Carrier Synchronization, Impairment Estimation and Interference Alignment for Wireless Communication Systems

by

Mingda Zhou

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Doctor of Philosophy

in

Electrical and Computer Engineering

by

December 2019

APPROVED:

_____

Professor Xinming Huang, Major Thesis Advisor

_____

Professor Kaveh Pahlavan, Committee Member

_____

Professor Youjian Liu, Committee Member

_____

Professor Donald Richard Brown, Head of Department

“To whom I love, learn from and respect.”

Abstract

Wireless communication systems utilize the wireless medium to perform over-the-air (OTA) data transfer. There are many factors that can impact the quality of wireless communications, such as medium imperfection, interfering environment, mismatch of transceivers, etc. To mitigate these problems and improve the quality of service (QoS), this research study is conducted on three important topics including synchronization techniques, impairment estimation theory and techniques, and interference alignment techniques.

In this thesis, it firstly present a dual link algorithm to align and manage the interference of multiple-input and multiple-output (MIMO) networks. A field-programmable gate array (FPGA) prototype is designed for software defined radio (SDR) platforms. As one of the key components, a hardware efficient architecture is proposed for the implementation of singular value decomposition (SVD). Secondly, it proposes a maximum-likelihood (ML) based synchronization approach for carrier frequency synchronization for MIMO systems. The algorithm is also implemented on FPGA for real-time performance evaluation. Finally, as an exemplary study of machine learning techniques for wireless communications, a neural network (NN) based estimator is proposed to perform coarse frequency offset estimations for MIMO systems. The proposed NN based estimator can accommodate various channel models and the results show promising performance in terms of accuracy and estimation range.

In summary, this thesis provides a comprehensive study on interference alignment, carrier synchronization, and impairment estimation using different approaches. Efficient hardware implementations for the key algorithms are also presented.

Acknowledgments

In this thesis I describe the research I conducted in pursuit of my Doctor of Philosophy in Electrical and Computer Engineering in Worcester Polytechnic Institute.

Firstly, I would like to offer my sincerest gratitude to my research advisor, Professor Xinming Huang, for his comprehensive guidance leading me to this research field, for sharing his experience and for providing financial support so that I could fully devote myself into graduate study and research. Professor Huang is a great mentor and a wise man. It is my honor to have him as my PhD advisor.

I am really grateful that I can have Professor Kaveh Pahlavan and Professor Youjian Liu as my committee members. Thank you for your patient, technical and intelligent guidance and instructions during the period of time since the day I step into WPI. I would also express my greatest appreciation for your valuable comments and reviewing of this thesis.

Besides, I would say thank you to my wife and my beloved parents, who offer me total understanding, support and love. There is nothing I could achieve without your infinite mental, physical and financial support.

I also want to thank all the peers in embedded computing lab, Zhilu Chen, Yuteng Zhou, Yecheng Lyu, Lin Bai, Yiming Zhao and Mahdi Elhousni. Thank you so much for offering me a friendly and technical atmosphere in the lab like a big family. Your technical support encourages me a lot to explore the area of my research interests.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The demand of communications from a far location and the development of electrical and electronic technology gave birth to the telecommunications in modern society where telecommunication is defined by the International Telecommunication Union (ITU) as the transmission, emission or reception of any signs, signals or messages by electromagnetic systems [2]. Later in 1864, the postulation of Maxwell's equations laid the foundations of wireless systems which was patented first by Marconi in 1897. For the first many years of wireless systems, the term "wireless" was treated equal as radio since the receiver of a radio was called wireless because there were no wires between a radio station and a listening radio device. In the same era, telephone patented by Bell and inventions of diode and triode stimulated the development of long distance telephony. These inventions, along with micro-electronic circuits, brought the possibility of reducing the size of receiving devices, as well as the size of antennas for reception of wireless signals. In 1979, the first commercial automated cellular network, launched in Japan, indicated that wireless communications officially stepped into the times of personal use. Around two decades, the original version IEEE 802.11 standard was released in 1997 and clarified in 1999, started the blossom of local wireless networks and rapidly grew into a family of standards

for local wireless networks.

1.1 Motivation

With the rapid development of wireless communication systems, newly invented and deployed high-technologies, such as wider band and complicated modulation methods, make wireless systems more and more complicated [3]. In the meantime, to accommodate a larger number of users and provide better user experience, the standards for wireless communications become rigorous and harder to comply, which also requires the advancement of technologies in physical layer. Therefore to deal with the issues above, some of the important conditions that should be taken into consideration are listed as follows:

- Interference signal brought by other nearby users in the same network [4–6]. In a broadcasting multiuser network, each user broadcasts the message. Therefore if multiple users broadcast at the same time, the received signal could be ruined that nobody receives the correct message. To solve this problem, it is important to manage the interference so that each user can catch the correct message from wireless media.

- Carrier frequency offset (CFO) introduced because of mismatch of oscillators, relative displacement of transceivers, or other reasons [7–9]. The carrier frequency offset makes the transmitted signal drifting and consequently results in the failure of message representation. To recover the CFO, we need to estimate it according to the characteristics of received signal and then perform the compensation. CFO estimation usually consists of two steps: coarse and fine estimations.

- Symbol timing offset (STO) introduced due to asynchronous sampling moments [10, 11]. In a wireless communication system, the data is oversampled to resist

the inter symbol interference (ISI). However, the introduced timing offset holds the chance to bring samples far away from the maximum amplitude points in the waveform. Hence, well designed STO estimation method could help the samples stay close to expected constellation points and thus obtain lower bit error rate (BER).

- Impact of channel during over-the-air transmission [12–15]. Channel has many different kinds of influences on the transmitted signal. The most commonly discussed channels are additive white Gaussian noise (AWGN), fast fading, slow/flat fading, multipath and time-varying channels. If complex surrounding environment and strong interference plus noise occurs, the impact of channels can be extremely harmful to the signal. Therefore, a well modeled channel and corresponding channel estimation method are an important step in the receiver.

1.2 Contributions of The Thesis

Being motivated by the description above and to meet the challenges, we conduct research studies in interference alignment, baseband synchronizations and wireless system SDR implementations. The contributions of this thesis with these research studies in wireless communications are:

- We present a dual link algorithm [16] and the design of an FPGA prototype targeting SDR platforms. Compared with execution of the algorithm on CPU as most of the SDR implementations do, our FPGA based prototype requires less time on algorithm convergence, resulting in a speedup factor 2 or higher. Meanwhile, the dual link algorithm has the advantage that converges monotonically to a stationary point with very high convergence speed. Detailed contribution of this research work is reflected in [17]:

1. *Zhou, Mingda, Xinming Huang, Yuteng Zhou, Xing Li and Youjian Liu, "An FPGA prototype of dual link algorithm for MIMO interference network". 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2017.*

- In the FPGA prototype of dual link algorithm, Cholesky decomposition is applied to perform signal covariance matrix and adjust the transmit power, but the complexity of Cholesky decomposition grows exponentially. Therefore in addition, we proposed a hardware-efficient architecture for singular value decomposition (SVD). The proposed architecture accommodates arbitrary matrix order and the matrix order becomes an input so that it can be changed according to users' demand. In the mean time, the hardware utilization of proposed SVD architecture is very low and does not change along with the order of matrices. Detailed contribution of this research work is reflected in [18]:

  1. *Zhou, Mingda, Youjian Liu, Tian Xia and Xinming Huang. "An efficient and scalable hardware architecture for singular value decomposition towards massive MIMO communications." 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2017.*

- We propose a maximum likelihood (ML) based frequency offset estimation algorithm for multiple-input & multiple-output (MIMO) system, with an asymptotically optimal closed form solution to the estimator, as well as an efficient hardware architecture for our ML based CFO estimation algorithm. The proposed method has close-to-bound performance and large acquisition range, where the corresponding architecture is hardware efficient (can be easily fitted in a small FPGA), pipelined (minimal processing latency) and reconfigurable (compatible with various frame structures). Detailed contribution of this research work is reflected in [19]:

1. *Zhou, Mingda, Zhe Feng, Youjian Liu and Xinming Huang. "An efficient algorithm and hardware architecture for maximum-likelihood based carrier frequency offset estimation in MIMO systems". IEEE Access 6 (2018): 50105-50116.*

- We study the signal distortion brought by CFO and propose a neural network which consists of long short-term memory (LSTM) and dense layers, to perform coarse CFO estimations for single carrier (SC) systems and integer frequency offset (IFO) estimations for orthogonal frequency-division multiplexing (OFDM) systems. The proposed neural network based estimator has the ability to accommodate a variety of system models without changing the network architecture. It also has higher accuracy on coarse/integer frequency offset estimations, under some certain conditions, comparing to other estimation methods Detailed contribution of this research work is reflected in [20]:

    1. *Zhou, Mingda, Xinming Huang, Zhe Feng and Youjian Liu, "Coarse Frequency Offset Estimation in MIMO Systems Using Neural Networks: A Solution with Higher Compatibility". IEEE Access 7 (2019): 121565 - 121573.*

- We also propose a joint MAP frequency offset and channel estimation for MIMO systems. The solution allows the time invariant channel to have arbitrary spatial correlation and mean with a circularly symmetric complex Gaussian distribution. The Bayesian Cramér-Rao Lower bound (BCRLB) is also derived in closed form for the frequency offset estimation with prior knowledge. The BCRLB provides insight on the pilot/training signal design, including the effect of time spreading, and structures of periodic pilot and time division pilot. Detailed contribution of this research work is reflected in:

    1. *Zhou, Mingda, Zhe Feng, Xinming Huang and Youjian Liu, "Maximum A Pos-*

teriori Probability (MAP) Joint Fine Frequency Offset and Channel Estimation for MIMO Systems with Channels of Arbitrary Correlation". 2019 IEEE Global Communications Conference (GLOBECOM), 2019

Some other subdominant contributions of mine in this area are reflected by the list of publications below [21–23]:

1. *Cai, Xin, Mingda Zhou, and Xinming Huang, "Model-based design for software defined radio on an FPGA." IEEE Access 5 (2017): 8276-8283.*

2. *Cai, Xin, Mingda Zhou, Tian Xia, Wai Fong, Wing Lee and Xinming Huang, "Low-Power SDR Design on an FPGA for Intersatellite Communications." IEEE Transactions on Very Large Scale Integration (VLSI) Systems 99 (2018): 1-12.*

3. *Zhe Feng, Xing Li, Victor Palacios, Peter Mathys, Youjian Liu, Mingda Zhou, Xinming Huang and Xin Cai, "Software Defined Radio Implementation of the Dual Link Algorithm in TDD Mode using USRP E310." Proceedings of the GNU Radio Conference. Vol. 1. No. 1. 2016.*

1.3 Outline of The Thesis

This thesis primarily focus on specific topics in the physical layer baseband communications of wireless communication systems, including interference alignment, synchronization approaches and SDR implementations. The rest of this thesis is organized as follows: in Chapter 2, we present an overview of wireless communication systems, including a glance at different types of wireless systems and impairments that making the wireless transmission imperfect. In Chapter 3, the development history of wireless communication systems, related literature and research studies are reviewed. The review of literature and research studies in Chapter 3 mainly focuses on carrier frequency offset estimation

techniques, MIMO interference alignment techniques and the hardware implementations of these techniques. Related works on other estimations and synchronizations of other common impairments, as well as SDR techniques are also reviewed and discussed in this chapter. Chapter 4 presents the methods we proposed during our research studies. This chapter covers the methods and hardware architectures of carrier frequency offset synchronizations, hardware architectures proposed for improvement of MIMO systems. Following Chapter 4, we conduct simulations and experiments in Chapter 5 to verify the performance of proposed methods and architectures. Comparisons with related methods are also made in this chapter. Finally in Chapter 6, we conclude our work and present the future directions of our research at the same time.

Chapter 2

Overview of Wireless Communication Systems and Impairments

In this chapter, we discuss the back ground of wireless communication systems. First, wireless communication systems is categorized according to different disciplines such as modulation scheme, number of antennas and so on. Then, we focus on modulation schemes and take a more comprehensive look at specific system models. Finally, we discuss the impairments that a robust wireless system should take care of to obtain correct messages. In addition, analysis of advantages and disadvantages of these systems are also posted in this chapter.

2.1 Overview of Wireless Communication Systems

Following the establishment of Maxwell's equations, scientists and engineers realized the huge potential of wireless media [24]. With the rapid development of technology, wireless transmission in nowadays is one of the most commonly used technologies in a variety of applications [25–29]. For instance, Wi-Fi [30] and cellular networks [31] are essential

parts of our life, we can simply dial the number on our cellphones and call friends and family members living far away, or connect the Wi-Fi at home to entertain ourselves. These common activities are supported heavily by a variety of wireless communication systems.

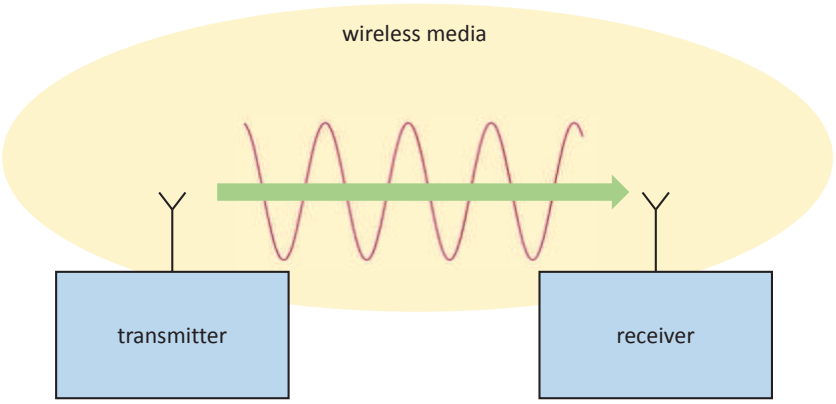

Figure 2.1: General model of a wireless communication system

A wireless communication system contains at least one transmitter and one receiver [32], the transmitter generates information bits and emit the signal out while at the receiver the messages are captured and represented, as shown in Figure 2.1. However, the structures of transmitters and receivers may vary according to system models. The details of different types of systems regarding to modulation scheme will be demonstrated explicitly in following two sections.

2.2 Single Carrier Systems

Traditionally, a certain bandwidth at a certain time is assigned to a single user, and the user could make use of the allocated bandwidth for information exchange [21]. Therefore, the information bits is modulated into symbols and entirely loaded to the single carrier. The wireless communication systems in this kind is called single carrier (SC) system.

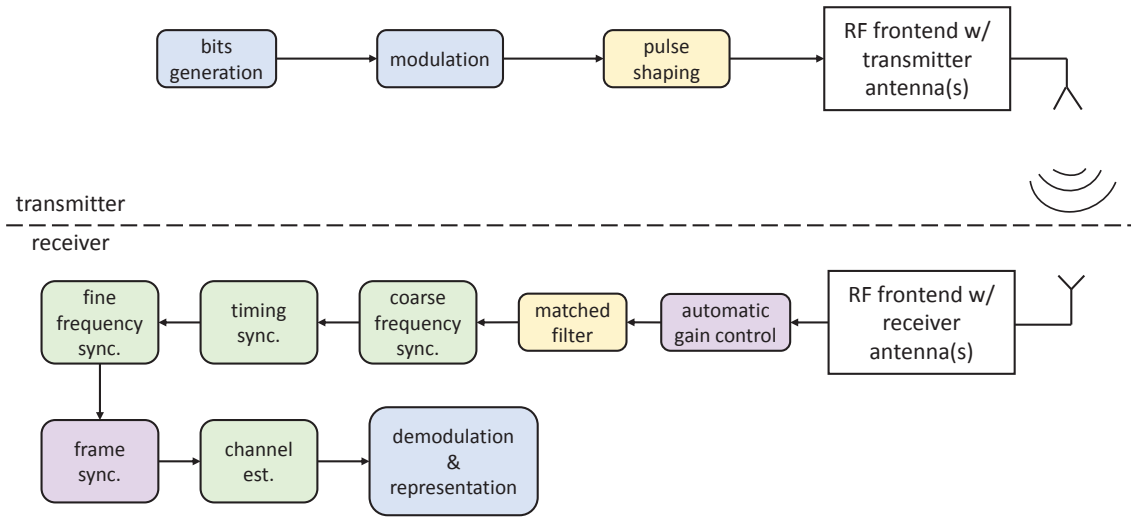

Figure 2.2: Structure of general transmitter and receiver in a single carrier system

A general structure of single carrier system is illustrated in Figure 2.2. The transmitter is responsible for data generation, baseband modulation and pulse shaping which is against inter-symbol interference (ISI) effects [33], while the receiver is in charge of recovering the signals from all kinds of impairments.

When a transmitted frame is received, the first thing is to adjust the received signal's amplitude using automatic gain control (AGC) [34], then a matched filter is applied to detect the transmitted digital symbols [35]. Later, the symbols at hand is combined with frequency offset, timing offset and channel influence, therefore frequency synchronization (coarse and fine), timing synchronization units are employed to eliminate the offsets. Frame synchronization (or frame detection) is performed so that the receiver could find the pilot sequence in a frame to estimate the channel. Finally, recovered symbols pass the demodulator and are represented to users.

2.3 OFDM Systems

With the rapid development of wireless communication technology and increase of customers of wireless communication products, the bandwidth usage regulated by federal communication commission (FCC) is limited comparing to the huge demand. Also, the complex metropolitan environment causes severe multipath effect. Therefore, orthogonal frequency division multiplexing (OFDM) technology has grown as one of the key modulation schemes in modern wireless communications [36].

In general, an OFDM system occupies the bandwidth using a number of subcarriers each with a small portion of the entire allocated spectrum [37]. To mange the subcarrier in a easier way, the information symbols are loaded to subcarriers in frequency domain instead of directly manipulated in time domain.

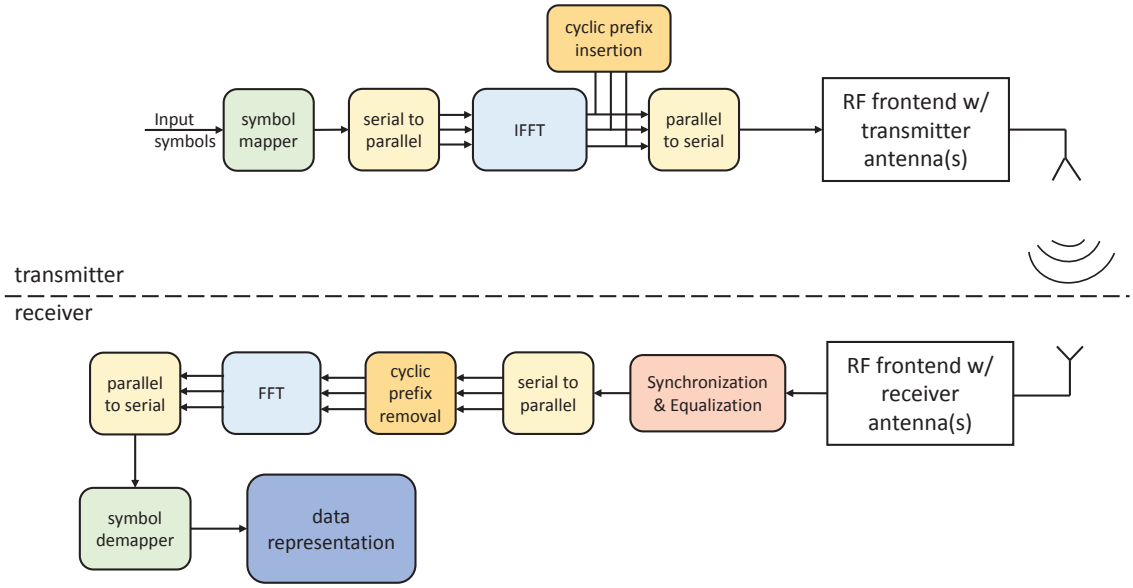

Figure 2.3: Structure of general transmitter and receiver in an OFDM system

Figure 2.3 demonstrates the structure of a general OFDM system. From this figure we see that the concept of OFDM system is making the use of separated subcarrier so that interference and crosstalk are reduced.

2.4 Common Impairments in Wireless Communication Systems

Under most conditions, transmission over wireless medium is not simple as it sounds like [38]. Since transmitter and receiver are separate devices, the oscillators, clocks and other components could work asynchronous at transmitter an receiver, which leads to frequency and timing offset. On the other hand, the wireless media are not ideal, either. There exists interference from other users whose spectrum may overlap with yours, background noise, and fading or multipath channels. These factors also result in degradation of wireless communication systems' performance.

2.4.1 Carrier frequency offset

Carrier frequency offset (CFO) often occurs because of the frequency mismatch between oscillators in transmitter and receiver, or the Doppler shift caused by relative displacement of transmitter and receiver. If left unattended, the carrier frequency offset will lead to drifting of the signal and finally ruin the received signal. The symbols points affected by CFO is shown in Figure 2.4. As can be seen in the figure (left), the symbols with additive noise are still gathered around constellation points, while the CFO causes rotation and therefore it is hard to demodulate.

2.4.2 Symbol timing offset

Symbol timing offset (STO) often occurs due to the mismatch of sampling moments. In a wireless communication systems, pulse shaping is usually performed to avoid inter symbol interference (ISI), and therefore requires upsampling and downsampling. Unfortunately, if upsampling and downsampling moments don't match up, the recovered

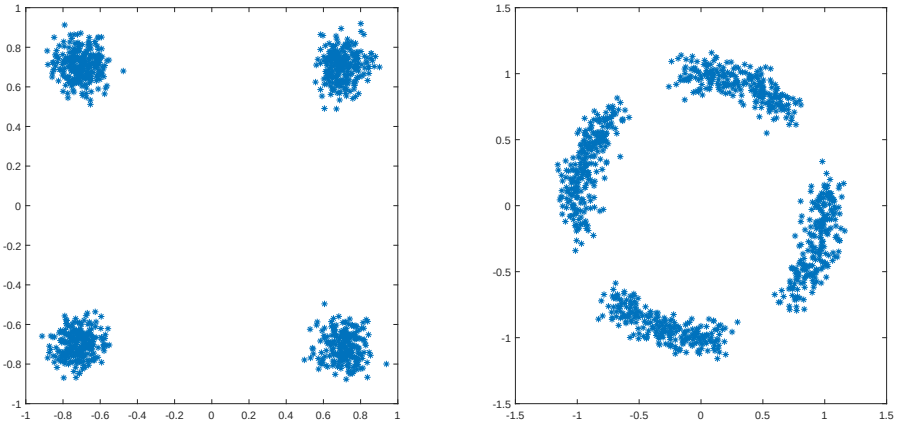

Figure 2.4: Effects of CFO on noised symbols

symbols may stay far away from correct sample points and therefore the amplitudes of samples are scattered everywhere, as shown in Figure 2.5.

