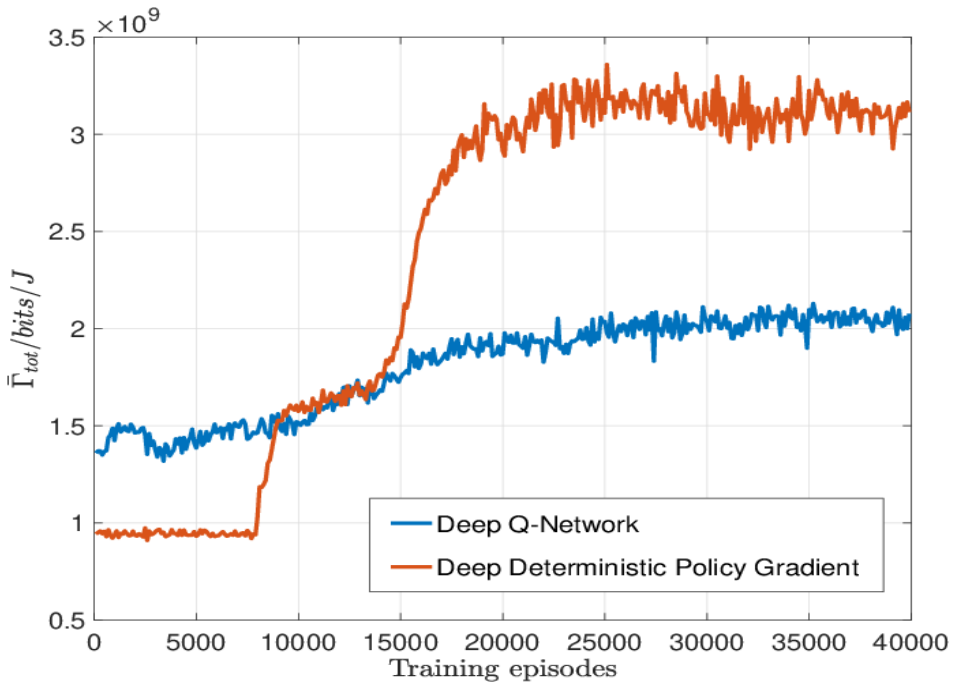

Figure 6.2: The convergence on average total energy efficiency $\bar{\Gamma}_{tot}$(moving average of Γ_{tot}) in the training process of both Deep Deterministic Policy Gradient and Deep Q-Network. $K = 2$. $N = 3$. $B = 24$Kbits.

For each evaluation network, the network connectivity is fully connected. For each evaluation network, the activation function for the hidden layers of is ReLU. For the evaluation actor network, the activation function for the output layers is sigmoid. Both the target actor network and the evaluation actor network have the same structure of the neural network. Both the target critic network and the evaluation critic network also have the same structure of the neural network. The learning rate for both the actor networks and critic

networks is defined as 0.0005. The replacing interval for actor networks and critic networks are 1700 and 1500, respectively. The mini batch size is 32. The memory capacity is more than 200000. The reward discount $\gamma \geq 0.8$. The training episodes is more than 120000. Each episode starts at $t = 0$ and ends at $t = T_b$.

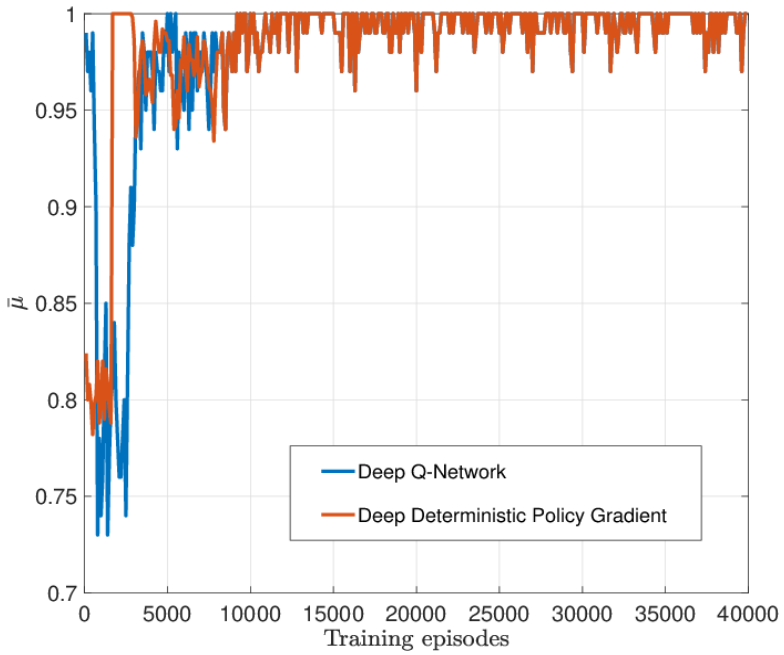

Figure 6.3: The convergence on payload delivery satisfaction $\bar{\mu}$(moving average of μ) in the training process of both Deep Deterministic Policy Gradient and Deep Q-Network. $K = 2$. $N = 3$. $B = 24$Kbits.

The DQN is implemented in order to compare with the DDPG algorithm. When $N = 3$, the action space $\mathcal{A}$ is generated as $P_n(t)$ is selected from $[2,1,0.5,0.25]$mW. $\sum_{n=1}^{N} P_n(t) \leq P$.

Both the evaluation network and the target network in DQN are structured with 4 hidden layers, each hidden layer has 100 hidden nodes.

The total number of the weights is 37500. The connectivity of each neural network is fully connected. The learning rate is defined as 0.00005.

The mini batch size is 10. The size of the experience pool is 60000. Initially, the exploration rate $\varepsilon_c = 1$ and it decreases with 0.001 at each training interval, and finally stops at $\varepsilon_{ch} = 0.1$.