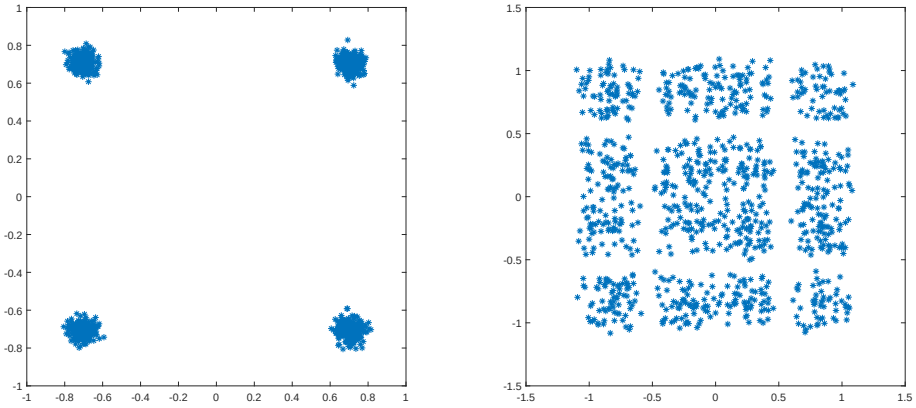

Figure 2.5: Effects of STO on noised symbols

2.4.3 Channel impacts

Channel usually refers to the wireless transmission medium in wireless communications systems, and has important influences to received signal. Since channel is complex, it may affect the magnitude and phase during the transmission. If the environment is complicated

where multipath channel appears, your transmitted signal may suffer reflections and you even may have multiple received signals overlaying with each other.

2.4.4 Interference and noise

Interference and noise are disturbance from different sources. Interference is the signal from other transmitters occupying the same bandwidth, thus it is hard to simply split the interference out by implementing filters. On the other hand, noise is usually from physical surrounding objects, passively or actively. Slightly different from interference, signals can be relieved from noise by applying filters so that the noise outside signal bandwidth could be removed. Technically, by applying interference aligning method, some specific interference can be aligned while some other interference cannot. In addition, unless background noise is dominating, it can be left in the system as long as it does not result in severe degradation in general cases.

Chapter 3

Background and Related Works

3.1 History of Wireless Communication Systems

The history of "wireless world" started with understanding of electromagnetic phenomena around us. In 1865, Maxwell predicted the existence of electromagnetic waves by deriving the electromagnetic wave equation in his paper "*A Dynamical Theory of the Electromagnetic Field*". Since then, the history of wireless communications started, and finally human stepped into the era of wireless communications. In the first several decades, however, the development of wireless communication systems didn't ramp to its highway until Marconi demonstrated wireless telegraph in 1896. In the next decade, Marconi continued his experiments on wireless radio signal transmission, and started commercializing his wireless telegraphy equipment.

With the first radio transmission of voice in 1914, the development of wireless communications pushed the content of wireless transmission to a more instant form. 13 years later in 1927, the first long-distance TV transmission was conducted by AT&T Bell Labs, opening the gate to modern life with TV shows. Since then, the market of commercial service of wireless communications grew exponentially: The first TV station settled in

New York in 1928, first FM radio patented in 1935. Growth of the market stimulated the development of wireless communication services. In 1946, public switched telephone network (PSTN) operated its first interconnection of mobile users and soon in 1946, Federal Communications Commission recognized mobile radio as a new class of wireless service. The number of mobile users of PSTN in the next 20 years increased dramatically from less than 50,000 to more than 1,400,000. With the explosion of mobile user population, improved mobile telephone service (IMTS) was merged to PSTN in 1964 to support functions such as full-duplex and auto dial services.

Later on, the wireless communications technologies are being overwhelmingly developed and deployed [31, 39, 40]. First cellular communication system was deployed in Tokyo in 1979, initiated the new age of personal cellular communications. In the following 40 years, cellular communications has rapidly evolved from the first generation (1G) to the fifth generation (5G). During the evolution from 1G to 2G, the radio signal evolved from analog to digital, and short message service (SMS) was introduced. Then with the development of code-division multiple access (CDMA) technology, 2G evolved to 3G and the 3G network higher and higher data rate. In addition to 3G, 4G technology provides broadband internet access and the orthogonal frequency-division multiple access brought 4G users much higher data rate. 40 years after the birth of 1G, we are now standing at the edge of 5G, which is promising to bring users $1Gbit/s$ data rate and way lower communication latency.

Along with the development of cellular communication, wireless local area network (WLAN) also gained its popularity in the past 30 years [41, 42]. The first wireless product fitting the IEEE 802.11 standards, WaveLAN, was invented in 1991. In the following years, IEEE 802.11 family grew up quickly with evolving standards. The modulations used in the standards also brought higher data rate by making use of OFDM and MIMO technologies. Nowadays, people can enjoy the perks brought by most recent Wi-Fi tech-

nology at home, such as ultra high definition (UHD) video, wireless video casting and so on.

3.2 Literature Review

In this section, the literature review covers a variety of topics, but mainly focuses on carrier frequency offset estimation techniques, MIMO interference alignment techniques and the hardware implementations of these techniques. Related works on other synchronization techniques, as well as SDR techniques are also briefly reviewed and discussed in this section.

3.2.1 High-level wireless communication system design and synchronization

As discussed in last section, wireless communications has been under development and investigation for many decades. Every generation of engineers and researchers have their understanding of wireless communications and these precious knowledge is recorded to benefit the next generation. For the many years of wireless communication systems development, [1, 43–45] are widely used textbooks for wireless receiver design on baseband. Besides, [46] is a well know book that covers more comprehensive topics of the entire procedure of wireless communications. These books draw the general picture of how wireless communication systems work, what kinds of imperfections exist in practical wireless communication systems and how we should recover the wireless systems from the imperfections. Researchers have devoted their efforts on developing and improving techniques to fix the imperfections.

3.2.2 Carrier frequency synchronization

CFO is one of the most common impairments in wireless communication systems, which often occurs due to the oscillator frequency mismatch or Doppler shift [47]. Once CFO occurs, the received signal is shifted by a certain amount in frequency domain and is distorted in time domain, preventing users from obtaining correct messages. Therefore, CFO estimation is usually the first step at receiver side. CFO estimation usually consists of two steps: coarse estimation and fine estimation [48]. The coarse CFO estimation roughly estimate the carrier frequency offset so that the residual after coarse CFO compensation has minimal negative influence on next synchronization module. On the other hand, fine CFO estimation is expected perform very high-accuracy estimation of the residual CFO so that the the received signal can be perfectly recovered to avoid message errors.

Nasraoui et al [49] proposed a pilot aided correlation method, taking the sliding correlation of structured preamble to estimate the frequency offset in a closed form solution. The method had low complexity, however its estimation accuracy was not the best among all competitive algorithms. Pilot aided correlation method provides a tradeoff between performance and complexity. A correlation-based estimation method was proposed for MIMO systems in [50]. It took the advantages of orthogonal pilots to eliminate the interference among antennas and calculate the frequency offsets between each pair of antennas. Due to this fact, the complexity of this algorithm was relatively high. The other disadvantage of this algorithm was that the mean squared error (MSE) of frequency estimation diverged from given Cramer-Rao bound when the signal-to-noise ratio (SNR) became higher, showing that the algorithm was not optimal.

In [51], a maximum-likelihood based estimator (MLE) for frequency offset was proposed for single-input single-output (SISO) system, followed by different solutions to the MLE. Among these solutions, the linear approximation utilizes only small part of the received signal to avoid phase unwrapping, which leads to a degradation of its performance.

Newton search and local grid search were also applied in [51] to refine the estimates of linear approximation method, while the Newton search frequently fails because of local maximums and the accuracy of uniform search depends heavily on resolution and search range. Recently, an high resolution CFO estimation algorithm was proposed in [52]. This algorithm uses Golay sequence as pilot signal and provides closed form solution to CFO estimation. Another advantage of this algorithm is that the MSE of estimated CFO keeps dropping along with the SNR increase while that of the method in [51] almost stops decreasing at high SNR. Unfortunately, in spite of its competitive performance, this algorithm is designed with the assumption that the communication system is SISO. On the other hand, by avoiding phase wrapping and unwrapping, this algorithm has limited CFO acquisition range.

Besides the CFO estimation algorithms for non-OFDM systems, CFO estimation methods are also proposed for OFDM systems. In [53], a maximum likelihood (ML) based CFO estimation method using null subcarrier insertion scheme was proposed. The advantage of this method is that the inserted null subcarriers extend the frequency acquisition range so that the proposed method can estimate both integer and fractional CFO of an OFDM system. However, the expanded acquisition range is achieved at the expense of spectral efficiency, resulted by the inserted subcarriers.

In last decade, the combination of OFDM and MIMO system brought us a dramatic increase of wireless data transmission rate, and CFO estimation techniques were also investigated in a number of research studies. In [54], an CFO estimator for MIMO OFDM system based minimum the channel residual energy (CRE) is proposed while the corresponding solution to finding the minimum CRE is not given. Grid search of the maximum was employed in [55]. Iterative search via alternating projection frequency estimator (APFE) and approximate alternating projection frequency estimator (AAPFE) were employed in [56]. The grid search approach in [55] showed a clear gap between MSE of

frequency estimation and corresponding CRLB while the approaches in [56] got closer to corresponding CRLB. All these searching methods showed divergence from CRLB at high or low SNR. Besides, in [57], authors proposed a closed form data-aided least square (LS) estimator to estimate the phase change due to the frequency mismatch in MIMO system. This closed form estimator, however, still diverged from corresponding CRLB at high SNR. In [58–61], a group of researchers investigated the CFO estimation problem in massive MIMO multiuser OFDM systems where multipath channel models are considered. In [58], a computationally efficient blind CFO estimator was presented for MIMO OFDM system where the minimization of cost function is computationally efficient. The proposed method in [59] utilized scattered pilot subcarriers to separately estimate CFO for each user so that multidimensional search can be avoided. Another CFO estimator for multi-user MIMO OFDM systems was proposed in [60], taking a joint spatial-frequency alignment procedure so that the CFO can be estimated for users separately. Later in [61], an angle-domain adaptive filtering based CFO synchronization method was proposed. This method performs a two-stage multiuser interference (MUI) suppression so that the CFO of users with separate or overlapped angle-of-arrival regions can be estimated individually.

3.2.3 Interference alignment

The available radio spectrum under current FCC standards is becoming more and more limited, an important method of increasing network capacity has been increasing spatial reuse by reducing the cell size and letting cells reuse the same spectrum. However, increasing the density of access points or base stations cannot truly resolve the capacity crunch without proper interference management, meaning the transmit signals should be designed to strike an optimal balance between maximizing a link's own rate and reducing interference to other links [62, 63]. A simple example of MIMO interference networks is

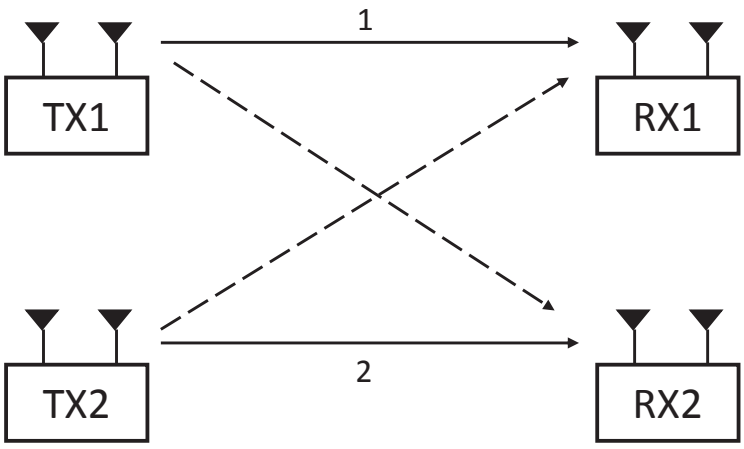

Figure 3.1: A simple example of interference networks

shown in Figure 3.1, where data links mutually interfere with each other.

For MIMO interference networks, two important algorithms had been proposed to maximize the weighted sum rate of the wireless network. They are the weighted minimum mean square error (WMMSE) algorithm [64] and polite water-filling (PWF) algorithm [65, 66]. However, both the WMMSE and PWF algorithms have drawbacks. The PWF algorithm has a fast speed of convergence while it is not guaranteed to be monotone. The WMMSE algorithm converges to a stationary point while it converges slower. A dual link algorithm for MIMO interference networks was proposed in [16], in order to manage the interference and maximize the weighted sum rate of interference networks. The dual link algorithm is scalable and has the advantages of both WMMSE and PWF algorithms, i.e., fast monotone convergence. It jointly optimizes the covariance matrices, or equivalently, the beamforming matrices, of transmit signals of multiple transmitters, assuming Gaussian transmit signal and the availability of all channel state information. The dual link algorithm is ideally suited for distributed implementation that only requires local channel information in a time division duplex (TDD) system.

To maximize the data rate of MIMO interference networks, decomposition of matrices are commonly employed. The singular value decomposition (SVD) is a key technique

in many signal processing algorithms that perform factorization of real and complex matrices. It is employed in a number of applications in wireless communications such as channel estimation, interference alignment and so on. With the rapid development of MIMO technology, SVD is more and more widely engaged with modern communication systems. Because of its importance, a number of researches of SVD algorithm and its hardware implementation for MIMO systems are conducted in recent years. In [67] the authors proposed an adaptive SVD algorithm with low complexity that can track the channel response matrix. This algorithm, however, causes accuracy degradation in hardware implementation because of a sequence of division operations. And the authors of [68] presented a steering matrix computation architecture performs SVD operation yet its convergence speed may missed the convergence requirement. To avoid these drawbacks, in [69], the authors proposed a super-linear SVD algorithm that can be implemented on hardware with enough accuracy for MIMO size up to 4-by-4.

More recently, massive MIMO makes a clean break with traditional MIMO through the implementation of a very large number of service antennas that are operated fully coherently and adaptively. Extra antennas help by focusing the transmission and reception of signal energy into smaller regions of space. This brings huge improvements in network throughput and energy efficiency, especially when combined with simultaneous scheduling of a large number of user terminals [70]. A massive MIMO system, though differs from traditional MIMO systems, still requires SVD operations as long as the system contains precoding schemes or any other matrix factorization. Therefore, the SVD with large complex matrices becomes an important part of massive MIMO.

3.2.4 Hardware implementation of algorithms and applications

As discussed above, the algorithms and applications proposed by researchers sometimes are only verified in simulations. To realize these algorithms and applications, directly fab-

ricate them on matured commercial wireless communication systems could be way too costly. Under this condition, SDR might be a better choice. SDR was initially regarded as a promising solution to the tremendously increased demand of customized radio system and has become a popular platform for realizing and prototyping wireless communication systems with special functionalities. The SDR systems, in most cases, implement only basic digital signal processing operations on FPGA while the algorithms are usually running on CPUs [71] [72]. But for high performance SDR system, field programmable gate array (FPGA) is often employed as a key component where customized hardware logic can be implemented. Therefore, the discussed algorithms and applications can be implemented on the hardware of SDR platforms. When compared with a general purpose processor, FPGA has the advantages of higher computing performance at lower power consumption for an application-specific task. Utilizing its rich resources of computational units, an FPGAs can accelerate the implementation of complex algorithms even with strict real-time requirement [73], such that the customized algorithms can be all implemented on the FPGA. In [21], a modular designed wireless communication system was implemented on an FPGA-based SDR platform.

Some other hardware implementations of wireless communication systems were proposed in [74–77], but these implementations focused more on decoding where frequency synchronization was not considered. As mentioned in Section 3.2.2, without accurate frequency synchronization, communications may suffer higher bit error rate (BER) because of the mismatch between oscillators, or the transmission could even be ruined. Some other recent works [78–81] proposed architectures and implementations of CFO synchronizations for orthogonal frequency division duplex (OFDM) based or non-OFDM based SISO systems.

Chapter 4

Methodology

In this chapter, our approaches for CFO estimation and MIMO interference alignment are presented in detail. In Section 4.1, we propose an hardware architecture of dual link rate maximization algorithm. However the decomposition employed in Section 4.1 is Cholesky decomposition, the hardware consumption of which increases dramatically along with the number of antennas. Therefore in Section 4.2, we propose a hardware-efficient SVD implementation which has almost invariant resource consumption for any number of antenna. On the other hand, transceivers in a wireless communication system can not communicate if there exists carrier frequency offset. Thus, in Section 4.4, a asymptotically optimal CFO estimation algorithm as well as hardware efficient architecture is demonstrated. Later in Section 4.5, we dive into the area of coarse CFO estimation and propose a neural network architecture for coarse CFO estimation.

4.1 Hardware Implementation of Dual Link Algorithm

4.1.1 Algorithm Description

4.1.1.1 The Dual Link Algorithm

We consider a general interference network with L interfering data links. Link l's physical transmitter is T_l, which has L_{T_l} many antennas. Its physical receiver is R_l, which has L_{R_l} many antennas. The received signal at R_l is

$$\mathbf{y}_l \;=\; \sum_{k=1}^{L} \mathbf{H}_{l,k}\mathbf{x}_k + \mathbf{n}_l, \tag{4.1}$$

where $\mathbf{x}_k \in \mathbb{C}^{L_{T_k}\times 1}$ is the transmit signal of link k and is modeled as a circularly symmetric complex Gaussian vector; $\mathbf{H}_{l,k} \in \mathbb{C}^{L_{R_l}\times L_{T_k}}$ is the channel state information (CSI) matrix between T_k and R_l; and $\mathbf{n}_l \in \mathbb{C}^{L_{R_l}\times 1}$ is a circularly symmetric complex Gaussian noise vector with identity covariance matrix.

The optimization problem to be solved is the weighted sum-rate maximization under a total power constraint:

$$\begin{aligned} \textbf{WSRM\_TP}: \quad \max\nolimits_{\mathbf{\Sigma}_{1:L}} \quad & \sum_{l=1}^{L} w_l \mathcal{I}_l\left(\mathbf{\Sigma}_{1:L}\right) \\ \text{s.t.} \quad & \mathbf{\Sigma}_l \succeq 0, \forall l, \\ & \sum_{l=1}^{L} \mathrm{Tr}\left(\mathbf{\Sigma}_l\right) \le P_{\mathrm{T}}, \end{aligned} \tag{4.2}$$

where $w_l > 0$ is the weight for link l, and we can set $w_l = 1$ such that each link are equally weighted. Assuming the channels are known at both the transmitters and receivers (CSITR), an achievable rate of link l is

$$\mathcal{I}_l\left(\mathbf{\Sigma}_{1:L}\right) = \log\left|\mathbf{I} + \mathbf{H}_{l,l}\mathbf{\Sigma}_l\mathbf{H}_{l,l}^{\dagger}\mathbf{\Omega}_l^{-1}\right| \tag{4.3}$$

where $\mathbf{\Sigma}_l$ is the covariance matrix of $\mathbf{x}_l$; and $\mathbf{\Omega}_l$ is the interference-plus-noise covariance matrix of the l^{th} link,

$$\mathbf{\Omega}_l = \mathbf{I} + \sum_{k=1,k\neq l}^{L} \mathbf{H}_{l,k}\mathbf{\Sigma}_k\mathbf{H}_{l,k}^{\dagger}. \tag{4.4}$$

The Dual Link algorithm for this optimization problem is given in Algorithm 1. It is an iterative algorithm with fast and monotone convergence [16]. Since the problem is non-convex, the algorithm converges to a local optimal point. Name the original channel forward link channel. The terms $\hat{\mathbf{\Sigma}}_l$ and $\hat{\mathbf{\Omega}}_l$ in the algorithm are corresponding terms in a reverse link channel, where the roles of the transmitters and receivers are exchanged and the channel $\mathbf{H}_{l,k}$ is replaced by $\mathbf{H}_{l,k}^{\dagger}$. The reverse links can be virtual links for the description of the algorithm. But in a TDD system, the reverse links exist physically, leading to a distributed implementation of the algorithm.

Algorithm 1: The Dual Link Algorithm

1. Initialize $\mathbf{\Sigma}_l$'s, s.t. $\sum_{l=1}^{L} \text{Tr}\left(\mathbf{\Sigma}_l\right) = P_{\text{T}}$
2. $R \Leftarrow \sum_{l=1}^{L} w_l \mathcal{I}_l\left(\mathbf{\Sigma}_{1:L}\right)$
3. Repeat
4. $R^{'} \Leftarrow R$
5. $\mathbf{\Omega}_l \Leftarrow \mathbf{I} + \sum_{k\neq l} \mathbf{H}_{l,k}\mathbf{\Sigma}_k\mathbf{H}_{l,k}^{\dagger}$
6. $\hat{\mathbf{\Sigma}}_l \Leftarrow \frac{P_{\text{T}} w_l \left(\mathbf{\Omega}_l^{-1} - \left(\mathbf{\Omega}_l + \mathbf{H}_{l,l}\mathbf{\Sigma}_l\mathbf{H}_{l,l}^{\dagger}\right)^{-1}\right)}{\sum_{k=1}^{L} w_k \text{tr}\left(\mathbf{\Omega}_k^{-1} - \left(\mathbf{\Omega}_k + \mathbf{H}_{k,k}\mathbf{\Sigma}_k\mathbf{H}_{k,k}^{\dagger}\right)^{-1}\right)}$
7. $\hat{\mathbf{\Omega}}_l \Leftarrow \mathbf{I} + \sum_{k\neq l} \mathbf{H}_{k,l}^{\dagger}\hat{\mathbf{\Sigma}}_k\mathbf{H}_{k,l}$
8. $\mathbf{\Sigma}_l \Leftarrow \frac{P_{\text{T}} w_l \left(\hat{\mathbf{\Omega}}_l^{-1} - \left(\hat{\mathbf{\Omega}}_l + \mathbf{H}_{l,l}^{\dagger}\hat{\mathbf{\Sigma}}_l\mathbf{H}_{l,l}\right)^{-1}\right)}{\sum_{k=1}^{L} w_k \text{tr}\left(\hat{\mathbf{\Omega}}_k^{-1} - \left(\hat{\mathbf{\Omega}}_k + \mathbf{H}_{k,k}^{\dagger}\hat{\mathbf{\Sigma}}_k\mathbf{H}_{k,k}\right)^{-1}\right)}$
9. $R \Leftarrow \sum_{l=1}^{L} w_l \mathcal{I}_l\left(\mathbf{\Sigma}_{1:L}\right)$
10. until $\left|R - R^{'}\right| \leq \epsilon$ or a fixed number of iterations are reached.

4.1.1.2 Distributed Algorithm and Local Channel Information Estimation

In a TDD system, the dual link algorithm can be readily implemented in a distributed and low complexity fashion. We take advantage of the physical reverse link for the distributed algorithm. For example, in Step 6 of Algorithm 1, to update the reverse link l's transmit

signal covariance $\hat{\mathbf{\Sigma}}_l$, we only need to estimate local interference plus noise covariance $\mathbf{\Omega}_l$ and local total received signal covariance $\mathbf{\Omega}_l + \mathbf{H}_{l,l}\mathbf{\Sigma}_l\mathbf{H}_{l,l}^\dagger$ from the forward link received signal. This can be done with low complexity because the channel has summed the interference for us for free. There is no need to estimate $\mathbf{H}_{k,l}$ for all k and l. The reverse link calculation in Step 8 can be done similarly using the physical reverse link received signal.

The distributed algorithm and local channel information estimation for Step 6 is as follows. Assume forward link l uses precoding matrix $\mathbf{V}_l$ to transmit orthogonal pilot signal $\mathbf{P}_l \in \mathbb{C}^{L_{T_l} \times n}$ with n channel uses, where $\mathbf{V}_l\mathbf{V}_l^\dagger = \mathbf{\Sigma}_l$ and $\mathbf{P}_l\mathbf{P}_l^\dagger = n\mathbf{I}_{L_{T_l} \times L_{T_l}}$. For example, Hadamard matrices may be used for the pilots. In practice, pilots of different users can be near orthogonal. The received signal of forward link l is

$$\mathbf{Y}_l \;=\; \sum_{k=1}^{L} \mathbf{H}_{l,k}\mathbf{V}_k\mathbf{P}_k + \mathbf{N}_l \in \mathbb{C}^{L_{R_l} \times n}.$$

The least square based estimation of link l's own signal covariance is

$$\mathbf{H}_{l,l}\mathbf{\Sigma}_l\mathbf{H}_{l,l}^\dagger \doteq \mathbf{A}_l \;=\; \left(\frac{\mathbf{Y}_l\mathbf{P}_l^\dagger}{n}\right)\left(\frac{\mathbf{Y}_l\mathbf{P}_l^\dagger}{n}\right)^\dagger. \tag{4.5}$$

The estimated total received signal covariance of link l is

$$\mathbf{\Omega}_l + \mathbf{H}_{l,l}\mathbf{\Sigma}_l\mathbf{H}_{l,l}^\dagger \doteq \mathbf{B}_l \;=\; \frac{\mathbf{Y}_l\mathbf{Y}_l^\dagger}{n}. \tag{4.6}$$

Then, instead of using Step 5, which requires global channel knowledge, the interference plus noise covariance $\mathbf{\Omega}_l$ can be estimated as

$$\mathbf{\Omega}_l \;\doteq\; \mathbf{B}_l - \mathbf{A}_l. \tag{4.7}$$

Using (4.6) and (4.7), Step 6 can be calculated. Note that the normalization in Step 6 is to satisfy the total power constraint and can be implemented by adjusting in small steps and sharing one scalar constant

$$\mu = \frac{1}{P_{\mathrm{T}}} \sum_{l=1}^{L} w_l \mathrm{tr} \left(\mathbf{\Omega}_l^{-1} - \left(\mathbf{\Omega}_l + \mathbf{H}_{l,l} \mathbf{\Sigma}_l \mathbf{H}_{l,l}^{\dagger} \right)^{-1} \right), \tag{4.8}$$

similar to power control in CDMA networks. Step 8 of Algorithm 1 can be similarly calculated using reverse link pilot and received signals. Once the covariance matrices $\mathbf{\Sigma}_l$ and $\hat{\mathbf{\Sigma}}_l$ are calculated, the precoding or beamforming matrices $\mathbf{V}_l$ and $\hat{\mathbf{V}}_l$ can be calculated using Cholesky decomposition using FPGA [82].

As seen from the above, the distributed algorithm is scalable and only needs local information, except for the sharing of the normalization constant μ.

4.1.1.3 Algorithm Simulation

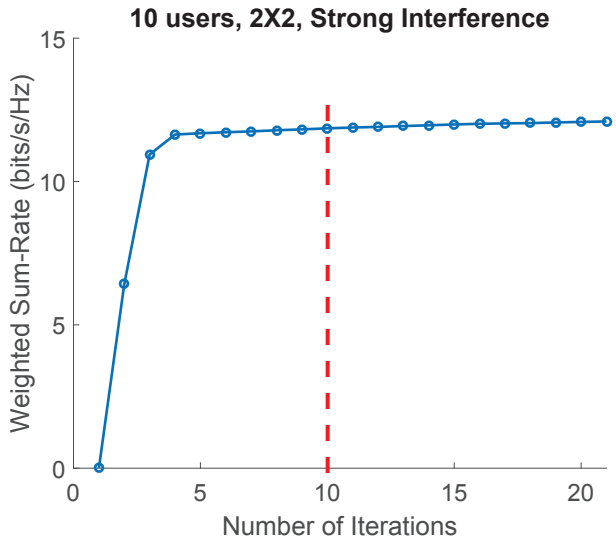

Figure 4.1: Simulation results of rate maximization algorithm

After applying our rate maximization algorithm with a cluster of 10 users each with 2-by-2 MIMO and the pilot length is set to be 4000, the relationship between sum rate of

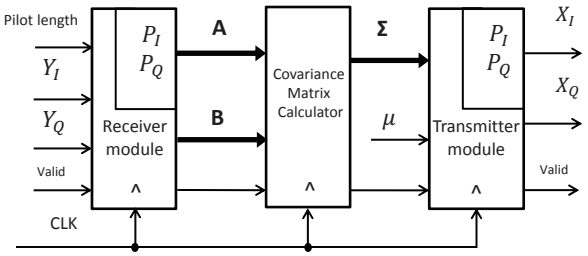

Figure 4.2: Overall system diagram of FPGA prototype

the network and the number of iterations is illustrated in Figure 4.1. It can be observed that the dual link sum rate maximization algorithm converges to its final result fast yet smoothly. Meanwhile, the sum rate of the network goes very close to its final value after around 10 iterations.

4.1.2 System Prototyping Using FPGA

A typical MIMO system usually consists of two parts: receiver and transmitter [83]. Receiver is responsible for receiving signals from antennas and transmitter is responsible for sending out designated signals through multiple antennas. Resembling basic radio structures, in our prototype design, we introduce an additional computational module called covariance matrix calculator that computes $\mathbf{\Omega}$ and $\mathbf{\Sigma}$ in every iteration. If we allocate multiple users in a mutually detectable environment and assign them initialized signal covariance matrices, these users will send constructed pilot signals to each other, train signal covariance matrices on each link, and finally force those matrices converge to their optimal values. The top-level system diagram of an individual user is illustrated in Figure 4.3. Note that Y_I, Y_Q, X_I and X_Q represent the real and imaginary part of received signal and transmitted signal, respectively.

4.1.2.1 Receiver Module Design

Figure 4.3 is the diagram of receiver module of a single user which receives the signals from the wireless channel and prepares necessary matrices for computation of signal covariance matrix. As the pilot $\mathbf{Y}$ is received, multiply accumulator computes the signal covariance and accumulator outputs are then dividThe order of sections in this chapter is the same as in Chapter 4.ed by the length to produce the intermediate matrices $\mathbf{B}$ and $\mathbf{A}$ as in (5) and (6). Owing to the independent Gaussian noise and mutual orthogonality of pilots, it is not necessary to find the summation of interference and noise by adding them up. We can simply exploit received signal and foregone pilots to compute intermediate matrices and process them later in the covariance matrix calculator. Part (a) of Figure 4.3 depicts the receiver module of fixed point structure and part (b) depicts that of floating point structure. The difference between two structures is that two "constant & matrix multiplier" units in floating point structure are employed working parallelly while a mux controller is used in fixed point structure to make it work serially. Detailed comparison between the parallel structure and serial structure is discussed at the end of Section 4.1.2.

4.1.2.2 Covariance Calculator Module Design

Figure 4.4 illustrates the hardware processing steps leading to signal covariance matrix $\mathbf{\Sigma}$. After computation of $\mathbf{A}$ and $\mathbf{B}$, we subtract $\mathbf{A}$ from $\mathbf{B}$ to find the interference-plus-noise matrix $\mathbf{\Omega}$ as in (7). Applying matrix inversion and another subtraction, the signal covariance matrix $\mathbf{\Sigma}$ can be obtained. In terms of the matrix inversion, adjoint matrix method [84] is employed in our design, which requires to calculate the reciprocal of the determinant of a complex matrix. Since the determinant of a complex matrix is a complex number under most conditions, we need to compute the reciprocal of a complex number during matrix inversion module. We write our own complex reciprocal submodule for fixed point structure owing to the fact that there is no complex reciprocal Xilinx IP core

provided for fixed point calculation. Assume we have a complex number $a_I + a_Q j$, and its reciprocal is $b_I + b_Q j$, where $b_I = \frac{a_I}{a_I^2 + a_Q^2}$ and $b_Q = \frac{-a_Q}{a_I^2 + a_Q^2}$.

4.1.2.3 Transmitter Module Design

The transmitter module shown in Figure 4.5 is simply a serial structure. With intermediate input $\mathbf{\Sigma}$ and the power adjustment coefficient μ (as in (7) and (13)) which is given by a central controller in the network, it is easily to obtain adjusted signal covariance matrix $\hat{\mathbf{\Sigma}}$. From the derivation in Section 4.1.1, it is obvious that both $\mathbf{\Sigma}$ and $\hat{\mathbf{\Sigma}}$ are positive semi-definite matrix. Although strictly speaking Cholesky decomposition is only applicable to positive definite matrix, we can still apply Cholesky decomposition to $\mathbf{\Sigma}$ and obtain the decomposed lower triangular matrix $\mathbf{V}$ [85]. We then transmit the product of $\mathbf{V}$ and pilot $\mathbf{P}_l$.

4.1.3 Comparison between parallel and serial structure

Many units in the fixed point structure perform computations serially while the units in the floating point structure work in parallel. Every of these units performs more than one complex multiplication operations. The parallel structure of the floating point is shown in Figure 4.6. Each unit with parallel structure instantiates multiple complex multipliers that operate in parallel, and each complex multiplier requires 4 multipliers computing real and imaginary parts. On the contrary in Figure 4.7, floating point units only instantiate one complex multiplier which consists of only a single multiplier, but this multiplier is reused frequently in the operation.

Therefore, the emphasis of parallel structure and serial structure differs — parallel structure clings to timing performance while serial structure sticks to resource utilization efficiency. For instance, computational latency of a floating point multiplier is 8 cycles while that of a fixed point multiplier is 1 cycle. However, each 2×2 complex matrix mul-

tiplication requires 8 complex multiplications and each complex multiplication requires 4 multiplication operations. Consequently, a fixed point serial structure takes at least 32 cycles to complete a complex matrix multiplication, while a parallel structured floating point complex matrix multiplier takes only 8 cycles to complete the same operation since it has 32 multipliers working simultaneously.

4.2 Hardware-Efficient Scalable Singular Value Decomposition

Suppose we have a MIMO system with N transmitter antennas and N receiver antennas equipped, and the channel between transmitter and receiver antennas is $\mathbf{H} \in \mathbb{C}^{N \times N}$. If we perform a channel matrix decomposition using SVD technique [13], we have

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\dagger} \quad \Leftrightarrow \quad \mathbf{\Sigma} = \mathbf{U}^{\dagger}\mathbf{H}\mathbf{V} \tag{4.9}$$

where $\mathbf{U} \in \mathbb{C}^{N \times N}$ and $\mathbf{V} \in \mathbb{C}^{N \times N}$ are unitary matrices . $\mathbf{\Sigma} \in \mathbb{C}^{N \times N}$ is the singular value matrix with non-negative real entries on diagonal.

4.2.1 The Jacobi SVD Method

Luk proved that by taking orthogonal two-sided Jacobi rotations to the channel matrix $\mathbf{H}$ on both sides iteratively, the Jacobi method helps generate the matrix $\mathbf{\Sigma}$ [86]. Suppose we are in the k^{th} sweep with $\mathbf{H}_k$, then by taking the k^{th} left-side and right-side Jacobi rotation we have

$$\mathbf{H}_{k+1} = \mathbf{J}_k^{L\dagger}\mathbf{H}_k\mathbf{J}_k^{R} \tag{4.10}$$

which finally results in:

$$\mathbf{\Sigma} = \mathbf{H}_n; \quad \mathbf{U} = \prod_{k=0}^{n-1} \mathbf{J}_i^L; \quad \mathbf{V} = \prod_{k=0}^{n-1} \mathbf{J}_i^R \tag{4.11}$$

The Jacobi rotation matrices $\mathbf{J}^L$ and $\mathbf{J}^R$ are generally in the form

$$J_{ij} = \begin{cases} J_{pp} = cos\phi; \\ J_{pq} = sin\phi; \quad p < q \\ J_{qp} = -sin\phi; \quad p < q \\ J_{qq} = cos\phi; \\ \delta_{ij}; \quad elsewhere \end{cases} \tag{4.12}$$

The rotation angle ϕ is selected to diagonalize the 2×2 submatrix formed by the elements in channel matrix H at indices (p,p), (p,q), (q,p), (q,q):

$$\begin{bmatrix} cos\phi_L \; sin\phi_L \\ -sin\phi_L \; cos\phi_L \end{bmatrix}^{\dagger} \begin{bmatrix} H_{pp} \; H_{pq} \\ H_{qp} \; H_{qq} \end{bmatrix} \begin{bmatrix} cos\phi_R \; sin\phi_R \\ -sin\phi_R \; cos\phi_R \end{bmatrix} = \begin{bmatrix} H'_{pp} & 0 \\ 0 & H'_{qq} \end{bmatrix} \tag{4.13}$$

where ϕ_L and ϕ_R are the left and right rotation angles obtained by:

$$\phi_L + \phi_R = tan^{-1} \frac{H_{qp} + H_{pq}}{H_{qq} - H_{pp}} \quad \phi_L - \phi_R = tan^{-1} \frac{H_{qp} - H_{pq}}{H_{qq} + H_{pp}} \tag{4.14}$$

4.2.2 Complex SVD With Real Processing Elements

As discussed in subsection 4.2.1, a 2×2 SVD operation is performed in each iteration, which diagonalize a specific submatrix in $\mathbf{H}$. This 2×2 SVD unit is defined as a processing element (PE). In MIMO communication, the signals and channels are all in complex form, which leads to the fact that the SVD computations are expected to be compatible

with complex numbers. A SVD design with complex PEs, however, is obviously more complicated than that with real PEs. To reduce the hardware resource consumption, a complex SVD design with real PEs becomes an urgent need.

Consider a complex matrix $\mathbf{H} = \mathbf{H}_I + j\mathbf{H}_Q$, where $\mathbf{H}_I$ is the real part and $\mathbf{H}_Q$ is the imaginary part of $\mathbf{H}$, respectively. Suppose we have SVD decomposition of $\mathbf{H}$ that $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\dagger$, the singular value matrix $\mathbf{\Sigma}$ must be a real matrix. Then we have

$$\begin{aligned} \mathbf{H} &= \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\dagger = (\mathbf{U}_I + j\mathbf{U}_Q)\mathbf{\Sigma}(\mathbf{V}_I + j\mathbf{V}_Q)^\dagger \\ &= (\mathbf{U}_I\mathbf{\Sigma}\mathbf{V}_I^T + \mathbf{U}_Q\mathbf{\Sigma}\mathbf{V}_Q^T) + j(\mathbf{U}_Q\mathbf{\Sigma}\mathbf{V}_I^T - \mathbf{U}_I\mathbf{\Sigma}\mathbf{V}_Q^T) \end{aligned} \tag{4.15}$$

Taking (4.15) into an extended real matrix $\mathbf{H}_e = \begin{bmatrix} \mathbf{H}_I & -\mathbf{H}_Q \\ \mathbf{H}_Q & \mathbf{H}_I \end{bmatrix}$ with both real and imaginary parts of $\mathbf{H}$, we have

$$\mathbf{H}_e = \begin{bmatrix} \mathbf{U}_I & -\mathbf{U}_Q \\ \mathbf{U}_Q & \mathbf{U}_I \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma} & \\ & \mathbf{\Sigma} \end{bmatrix} \begin{bmatrix} \mathbf{V}_I & \mathbf{V}_Q \\ -\mathbf{V}_Q & \mathbf{V}_I \end{bmatrix}^T \tag{4.16}$$

Therefore, the SVD design with real PEs can also perform complex SVD calculation with help of the extended real matrix in (4.16). Note that if the singular value in $\mathbf{\Sigma}$ is large-to-small ordered, column permutation is necessary so that $\mathbf{U}_I$, $\mathbf{U}_Q$, $\mathbf{V}_I$ and $\mathbf{V}_Q$ can be ordered correctly.

4.3 Design and Implementation

This section provides an overview of structure of the hardware-efficient complex SVD implementation. To avoid heavy resource utilization, the system is designed in serial

structure and PEs are reused for computational operations. Meanwhile, only two layers are constructed in order to simplify the whole structure. The top layer monitors and controls the system with a finite state machine (FSM) and updates the matrices in every iteration. The bottom layer is simply a 2×2 SVD module that only processes real matrices. Figure 4.8 shows the generic structure of the our design. The coordinate rotation digital computer (CORDIC) based 2×2 SVD module within the red bounding box is the bottom layer and the rest in the green bounding box is on the top layer.

4.3.1 Top Level Design

The top level has two different functions: controlling and updating. The controller is a FSM that monitors the current indices and decide what to do next. The updating module is responsible for taking rotations to the matrices so that H, U and V can be updated properly.

4.3.1.1 Controlling

The system is running under the control of a FSM. In each iteration, the FSM tracks the row and column indices of the extracted entries from extended channel matrix $\mathbf{H}_e$. For instance, H_{pp}, H_{pq}, H_{qp} and H_{qq} are extracted when the tracked indices are p and q. With these tracked indices, the FSM helps determine where to start, what to do in the next iteration and when to stop. The diagram of FSM is shown on the left hand side in Figure 4.9 and the sequence flow of tracked indices is shown on the right hand side.

4.3.1.2 Updating

After the 2×2 SVD calculation, matrices are expected to be updated. Instead of doing matrix multiplications which require fixed matrix size as well as a larger amount of resources, only p^{th} and q^{th} rows and columns are involved in matrix update. Figure 4.10

shows the flow chart of update process of singular value matrix which finally leads to $\mathbf{\Sigma}$ after the convergence requirement is met. Note that $off(\mathbf{H}) = \sum_{p \neq q} H_{pq}$ is the off-diagonal norm of the matrix, and ϵ is the threshold for convergence. The update processes of left-side and right-side Jacobi rotation matrices only hold the process inside left and right bounding box in Figure 4.10, respectively.

4.3.2 Bottom Level Design

The bottom level is a 2×2 SVD module for real elements. The trigonometric functions operated in (4.13) and (4.14) are accomplished using CORDIC provided by the design tool (Vivado Design Suite from Xilinx Inc.). Diagram of the bottom level is shown in Figure 4.11. In each iteration, the FSM controller select proper elements to form a 2×2 submatrix $\mathbf{A}$ and send it to the bottom level. After the rotation matrices ($\mathbf{R}_L$ and $\mathbf{R}_R$) are calculate, the input matrix $\mathbf{A}$ can be diagonalized by $\mathbf{S} = \mathbf{R}_L \mathbf{A} \mathbf{R}_R$.

4.4 Maximum likelihood based CFO estimation and its hardware efficient architecture

In this section, we propose an asymptotically optimal algorithm that can accurately estimate the carrier frequency offset as well as an efficient hardware implementation for proposed algorithm. More practically, since the proposed algorithm assumes one sample per symbol where timing synchronization is a prerequisite, we consider the case that the timing synchronization is completed so that our algorithm can be directly applied without further treatment. The advantages of the proposed approach are as follows:

1. We propose a maximum likelihood based frequency offset estimation algorithm for MIMO system, with an asymptotically optimal closed form solution to the estima-

tor, giving the result that the MSE of proposed method approaches corresponding lower bound as SNR increases. The proposed method has large acquisition range up to 25% of bandwidth.

2. We present an efficient hardware architecture for our ML based CFO estimation algorithm. By avoiding matrix inversion and searching for maximum which are common in the existing solutions to CFO estimation, hardware implementation of our algorithm is efficient and therefore can be easily fitted in a small FPGA.

3. The proposed hardware architecture is pipelined so that the processing delay is minimal. Meanwhile, the architecture is reconfigurable to different pilot lengths, which makes it possible to accommodate various frame structures.

4.4.1 System Model

The notations that are used in Section 4.4 for system model description and algorithm derivation are summarized in Table 4.1. Besides, The operators in this section are as follows: $[\vec{a}_x]_x$ denotes a tall vector whose x-th row of vector is $\vec{a}_x$, $[\boldsymbol{A}]_{x_1,x_2}$ denotes a block matrix whose block at x_1-th row and x_2-th column is $\boldsymbol{A}$, $\Re[\cdot]$ and $\Im[\cdot]$ stand for real and imaginary part, $\mathbb{E}[\cdot]$, $(\cdot)^\dagger$ and $\mathrm{Tr}(\cdot)$ denote expectation, conjugate transpose and trace, respectively.

Table 4.1: Notation List

Parameter	Notation
Pilot Length	n
# of Transmit Antennas	l_t
# of Receiver Antennas	l_r
Sample Period	t_b
Symbol Energy	ρ
Channel Variance	$\sigma^2_{\boldsymbol{h}}$
Noise Variance	$\sigma^2_{\boldsymbol{n}}$
Identity Matrix	$\boldsymbol{I}$
Fisher Information Matrix	$\boldsymbol{\mathcal{I}}$
CFO Estimate	$\hat{f}_\delta$
Symbol Energy	E_s
Kronecker Delta	$\delta[r_1 - r_2]$

To investigate the frequency synchronization of a MIMO systems, this section considers the system model as follows. Suppose the transmitter has l_{t} antennas, the receiver has l_{r} antennas and the MIMO system is located where the channel has flat-fading or slow fading that the channel state can be assumed to be constant during the transmission of pilot [87]. Let $\boldsymbol{h}_{r,t}$ be the channel coefficient from the t-th transmit antenna to the r-th receive antenna and define $\vec{\boldsymbol{h}}_r = [\boldsymbol{h}_{r,t}]_{t=1:l_{\mathrm{t}}} \in \mathbb{C}^{l_{\mathrm{t}}\times 1}$ as the vector of channel states at the r-th receive antenna. The channel state matrix $\boldsymbol{H} = \left[\vec{\boldsymbol{h}}_r^T\right]_{r=1:l_{\mathrm{r}}} \in \mathbb{C}^{l_{\mathrm{r}}\times l_{\mathrm{t}}}$ comprises circularly symmetric complex Gaussian random variables with joint distribution $\mathcal{CN}\left(\vec{0}, \left[\mathbb{E}\left[\boldsymbol{h}_{r_1}\boldsymbol{h}_{r_2}^\dagger\right]\right]_{r_1=1:l_{\mathrm{r}},r_2=1:l_{\mathrm{r}}}\right)$. Define $\boldsymbol{N} = \left[\vec{\boldsymbol{n}}_r^T\right]_{r=1:l_{\mathrm{r}}} \in \mathbb{C}^{l_{\mathrm{r}}\times n}$ as the noise at receiver with distribution $\mathcal{CN}\left(\vec{0}, \left[\mathbb{E}\left[\vec{\boldsymbol{n}}_{r_1}\vec{\boldsymbol{n}}_{r_2}^\dagger\right]\right]_{r_1,r_2\in\{1,2,\ldots,l_{\mathrm{r}}\}}\right)$; $\boldsymbol{H}$ and $\boldsymbol{N}$ are independent. We send length n pilot signals $\boldsymbol{S}^T = \left[\vec{\boldsymbol{s}}_t^T\right]_{t=1:l_{\mathrm{t}}} \in \mathbb{C}^{l_{\mathrm{t}}\times n}$, through the channel, with sample period t_b. The pilot signals, known to the receivers, for different transmit antennas are orthogonal, i.e., $\vec{\boldsymbol{s}}_{t_1} \perp \vec{\boldsymbol{s}}_{t_2},\ \forall t_1 \neq t_2$, $\boldsymbol{S}^\dagger\boldsymbol{S} = \frac{n\rho}{l_{\mathrm{t}}}\boldsymbol{I}_{l_{\mathrm{t}}\times l_{\mathrm{t}}}$, where ρ is the symbol energy and $\boldsymbol{S}$'s energy satisfies $\mathrm{Tr}\left(\boldsymbol{S}^\dagger\boldsymbol{S}\right) = \sum_{i=1}^{l_{\mathrm{t}}}\sum_{k=1}^{n}|s_{i,k}|^2 = \rho n > 0$. At the receiver, assuming perfect symbol timing recovery is applied so that with a frequency offset $f_\delta$and

additive white Gaussian noise (AWGN) with double sided power spectral density $2N_0$, the received signal can be written as:

$$\begin{aligned} \boldsymbol{Y}^T &= \boldsymbol{H}\boldsymbol{S}^T\boldsymbol{F} + \boldsymbol{N}^T \\ &= \left[\vec{\boldsymbol{h}}_r^T\right]_r \left[\vec{\boldsymbol{s}}_t^T\right]_t \boldsymbol{F} + \left[\vec{\boldsymbol{n}}_r^T\right]_r \end{aligned} \tag{4.17}$$

where the frequency offset matrix

$$\boldsymbol{F} = \begin{bmatrix} e^{j2\pi f_\delta t_b \cdot 0} & 0 & \cdots & 0 \\ 0 & e^{j2\pi f_\delta t_b \cdot 1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & e^{j2\pi f_\delta t_b (n-1)} \end{bmatrix} \tag{4.18}$$

is a diagonal matrix, which rotates the pilot signal according to the frequency offset f_δ.