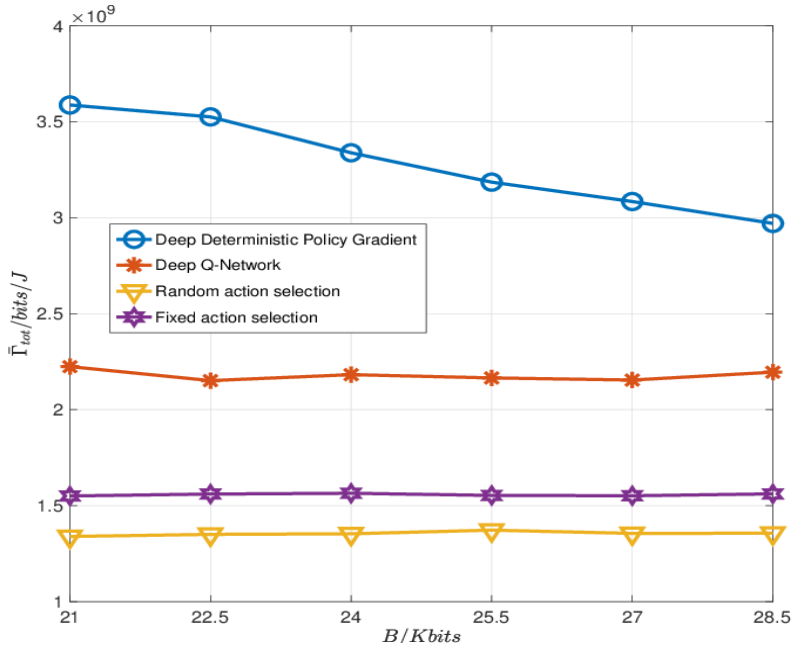

Figure 6.4: The total energy efficiency performance comparison between Deep Deterministic Policy Gradient algorithm, Deep Q-Network, Random action selection and Fixed action selection algorithms on $\bar{\Gamma}_{tot}$. $K = 2, N = 3$.

The target network's weight replacement interval is 500 training epochs.

The training process is counted by episodes.

The total training episodes is 60000. Each episodes starts at $t = 0$ and ends at $t = T$.

In a training episodes u, the payload satisfactory parameter $\mu(u)$ is defined. $\mu(u) = 1$ if $t \leq T_b$ and $\forall k, B_k^{acc}(T) \geq B$.

Otherwise $\mu(u) = 0$. Throughout the training process, the moving average of 100 episodes' $\mu(u)$ is calculated as

$$\bar{\mu} = \frac{\mu(u-99)+\ldots+\mu(0)}{100} \tag{6.44}$$

Higher $\bar{\mu}$ indicates a higher successful payload delivery rate. The moving average of 100 episodes' $\Gamma_{tot}(u)$ is calculated as

$$\bar{\Gamma}_{tot} = \frac{\Gamma_{tot}(u-99)+\ldots+\Gamma_{tot}(0)}{100} \tag{6.45}$$

From Fig. 6.2 and Fig. 6.3, it can be observed that by the training episodes increase, both the average total energy efficiency $\bar{\Gamma}_{tot}$ and the payload delivery satisfaction $\bar{\mu}$ converge.

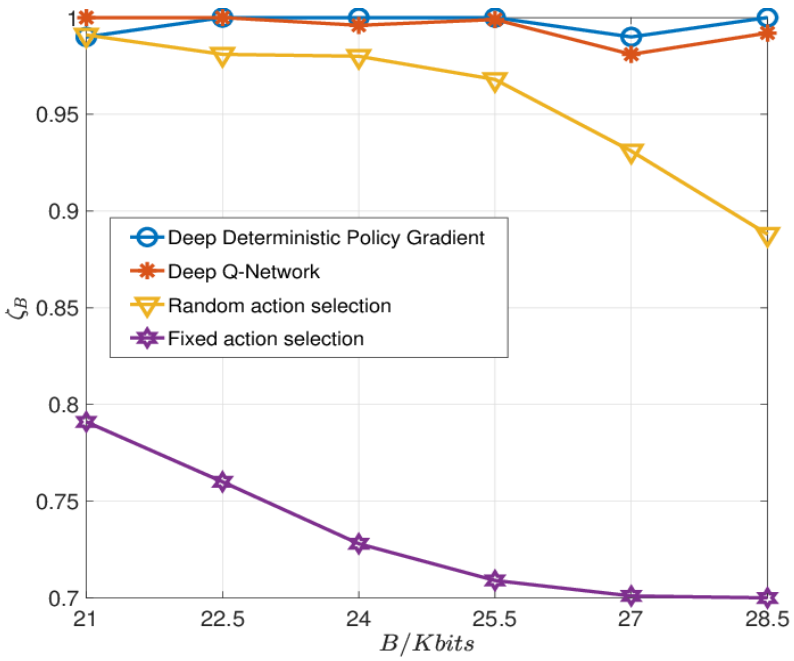

Figure 6.5: The information payload delivery performance comparison between Deep Deterministic Policy Gradient algorithm, Deep Q-Network, Random action selection and Fixed action selection algorithms on ζ_B. $K = 2, N = 3$.

Both DDPG and DQN achieve high payload delivery satisfaction. However, DDPG outperforms DQN on the total energy efficiency.

That can be explained as DDPG can output the continuous action values, however, DQN can only choose discretized action.