We consider the simplest case that channels are independent where $\mathbb{E}\left[\vec{\boldsymbol{h}}_{r_1}\vec{\boldsymbol{h}}_{r_2}^\dagger\right] = \sigma_{\boldsymbol{h}}^2\boldsymbol{I}\delta[r_1 - r_2]$ and i.i.d. noise $\mathbb{E}\left[\vec{\boldsymbol{n}}_{r_1}\vec{\boldsymbol{n}}_{r_2}^\dagger\right] = \sigma_{\boldsymbol{n}}^2\boldsymbol{I}\delta[r_1 - r_2]$. It is the case of choice when we have little knowledge of the channel. To write the model in the familiar linear transformation form, we define tall vectors $\vec{\boldsymbol{y}} = \left[\vec{\boldsymbol{y}}_r\right]_r$, $\vec{\boldsymbol{h}} = \left[\vec{\boldsymbol{h}}_r\right]_r$, $\vec{\boldsymbol{n}} = \left[\vec{\boldsymbol{n}}_r\right]_r$, and block matrix

$$\begin{aligned} \ddot{\boldsymbol{X}} &= \begin{bmatrix} \boldsymbol{X} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \ddots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{X} \end{bmatrix} = [\boldsymbol{X}]_{r,r=1:l_{\mathrm{r}}} \\ &= \boldsymbol{I}_{l_{\mathrm{r}}} \otimes \boldsymbol{X} \end{aligned} \tag{4.19}$$

where $\boldsymbol{I}_{l_{\mathrm{r}}}$ is a $l_{\mathrm{r}} \times l_{\mathrm{r}}$ identity matrix and $\otimes$ is the Kronecker product. With these notation,

the received signal can be written as

$$\vec{\boldsymbol{y}} = \ddot{\boldsymbol{X}}\vec{\boldsymbol{h}} + \vec{\boldsymbol{n}}. \tag{4.20}$$

For r-th receive antenna, the received signal can be written as:

$$\begin{aligned} \boldsymbol{y}_r(k) &= \boldsymbol{x}_r(k) + \boldsymbol{n}_r(k) \\ &= \sum_k \sum_t \boldsymbol{h}_{r,t} s_{t,k} e^{j2\pi f_\delta k t_b} + \boldsymbol{n}_r \end{aligned} \tag{4.21}$$

4.4.2 Proposed CFO Estimation Algorithm

To perform ML estimation of frequency offset, we need to obtain the estimate from

$$\hat{f}_\delta = \arg\max_{f_\delta} f_{\vec{\boldsymbol{y}}|f_\delta}(\vec{\boldsymbol{y}}|f_\delta) \tag{4.22}$$

We observe that $\vec{\boldsymbol{y}}$ is a summation of Gaussian random variables and has distribution $\mathcal{CN}\left(\vec{0}, \Sigma_{\vec{\boldsymbol{y}}}(f_\delta)\right)$, where the covariance matrix of $\vec{\boldsymbol{y}}$ can be written as

$$\begin{aligned} \boldsymbol{\Sigma}_{\vec{\boldsymbol{y}}}(f_\delta) &= \sigma^2_{\boldsymbol{h}} \ddot{\boldsymbol{X}} \ddot{\boldsymbol{X}}^\dagger + \sigma^2_{\boldsymbol{n}} \boldsymbol{I} \\ &= \sigma^2_{\boldsymbol{h}} \boldsymbol{I}_{l_\mathrm{r}} \otimes (\boldsymbol{F}\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}^\dagger) + \sigma^2_{\boldsymbol{n}} \boldsymbol{I} \end{aligned} \tag{4.23}$$

Therefore,

$$\begin{aligned}
\hat{f}_\delta &= \arg\max_{f_\delta} f_{\vec{\boldsymbol{y}}|\boldsymbol{f}_\delta}(\vec{\boldsymbol{y}}|f_\delta) \\
&\propto \arg\max_{f_\delta} e^{-\vec{\boldsymbol{y}}^\dagger \boldsymbol{\Sigma}_{\vec{\boldsymbol{y}}}(f_\delta)^{-1}\vec{\boldsymbol{y}}} \\
&= \arg\min_{f_\delta} \vec{\boldsymbol{y}}^\dagger \boldsymbol{\Sigma}_{\vec{\boldsymbol{y}}}(f_\delta)^{-1}\vec{\boldsymbol{y}} \\
&= \arg\min_{f_\delta} \sum_r \vec{\boldsymbol{y}}_r^\dagger \frac{1}{\sigma_{\boldsymbol{n}}^2}\left(\frac{\sigma_{\boldsymbol{h}}^2}{\sigma_{\boldsymbol{n}}^2}\boldsymbol{F}\boldsymbol{S}\boldsymbol{S}^\dagger\boldsymbol{F}^\dagger + \boldsymbol{I}\right)^{-1}\vec{\boldsymbol{y}}_r
\end{aligned} \tag{4.24}$$

Apply Woodbury identity [88] to (4.24) and bear in mind that $\boldsymbol{F}^\dagger\boldsymbol{F} = \boldsymbol{I}$, $\boldsymbol{S}^\dagger\boldsymbol{S} = \frac{n\rho}{l_{\mathrm{t}}}\boldsymbol{I}$,we can obtain

$$\begin{aligned}
\hat{f}_\delta &= \arg\min_{f_\delta} \sum_r \vec{\boldsymbol{y}}_r^\dagger \\
&= \left(\boldsymbol{I} - \boldsymbol{F}\boldsymbol{S}\left(\boldsymbol{S}^\dagger\boldsymbol{F}^\dagger\boldsymbol{F}\boldsymbol{S} + \frac{\sigma_{\boldsymbol{n}}^2}{\sigma_{\boldsymbol{h}}^2}\boldsymbol{I}\right)^{-1}\boldsymbol{S}^\dagger\boldsymbol{F}^\dagger\right)\vec{\boldsymbol{y}}_r \\
&= \arg\max_{f_\delta} \sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}\boldsymbol{S}\left(\frac{n\rho}{l_{\mathrm{t}}}\boldsymbol{I} + \frac{\sigma_{\boldsymbol{n}}^2}{\sigma_{\boldsymbol{h}}^2}\boldsymbol{I}\right)^{-1}\boldsymbol{S}^\dagger\boldsymbol{F}^\dagger\vec{\boldsymbol{y}}_r \\
&= \arg\max_{f_\delta} \underbrace{\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}\boldsymbol{S}\boldsymbol{S}^\dagger\boldsymbol{F}^\dagger\vec{\boldsymbol{y}}_r}_{g(\vec{\boldsymbol{y}},f_\delta)}
\end{aligned} \tag{4.25}$$

In general, (4.25) is not explicitly solvable. Exhaustive search over the candidates may produce the estimate, but it requires extensive resources for hardware implementation, as well as has potentially large quantization errors. A common way to solve this problem is to find an explicit solution to the first derivative of the estimator.

First, we need to construct the pilot sequence. In this section, we employ periodic structure:

$$\boldsymbol{S} \;=\; \sqrt{\rho}\underbrace{[c_i]_{i,i=1:n}}_{\boldsymbol{C}}\underbrace{[\boldsymbol{I}_{l_t}]_{i=1:m}\,\boldsymbol{O}}_{[\boldsymbol{O}]_{i=1:m}} \tag{4.26}$$

has a structure of scrambled periodic matrix $[\boldsymbol{O}]_{i=1:m} = \begin{bmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \vdots \end{bmatrix} \in \mathbb{C}^{n\times l_t}$, which is a block matrix with m copies of an unitary matrix $\boldsymbol{O} \in \mathbb{C}^{l_t\times l_t}$ on top of each other. Matrix $\boldsymbol{O}$ satisfies $\boldsymbol{O}^\dagger\boldsymbol{O} = \boldsymbol{O}\boldsymbol{O}^\dagger = \boldsymbol{I}_{l_t}$. We assume the pilot length $n = ml_t,\ m \in \mathbb{Z}^+$. The scrambling code is $\vec{\boldsymbol{c}} = [c_i]_{i=1:n} \in \mathbb{C}^{n\times 1}$, where $|c_i| = 1,\ \forall i$. Diagonal matrix $\boldsymbol{C}$'s diagonal elements are from the scrambling code $\vec{\boldsymbol{c}}$.

Due to the considerations that searching algorithms require potentially large quantization errors (exhaustive search) or indefinite convergence time (iterative search), we can attempt to solve the first derivative of $g(\vec{\boldsymbol{y}}, f_\delta)$ to avoid these drawbacks.

To find $\frac{\partial g(\vec{\boldsymbol{y}},f_\delta)}{\partial f_\delta} = 0$, we calculate the differential first:

$$\begin{aligned}
& g(\vec{\boldsymbol{y}}, f_\delta + df_\delta) - g(\vec{\boldsymbol{y}}, f_\delta) \\
= \;& \sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta + df_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta + df_\delta)^\dagger \vec{\boldsymbol{y}}_r \\
& - \sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta)^\dagger \vec{\boldsymbol{y}}_r \\
= \;& 2\Re\left[\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \left(\boldsymbol{F}(f_\delta)\left(j2\pi df_\delta t_b \boldsymbol{J}\right)\right)^\dagger \vec{\boldsymbol{y}}_r\right] \\
& + o(df_\delta) \\
= \;& 4\pi t_b \Im\left[\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta)^\dagger \boldsymbol{J}\vec{\boldsymbol{y}}_r\right] df_\delta \\
& + o(df_\delta)
\end{aligned} \tag{4.27}$$

where we have used

$$\begin{aligned}\boldsymbol{F}(f_\delta + df_\delta) &= \boldsymbol{F}(f_\delta)\boldsymbol{F}(df_\delta) \\ &= \boldsymbol{F}(f_\delta)\left(\boldsymbol{I} + j2\pi df_\delta t_b \boldsymbol{J} + o(df_\delta)\right) \end{aligned} \tag{4.28}$$

$$\begin{aligned}\boldsymbol{F}(df_\delta) &= \left[1 + j2\pi df_\delta t_b(i-1) + o(df_\delta)\right]_{i,i=1:n} \\ &\doteq \boldsymbol{I} + j2\pi df_\delta t_b \boldsymbol{J} \end{aligned} \tag{4.29}$$

and

$$\boldsymbol{J} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & (n-1) \end{bmatrix} \tag{4.30}$$

From the differential, the first order derivative can be expressed as

$$\frac{\partial g(\vec{y}, f_\delta)}{\partial f_\delta} = 4\pi t_b \Im\left[\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta)^\dagger \boldsymbol{J}\vec{\boldsymbol{y}}_r\right] \tag{4.31}$$

and therefore to find the frequency offset, we solve

$$0 = \Im\left[\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(\hat{f}_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(\hat{f}_\delta)^\dagger \boldsymbol{J}\vec{\boldsymbol{y}}_r\right] \tag{4.32}$$

It is obvious that directly solve (4.32) requires matrix operations which is not easy. Therefore, we transform these matrix operations into scalar form. Define $d \triangleq e^{j2\pi\hat{f}_\delta t_b l_\mathfrak{t}}$, $\left[d^{k-1}\right]_{1,k=1:m} = \begin{bmatrix} d^0 & \cdots & d^{m-1} \end{bmatrix}$, and let $\vec{\boldsymbol{y}}_{c,r} = \boldsymbol{C}^\dagger \vec{\boldsymbol{y}}_r$ to be the unscrambled received signal which is partitioned into length $l_\mathfrak{t}$ vectors $\vec{\boldsymbol{y}}_{c,r}(k) \in \mathbb{C}^{l_\mathfrak{t}\times 1}$such that $\vec{\boldsymbol{y}}_{c,r} = \left[\vec{\boldsymbol{y}}_{c,r}(k)\right]_{k=1:m}$. Define

the sample correlation $\beta_{k,i} \triangleq \sum_{r=1}^{l_\mathrm{r}} < \vec{y}_{c,r}(i), \vec{y}_{c,r}(k) >$ with the property $\beta_{k,i} = \beta_{i,k}^*$. Also bearing in mind that for a matrix $\boldsymbol{A}$, $\Im\left[\boldsymbol{A}\right] = 0$ is equivalent to $\boldsymbol{A} - \boldsymbol{A}^\dagger = 0$. Then (4.32) is equivalent to

$$\begin{aligned} 0 &= \left[\left(\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta)^\dagger \boldsymbol{J}\vec{\boldsymbol{y}}_r\right)\right. \\ &\quad \left. - \left(\sum_r \vec{\boldsymbol{y}}_r^\dagger \boldsymbol{F}(f_\delta)\boldsymbol{S}\boldsymbol{S}^\dagger \boldsymbol{F}(f_\delta)^\dagger \boldsymbol{J}\vec{\boldsymbol{y}}_r\right)^\dagger\right] \\ &= \rho \sum_r \left[d^{k-1}\right]_{1,k=1:m} \\ &\quad \left[\vec{\boldsymbol{y}}_{c,r}(k_1)^\dagger \left(k_2 - k_1\right) \boldsymbol{I}_{l_\mathrm{t}} \vec{\boldsymbol{y}}_{c,r}(k_2)\right]_{k_1=1:m, k_2=1:m} \\ &\quad \left[\left(d^\dagger\right)^{k-1}\right]_{k=1:m} \\ &= -\rho l_\mathrm{t} \sum_{k_1=1}^{m}\sum_{k_2=1}^{m} d^{k_1-k_2}(k_1-k_2)\beta_{k_1,k_2} \end{aligned} \tag{4.33}$$

Take a further step simplification of (4.33), let $r_i e^{-j\theta_i} = \sum_{k=i+1}^{m} \beta_{k,k-i}$, we have

$$\begin{aligned} 0 &= -2j\rho l_\mathrm{t} \Im\left[\sum_{k=2}^{m}\sum_{i=1}^{k-1} i \cdot d^i \beta_{k,k-i}\right] \\ &\quad -2j\rho l_\mathrm{t} \Im\left[\sum_{i=1}^{m-1} e^{ji2\pi \hat{f}_\delta t_b l_\mathrm{t}} \left(i \cdot r_i e^{-j\theta_i}\right)\right] \\ &\quad -2j\rho l_\mathrm{t} \left[\sum_{i=1}^{m-1} i \cdot r_i \sin\left(i2\pi \hat{f}_\delta t_b l_\mathrm{t} - \theta_i\right)\right] \end{aligned} \tag{4.34}$$

Consequently, if we denote $\alpha = 2\pi t_b l_t \hat{f}_\delta$, the solution to (4.32) is

$$0 = \sum_{i=1}^{m-1} i \cdot r_i \sin\left(i\alpha - \theta_i\right) \tag{4.35}$$

For high SNR, $i\alpha - \theta_i$ approaches 0 for all possible i, under which condition $sin(x) \approx x$ can be applied to (4.35) obtain (4.36)

$$0 \quad = \quad \sum_{i=1}^{m-1} i \cdot r_i (i 2\pi \hat{f}_\delta t_b l_\mathrm{t} - \theta_i) \tag{4.36}$$

and we can calculate the estimate of frequency offset easily from (4.36).

However the approximation $i\alpha - \theta_i \approx 0$ may not be suitable for lower SNR case. We consider the Taylor series $sin(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} x^{2k+1}$ which stands under all conditions, (4.36) may be expanded as

$$0 \quad = \quad \sum_{i=1}^{m-1} i \cdot r_i (\sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} (i\alpha - \theta_i)^{2k+1}) \tag{4.37}$$

Since there is no general solution to a quintic or higher order equation, a third order Taylor series approximation is applied to (4.36) and we obtain (4.39)

$$0 \quad = \quad \sum_{i=1}^{m-1} i \cdot r_i (\sum_{k=0}^{1} \frac{(-1)^k}{(2k+1)!} (i\alpha - \theta_i)^{2k+1}) \tag{4.38}$$

and therefore we have

$$\begin{aligned} \alpha^3 \sum_{i=1}^{m-1} \frac{-i^4 r_i}{6} + \alpha^2 \sum_{i=1}^{m-1} \frac{i^3 r_i \theta_i}{2} + \\ \alpha \sum_{i=1}^{m-1} i^2 r_i (1 - \frac{\theta_i^2}{2}) + \sum_{i=1}^{m-1} i r_i \left(\frac{\theta_i^3}{6} - \theta_i \right) \quad = \quad 0 \end{aligned} \tag{4.39}$$

Define $a = \sum_{i=1}^{m-1} \frac{-i^4 r_i}{6}$, $b = \sum_{i=1}^{m-1} \frac{i^3 r_i \theta_i}{2}$, $c = \sum_{i=1}^{m-1} i^2 r_i (1 - \frac{\theta_i^2}{2})$ and $d = \sum_{i=1}^{m-1} i r_i \left(\frac{\theta_i^3}{6} - \theta_i \right)$, (4.39) now is

$$a\alpha^3 + b\alpha^2 + c\alpha + d \quad = \quad 0 \tag{4.40}$$

Algorithm 2: ML based CFO estimation algorithm for MIMO system

1. **Input:** $\vec{\boldsymbol{y}}_r \in \mathbb{C}^{n\times 1}$, $r = 1, ..., l_{\text{r}}$, one sample per symbol
2. Unscramble $\vec{\boldsymbol{y}}_r$ and perform a partition so that $\vec{y}_{c,r} = C^{\dagger}\vec{\boldsymbol{y}}_r = \begin{bmatrix} \vec{\boldsymbol{y}}_{c,r}(1) \\ \vdots \\ \vec{\boldsymbol{y}}_{c,r}(m) \end{bmatrix}$, $\vec{\boldsymbol{y}}_{c,r}(k) \in \mathbb{C}^{l_{\text{t}}\times 1}$, $r = 1, ..., l_{\text{r}}$, $ml_{\text{t}} = n$
3. **For** $i = 1, ..., m-1$
   (a) $\beta_{\text{sum}} = 0$
   (b) **For** $k = i+1, ..., m$
      i. $\beta_{k,k-i} = \sum_{r=1}^{l_{\text{r}}} < \vec{\boldsymbol{y}}_{c,r}(k-i), \vec{\boldsymbol{y}}_{c,r}(k) >$
      ii. $\beta_{\text{sum}} = \beta_{\text{sum}} + \beta_{k,k-i}$
   (c) $r_i = |\beta_{sum}|\,,\ \theta_i = \angle\beta_{sum}$
4. $\theta_i = \text{unwrap}\,(\theta_i)\,,\ i = 1, ..., m-1$
5. $\alpha = \frac{\sum_{i=1}^{m-1} i^3 \cdot r_i\theta_i}{\sum_{i=1}^{m-1} i^4 r_i}$
6. $\hat{f}_\delta = \frac{\alpha}{2\pi t_b l_{\text{t}}}$
7. **Output:** $\hat{f}_\delta$.

Since the frequency offset is a real number and there should exist only one solution, the real valued solution to (4.40) is $\alpha = -\frac{b}{3a}$, plug in $\alpha = 2\pi t_b l_t \hat{f}_\delta$, we have

$$\hat{f}_\delta \doteq \frac{1}{2\pi t_b l_t} \frac{-\sum_{i=1}^{m-1} \frac{i^3 r_i\theta_i}{2}}{3\sum_{i=1}^{m-1} \frac{-i^4 r_i}{6}} = \frac{1}{2\pi t_b l_t} \frac{\sum_{i=1}^{m-1} i^3 r_i\theta_i}{\sum_{i=1}^{m-1} i^4 r_i} \tag{4.41}$$

The entire procedure of proposed method is shown in Algorithm 2.

4.4.3 CRLB: Performance Guideline

The CRLB of a random variable λ is

$$\sigma_{\boldsymbol{\lambda}}^2 \geqslant \frac{1}{\boldsymbol{\mathcal{I}}_{\vec{\boldsymbol{y}}(k)|\lambda}} \tag{4.42}$$

where $\vec{\boldsymbol{y}}(k)$ is the received signal and $\boldsymbol{\mathcal{I}}_{\vec{\boldsymbol{y}}(k)|\lambda}$ is the Fisher information matrix with λ to be estimated. $\boldsymbol{\mathcal{I}}_{\vec{\boldsymbol{y}}(k)|\lambda} = \mathbb{E}\left[(\frac{\partial f}{\partial \lambda})^2|\lambda\right] = \mathbb{E}\left[\frac{\partial^2 f}{\partial \lambda^2}|\lambda\right]$ where f is the log likelihood function. Consider a channel with circularly symmetric complex Gaussian random noise, the received signal observed during an interval nt_b can be expressed as

$$\vec{\boldsymbol{y}}(k) = \vec{\boldsymbol{x}}(k) + \vec{\boldsymbol{n}}(k) \tag{4.43}$$

In (4.43), $\vec{\boldsymbol{x}}(k)$ is the information-bearing signal with frequency offset and $\vec{\boldsymbol{n}}(k)$ is the additive white Gaussian noise with double sided power spectral density $2N_0$. Then the Fisher information matrix can be written as

$$\boldsymbol{\mathcal{I}}_{\vec{\boldsymbol{y}}(k)|\boldsymbol{\lambda}} = -\mathbb{E}\left[\frac{2}{N_0}\sum_k (\vec{\boldsymbol{y}}(k) - \vec{\boldsymbol{x}}(k))\frac{\partial^2 \vec{\boldsymbol{x}}}{\partial \lambda^2} - (\frac{\partial \vec{\boldsymbol{x}}}{\partial \lambda})^2\right] \tag{4.44}$$

Consider that the noise is AWGN, we have$\mathbb{E}\left[\vec{\boldsymbol{y}}(k) - \overrightarrow{x}(k)\right] = -\mathbb{E}\left[\vec{\boldsymbol{n}}(k)\right] = 0$, therefore the Fisher information matrix in (4.44) is

$$\boldsymbol{\mathcal{I}}_{\vec{\boldsymbol{y}}(k)|\boldsymbol{\lambda}} = \frac{2}{N_0}\mathbb{E}\left[\sum_k (\frac{\partial \vec{\boldsymbol{x}}}{\partial \lambda})^2\right] \tag{4.45}$$

Since we are seeking the CRLB of frequency offset and consider a MIMO case in which an user has l_t transmitter antennas and l_r receiver antennas, let λ to be the frequency offset

then we have [89]

$$\sigma_{f_\delta}^2 \geqslant \frac{N_0}{2\sum_{r=1}^{l_r}\mathbb{E}\left[\sum_k \left|\frac{\partial \vec{\boldsymbol{x}_r}}{\partial f_\delta}\right|^2\right]} \tag{4.46}$$

Under MIMO case, suppose the observation and estimation process starts at t_0, the received signal at the r-th receiver antenna with frequency offset is (as also shown in 4.21, Section 4.4.1)

$$\begin{aligned} \boldsymbol{y}_r(k) &= \boldsymbol{x}_r(k) + \boldsymbol{n}_r(k) \\ &= \sum_k \sum_{t=1}^{l_t} \boldsymbol{h}_{r,t,k} s_{t,k} e^{j(2\pi f_\delta k t_b - t_0))} + \boldsymbol{n}_r \end{aligned} \tag{4.47}$$

Assume that the channel on a transmission link is invariant during the observation time nt_b and has a Gaussian distribution with variance $\sigma_{\boldsymbol{h}_{r,t}}^2$, we have

$$\boldsymbol{x}_r(k) = \sum_t \boldsymbol{h}_{r,t} \sum_k s_{t,k} e^{j(2\pi f_\delta (k t_b - t_0))} \tag{4.48}$$

Using (4.48) it is found that

$$\sum_k \left|\frac{\partial \vec{\boldsymbol{x}}}{\partial f_\delta}\right|^2 = 4\pi^2 \sum_s \boldsymbol{h}_{r,t} \sum_k s_{t,k} (k t_b - t_0)^2 \tag{4.49}$$

Since the channel is independent from pilot symbols, we have

$$\mathbb{E}\left[\sum_k \left|\frac{\partial \vec{\boldsymbol{x}}}{\partial f_\delta}\right|^2\right] = \mathbb{E}\left[\boldsymbol{h}_{r,t}^2\right] \mathbb{E}\left[(\sum_k s_{t,k}(k t_b - t_0)^2)^2\right] \tag{4.50}$$

Suppose that the pilot for fine frequency estimation has n symbols and is uniformly distributed on the constellation points, also according to Section 4.4.1 and (4.50), we can

easily know that

$$\mathbb{E}\left[\boldsymbol{h}_{r,t}^2\right] = \sigma_{\boldsymbol{h}_{r,t}}^2 \tag{4.51}$$

$$\mathbb{E}\left[(\sum_k s_{t,k}(kt_b - t_0)^2)^2\right] = E_s\mathbb{E}\left[\sum_k (kt_b - t_0)^2\right] \tag{4.52}$$

Note that the symbol is fixed where the expectation of its square is symbol energy. Substituting (4.52) and (4.51) into (4.49) yields

$$\begin{aligned}\mathbb{E}&\left[\sum_k \left|\frac{\partial \vec{\boldsymbol{x}_r}}{\partial f_\delta}\right|^2\right] \\ &= 4\pi^2 \textstyle\sum_t \sigma_{\boldsymbol{h}_{r,t}}^2 E_s \mathbb{E}\left[\sum_k (kt_b - t_0)^2\right]\end{aligned} \tag{4.53}$$

Since CRLB is a lower bound, we are interested in the maximum achievable value of the bound, which can be obtained by choosing t_0 as the midpoint of observation time where we have $t_0 = \frac{nt_b}{2}$. Then with the summation in (4.53), we have

$$\begin{aligned}\mathbb{E}&\left[\sum_k \left|\frac{\partial \vec{\boldsymbol{x}_r}}{\partial f_\delta}\right|^2\right] \\ &= 4\pi^2 \textstyle\sum_t \sigma_{\boldsymbol{h}_{r,t}}^2 E_s \frac{t_b^2(n^3+3n^2+2n)}{12}\end{aligned} \tag{4.54}$$

Plug (4.54) into (4.46), we eventually obtain

$$\begin{aligned}\sigma_{f_\delta}^2 &\geqslant \frac{N_0}{2\sum_r (4\pi^2 \sum_t \sigma_{\boldsymbol{h}_{r,t}}^2 E_s \frac{t_b^2(n^3+3n^2+2n)}{12})} \\ &= \frac{3}{2\pi^2(n^3+3n^2+2n) l_r t_b^2 \frac{\rho\sigma_{\boldsymbol{h}}^2}{\sigma_{\boldsymbol{n}}^2}}\end{aligned} \tag{4.55}$$

where $\sigma_{\boldsymbol{h}_{r,t}}^2 = \sigma_{\boldsymbol{h}}^2, \forall r, t$ and $\frac{l_t E_s}{N_0} = \frac{\rho\sigma_{\boldsymbol{h}}^2}{\sigma_{\boldsymbol{n}}^2}$(see also Section 4.4.1).

4.4.4 Architecture for Proposed Algorithm

Figure 4.12(a) shows the architecture of proposed method, and Figure 4.12(b) shows the architecture of method in [1] which is a commonly adopted architecture for searching algorithms. From the figures we can observe that for the architecture of method in [1], each potential CFO candidate requires an extra processing unit while a large number of candidates could consume significant resources. Furthermore, multiple antennas require multiple times of resources for a single antenna architecture. Therefore, it is hard to find the balance between hardware consumption and searching accuracy for the method in [1]. To limit the resource consumption, the proposed method avoids searching techniques so that the resource consumption is only related to the number of receivers. From 4.12(a) it is clear that when pilot length changes, the overall architecture keeps the same as long as the memories that store unscrambled received signal are not overflowed. In the proposed architecture, the pilot length is an coefficient for control logic that tells what to do at specific states. Therefore, the architecture is reconfigurable to different pilot lengths. To accommodate frames with various structures, frame lengths and pilot lengths, only the pilot length and corresponding valid signal are required for proposed architecture to reconfigure the FSM.

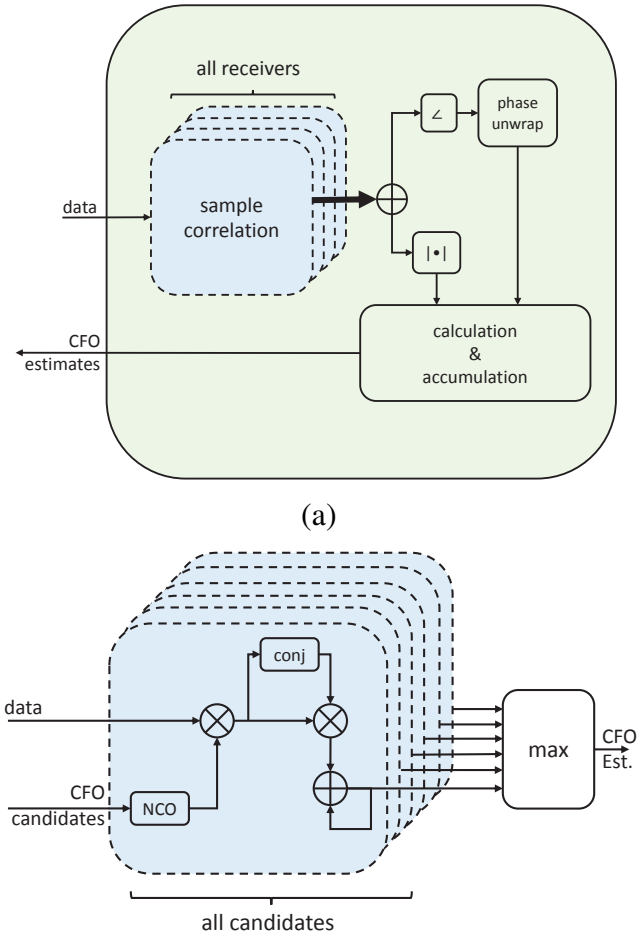

Figure 4.12: Architectures of ML CFO estimators. (a)Architecture of proposed method. (b)Architecture of approach in [1]

4.4.4.1 Memory based sample correlation

Take a glance at (4.33) and (4.41), we find that sample correlation β is an important step when approaching the final solution. Considering that each unscrambled received pilot symbol/sample may appear more than once during the process, we allocate a dual port block memory or two single port block memories for each antenna to store the received pilot. After receiving the pilot signal, multiplication with the conjugate pilot signal is

performed to acquire the unscrambled received signal $\vec{y}_{c,r}$ for each antenna, and then the unscrambled received signal is stored in the memory bank, waiting to be read for further operations. After all pilot symbols are received, a finite state machine (FSM) controlled accumulator works on preparing the sample correlation for next steps. Figure 4.13 shows the architecture of memory based sample correlation. The control signals consists of valid signal for incoming data and memory read and write address indices along with the valid signal. Once the incoming data is valid, it is unscrambled with pilot symbols and then stored in specific entries of the BRAMs. Once the unscrambled pilot signals are stored, the FSM controlled accumulator accumulates the sample correlation. The states transitions of FSM during the accumulation process is controlled by the indices from control logic.

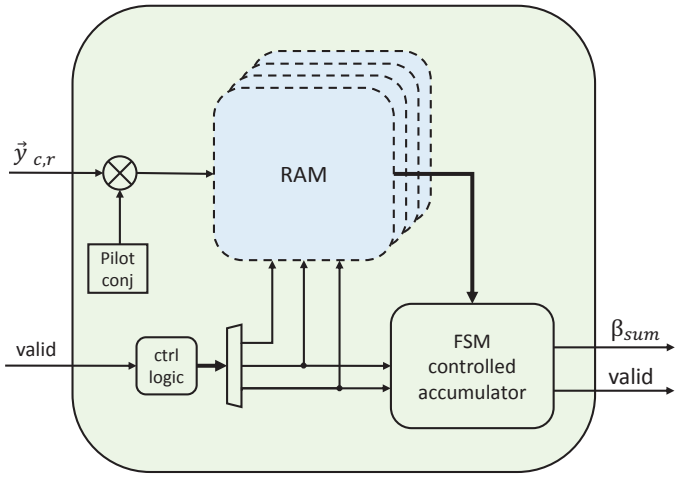

Figure 4.13: Architecture of memory based sample correlation

4.4.4.2 Sample based phase unwrapping

From Figure 4.12(a), it is easy to observe that the output of memory based sample correlation is then turned into magnitude and angle. The constantly increasing or decreasing

angle reflects the impact of frequency offset, but the complex-to-angle module output always stays within the range $(-\pi, +\pi)$. To solve this issue, we design a state controller that tracks the phase change and compensates for the phase increment. Every time the angles of two consecutive output cross π or $-\pi$, the coefficient for compensation increase or decrease by 1. Therefore, we can always know how much we need to compensate to the angle. After the phase unwrapping, parameters are delivered for CFO calculation, which contains only arithmetic operations, as shown exactly the same in (4.41). The architecture of implemented phase unwrapping is shown in Figure 4.14. Different from frame based phase unwrapping which takes a whole frame of symbols as input and unwraps the phase, sample based phase unwrapping is uses delay-and-compare to find out whether the phase exceeds detection range, performs phase unwrapping accordingly, and therefore has shorter processing latency.

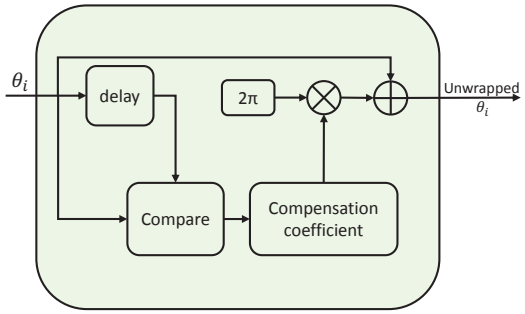

Figure 4.14: Architectures of phase unwrapping

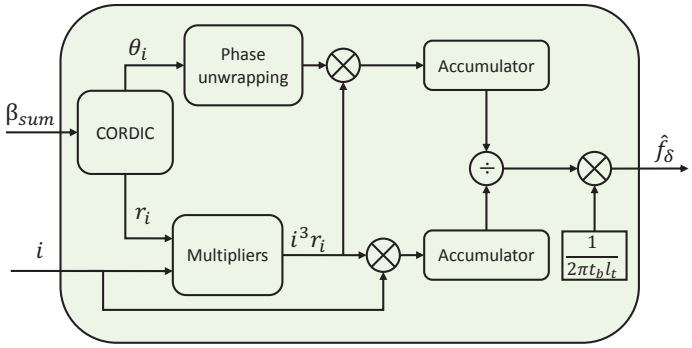

Figure 4.15: Architecture of CFO calculation

4.4.4.3 CFO calculation with accumulators

For the CFO estimate shown in (4.41), it is a weighted average of θ_i. However, instead of keeping all calculated weights and doing average which may causes longer processing delay higher resource consumption, two self-defined accumulators are implemented to realized the weighted average of θ_i. And also, noticing that both the numerator and denominator in (4.41) have the specific term $i^3 r_i$, the resources can be further optimized. Figure 4.15 shows the architecture of CFO calculation with the accumulators. When β_{sum}(definition given in Algorithm 2) and corresponding index i are provided, the elements of numerator and denominator are calculated and accumulated, until the index value increases to a certain point. The coefficient $\frac{1}{2\pi t_b l_t}$ is a fixed constant as long as the properties of the communication system do not change.

4.4.4.4 Pipeline and throughput

Table 4.2: Number of Cycles for Stages in Estimation Process

	T_W	T_R	T_S	T_{CO}	T_P	T_C
Cycles	$n+2$	$\frac{n^2-nl_t}{2l_t^2}$	4	7	2	4

In spite of unscrambled pilot read and write, the architecture of proposed method is pipelined. Figure 4.16 sketches the timing series and processing latency of proposed architecture. The latency of an entire estimation process consists of 6 parts: unscrambled pilot write and read (T_W and T_R), sample correlation (T_S), coordinate rotation digital computer (CORDIC) based complex to magnitude and angle conversion (T_{CO}), phase unwrapping (T_P) and CFO computation (T_C). In Figure 4.16, the read process starts after all unscramble pilot symbols are written in the BRAMs, while other modules start immediately when the data comes out of memories, taking the advantage of pipeline architecture. Therefore, the processing latency of an estimation process is $T_{single} = T_W + T_R + T_S + T_{CO} + T_P + T_C$. The number of cycles for each stage in the estimation process is shown in Table 4.2. Note that n and l_t are length of pilot and number of transmit antennas, as described in Section 4.4.1. Consider a burst mode transmission, where the burst data is first loaded to buffer after reception, and the estimation process starts with full speed, as shown in Figure 4.16. Therefore, suppose there are N frames that require CFO estimations, the achievable throughput S of the proposed architecture can be expressed with processing latency:

$$S = \frac{1}{N(T_W + T_R) + T_S + T_{CO} + T_P + T_C}$$

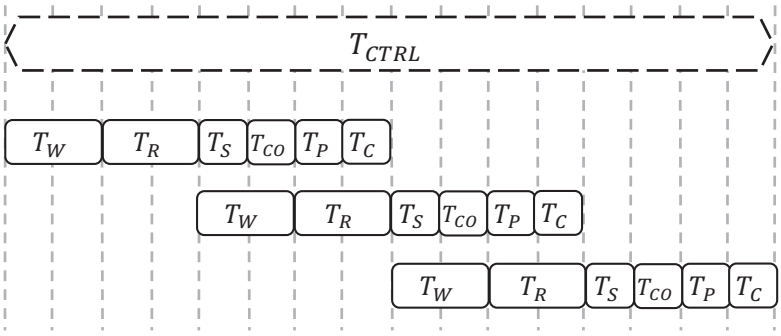

Figure 4.16: Process of pipeline architecture

4.5 Coarse Carrier Frequency Offset Estimation Using Neural Networks

4.5.1 System and Channel Models

We send length n pilot signal $\boldsymbol{S} = [\vec{s}_{k,t}]_{k,t} \in \mathbb{C}^{N \times l_t}$ through the channel, with sample period t_b. The pilot signals, known to the receivers, are orthogonal for different transmit antennas , i.e., $\vec{s}_{t_1} \perp \vec{s}_{t_2},\ \forall t_1 \neq t_2$, $\boldsymbol{S}^\dagger \boldsymbol{S} = \frac{N\rho}{l_t} \boldsymbol{I}_{l_t \times l_t}$, where ρ is the symbol energy, $\vec{s}_{t_1}$is the transmitted pilot signal at t_1-th transmit antenna (similar definition for $\vec{s}_{t_2}$), and the energy of pilot satisfies $\text{Tr}\left(\boldsymbol{S}^\dagger \boldsymbol{S}\right) = \sum_{t=1}^{l_t} \sum_{k=1}^{n} |s_{t,k}|^2 = \rho n > 0$. Define $\boldsymbol{x} \in \mathbb{C}^{N \times l_t}$ as the pilot signal with the effect of carrier frequency offset, t_b denotes the sample period and f_δ denotes the CFO, we have

$$\boldsymbol{x} \ = \ \boldsymbol{F}\boldsymbol{S} \tag{4.56}$$

where

$$\boldsymbol{F}(f_\delta) = \begin{bmatrix} e^{j2\pi f_\delta t_b \cdot 0} & 0 & \cdots & 0 \\ 0 & e^{j2\pi f_\delta t_b \cdot 1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & e^{j2\pi f_\delta t_b (N-1)} \end{bmatrix} \tag{4.57}$$

is the CFO matrix.

The general model of a MIMO system is shown in Figure 4.17.

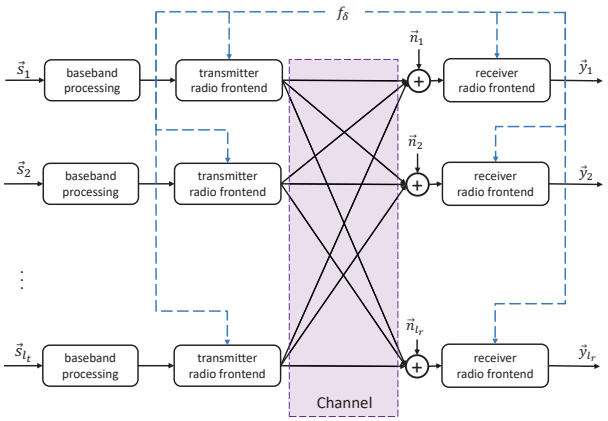

Figure 4.17: General model of a MIMO system

4.5.1.1 AWGN Channel

AWGN channel is the the most basic channel model of wireless communication systems. In an AWGN channel, additive white Gaussian noise is the only disturbance and channel does not bring any phase rotation to the transmitted signal. Assume the transmitter has l_{t} antennas, the receiver has l_{r} antennas and the MIMO system is in a place where the channel of each link is an AWGN channel. Let $h_{r,t}$ be the channel coefficient from the t-th transmit antenna to the r-th receive antenna, then we have $\boldsymbol{h} = [h_{t,r}]_{t,r} \in \mathbb{C}^{l_t \times l_r}$ where $h_{t,r} = 1, \ \forall t, r$. Define $\boldsymbol{n} = [\vec{n}_r]_{r=1:l_r} \in \mathbb{C}^{N \times l_r}$ as the noise at receiver, and the noise has complex normal distribution $\mathcal{CN}\left(\vec{0}, \mathbb{E}\left[\vec{n}_r^{\dagger}\vec{n}_r\right]\right)$. Taking the signal model in Equation (4.56), the received signal can be expressed as

$$\begin{aligned} \boldsymbol{y} &= \boldsymbol{x}\boldsymbol{h} + \boldsymbol{n} \\ &= \boldsymbol{x} + \boldsymbol{n} \end{aligned} \tag{4.58}$$

For r-th receive antenna, the received signal can be written as [90]

$$\begin{aligned} y_r(k) &= \sum_{t=1}^{l_t} h_{t,r} s_t(k) e^{j2\pi f_\delta k t_b} + n_r(k) \\ &= \sum_{t=1}^{l_t} s_t(k) e^{j2\pi f_\delta k t_b} + n_r(k) \end{aligned} \tag{4.59}$$

4.5.1.2 Flat/slow Fading Channel

Suppose the transmitter has l_t antennas, the receiver has l_r antennas and the MIMO system is in a flat fading or slow fading channel where the channel state is assumed to be constant during the transmission of a single frame. Define channel coefficient matrix as $\boldsymbol{h} = [h_{t,r}]_{t,r} \in \mathbb{C}^{l_t \times l_r}$ where $h_{t,r}$ is the channel coefficient from the t-th transmit antenna to the r-th receive antenna. Define $\boldsymbol{n} = [\vec{n}_r]_{r=1:l_r} \in \mathbb{C}^{N \times l_r}$ as the noise at receiver, and the noise has complex normal distribution $\mathcal{CN}\left(\vec{0}, \mathbb{E}\left[\vec{n}_r^\dagger \vec{n}_r\right]\right)$. Taking the signal model in Equation (4.56), the signal at receiver is

$$\boldsymbol{y} = \boldsymbol{x}\boldsymbol{h} + \boldsymbol{n} \tag{4.60}$$

For r-th receive antenna, the received signal can be written as [19]

$$y_r(k) = \sum_{t=1}^{l_t} h_{t,r} s_{t,k} e^{j2\pi f_\delta k t_b} + n_r(k) \tag{4.61}$$

4.5.1.3 Multipath Channel

In wireless communications, multipath channel is a channel through which the signal has two or more paths. The causes of multipath fading include reflection, diffraction, scattering and some other physical phenomena. The multipath channel model is demonstrated as follows. Consider the transmitted pilot signal as described at the beginning of Section 4.5.1. During the transmission, the signal travels through a multipath channel. For

the channel between t-th transmit antenna and r-th receive antenna, the length of channel response $\vec{h}_{t,r}$ is m. Define $\boldsymbol{n} = [\vec{n}_r]_{r=1:l_r} \in \mathbb{C}^{N \times l_r}$ as the noise at receiver, and the noise has complex normal distribution $\mathcal{CN}\left(\vec{0}, \mathbb{E}\left[\vec{n}_r^{\dagger}\vec{n}_r\right]\right)$. Assume frequency offset f_δ does not change during each frame, then the received signal at r-th receive antenna can be expressed as

$$\vec{y}_r = \sum_{t=1}^{l_t} \vec{h}_{t,r} * \vec{x}_t + \vec{n}_r \tag{4.62}$$

or in scalar form as

$$y_r(k) = \sum_{t=1}^{l_t}\sum_{n=1}^{N} h_{t,r}(m-k)s_t(k)e^{j2\pi f_\delta k t_b} + n_r(k) \tag{4.63}$$

where $h_{t,r}(m-k) = 0,\ \forall k > m$.

4.5.2 Neural Network Based Estimator

In this section, some traditional coarse CFO estimation methods are briefly reviewed and discussed, and then NN based coarse CFO estimator is illustrated, including architecture of proposed NN estimator and details about layers in the network. Due to limited space, however, some details such as loss function and activation layer are demonstrated in short.

4.5.2.1 Traditional coarse CFO/IFO Estimation Methods

Before machine learning methods are emerging into the blueprint, traditional CFO estimation methods of wireless communications were mainly mathematical methods. For the coarse CFO estimation in wireless systems, the most intuitive method is fast Fourier transform (FFT). The receiver takes FFT of the received signal, finds the peak of spectrum

and roughly calculates how much frequency the signal has been shifted. The FFT based method is intuitively straightforward, but the FFT module is very complex and sometimes noise could ruin the estimation results.

There are some other methods besides the FFT based estimator. For example, the distortion introduced by CFO results in a phase shift between adjacent samples, therefore by tracking the phase shift using phase-lock-loop (PLL) or Costas loop based estimator, one can compensate the impact brought by CFO. Decision-directed method, which is a frequently used constellation-dependent CFO estimation method, compares the phase difference between the received signal and constellation to estimate the CFO.

ML based estimator is another promising solution to coarse CFO estimations. Researchers formulate the ML based estimators for different MIMO systems and propose the solutions to find the corresponding coarse CFO estimates which maximize the proposed likelihood functions.