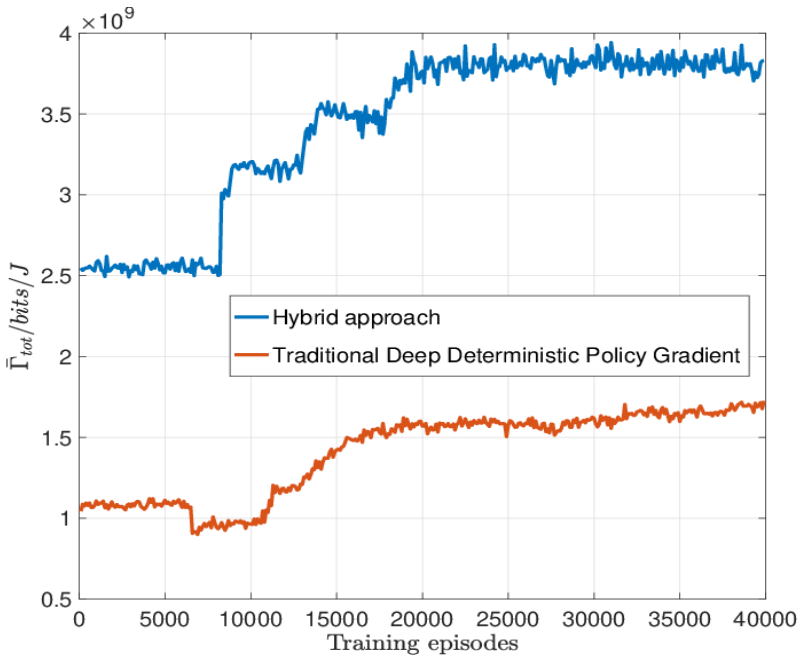

Figure 6.6: The convergence on average total energy efficiency $\bar{\Gamma}_{tot}$(moving average of Γ_{tot}) of hybrid approach and traditional Deep Deterministic Policy Gradient algorithm in the training process. $B = 24$Kbits. $K = 3$. $N = 16$.

In the simulation, $N_t = 1000$ test data are applied to test the performance of the well training DDPG algorithm. It is defined that $\zeta_B = \frac{N_s}{N_t}$, which denotes that N_s of N_t test data can successfully deliver the payload.

$\bar{\Gamma}_{tot}$ is used to evaluate the energy efficiency performance of 1000 test data. The system performance of DDPG algorithm is compared with DQN,

Random action selection and Fixed action selection when there are 2 users and 3 available subchannels $K = 2, N = 3$. In Random transmission, a random action is selected from the discretized action set $\mathcal{A}$ for transmission in each time slot.

For Fixed action selection, $\frac{1}{6}$ mW power is evenly allocated on each subchannel. In the simulation, different payload requirements are considered.

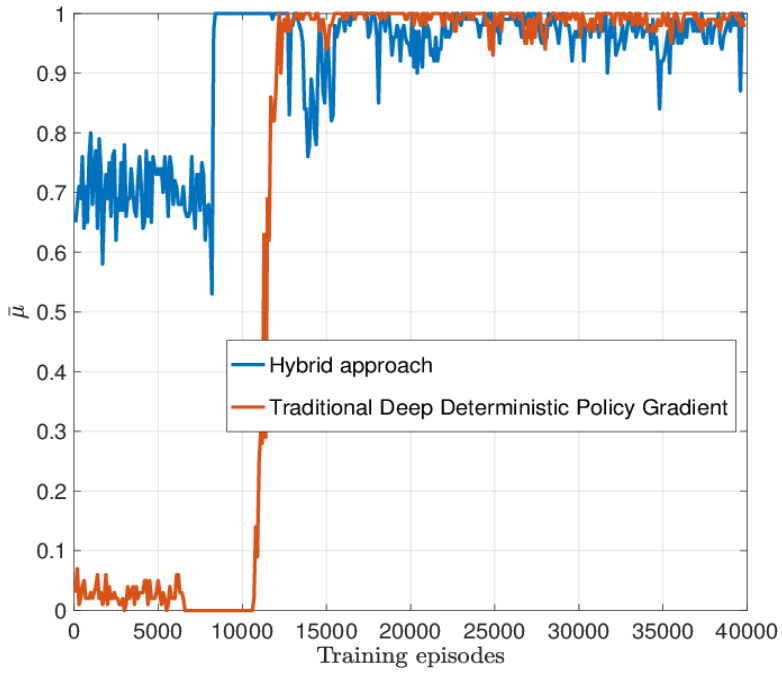

Figure 6.7: The convergence on payload delivery satisfaction $\bar{\mu}$(moving average of μ) of hybrid approach and traditional Deep Deterministic Policy Gradient algorithm in the training process. $B = 24$Kbits. $K = 3$. $N = 16$.

From Fig. 6.4 and Fig. 6.5, it can be observed that of all four algorithms, DDPG algorithm accomplishes the highest energy efficiency while maintaining near 100% successful payload delivery probability. By the information payload requirement increases, the achieved energy efficiency slightly decreases. That can be explained as that DDPG consumes more power for transmission in order to satisfy the payload requirement, which results in a decrease in energy efficiency.

When considering the large number of available subchannel condition, DDPG algorithm is utilized to determine the power allocation of each mobile user. The subchannel assignment is calculated by a heuristic approach. The subchannel selection threshold is $\hat{p}_s = 10^{-7}$. The OFDMA system model is established with multiple mobile users $K = 3,4$ and multiple available subchannels $N = 16,32,64$. The traditional DDPG algorithm is also implemented as, it can output both the power allocation and subchannel assignment.

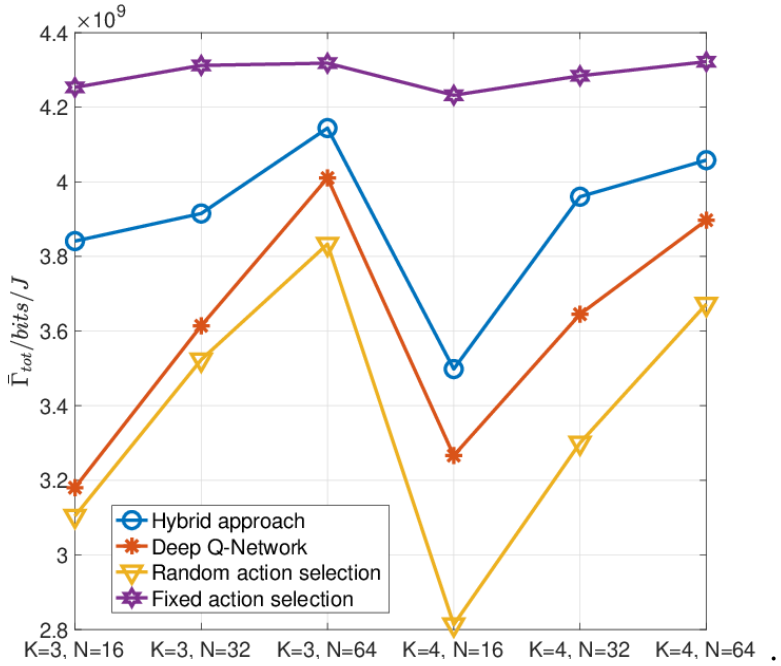

Figure 6.8: The performance comparison of average energy efficiency $\bar{\Gamma}_{tot}$. $K = 3,4$, $N = 16,32,64$. $B = 24$ Kbits.

Fig. 6.6 and Fig. 6.7 compare the performance of the hybrid approach and traditional DDPG algorithm. Both the hybrid approach and traditional DDPG can achieve high payload delivery satisfaction. However, the hybrid approach outperforms traditional DDPG algorithm in energy efficiency. The reasons are explained as follows. In the simulation, the number of the users is $K = 3$ and the total number of the subchannels is

$N = 16$. If implementing traditional DDPG algorithm, the dimension of the action is $\mathbf{a}_t = \mathbb{R}^{1\times 32}$. However, when implementing the hybrid approach, $\mathbf{a}_t = \mathbb{R}^{1\times 3}$. High dimensional output results in training difficulty and bad training performance. Beside, in order to implement traditional DDPG algorithm, the system state has to contain precise channel gain on each subchannel, which is not practical to be acquired in the real-time system [5]. Henceforth, the hybrid approach is better than traditional DDPG in solving the long-term optimization in condition of a large number of available subchannels.

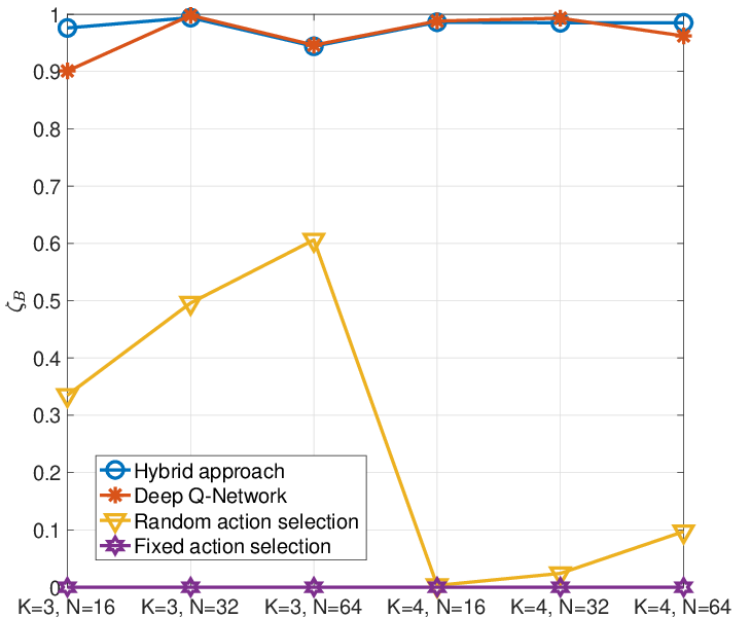

Figure 6.9: The performance comparison of successful payload delivery probability ζ_B. $K = 3,4$, $N = 16,32,64$. $B = 24$ Kbits.

In Fig. 6.8 and Fig. 6.9, the system performance of the hybrid approach is compared with the other algorithms under different conditions of the number of users and available subchannels. For DQN, the power P_k is discretized between [0,3] mW, and the action space is formulated as $\mathcal{A}$ under the constraint of $\sum_{k=1}^{K} P_k(t) \leq P$. In Random transmission, a random action is selected from action set $\mathcal{A}$ for transmission at each time

slot. For Fixed action selection, 0.125 mW power power are evenly allocated to each user, respectively. Of all the algorithms, the hybrid approach accomplishes higher energy efficiency than DQN and Random action selection algorithms while maintaining near 100% payload delivery probability. The Fixed action selection achieves higher energy efficiency than the hybrid approach, however, it cannot satisfy the payload requirement.