For all the methods above, however, there are still drawbacks need to be improved. First, the CFO acquisition range of tracking approaches such as PLL is limited. If a wireless system suffers severe CFO, the tracking methods do not function well. Second, the likelihood functions for maximum-likelihood estimators are derived for specific systems models, leading to the fact that the estimator derived for one system model cannot be applied to another, or vise versa.

4.5.2.2 Neural Network for Coarse/Integer Frequency Offset Estimation

In recent years, traditional wireless communications is taking a shift towards "smart" wireless communications, which indicates the convergence of machine learning technologies and wireless communications. Therefore, applying neural network to solve the coarse CFO estimation problem may contribute to the transition progress. For the system and channel models demonstrated in Section 4.5.1, the NN estimator for coarse CFO estima-

tions is expected to take the received signal, learn the estimation rules accordingly and find out the CFO estimate $\hat{f}_\delta$. In traditional wireless communications, ML based grid search is one of the most popular solutions to coarse CFO estimation problem [1, 55, 56]. The grid search method calculates the value of likelihood function with a number of possible CFO candidates and searches for the optimal CFO candidate that maximizes the likelihood function. Now reconsider the idea of ML based grid search method in the way of classification. Instead of trying out all candidates and searching for the optimal one, using the classification method we can classify which CFO candidate is the one closest to ground truth. Therefore, inspired by the ML based grid search method, we propose an NN estimator, and the NN estimator classifies the optimal CFO candidate.

Figure 4.18 shows the general process of coarse CFO estimations using proposed NN estimator. During the training process, training data is captured and splitted for training and validation. The data for training is first fed to the proposed NN estimator to train the network parameters, then the data for validation is fed to the network to examine the performance. Later, the next epoch starts to modify and update the network for a smaller value of the loss function. Once the entire training process is finished, one can feed the test data to the trained NN estimator to perform coarse CFO estimations. As can be seen from Figure 4.18, the output of proposed NN estimator is a vector indicating which candidate is the one closest to true CFO. Therefore by applying the proposed NN estimator, the coarse CFO estimation problem is transformed into a classification problem.

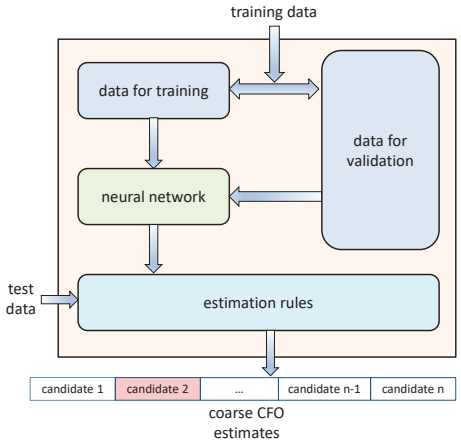

Figure 4.18: Coarse CFO estimations using proposed NN estimator

4.5.2.3 Neural Network Layers in The NN Estimator

The architecture of proposed NN estimator is shown in 4.19. The proposed NN estimator contains an LSTM layer, multiple dense layers with a sigmoid activation layer and a dense layer with a softmax activation layer for output. In the proposed NN estimator, LSTM layer is firstly employed to extract the time sensitive features brought by CFO. Then dense layers are employed to learn the rules for coarse CFO estimation. It can also be observed that a data reshaping and selection module is employed before proposed NN network.

Preprocessing: data reshaping and selection module We assume that the network has no prior knowledge about the number of antennas in the MIMO system, therefore a data reshaping and selection module is introduced to make the NN estimator more generic and adaptive to different numbers of antennas.

The data reshaping and selection module takes the received signal and format it to a certain shape. Since the number of receive antennas l_r is not know to the NN estimator,

the data reshaping and selection module conducts a cross-antenna sort so that data fed to NN estimator contains CFO information gathered from all receive antennas. Consider the received signal is $\boldsymbol{y} \in \mathbb{C}^{N \times l_r}$ (see also Section 4.5.1), the input size of proposed NN estimator is $L \times 2$. The input has two channels, one for real part of the received signal and the other for imaginary part. Once the signal is detected, the data reshaping and selection module takes one block of symbols from each antenna in turn, concatenates the blocks of symbols and feeds the concatenated sequence to input memory until full. The diagram of how data reshaping and selection module works is shown in Figure 4.20. If the received signal does not fill up the input memory, zeros padding is employed.

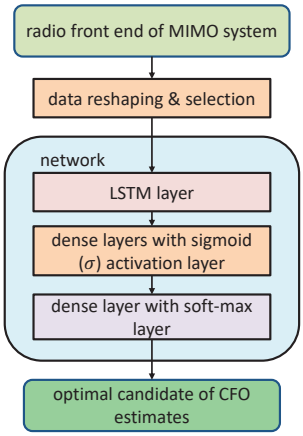

Figure 4.19: Architecture of proposed NN estimator for coarse CFO estimation

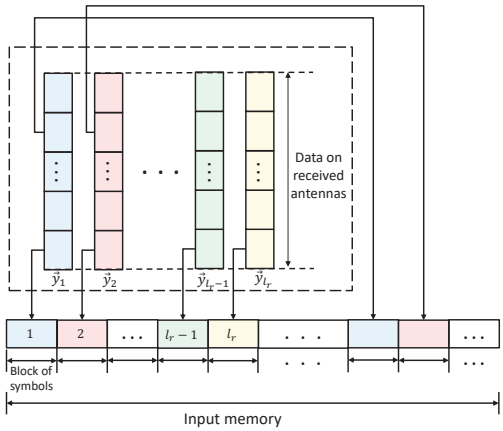

Figure 4.20: Diagram of data reshaping and selectin module

LSTM layer An LSTM layer, as shown in Figure 4.21, has a number of connected LSTM units (σ is a sigmoid function). Each general LSTM unit consists of a cell, a forget gate, an input gate and an output gate. As can be seen, when the output of last cell and the input of current cell are obtained, the forget gate tells how much information will be removed from the last cell state. Later, the updated cell state is added by a value given from input gate. The value is obtained from a function of the input of current cell and the output of last cell; Finally, the output gate computes the output of current LSTM unit, and transfers the output as well as current cell state to the next LSTM unit. Since the architecture of LSTM units makes them sensitive to important events in a time series, the LSTM layer is suitable for time series data processing and classification.

Usually, the larger number of LSTM units are cascaded in an LSTM layer, the more features of time series data can be collected. However, when the number of LSTM units is greater than the size of input data under which case zero padding is needed, no extra amount of features should be expected. In our case, the number of LSTM units in the LSTM layer is 128. However it is adjustable according to different frame structure.

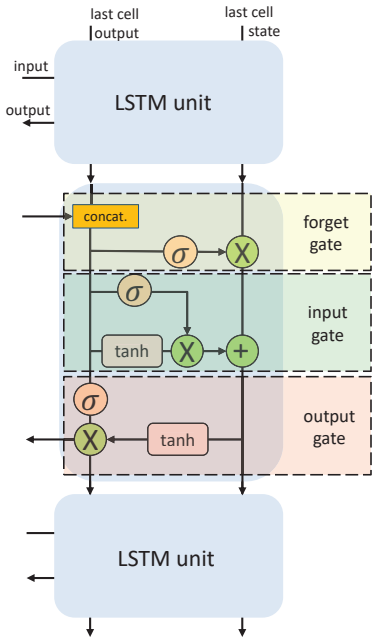

Figure 4.21: Structure of an LSTM layer

Dense layers The dense layer has a straightforward architecture, easy-to-use features for mapping and regression and it has been proven to be a universal approximator as long as the number of nodes is sufficient [91]. On the other hand, unfortunately, due to the fact that a dense layer does not take any sequential data, it is insensitive to time related features of the input signal. Therefore, the dense layers (or hidden layers in this letter) are responsible for mapping the LSTM layer output features to the classification results.

In dense layers the information always moves forward and never goes backwards. The structure of dense layers in the proposed NN estimator is shown in Figure 4.22. The first dense layer takes input data and then in each hidden layer, every single node takes data fed from the nodes in previous layer, multiplies the data with weights, adds biases and feeds to the next layer.

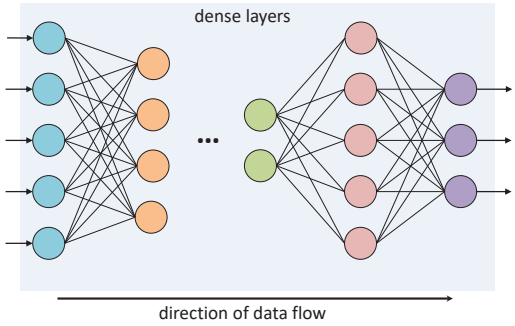

Figure 4.22: Structure of dense layers

Activation layers Activation layer is a layer made of a number of activation functions, and it is an important part of a neural network. The activation function defines whether a node is activated or to what level the node is activated. An activation layer takes the output data from previous layer and maps the values into a bounded range. The activation functions used in the activation layers in proposed NN estimator are listed in Table 4.3. Note that both the input and output of softmax layer are vectors.

As may be noticed in Figure 4.19, the activation layer for the last dense layer is a soft-max layer. This soft-max layer also prepares the data for the calculation of loss of the proposed neural network. Since the number of CFO candidates (classes) is more than 2, the loss function used in the proposed NN estimator is a categorical cross-entropy function.

Table 4.3: Activation functions used in proposed NN estimator

Name	Function
Sigmoid	$\sigma(x) = \frac{1}{1+e^{-x}}$
Softmax	$\eta(\vec{x}) = \frac{e^{\vec{x}}}{\sum e^{\vec{x}}}$

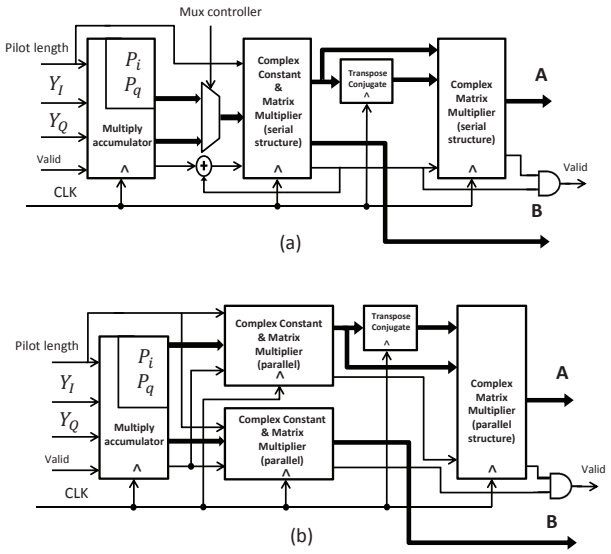

Figure 4.3: (a): Receiver module of fixed point structure; (b) Receiver module of floating point structure

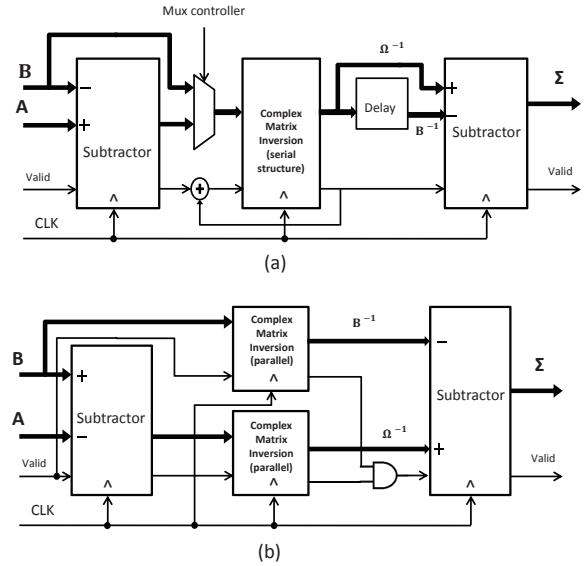

Figure 4.4: (a): Covariance calculator of fixed point structure; (b) Covariance calculator of floating point structure

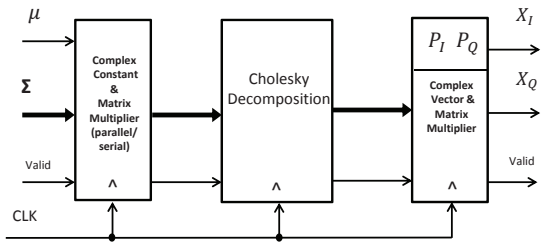

Figure 4.5: Diagram of transmitter module

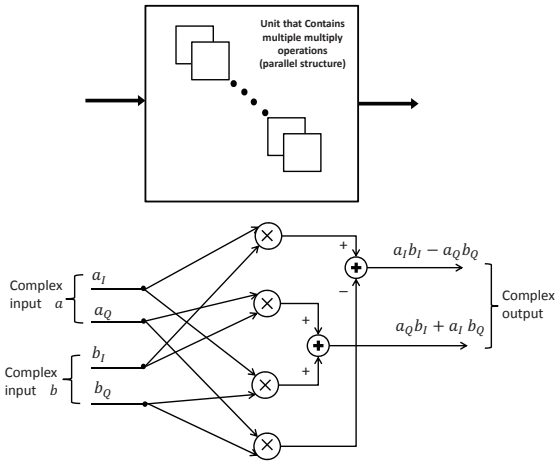

Figure 4.6: Diagram of parallel structure units containing multiple multiplication operations

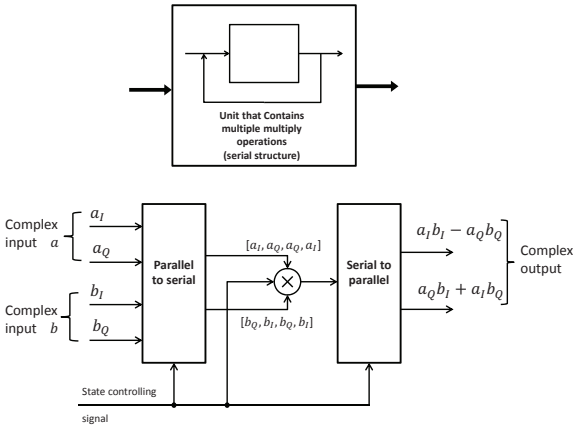

Figure 4.7: Diagram of serial structure units containing multiple multiplication operations

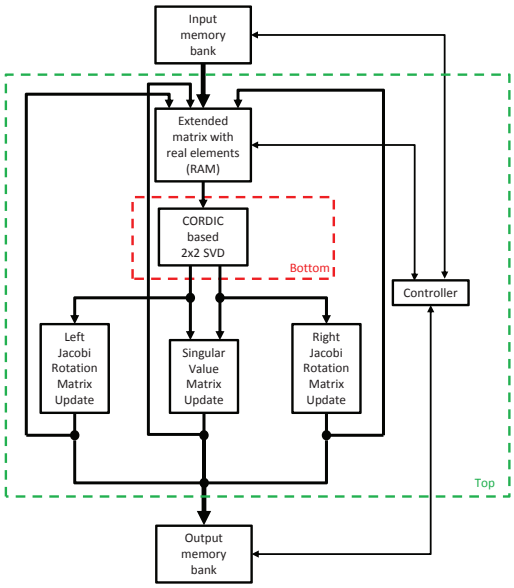

Figure 4.8: Generic system structure

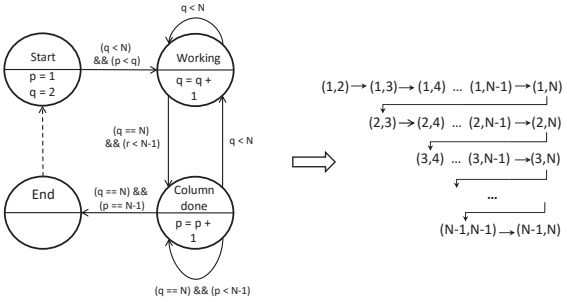

Figure 4.9: Diagram of top layer FSM

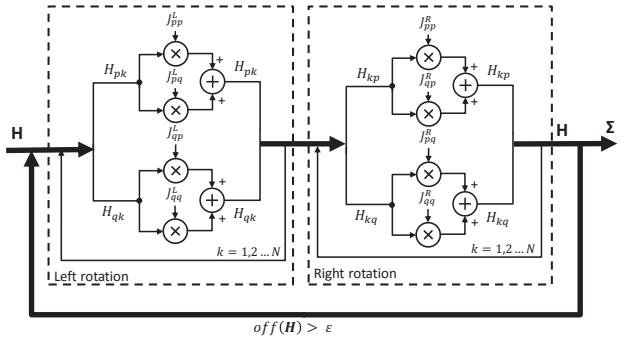

Figure 4.10: Flow chart of singular value matrix update

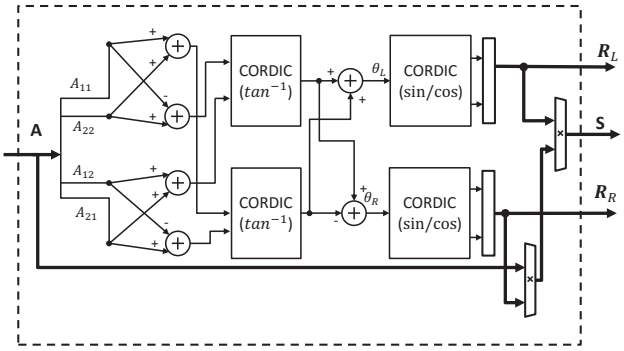

Figure 4.11: Diagram of 2×2 CORDIC based SVD

Chapter 5

Results and Analysis

In this chapter, results of the proposed approaches in Chapter 4 are illustrated and analyzed. The order of sections in this chapter is the same as in Chapter 4.

5.1 Results of Dual Link Algorithm Implementation

The performance comparison is listed between two different structures proposed in Sections 4.1.2. To compare the performance of our prototype designs, simulation and synthesis for FPGAs are targeted on Xilinx ZYNQ-7 ZC706 which is one of the designate FPGA boards for Analog Devices AD-FMCOMMSx-EBZ series SDR platforms.

5.1.1 Synthesis and Resource Utilization

Table 5.1: FPGA resource utilization of an individual user on ZC706

Structure	Resource	Utilization	Available	Utilization %	Max Clock Rate
Fixed Point	LUTs	6654	218600	3.04	181.82 MHz
	Registers	9923	437200	2.27	
	DSP	160	900	17.78	
	Block RAM	4	545	0.73	
Floating Point	LUTs	30933	218600	14.15	337.84 MHz
	Registers	54466	437200	12.46	
	DSP	408	900	45.33	
	Block RAM	4	545	0.73	

Table 5.1 illustrate the resource utilization of both fixed-point and floating-point designs on ZC706, respectively. It is obvious that the resource usage of floating-point structure is much higher than that of fixed-point structure, except for the block memory which stores pilot signals. On the other hand, the achievable clock rate of floating-point structure is 337.84 MHz while fixed point structure only achieves 181.82 MHz clock rate, only around half of 337.84 MHz. This is because the fixed-point structure with resource reuse needs strict timing control where the paths of controlling signal becomes the critical path, hence the maximum clock rate of fixed-point structures drops down. This controller design will be improved in our future work.

5.1.2 Performance Evaluation

Simulations are performed to evaluate the processing latency and accuracy of two structures. We also conduct simulations of single user single iteration $1,000,000$ times on an Intel i5 quad-core CPU with 8GB memories as performance guideline. Table 5.2 shows a single iteration processing latency of a single user which includes all process time after receiving the signal and before transmitting the signal out.

Table 5.2: Comparison between FPGA and CPU

Platform	FPGA		CPU
Device	ZC706	ZC706	i5 quad core
Structure	fixed point	floating point	floating point
Error range	$\approx 3 \times 10^{-3}$	$10^{-4} - 10^{-3}$	$\leq 10^{-4}$
Single user single iteration processing latency	$1579ns$	$880ns$	$3380ns$
Speedup factor	2.14	3.84	1

From Table 5.2 we find that when comparing with floating point structure, the fixed point structure has both advantages and disadvantages. The fixed point structure with many serial structured computational units utilizes less FPGA resources while it generates a wider range of errors, as well as long latency. Both proposed structures on an FPGA, however, can accommodate a shorter processing latency than that performed by CPU, with a speed up factor of 2.14 and 3.84, respectively.

5.2 Results of Hardware-Efficient SVD Architecture

5.2.1 Simulation Results

The performance of the proposed architecture is compared against the BLV in terms of average number of sweeps needed to reduce the off diagonal norm to a specified threshold. When the off diagonal norm is reduced to the threshold, the singular value matrix is regarded as convergent and the SVD process is done. The performance of BLV architecture is obtained from Reference [92] while that of the proposed architecture comes from Monte-Carlo simulations with same configurations: uniformly distributed complex matrix elements and same threshold (10^{-12} of its original value). Figure 5.1 illustrates the average number of sweeps required for convergence with various matrix orders.

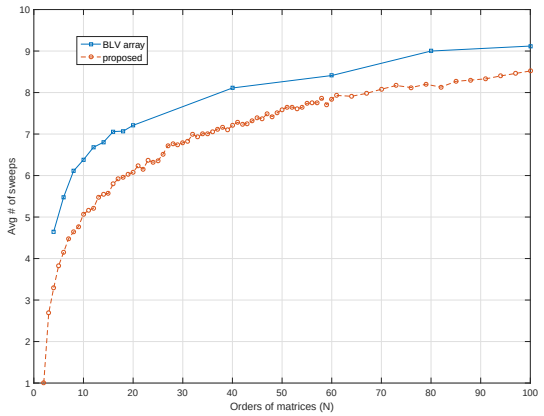

Figure 5.1: Average number of sweeps needed for convergence

It can be observed from Figure 5.1 when the matrix order is greater than 40, the number of sweeps of both architecture is nearly a proportion of the matrix order. On the other hand, the average number of sweeps of proposed architecture is smaller than that of BLV architecture at all matrix orders.

5.2.2 Hardware Resource Utilization

Since the primary goal of the proposed architecture is to provide a hardware-efficient solution for SVD in massive MIMO system, the hardware resource utilization is illustrated and compared with that from Reference [93] and [94]. Unfortunately, since the targeted FPGA board in this section is Zynq 7 ZedBoard and Xilinx's new product Vivado does not support Vertex 5 or 6 series anymore, we are not able compare our design with Reference [93] and [94] on same devices. Even with different FPGA platforms, however, the hardware resource utilization still reflects the hardware efficiency in a certain condition. In other words, if we implement the proposed design on Vertex 5 or 6, the hardware resource utilization will reported to be almost the same as on ZedBoard. The hardware resource utilization is shown in Table 5.3.

Table 5.3: Resource utilization

	Proposed Architecture	BLV array (8×8) [93]	Mrudula (8×8) [93]	Architecture I Ibrahim [94]
Selected Device	Znyq 7 ZedBoard	Virtex 6 VLX365	Virtex 6 VLX365	Virtex 5 VLX350
Achievable Frequency	128.2 MHz	147.0 MHz	146.6 MHz	146.0 MHz
Slice LUTs	1098	45343	51254	19542
Slice Registers	1333	42322	47749	25113
DSP48E1s	23	160	160	N/A

From Table 5.3 it can be found that the slices engaged in BLV and Reference [93] are around 40 times more than that in the proposed architecture. And in terms of DSPs, the proposed architecture consumes less than one fifth of the total number occupied in Reference [93]. Meanwhile, though the number of DSPs is not given in Reference [94], the number of slices is still around 20 times more than that in the proposed architecture. In addition, from Table 5.3 we can observe that the achievable clock rate of proposed design is slightly lower than that in [94], however, we still observe a better timing performance of proposed design, which is shown in Section 5.2.3.