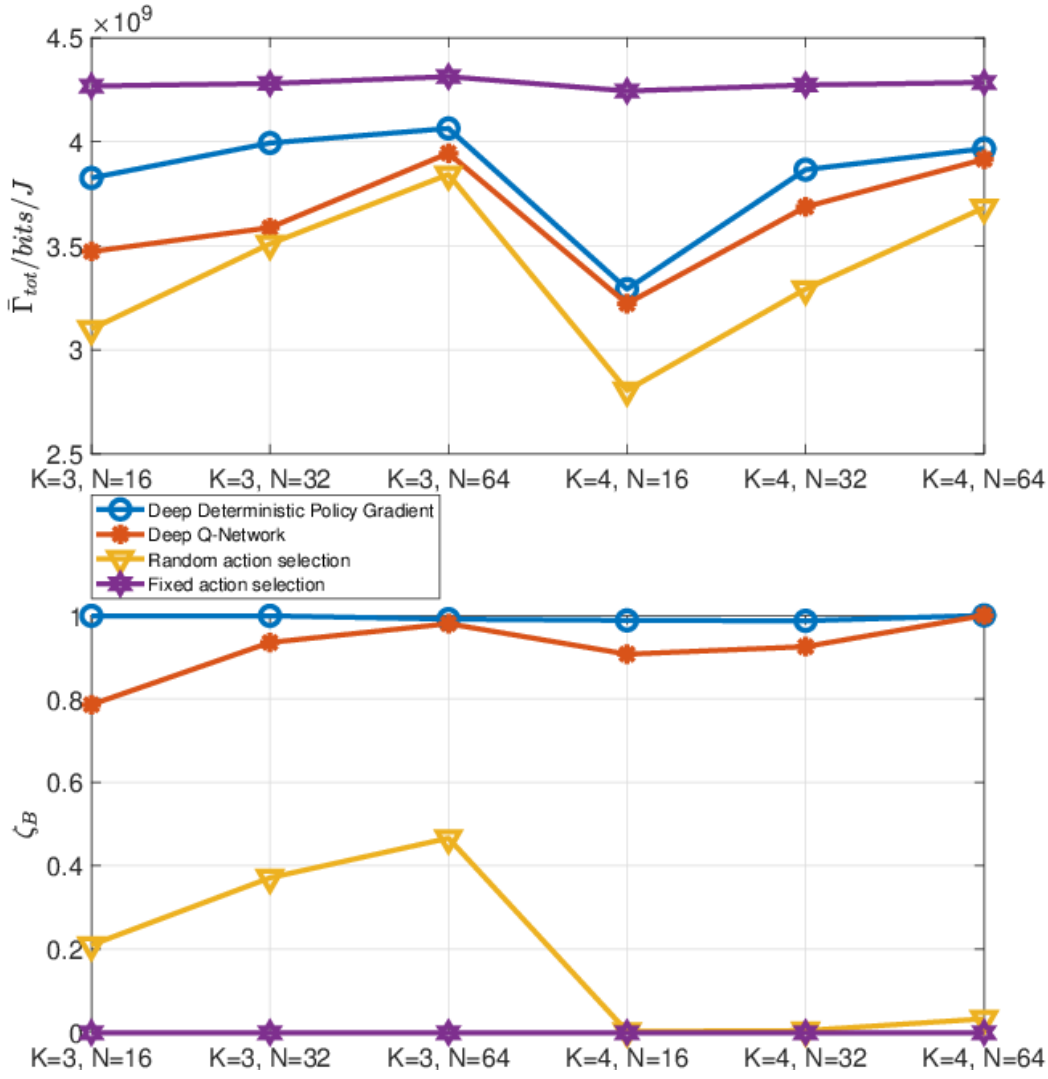

Figure 6.10: The performance comparison of average energy efficiency $\bar{\Gamma}_{tot}$ and successful payload delivery probability ζ_B. $K = 3,4$, $N = 16,32,64$. $B = 27$ Kbits.

In Fig. 6.10, the payload requirement is increased from 24 Kbits to 27 Kbits and the system performance of the hybrid approach is compared with the other algorithms. When the payload requirement increases, the difference in energy efficiency achieved between the hybrid approach and DQN becomes smaller. However, the successful payload delivery rate of DQN gets lower, which still proves the superiority of the hybrid approach.

Conclusions

In Chapter Six, a real-time multiuser OFDMA energy-efficient transmission system is implemented. With dynamic power allocation and subchannel assignment, the optimization problem is formulated as maximizing the total energy efficiency while delivering information payloads to all mobile users. DDPG algorithm is applied to make the optimal resource allocation decision effectively in a real-time system. With proper design of the action, both the continuous power allocation and discretized subchannel assignment are simultaneously determined by DDPG algorithm. Compared with DQN and other algorithms, DDPG shows the excellent performance in achieving high energy efficiency and delivering information payloads to the mobile users. As the number of available subchannels increases, a hybrid approach is invented: the power allocation is determined by DDPG and the subchannel assignment is determined by a heuristic algorithm. Compared with the traditional DDPG algorithm, the hybrid approach dramatically decreases the dimension of the action and enhances the training effect. The simulation results demonstrate that the hybrid approach outperforms the traditional DDPG in energy efficiency. Compared with DQN, Random action selection and Fixed action selection, the hybrid approach shows advantages in energy saving and information payloads delivery.

CHAPTER SEVEN

Conclusions

Dissertation Summary

This dissertation mainly focuses on applying DL algorithms to solve practical energy-efficient wireless communication and spectrum management problems. Specifically, two promising wireless communication systems are explored: SWIPT system and multiuser downlink OFDMA data transmission system. In the SWIPT system, different DL methods are utilized to determine the optimal transmission strategies, which can improve the energy harvesting performance. In the multiuser downlink OFDMA data transmission system, the optimal spectrum management strategies are optimized by the DL algorithms in order to achieve the energy-efficient data transmission.

In the SWIPT systems, the operating mechanism of simultaneous wireless information and power transfer is discussed. The base station aims at fairly charging multiple energy harvesters while maintaining high communication quality. A DNN is trained to solve the optimization problem, which efficiently accelerates the operating speed. It consumes very long time to train a DNN with a large number of training data. However, the well trained DNN can be utilized online for an immediate reaction. Besides the DNN, a K-means clustering algorithm is applied to classify each of the channel conditions into one of several classes. For each individual class, a specific DNN is trained. Once a channel condition is classified, the corresponding DNN is called to output the power

allocation strategy. The cooperation between DNN and K-means clustering can help the transmitter effectively adjust its transmission strategy.

Then, wireless charging is formulated as a continuous process. With very limited environment information, a DQN guides the transmitter to continuously and fairly charge multiple energy harvesters while maintaining the communication quality. The DQN shows superiority in dynamically determining the transmission strategy and avoiding short-sighted suboptimal solutions. Even with different channel statistics, the DQN can always learn the optimal power allocation strategy in order to fully charge all energy harvesters in the least time without sacrificing any communication quality.

Finally, the system model is extended from a single communication pair (one transmitter and one information receiver) to the multiple communication pairs. Since multiple information transmitters cannot acquire the precise channel information to determine the real-time transmission decision, a MAB approach is applied to analyze the fair energy harvesting optimization problem. In particular, an improved UCB_1 algorithm is used to deal with a large number of bandit arms.

In the multiuser downlink OFDMA data transmission system, the spectrum management strategies are adjusted at the base station to accomplish data transmission from the base station to multiple mobile users. The base station aims at maximizing the energy efficiency while maintaining a good communication quality from the base station to each user. Since both the subchannel assignment and power allocation on each subchannel have to be determined, two different DNNs are trained separately. Due to the non-convexity of the proposed optimization problem, generating the training data consumes a lot of time. Therefore, a Refined Exhaustive Search algorithm is proposed,

which can efficiently reduce the data generation time and ensure sufficient training data. DNNs can dramatically reduce the execution time in solving the optimization problem while assuring an excellent system performance.

Next, the data transmission is modeled as a continuous process. The long-term optimization is formulated as maximizing the total energy efficiency while delivering information payload to each mobile user within the time budget. The real-time power allocation and subchannel assignment have to be determined. A DDPG algorithm is applied to solve the proposed long-term optimization problem. With limited channel information, the DDPG algorithm can optimize the resource allocation strategy for each time slot to meet global optimization goal and constraints. In order to solve the optimization problem, both the continuous power control and discretized subchannel assignment strategies have to be determined by the DDPG algorithm. As the number of available subchannels increases, traditional DDPG algorithm cannot solve the proposed problem well because high dimensional action leads to poor training result. A hybrid algorithm is invented to solve that problem: a DDPG algorithm is utilized to determine the power allocation, while a heuristic approach is used to determine the subchannel assignment in a timely manner.

Future Research

For the SWIPT system, the experimental results indicate that the amount of harvested energy is tiny. In order to improve the amount of harvested energy, a massive MIMO communication system is considered in the future research. The core technology of 5G communication is massive MIMO. A large number of antennas equipped on both the information transmitters and receivers can guarantee an extremely fast data speed, while it can result in a considerable increase in harvested energy. However, it becomes more

difficult to adjust the power allocation strategy on a large number of transmit antennas. Solving the previously proposed optimization problems, the DNN and DRL have to be refined. Due to the large number of antennas, the dimensions of both input and output of the DNNs dramatically increase. A more precise training process has to be carried out to guarantee a better training effect. In the future, the proposed SWIPT is implemented in the real 5G communication systems as an important power supply for the IoT devices.

For the multiuser downlink OFDMA data transmission system, the current proposed single base station system is extended to a multiuser scenario. Since in 5G communication systems, the number of base stations is largely increased in order to assure a high data speed and low latency communication. As multiple base stations coexist in the systems, they have to dynamically adjust their transmission strategies to transmit data energy-efficiently. Each base station aims to provide high quality communication to the responsible mobile users while avoiding the possible interference with the other base stations. A decentralized DRL framework is applied at each base station to determine the optimal transmission strategy in a timely manner. In order to approximate the real communication systems, mobile users are not stationary, which increases the complexity of the system. The proposed OFDMA scheme is operated at the real base stations as a solution to enormous energy waste existing in present communication systems.

Besides two continued projects, the DL algorithms are exploited in more practical communication systems, such as V2X (Vehicle-to-Everything) and UAV (Unmanned aerial vehicle) communication systems. DL methods have shown superiority in solving complicated optimization problems, which will be seriously considered in different industries in the future.

REFERENCES

[1] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," China Communications, vol. 14, no. 11, pp. 92–111, 2017.

[2] U. Challita, L. Dong, and W. Saad, "Proactive resource management in lte-u systems: A deep learning perspective," arXiv preprint arXiv:1702.07031, 2017.

[3] Y. He, Z. Zhang, F. R. Yu, N. Zhao, H. Yin, V. C. Leung, and Y. Zhang, "Deep reinforcement learning-based optimization for cache-enabled opportunistic interference alignment wireless networks," IEEE Transactions on Vehicular Technology, vol. 66, no. 11, pp. 10 433–10 445, 2017.

[4] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," Neural networks, vol. 2, no. 5, pp. 359–366, 1989.

[5] H. Sun, A. Nallanathan, C.-X. Wang, and Y. Chen, "Wideband spectrum sensing for cognitive radio networks: a survey," IEEE Wireless Communications, vol. 20, no. 2, pp. 74–81, 2013.

[6] J. Xu and R. Zhang, "A general design framework for mimo wireless energy transfer with limited feedback." vol. 64, no. 10, pp. 2475–2488, 2016.

[7] H. Li, T. Lv, and X. Zhang, "Deep deterministic policy gradient based dynamic power control for self-powered ultra-dense networks," in 2018 IEEE Globecom Workshops (GC Wkshps). IEEE, 2018, pp. 1–6.

[8] S. Maghsudi and E. Hossain, "Multi-armed bandits with application to 5G small cells," IEEE Wireless Communications, vol. 23, no. 3, pp. 64–73, 2016.

[9] Y. Xing and C. Tapparello, "Dynamic fountain codes for energy efficient data dissemination in underwater sensor networks," in OCEANS 2017-Anchorage. IEEE, 2017, pp. 1–6. © [2017] IEEE. Reprinted, with permission, from [Y. Xing and C. Tapparello, "Dynamic fountain codes for energy efficient data dissemination in underwater sensor networks", IEEE OCEANS, 2017].

[10] S. Buzzi, I. Chih-Lin, T. E. Klein, H. V. Poor, C. Yang, and A. Zappone, "A survey of energy-efficient techniques for 5g networks and challenges ahead," IEEE Journal Selected Areas in Communications, vol. 34, no. 4, pp. 697–709,2016.

[11] S. El Hassani, H. El Hassani, and N. Boutammachte, “Rf energy harvesting for 5g: An overview,” in 2017 International Renewable and Sustainable Energy Conference (IRSEC). IEEE, 2017, pp. 1–6.

[12] C. Xiong, G. Y. Li, S. Zhang, Y. Chen, and S. Xu, “Energy-efficient resource allocation in ofdma networks,” IEEE Transactions on Communications, vol. 60, no. 12, pp. 3767–3778, 2012.

[13] C. C. Zarakovitis, Q. Ni, and J. Spiliotis, “Energy-efficient green wireless communication systems with imperfect csi and data outage,” IEEE Journal on Selected Areas in Communications, vol. 34, no. 12, pp. 3108–3126, 2016.

[14] Y. Li, M. Sheng, C. W. Tan, Y. Zhang, Y. Sun, X. Wang, Y. Shi, and J. Li, “Energy-efficient subcarrier assignment and power allocation in ofdma systems with max-min fairness guarantees,” IEEE Transactions on Communications, vol. 63, no. 9, pp. 3183–3195, 2015.

[15] Y. Xing and L. Dong, “Passive radio-frequency energy harvesting through wireless information transmission,” in Proc. of IEEE DCOSS, Jun. 2017, pp. 73–80. © [2017] IEEE. Reprinted, with permission, from [Y. Xing, L. Dong, “Passive radio-frequency energy harvesting through wireless information transmission”, IEEE DCOSS, Jun. 2017].

[16] Y. Xing, Y. Qian, and L. Dong, “Deep learning for optimized wireless transmission to multiple rf energy harvesters,” in Proc. of IEEE VTC Fall, 2018. © [2018] IEEE. Reprinted, with permission, from [Y. Xing, Y. Qian, and L. Dong, “Deep learning for optimized wireless transmission to multiple rf energy harvesters”, IEEE VTC, Fall.2018].