5.2.3 Timing and Throughput

To facilitate a faster design and evaluation with different orders of matrices, we model the timing and throughput in this section. Figure 5.2 sketches the processing latency of a single iteration to provide an overview of the timing schedule. The latency of a single iteration consists of 4 primary parts: controlling (T_C, 15 cycles), raw data extraction (T_R), 2×2 SVD (T_{SVD}) and matrices update (T_H, T_U and T_V).

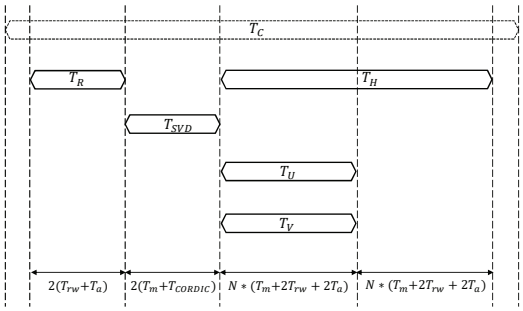

Figure 5.2: Timing schedule of a single iteration

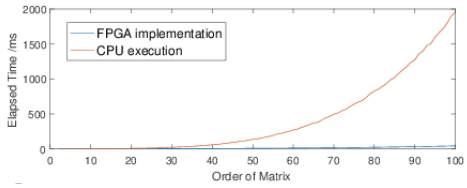

Figure 5.3: Comparison between CPU and proposed architecture

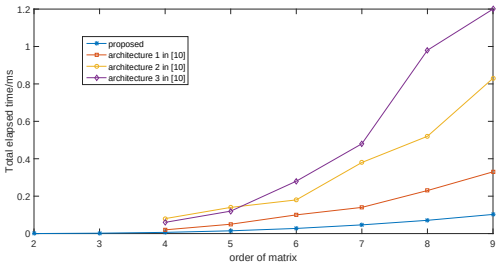

Figure 5.4: Comparison between different architectures

T_{rw} (1 cycle), T_a (1 cycle), T_m (1 cycle), T_{CORDIC} (13 cycle)in Figure 5.2 denote the latencies of memory read/write, address computation, multiplication and CORDIC operation, respectively. Hence, throughput S of one sweep for the proposed architecture can be expressed with processing latency:

$$S = \frac{1}{L_p(N(N-1)/2)} = \frac{2}{N(N-1)L_p} \tag{5.1}$$

where N is the order of factorized matrices and $L_p = 2(N+1)(T_m+T_{rw}+T_a)+2T_{CORDIC}$ represents the processing latency of a single iteration.

Based on our previous work, the proposed architecture is aiming to be implemented on FPGA based software defined radio (SDR) platforms instead of on CPU as most SDR platforms do. Therefore to evaluate the timing performance of proposed architecture, comparison with CPU (i5 660 dual core, 3.3GHz) simulation and the architectures in Reference [94] are conducted and illustrated in Figure 5.3 and Figure 5.4, respectively. As can be seen in both figures, the proposed architecture performs much faster than CPU, especially when matrix order becomes large, and also competitive to the architectures in Reference [94]. Meanwhile, the error of computation of SVD matrices goes up to 1% in Reference [94] while the error in our design is at most $2^{-8} \approx 0.4\%$ of the singular values of matrices.

5.3 Results of Hardware Efficient ML Based CFO Estimation

In this section, simulations are conducted to analyze the performance of the hardware efficient ML based CFO estimation approach. The relationships between normalized MSE and SNR, as well as pilot length, are discussed. Besides, the acquisition range is also tested. Simulations of other synchronization algorithms are also conducted under same conditions to compare with. This section also quantifies the hardware resource consumption for an FPGA-based implementation, showing that the low hardware resource utilization of proposed architecture makes it accessible for a number of target devices, especially for FPGA-based software defined radio (SDR) where the vacant hardware resource could be distributed to other essential parts in an entire communication system.

5.3.1 Simulation Results

We consider a general 2x2 MIMO system with random complex Gaussian channel with unit variance and i.i.d noise-plus-interference with unit variance as the scenario. To acquire accurate simulation results, we conduct Monte-Carlo simulation and gather the mean square error (MSE) over the simulations. With these given parameters, CRLBs with different SNRs are calculated to compare with simulation results as the performance guideline. For convenience, we normalize the CFO, where the normalized CFO is set to be a proportion to baseband bandwidth $\frac{1}{t_b}$. Simulations of our algorithm and those in [1, 51, 57] are conducted and compared under same conditions.

5.3.1.1 SNR and pilot length

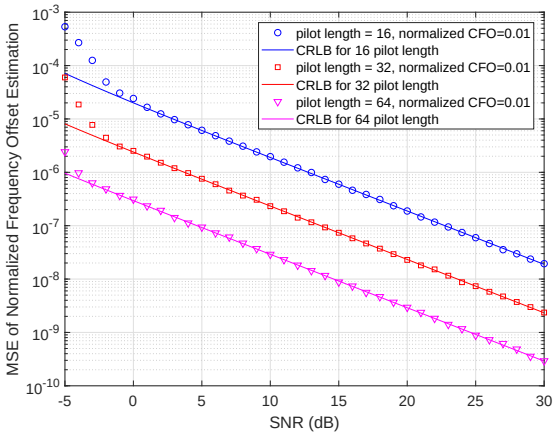

Figure 5.5: Influence of SNR and pilot length on MSE of CFO estimation

Figure 5.5 shows the relationship between SNR and normalized MSE of CFO estimation with different pilot lengths. To emphasize the influence of SNR, for each given pilot length, simulations are conducted with the normalized CFO held at 1% of the bandwidth, in a 2x2 MIMO system. As the SNR increases, the MSE for each pilot length drops fast between -5dB and 0 dB, and then keeps approaching corresponding CRLB. At the same

time, Figure 5.5 reveals that at each time the pilot length is doubled, both the MSE and corresponding CRLB have around 9 dB drops, which matches the derivation in (4.55).

5.3.1.2 Acquisition range

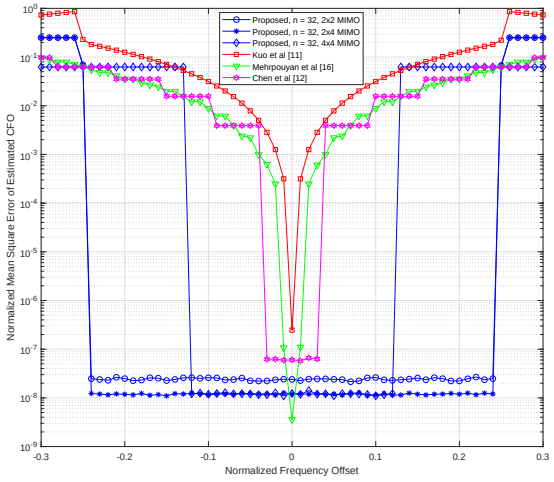

Figure 5.6: Acquisition range of CFO estimation methods

Figure 5.6 shows the acquisition range of different CFO estimation methods. Since the acquisition range of searching methods is also confined by the searching range, we exclude estimators with searching techniques from this comparison and compare the proposed method with those in [51, 52, 57] which are also closed form estimators. For all four methods, we conduct simulations under the same conditions: pilot with 32 symbols, 20 dB SNR, 2x2 MIMO system for proposed method and the method in [57], SISO system for the methods in [51, 52] which are dedicated for SISO. From Figure 5.6 we notice that by avoiding phase unwrapping, the acquisition range of CFO estimation techniques in [51, 52, 57] are limited, while the MSE of the method in [52] is stable within the acquisition range. If we recognize normalized MSE at 10^{-4} as a threshold for successful CFO estimations, the maximum detectable normalized CFOs are 25%, 3%, 3%

and 2%, respectively for proposed method, Chen's method, Mehrpouyan's method and Kuo's method. On the other hand, MIMO with different number of antennas using proposed method are also tested. From Figure 5.6 we can observe that due to the scrambled periodic matrix $[\boldsymbol{O}]_{i=1:m}$ in (4.26), the acquisition range is closely related to the number of transmit antennas where every time the number of transmit antennas is doubled, the acquisition range reduced by one half.

5.3.1.3 Mean square error

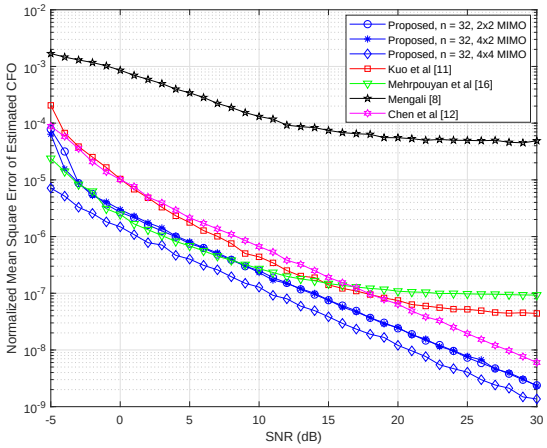

Figure 5.7: MSE comparison of different CFO estimation techniques

To examine the performance, we conduct simulations of different CFO estimation methods over a wide range of SNRs and obtain their normalized MSE to compare proposed method with others. To make the comparison more realistic, we extend the CFO estimator for SISO system in [1] to an estimator for MIMO system, so that the comparison could be made under same conditions. However, the methods in [51, 52] are designed for SISO systems. For the test cases, pilot length is 32 symbols, normalized CFO is 1% and all simulations run with a 2x2 MIMO system except for the methods in [51, 52] which runs in a SISO system where all system settings are the same except for the num-

ber of antennas. For the MLE in [1] which searches among a number of potential CFO candidates, the searching range lays between -30% and 30%, and the interval between adjacent candidates is 0.3%. The performance comparison of proposed method with the methods in [1, 51, 52, 57] is shown in Figure 5.7. It can be seen that the proposed method holds a competitive performance when the SNR is low, and the best performance when SNR goes higher. The linear approximation employed in [51] results in a degradation in the performance where the quantization error plays a vital role. At low SNRs ($\leq$10 dB), the performance of proposed method is competitive with the method in [57]. But the normalized MSE of methods in [51, 57] almost stop decreasing when SNR goes higher than 20 dB. Besides, the normalized MSE of the method in [1] is way higher than that of other methods, which is led by the fact that the searching grid is not fine enough. Besides 2x2 MIMO, the proposed method is also tested with 4x2 and 4x4 MIMO systems and the results verifies that the number of receive antennas has influence on the MSE (more receive antennas, lower MSE) while the number of transmit antennas does not, as shown in (4.55).

5.3.1.4 Bit error rate

With the CFO estimates, the communication systems compensate the CFOs and recover the distorted signals so that the transmitted data can be correctly revealed. Therefore, we simulate an entire communication system with other impairments corrected, add the CFO synchronization methods and perform CFO synchronization to examine the BER performance. The BER comparison of different CFO estimation methods is shown in Figure 5.8. The setup of communication system is the same as that described in Section 5.3.1.3. Each transmitted frame consists of 32 pilot symbols and 288 payload symbols where the symbols are QPSK modulated. From Figure 5.8 we can observe the same trend shown in Figure 5.7. There is a noticeable case that the BER performance of the method

in [1] stays on the same level throughout the entire SNR range. This is caused by the quantization error of grid search and therefore we conclude that the grid search without high resolution may not be suitable for fine frequency estimation.

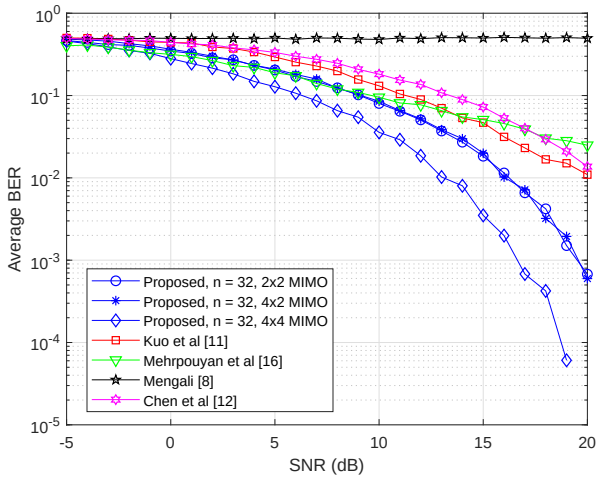

Figure 5.8: BER comparison of different CFO estimation techniques

5.3.2 Implementation Results

The circuits are synthesized and implemented using Xilinx Vivado 2017.2, targeting one of the popular FPGA development boards Xilinx Zynq XC7Z020 ZedBoard. The implementation results in terms of flip-flops (FFs), lookup tables (LUTs), DSP blocks, block RAMs (BRAMs) and achievable frequency are reported in Table 5.4.

Table 5.4: Resource Utilization and Achievable Frequency

Architecture	FFs	LUTs	DSPs	BRAMs	Frequency
Prop_2rec	1640 (2%)	2638 (5%)	13 (6%)	2 (1%)	125.8MHz
Prop_2recS	1640 (3%)	2638 (8%)	13 (11%)	2 (3%)	90.9MHz
Prop_4rec	1492 (1%)	2793 (5%)	16 (7%)	4 (3%)	125MHz
Search_10can	35280 (33%)	18550 (35%)	70 (32%)	10 (7%)	83.3MHz

Due to the fact that there is a lack of published ML based frequency synchronization implementation techniques, we extend the method in [1] to MIMO case and implement the corresponding architecture shown in Figure 4.12(b), in the case of a 2x2 MIMO system, for the purpose of comparison.

Search_10can refers to the architecture we implemented for the method in [1] with 10 potential CFO candidates. As is shown in Table 5.4, *Search_10can* shows significant resource utilization compared with proposed architectures *Prop_2rec* and *Prop_4rec*. Unfortunately, it is hard to find a balance between hardware resource consumption and estimation accuracy for *Search_10can*. Even though additional CFO estimation candidates result in smaller searching interval (assume same searching range) which leads to smaller quantization error, these additional candidates bring about higher hardware consumption, potentially exceeding the available amount.

Prop_2rec and *Prop_4rec* refer to the architecture of proposed method, with 2 receiver antennas and 4 receiver antennas respectively. According to Figure 4.12(a), there is extra hardware consumption brought by additional receiver antennas. However, the extra hardware resource consumption mainly consists of BRAMs for pilot storage and DSPs for received signal aggregation. The proposed architecture consumes only a trivial portion of available resources, and the rest can be allocated for other parts of an entire communication system. Or, on the other hand, the proposed architecture can be easily fitted in small FPGAs such as Xilinx Spartan-7 XC7S50CS (*Prop_2recS*). It is also worth noticing that the achievable frequencies of proposed architectures are around 40MHz higher than that of *Search_10can*. When CFO estimation unit is connected to other units in a communication system, lower achievable frequency may becomes the bottleneck of achieving high processing speed.

The maximum circuit frequency, reported as post-route, is $125.8MHz$ for proposed architecture with 2 receiver antennas implemented on ZedBoard. During our tests, pilot-

only frames are employed and each pilot-only frame has 32 QPSK symbols. Assume that the frames are received in burst mode, as discussed in Section 4.4.4.4, the proposed architecture can accomodate symbol rate up to $53.08Mbps$, comfortably exceeds the requirements for MIMO communication systems under most conditions.

In this section, simulations are conducted to analyze the performance of proposed NN estimator. The accuracy and performance of proposed NN estimator is presented and discussed in a variety of aspects including SNR range, channel models, number of antennas and CFO acquisition range. Simulations of some traditional synchronization algorithms are also conducted under same conditions as the guideline of performance evaluation. It is shown in the results that the proposed approach has consistent performance over a wide SNR range, and has the largest acquisition range among all tested coarse CFO estimation approaches. Meanwhile, the proposed NN estimator accommodates all the channel models we mentioned in Section 4.5.1, as well as tested numbers of antennas.

5.4 Results of Neural Network Based Coarse CFO Estimator

In this section, simulations are conducted to analyze the performance of the proposed NN estimator in Section 4.5.1. The accuracy and performance of proposed NN estimator is presented and discussed in a variety of aspects including SNR range, channel models, number of antennas and CFO acquisition range. Simulations of some traditional synchronization algorithms are also conducted under same conditions as the guideline of performance evaluation. It is shown in the results that the proposed approach has consistent performance over a wide SNR range, and has the largest acquisition range among all tested coarse CFO estimation approaches. Meanwhile, the proposed NN estimator accommodates all the channel models we mentioned in Section 4.5.1, as well as tested

numbers of antennas.

5.4.1 Configurations for NN estimator: training and testing

In Section 4.5.2, the proposed NN estimator is discussed while some details about the NN are not revealed for network training and testing. These detailed configurations for NN estimator training and testing are as follow.

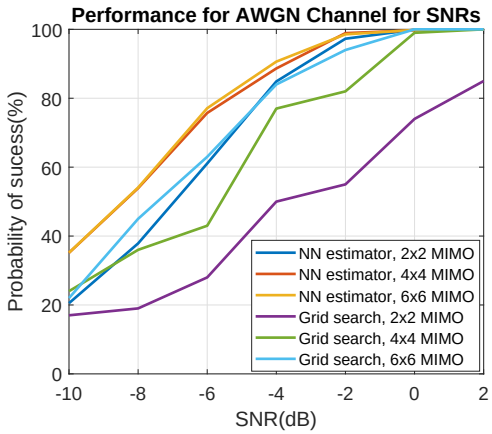

Figure 5.9: Performance over SNR range in AWGN channel

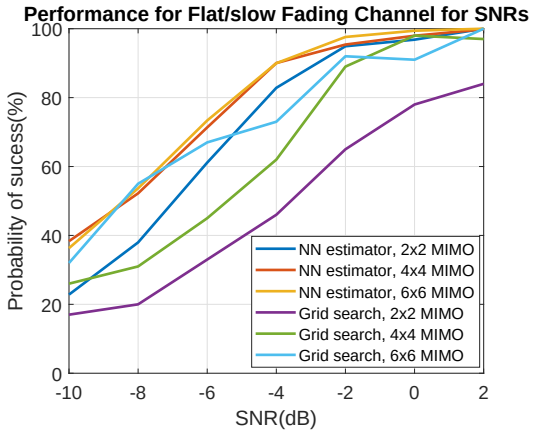

Figure 5.10: Performance over SNR range in flat/slow fading channel

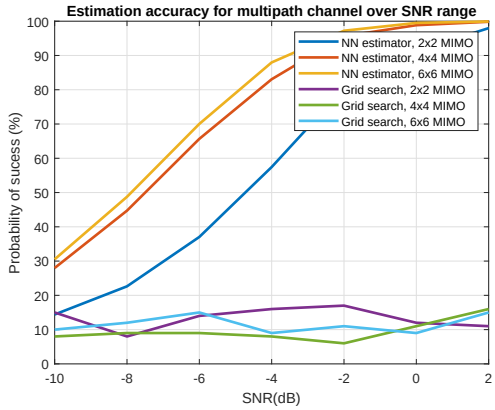

Figure 5.11: Performance over SNR range in multipath channel

5.4.1.1 Training configurations

- In simulations, the center frequency is $2.4GHz$ and sampling frequency is $10MHz$. However, instead of presenting CFO in Hz, we normalize the CFO so that the frequency offset presented as a ratio to sampling frequency: $f_\delta t_b$. The training dataset is generated across the normalized CFO range $(-0.5, 0.5]$ with 0.01 resolution $([-0.49, -0.48, ..., 0.5])$.

- The training dataset contains 5 different MIMO systems: 2×2, 3×3, 4×4, 5×5 and 6×6.

- All 3 channel models discussed in Section 4.5.1 are included in the training dataset.

- For flat/slow fading channel, the channel coefficient $h_{t,r}$ has symmetric complex Gaussian distribution $\mathcal{CN}\left(0, \sigma^2_{h_{t,r}}\right)$ and $\sigma^2_{h_{t,r}} = 1,\ \forall t, r$.2

- For multipath channel, we randomly generate the number of multipath and delay spread for each path. The delay spread should be less than the length of pilot signal

(32 symbol periods). In our case, we set the delay spreads to be 10 symbol periods or less for shorter simulation time.

- There are 32 QPSK pilot symbols in each frame, and the pilot signal on each antenna is orthogonal to others. The length of payload symbols in each frame is 64. The payload symbols are randomly generated for simulations, but since proposed NN estimator is a pilot-added method, the randomly generated payload symbols does not contribute to the training.

- The input size of proposed NN estimator is 128, and the length of each pilot block (defined at the beginning of Section 4.5.2.2) for data reshaping is 8. Therefore, the maximum number of receive antennas that the proposed NN can accommodate is $\frac{128}{8} = 16$.

- To avoid the disturbance of noise in training process, the SNR is assumed to be infinity. Under this case, the NN estimator could learn the estimation rules accordingly without the impairment from noise.

- There are 1000 frames per channel model per system per CFO candidate. According to the given conditions above, the datasets contains 100 normalized CFO candidates, 3 channel models and 5 MIMO systems. Therefore, the total number of frames contained in the training dataset is 1500000.

- When training the NN estimator, the batch size is set to be 15000 (10% of the training dataset) and the training process in total has 200 epochs.

5.4.1.2 Testing configurations

- Test cases are as follows: (1) Test for AWGN channel over SNR range $\{-10dB, -8dB, ..., 0dB, 2dB\}$, normalized CFO = 0.05; (2) Test for flat/slow fading channel over SNR range

$\{-10dB, -8dB, ..., 0dB, 2dB\}$ where $h_{t,r}$ has symmetric complex Gaussian distribution $\mathcal{CN}\left(0, \sigma^2_{h_{t,r}}\right)$ and $\sigma^2_{h_{t,r}} = 1, \ \forall t, r,$, normalized CFO = 0.05; (3) Test for multipath channel over SNR range $\{-10dB, -8dB, ..., 0dB, 2dB\}$ where the delay spread of each path is less or equal to 10, normalized CFO = 0.05; (4) Test for AWGN channel over different normalized CFOs in the set $\{-0.5, -0.45, ..., 0.45, 0.5\}$, SNR = $0dB$. (5) Test for flat/slow fading channel over different normalized CFOs in the set $\{-0.5, -0.45, ..., 0.45, 0.5\}$, where the channel is the same as described in (2), SNR = $0dB$; (6) Test for multipath channel over different normalized CFOs in the set $\{-0.5, -0.45, ..., 0.45, 0.5\}$, where the channel is the same as described in (3), SNR = $0dB$.

- The testing datasets are generated according to test cases instead of an ensemble of datasets containing all test cases.

- 10000 frames are generated and tested per SNR per antenna size in test case (1) – (3); 10000 frames are generated and tested per normalized CFO per antenna size in test case (4) – (6).