[17] Y. Xing, C. Tapparello, Y. Qian, and L. Dong, “Optimal wireless transmission to multiple rf energy harvesters using deep reinforcement learning,” submitted to IEEE Transaction on Vehicular Technology.

[18] Y. Xing, Y. Qian, and L. Dong, “A multi-armed bandit approach to wireless information and power transfer,” IEEE Communications Letters, 2020. © [2020] IEEE. Reprinted, with permission, from [Y. Xing, Y. Qian, and L. Dong, “A multi-armed bandit approach to wireless information and power transfer”, IEEE Communication Letters, 2020].

[19] Y. Xing, Y. Qian, and L. Dong, “Deep learning for optimized multiuser ofdma energy-efficient wireless transmission,” Submitted to IEEE Communication Letters.

[20] Y. Xing, Y. Qian, and L. Dong, “Energy-efficent multiuser transmission based on deep reinforcement learning,” Submitted to IEEE Transaction on Wireless Communication.

[21] Y. Xing, Y. Qian, and L. Dong, “Energy-efficent multiuser transmission based on deep deterministic policy gradient,” Under Revision. IEEE Communication Letters.

[22] K. W. Choi, P. A. Rosyady, L. Ginting, A. A. Aziz, and D. I. Kim, “Simultaneously charging multiple sensor nodes in multi-antenna wireless-powered sensor networks,” in Proc. of ICC Workshops, 2017, pp. 361–366.

[23] W. Wu, X. Zhang, S. Wang, and B. Wang, “Max–min fair wireless energy transfer for multiple-input multiple-output wiretap channels,” IET Communications, vol. 10, no. 7, pp. 739–744, 2016.

[24] C. K. Ho and R. Zhang, “Optimal energy allocation for wireless communications with energy harvesting constraints,” vol. 60, no. 9, pp. 4808–4818, Sep. 2012.

[25] A. Paulraj, R. Nabar, and D. Gore, Introduction to Space-Time Wireless Communications. Cambridge University Press, 2003.

[26] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, MIMO Wireless Communications. Cambridge University Press, 2007.

[27] S. Boyd and L. Vandenberghe, Convex optimization. Cambridge University Press, 2004.

[28] M. A. Wijaya, K. Fukawa, and H. Suzuki, “Intercell-interference cancellation and neural network transmit power optimization for MIMO channels,” in Vehicular Technology Conference (VTC Fall). IEEE, 2015, pp. 1–5.

[29] Y. Qian, Y. Xing, and L. Dong, “Wireless transmission design with neural network for radio-frequency energy harvesting,” in 2018 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, 2018, pp. 1–6. © 2018 IEEE.

[30] K. Q. Weinberger, J. Blitzer, and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” in Advances in Neural Information Processing Systems, 2006, pp. 1473–1480.

[31] M. Grant and S. Boyd, “CVX: Matlab software for disciplined convex programming, version 2.1,” http://cvxr.com/cvx, Mar. 2014.

[32] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” arXiv preprint arXiv:1312.5602, 2013.

[33] J. Foerster, I. A. Assael, N. de Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” in Advances in Neural Information Processing Systems, 2016, pp. 2137–2145.

[34] Z. Xu, Y. Wang, J. Tang, J. Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud rans," in Proc. of IEEE ICC. IEEE, 2017, pp. 1–6.

[35] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning." in AAAI, vol. 2. Phoenix, AZ, 2016, p. 5.

[36] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas, "Dueling network architectures for deep reinforcement learning," arXiv preprint arXiv:1511.06581, 2015.

[37] S. Timotheou, I. Krikidis, S. Karachontzitis, and K. Berberidis, "Spatial domain simultaneous information and power transfer for mimo channels," IEEE Trans. on Wireless Communications, vol. 14, no. 8, pp. 4115–4128, 2015.

[38] D. Mishra and G. C. Alexandropoulos, "Jointly optimal spatial channel assignment and power allocation for mimo swipt systems," IEEE Wireless Communications Letters, 2017.

[39] A. G. Barto, S. J. Bradtke, and S. P. Singh, "Learning to act using real-time dynamic programming," Artificial intelligence, vol. 72, no. 1-2, pp. 81–138, 1995.

[40] J. Cavers, Mobile channel characteristics. Springer Science & Business Media, 2006, vol. 555.

[41] T. Q. Wu and H. C. Yang, "On the performance of overlaid wireless sensor transmission with rf energy harvesting," IEEE Journal on selected areas in communications, vol. 33, no. 8, pp. 1693–1705, 2015.

[42] L. Dong and Y. Liu, "Parallel sub-channel transmission for cognitive radios with multiple antennas," Wireless personal communications, vol. 79, no. 3, pp. 2069–2087, 2014.

[43] J. Xu, L. Liu, and R. Zhang, "Multiuser MISO beamforming for simultaneous wireless information and power transfer," vol. 62, no. 18, pp. 4798–4810, Jul. 2014.

[44] L. Dong, "Optimization of multiple wireless transmissions for radio-frequency energy harvesting," vol. 22, no. 10, pp. 2140–2143, Oct. 2018.

[45] J. Park and B. Clerckx, "Joint wireless information and energy transfer in a K-user MIMO interference channel," vol. 13, no. 10, pp. 5781–5796, Oct. 2014.

[46] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless networks with RF energy harvesting: A contemporary survey," vol. 17, no. 2, pp. 757–789, 2014.

[47] W. Chen, Y. Wang, Y. Yuan, and Q. Wang, "Combinatorial multi-armed bandit and its extension to probabilistically triggered arms," Journal of Machine Learning Research, vol. 17, no. 50, pp. 1–33, Jan. 2016.

[48] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," Machine learning, vol. 47, no. 2-3, pp. 235–256, May 2002.

[49] Z. Ren, S. Chen, B. Hu, and W. Ma, "Energy-efficient resource allocation in downlink ofdm wireless systems with proportional rate constraints," IEEE Transactions on Vehicular Technology, vol. 63, no. 5, pp. 2139–2150, 2014.

[50] C. Nam, C. Joo, and S. Bahk, "Joint subcarrier assignment and power allocation in full-duplex ofdma networks," IEEE Transactions on Wireless Communications, vol. 14, no. 6, pp. 3108–3119, 2015.

[51] G. Wunder and T. Michel, "Optimal resource allocation for parallel gaussian broadcast channels: minimum rate constraints and sum power minimization," IEEE Transactions on Information Theory, vol. 53, no. 12, pp. 4817–4822, 2007.

[52] C. Isheden, Z. Chong, E. Jorswieck, and G. Fettweis, "Framework for link-level energy efficiency optimization with informed transmitter," IEEE Transactions on Wireless Communications, vol. 11, no. 8, pp. 2946–2957, 2012.

[53] Y. Wu, Y. Chen, J. Tang, D. K. So, Z. Xu, I. Chih-Lin, P. Ferrand, J.-M. Gorce, C.-H. Tang, P.-R. Li et al., "Green transmission technologies for balancing the energy efficiency and spectrum efficiency trade-off," IEEE Communications Magazine, vol. 52, no. 11, pp. 112–120, 2014.

[54] Q. D. Vu, L. N. Tran, M. Juntti, and E. K. Hong, "Energy-efficient bandwidth and power allocation for multi-homing networks," IEEE Transactions on Signal Processing, vol. 63, no. 7, pp. 1684–1699, 2015.

[55] Z. Han, T. Himsoon, W. P. Siriwongpairat, and K. R. Liu, "Resource allocation for multiuser cooperative ofdm networks: Who helps whom and how to cooperate," IEEE Transactions on Vehicular Technology, vol. 58, no. 5, pp. 2378–2391, 2008.

[56] C. Y. Wong, R. S. Cheng, K. B. Lataief, and R. D. Murch, "Multiuser ofdm with adaptive subcarrier, bit, and power allocation," IEEE Journal on selected areas in communications, vol. 17, no. 10, pp. 1747–1758, 1999.

[57] L. Dong, "Spectral-and energy-efficient transmission over frequency-orthogonal channels," in 2016 IEEE Online Conference on Green Communications (OnlineGreenComm). IEEE, 2016, pp. 13–20.

[58] J. Tang, D. K. So, E. Alsusa, and K. A. Hamdi, “Resource efficiency: A new paradigm on energy efficiency and spectral efficiency tradeoff,” IEEE Transactions on Wireless Communications, vol. 13, no. 8, pp. 4656–4669, 2014.

[59] Z. Wang and L. Vandendorpe, “Subcarrier allocation and precoder design for energy efficient mimo-ofdma downlink systems,” IEEE Transactions on Communications, vol. 65, no. 1, pp. 136–146, 2016.

[60] Y. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in Advances in neural information processing systems, 2015, pp. 1504–1512.

[61] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, “Learning to optimize: Training deep neural networks for interference management,” IEEE Transactions on Signal Processing, vol. 66, no. 20, pp. 5438–5453.

[62] M. Kim, W. Lee, J. Yoon, and O. Jo, “Toward the realization of encoder and decoder using deep neural networks,” IEEE Communications Magazine, vol. 57, no. 5, pp. 57–63, 2019.

[63] F. Chollet et al., “Keras,” https://keras.io, 2015.

[64] J. Li, H. Gao, T. Lv, and Y. Lu, “Deep reinforcement learning based computation offloading and resource allocation for mec,” in 2018 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, 2018, pp. 1–6.

[65] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L.-C. Wang, “Deep reinforcement learning for mobile 5g and beyond: Fundamentals, applications, and challenges,” IEEE Vehicular Technology Magazine, vol. 14, no. 2, pp. 44–52, 2019.

[66] L. Liang, H. Ye, and G. Y. Li, “Spectrum sharing in vehicular networks based on multi-agent reinforcement learning,” arXiv preprint arXiv:1905.02910, 2019.

[67] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous control with deep reinforcement learning,” arXiv preprint arXiv:1509.02971, 2015.

[68] Z. Ma, B. Li, Z. Yan, and M. Yang, “Qos-oriented joint optimization of resource allocation and concurrent scheduling in 5g millimeter-wave network,” Computer Networks, vol. 166, p. 106979, 2020.

[69] L. Ferdouse, A. Anpalagan, and S. Erkucuk, “Joint communication and computing resource allocation in 5g cloud radio access networks,” IEEE Transactions on Vehicular Technology, vol. 68, no. 9, pp. 9122–9135, 2019.

[70] J. Li, G. Lei, G. Manogaran, G. Mastorakis, and C. X. Mavromoustakis, “D2d communication mode selection and resource optimization algorithm with optimal throughput in 5g network,” IEEE Access, vol. 7, pp. 25 263–25 273, 2019.

[71] A. A. Hafez, Y. M. Jaamour, and K. I. Khorzom, “Resource allocation in ofdma femtocell based lte and 5g networks with qos guarantees,” Journal of Engineering and Applied Sciences, vol. 15, no. 2, pp. 643–652, 2020.

[72] S. Fu, F. Yang, and Y. Xiao, “Ai inspired intelligent resource management in future wireless network,” IEEE Access, 2020.

[73] C. Qiu, Y. Hu, Y. Chen, and B. Zeng, “Deep deterministic policy gradient (ddpg)-based energy harvesting wireless communications,” IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8577–8588, 2019.

[74] Y. H. Xu, C. C. Yang, M. Hua, and W. Zhou, “Deep deterministic policy gradient (ddpg)-based resource allocation scheme for noma vehicular communications,” IEEE Access, vol. 8, pp. 18 797–18 807, 2020.

[75] A. Bhatia, P. Varakantham, and A. Kumar, “Resource constrained deep reinforcement learning,” in Proceedings of the International Conference on Automated Planning and Scheduling, vol. 29, no. 1, 2019, pp. 610–620.

[76] D. T. Ngo, S. Khakurel, and T. Le-Ngoc, “Joint subchannel assignment and power allocation for ofdma femtocell networks,” IEEE Transactions on Wireless Communications, vol. 13, no. 1, pp. 342–355, 2013.