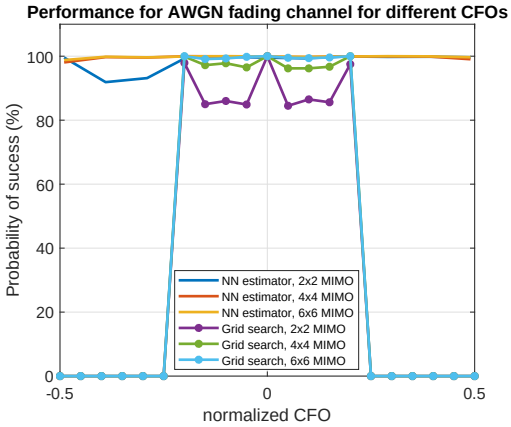

Figure 5.12: Performance with different CFOs in AWGN channel

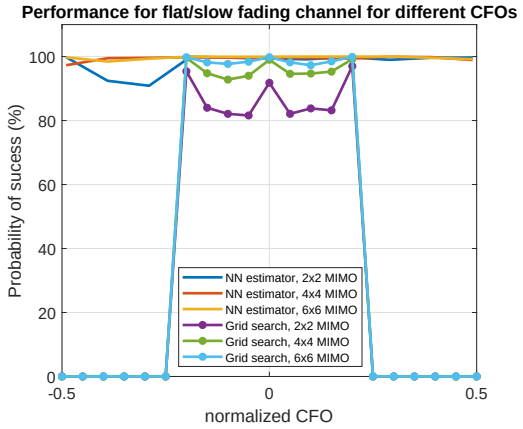

Figure 5.13: Performance with different CFOs in flat/slow fading channel

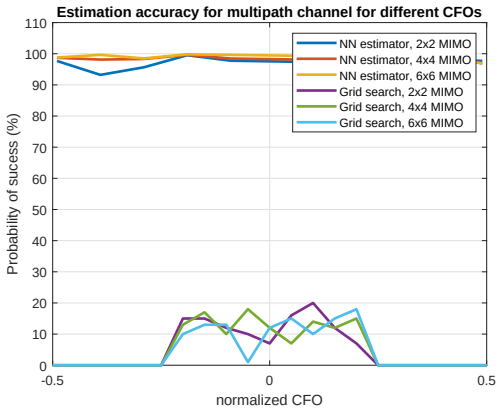

Figure 5.14: Performance with different CFOs in multipath channel

5.4.2 Performance over SNR range

To analysis the performance of proposed NN estimator, we generate testing dataset with different SNRs, ranging from $-10dB$ to $2dB$ at $2dB$ resolution. As mentioned in Section 4.5.2.2, the scheme of NN estimator is classifying the CFO candidate closest to real CFO value f_δ, which is inspired by the ML based grid search coarse CFO estimation method. Therefore, we also conduct simulations for the ML based grid search method [1] as per-

formance evaluation guideline. The performance evaluation metric used in this section is probability of success, which is the probability that estimators successfully classify the optimal coarse CFO candidate (candidate closest to f_δ).

Figure 5.9 (test case (1) in Section 5.4.1.2) shows the performance of proposed NN estimator and grid search method in AWGN channel. As can be seen, the probability of success increases rapidly along with SNR. The performance of proposed NN estimator, under the same condition is better than the grid search method, especially at low SNRs. In the meantime, we can observe that the number of antennas could improve the coarse CFO estimation performance of a MIMO system in AWGN channel.

Figure 5.10 (test case (2) in Section 5.4.1.2) and 5.11 (test case (3) in Section 5.4.1.2) show the performances of coarse CFO estimators in flat/slow fading and multipath channel, respectively. From Figure 5.10, we can draw similar conclusion that the probability of success of estimators grows with the increment of SNR, and the proposed NN estimator has a better performance comparing to grid search method.

For multipath channel, however, the probability of success of grid search method is less than 20% throughout the tested SNR range, regardless of the number of antennas. This performance degradation is brought by the limited assumptions of channel model. Therefore in Figure 5.11, it can be concluded that the ML based grid search method can not be applied directly to multipath channel. On the other hand, taking the advantage of adequate training dataset, the proposed NN estimator still shows a promising performance in multipath channel.

Taking Figure 5.9 - 5.11 together and we can observe that the probability of success for multipath channel is lower than AWGN and flat/slow fading channels. This observation also reflects the higher complexity of multipath channel. In addition, the proposed NN estimator can accurately select the optimal CFO candidate under all test cases for SNR $> 0dB$, showing a high probability of success (around 100%) as long as the signal

energy is higher than noise energy. Considering the fact that in real world, SNR $> 0dB$ is guaranteed under most of the conditions for successful transmissions, the performance of proposed NN estimator is promising.

5.4.3 CFO Acquisition range

The CFO acquisition range is another dimension for performance analysis. The wider the acquisition range, the better a MIMO system could survive large CFOs. Similar to Section 5.4.2, the tests of CFO acquisition range are performed in AWGN, flat/slow fading and multipath channels. As can be seen from Figure 5.12, 5.13 and 5.14 (test case (4) – (6) in Section 5.4.1.2 respectively), the probability of success of proposed NN estimator stays around 100% across the entire trained range (-0.5 to 0.5) and all tested channel models. The only exception is the 2×2 MIMO system around -0.3 normalized CFO in which case the probability of success is about 90%.

For ML based grid search method in a 2×2 MIMO system with AWGN and flat/slow fading channels, severe degradation can be observed in range $[-0.15, -0.05]$ and $[0.05, 0.15]$ of normalized CFO. The CFO acquisition range of ML based grid search method is up to 0.25 of the sampling frequency. Therefore, the CFO acquisition range of proposed NN estimator is twice as wide as the acquisition range of ML based grid search method. In multipath channel, the coarse CFO estimation process of grid search method is likely to be a "random selection" within acquisition range and shows a great performance degradation. This observation also verifies that the ML based grid search method in [1] does not work for multipath channel.

The proposed estimator also has the advantage of being compatible with multipath channel, comparing to grid search method.

As some other studies conducted in [19], the acquisition range of closed form ML based fine CFO estimation for flat/slow fading channel is up to 0.25 for 2×2 MIMO

system and up to 0.125 for 4×4 MIMO system. There are three other closed form CFO estimators mentioned and compared in [19], and the acquisition ranges of these estimators are less than 0.05.

5.4.4 Compatibility analysis: channel model and number of antennas

In this section, the proposed NN is trained with data from MIMO systems with different channel models and different numbers of antennas. From the results shown above, the proposed NN estimator is adaptive to all trained channel models as well as MIMO systems with different numbers of antennas. Consider a change of the wireless environment which results in a change of channel model, the methods derived for specific channel model will not work. Under this case, the proposed NN estimator may be considered as a better solution to accommodate the change. Meanwhile, the wide acquisition range of proposed NN estimator helps a MIMO system survive when it is suffering severe CFO impairment. With the ability of accommodating different channel models and change of number of antennas, it can be concluded that the proposed NN estimator has higher compatibility to different system models, and can easily fit in different MIMO systems if trained using an adequate dataset.

Chapter 6

Conclusion and Future Directions

In this chapter, an overall conclusion of this thesis is addressed. Besides, some future research directions are discussed, as well as possible improvements of proposed methods in this thesis.

6.1 Conclusion

In this thesis, we first introduce wireless communication systems and common impairments occur in the systems. Then our research studies concentrate on two common yet important impairments: interference and carrier frequency offset. The main body of this thesis consists of four proposed methods to solve the focused impairments. Two of the proposed methods are for interference alignment and the others are for carrier frequency offset. Based on the detailed results and analysis in Chapter 5, the conclusions of our work are drawn and listed below:

1. By designing the dual link rate maximization algorithm in a distributed manner and prototyping it on FPGA, it is running at least 2 times faster than on a CPU platform. We implement both fixed-point structure with serial units and floating-points struc-

ture with parallel units. The floating-point structure has a shorter processing time and higher accuracy comparing with fixed-point structure while it cannot fit into a small FPGA such as ZC702 on the ZedBoard. The fixed-point structure has longer processing time and its error range is wider, but resource efficiency makes it a more economical solution to realizing our dual link rate maximization algorithm.

2. The proposed hardware efficient SVD architecture consumes less hardware resource and can operate on matrices of arbitrary sizes. The processing time is competitive to existing designs when evaluated on an FPGA. The proposed architecture can be used to accelerate SVD computations in massive MIMO wireless communication systems.

3. Investigate CFO estimation in MIMO communication systems. The proposed ML based CFO estimation method is a closed form MLE and has close-to-bound performance. Since CFO estimation has significant influences on communication quality as well as the complexity of a MIMO synchronizer design, the proposed ML based CFO estimation method is important for MIMO systems where CFO estimation accuracy and time consumption of CFO estimation are sensitive. On the other handm, the hardware efficient architecture of our algorithm is also proposed. The proposed architecture is reconfigurable to different pilot lengths, and therefore can accommodate various frame structures without changing the architecture. Meanwhile, by avoiding searching techniques, the hardware resource consumption of proposed architecture is reduced hugely and is only related to the number of receiver antennas. Implementation results show that the proposed architecture consumes only a trivial amount of hardware resources, and therefore can be easily fitted into small FPGAs. The achievable frequency, at the same time, allows the system to approach high data rate.

4. The proposed NN based coarse CFO estimator is adaptive to different channel models, numbers of antennas, and shows promising performance on coarse CFO estimations. Meanwhile, the proposed NN estimator has wider acquisition range, which helps the MIMO system survive when severe CFO impairment is encountered. With its promising performance and higher compatibility, the proposed NN estimator could provide inspiration of designing traditional wireless systems in a "smarter" way in the future.

6.2 Future Directions

In this thesis, the topics covered in research studies are just a small portion of synchronization techniques in wireless communication systems. Therefore, the possible directions of future research studies are of broad interests in wireless communications. The directions of our future research studies include but not limited to:

1. Extension of proposed MIMO interference alignment to OFDM MIMO systems.

2. Extension of proposed carrier frequency offset estimation methods to OFDM MIMO systems.

3. Further investigation into baseband synchronization techniques in single carrier wireless systems, such as sampling frequency offset, sampling timing offset and so on.

4. End to end neural network powered baseband transceiver design of wireless communication systems.

Bibliography

[1] U. Mengali, *Synchronization techniques for digital receivers*. Springer Science & Business Media, 2013.

[2] T. Seymour and A. Shaheen, "History of wireless communication," *Review of Business Information Systems (RBIS)*, vol. 15, no. 2, pp. 37–42, 2011.

[3] V. K. Nassa, "Wireless communications: Past, present and future," *Dronacharya Research Journal*, vol. 50.

[4] J. Bruno, "Interference reduction in wireless networks," *Computing Research Topics, Computing Sciences Department, Villanova University*, 2007.

[5] A. Liu, Y. Liu, H. Xiang, and W. Luo, "Mimo b-mac interference network optimization under rate constraints by polite water-filling and duality," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 263–276, 2010.

[6] B. Cao, Y. Ge, C. W. Kim, G. Feng, H. P. Tan, and Y. Li, "An experimental study for inter-user interference mitigation in wireless body sensor networks," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3585–3595, 2013.

[7] J. Li, G. Liu, and G. B. Giannakis, "Carrier frequency offset estimation for ofdm-based wlans," *IEEE Signal Processing Letters*, vol. 8, no. 3, pp. 80–82, 2001.

[8] Y. Yao and G. B. Giannakis, "Blind carrier frequency offset estimation in siso, mimo, and multiuser ofdm systems," *IEEE Transactions on Communications*, vol. 53, no. 1, pp. 173–183, 2005.

[9] S. Sharma and K. Thakur, "Carrier frequency offset in ofdm systems," in *2018 2nd International Conference on Inventive Systems and Control (ICISC)*. IEEE, 2018, pp. 369–373.

[10] F. Gini and G. B. Giannakis, "Frequency offset and symbol timing recovery in flat-fading channels: A cyclostationary approach," *IEEE Transactions on Communications*, vol. 46, no. 3, pp. 400–411, 1998.

[11] H. Minn, M. Zeng, and V. K. Bhargava, "On timing offset estimation for ofdm systems," *IEEE Communications letters*, vol. 4, no. 7, pp. 242–244, 2000.

[12] J.-J. Van De Beek, O. Edfors, M. Sandell, S. K. Wilson, and P. O. Borjesson, "On channel estimation in ofdm systems," in *1995 IEEE 45th Vehicular Technology Conference. Countdown to the Wireless Twenty-First Century*, vol. 2. IEEE, 1995, pp. 815–819.

[13] O. Edfors, M. Sandell, J.-J. Van De Beek, S. K. Wilson, and P. O. Börjesson, "Ofdm channel estimation by singular value decomposition," *Communications, IEEE Transactions on*, vol. 46, no. 7, pp. 931–939, 1998.

[14] R. Ganesh, J. J. Kumari *et al.*, "A survey on channel estimation techniques in mimo-ofdm mobile communication systems," *International Journal of Scientific & Engineering Research*, vol. 4, no. 5, pp. 1850–1855, 2013.

[15] H. Xie, F. Gao, and S. Jin, "An overview of low-rank channel estimation for massive mimo systems," *IEEE Access*, vol. 4, pp. 7313–7321, 2016.

[16] X. Li, S. You, L. Chen, A. Liu, and Y. E. Liu, "A new algorithm for the weighted sum rate maximization in mimo interference networks," in *2015 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2015, pp. 147–152.

[17] M. Zhou, X. Huang, Y. Zhou, X. Li, and Y. Liu, "An fpga prototype of dual link algorithm for mimo interference network," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 3669–3673.

[18] M. Zhou, Y. Liu, T. Xia, and X. Huang, "An efficient and scalable hardware architecture for singular value decomposition towards massive mimo communications," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2017, pp. 667–670.

[19] M. Zhou, Z. Feng, Y. Liu, and X. Huang, "An efficient algorithm and hardware architecture for maximum-likelihood based carrier frequency offset estimation in mimo systems," *IEEE Access*, vol. 6, pp. 50 105–50 116, 2018.

[20] M. Zhou, X. Huang, Z. Feng, and Y. Liu, "Coarse frequency offset estimation in mimo systems using neural networks: A solution with higher compatibility," *IEEE Access*, vol. 7, pp. 121 565–121 573, 2019.

[21] X. Cai, M. Zhou, and X. Huang, "Model-based design for software defined radio on an fpga," *IEEE Access*, vol. 5, pp. 8276–8283, 2017.

[22] X. Cai, M. Zhou, T. Xia, W. H. Fong, W.-T. Lee, and X. Huang, "Low-power sdr design on an fpga for intersatellite communications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 11, pp. 2419–2430, 2018.

[23] Z. Feng, X. Li, V. Palacios, P. Mathys, Y. Liu, M. Zhau, X. Huang, and X. Cai, "Software defined radio implementation of the dual link algorithm in tdd mode using usrp e310," in *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.

[24] S. R. Saunders and A. Aragón-Zavala, *Antennas and propagation for wireless communication systems*. John Wiley & Sons, 2007.

[25] K. Pahlavan and A. H. Levesque, *Wireless information networks*. John Wiley & Sons, 2005, vol. 93.

[26] G. Baldini, S. Karanasios, D. Allen, and F. Vergari, "Survey of wireless communication technologies for public safety," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 2, pp. 619–641, 2013.

[27] A. Whitmore, A. Agarwal, and L. Da Xu, "The internet of things: A survey of topics and trends," *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.

[28] J. Moreno, J. M. Riera, L. De Haro, and C. Rodriguez, "A survey on future railway radio communications services: challenges and opportunities," *IEEE Communications Magazine*, vol. 53, no. 10, pp. 62–68, 2015.

[29] R. N. Mitra and D. P. Agrawal, "5g mobile technology: A survey," *ICT Express*, vol. 1, no. 3, pp. 132–137, 2015.

[30] WIKIPEDIA, "Wi-fi," https://en.wikipedia.org/wiki/Wi-Fi, [Online; accessed 10/14/2019].

[31] L. O. Walters and P. S. Kritzinger, "Cellular networks: past, present and future," *Crossroads*, vol. 7, no. 2, p. 5, 2000.

[32] L. Ahlin, J. Zander, and S. Ben Slimane, *Principles of wireless communications*. Studentlitteratur, 2006.

[33] B. P. Lathi, *Modern Digital and Analog Communication Systems 3e Osece*. Oxford University Press, Inc., 1998.

[34] K. V. Cai, J. L. Thomas, and P. L. Lim, "Receiver automatic gain control (agc)," Nov. 30 1993, uS Patent 5,267,272.

[35] G. Turin, "An introduction to matched filters," *IRE transactions on Information theory*, vol. 6, no. 3, pp. 311–329, 1960.

[36] S. B. Weinstein, "The history of orthogonal frequency-division multiplexing [history of communications]," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 26–35, 2009.

[37] N. LaSorte, W. J. Barnes, and H. H. Refai, "The history of orthogonal frequency division multiplexing," in *IEEE GLOBECOM 2008-2008 IEEE Global Telecommunications Conference*. IEEE, 2008, pp. 1–5.

[38] R. W. Heath Jr, *Introduction to Wireless Digital Communication: A Signal Processing Perspective*. Prentice Hall, 2017.

[39] M. L. Roberts, M. A. Temple, R. F. Mills, and R. A. Raines, "Evolution of the air interface of cellular communications systems toward 4g realization," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 1, pp. 2–23, 2006.

[40] N. Tripathi and J. H. Reed, *Cellular communications: a comprehensive and practical guide*. John Wiley & Sons, 2014.

[41] C. L. Jackson, K. J. Negus, and A. Petrick, "History of wireless local area networks (wlans) in the unlicensed bands," *info*, 2009.

[42] J. Kim and I. Lee, "802.11 wlan: history and new enabling mimo techniques for next generation standards," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 134–140, 2015.

[43] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital communication receivers: synchronization, channel estimation, and signal processing*. John Wiley & Sons, Inc., 1997.

[44] M. Rice, *Digital communications: a discrete-time approach*. Prentice Hall, 2009.

[45] C. R. Johnson Jr, W. A. Sethares, and A. G. Klein, *Software receiver design: build your own digital communication system in five easy steps*. Cambridge University Press, 2011.

[46] J. G. Proakis and M. Salehi, *Digital communications*. McGraw-hill New York, 2001, vol. 4.

[47] M. Morelli and U. Mengali, "Carrier-frequency estimation for transmissions over selective channels," *IEEE Transactions on Communications*, vol. 48, no. 9, pp. 1580–1589, 2000.

[48] C.-S. Peng and K.-A. Wen, "Synchronization for carrier frequency offset in wireless lan 802.11 a system," in *The 5th International Symposium on Wireless Personal Multimedia Communications*, vol. 3. IEEE, 2002, pp. 1083–1087.

[49] L. Nasraoui, L. N. Atallah, and M. Siala, "An efficient reduced-complexity two-stage differential sliding correlation approach for ofdm synchronization in the multi-path channel," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2012, pp. 2059–2063.

[50] Y. Yao and T.-S. Ng, "Correlation-based frequency offset estimation in mimo system," in *2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall (IEEE Cat. No. 03CH37484)*, vol. 1. IEEE, 2003, pp. 438–442.

[51] W.-Y. Kuo and M. P. Fitz, "Frequency offset compensation of pilot symbol assisted modulation in frequency flat fading," *IEEE Transactions on Communications*, vol. 45, no. 11, pp. 1412–1416, 1997.

[52] C. Chen, Y. Chen, Y. Han, H.-Q. Lai, and K. R. Liu, "High resolution carrier frequency offset estimation in time-reversal wideband communications," *IEEE Transactions on Communications*, vol. 66, no. 5, pp. 2191–2205, 2017.

[53] Y.-Y. Wang, C.-W. Huang, and W.-W. Chen, "Maximum likelihood carrier frequency offset estimation algorithm with adjustable frequency acquisition region," *Journal of the Franklin Institute*, vol. 355, no. 5, pp. 2978–2985, 2018.

[54] R. Singhai and P. Gupta, "Estimation and analysis of channel distortion and carrier frequency offset (cfo) in mimo-ofdm systems," *International Journal of Advanced Engineering Research and Science*, 2016.

[55] R. Jose and K. Hari, "Joint estimation of synchronization impairments in mimo-ofdm system," in *2012 National Conference on Communications (NCC)*. IEEE, 2012, pp. 1–5.

[56] M.-O. Pun, M. Morelli, and C.-C. Kuo, "Maximum-likelihood synchronization and channel estimation for ofdma uplink transmissions," *IEEE Transactions on Communications*, vol. 54, no. 4, pp. 726–736, 2006.

[57] H. Mehrpouyan, A. A. Nasir, S. D. Blostein, T. Eriksson, G. K. Karagiannidis, and T. Svensson, "Joint estimation of channel and oscillator phase noise in mimo

systems," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4790–4807, 2012.

[58] W. Zhang, Q. Yin, and F. Gao, "Computationally efficient blind estimation of carrier frequency offset for mimo-ofdm systems," *IEEE Transactions on Wireless Communications*, vol. 15, no. 11, pp. 7644–7656, 2016.

[59] W. Zhang, F. Gao, H. Minn, and H.-M. Wang, "Scattered pilots-based frequency synchronization for multiuser ofdm systems with large number of receive antennas," *IEEE Transactions on Communications*, vol. 65, no. 4, pp. 1733–1745, 2017.

[60] W. Zhang, F. Gao, S. Jin, and H. Lin, "Frequency synchronization for uplink massive mimo systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 1, pp. 235–249, 2017.

[61] Y. Ge, W. Zhang, F. Gao, and G. Y. Li, "Frequency synchronization for uplink massive mimo with adaptive mui suppression in angle domain," *IEEE Transactions on Signal Processing*, vol. 67, no. 8, pp. 2143–2158, 2019.

[62] N. Jindal, W. Rhee, S. Vishwanath, S. A. Jafar, and A. Goldsmith, "Sum power iterative water-filling for multi-antenna gaussian broadcast channels," *IEEE transactions on Information Theory*, vol. 51, no. 4, pp. 1570–1580, 2005.

[63] H. Huh, H. C. Papadopoulos, and G. Caire, "Multiuser miso transmitter optimization for intercell interference mitigation," *IEEE Transactions on Signal Processing*, vol. 58, no. 8, pp. 4272–4285, 2010.

[64] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, "An iteratively weighted mmse approach to distributed sum-utility maximization for a mimo interfering broadcast channel," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4331–4340, 2011.

[65] A. Liu, H. Xiang, W. Luo *et al.*, "Duality, polite water-filling, and optimization for mimo b-mac interference networks and itree networks," *arXiv preprint arXiv:1004.2484*, 2010.

[66] A. Liu, Y. Liu, H. Xiang, and W. Luo, "Polite water-filling for weighted sum-rate maximization in mimo b-mac networks under multiple linear constraints," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 834–847, 2011.

[67] T. J. Willink, "Efficient adaptive svd algorithm for mimo applications," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 615–622, 2008.

[68] C. Senning, C. Studer, P. Luethi, and W. Fichtner, "Hardware-efficient steering matrix computation architecture for mimo communication systems," in *2008 IEEE International Symposium on Circuits and Systems*. IEEE, 2008, pp. 304–307.

[69] C.-Z. Zhan, Y.-L. Chen, and A.-Y. Wu, "Iterative superlinear-convergence svd beamforming algorithm and vlsi architecture for mimo-ofdm systems," *IEEE Transactions on Signal Processing*, vol. 60, no. 6, pp. 3264–3277, 2012.

[70] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive mimo for next generation wireless systems," *IEEE communications magazine*, vol. 52, no. 2, pp. 186–195, 2014.

[71] H. Arslan and J. Mitola III, "Cognitive radio, software-defined radio, and adaptive wireless systems," *Wireless Communications and Mobile Computing*, vol. 7, no. 9, pp. 1033–1035, 2007.

[72] M. Dillinger, K. Madani, and N. Alonistioti, *Software defined radio: Architectures, systems and functions*. John Wiley & Sons, 2005.

[73] S. Che, J. Li, J. W. Sheaffer, K. Skadron, and J. Lach, "Accelerating compute-intensive applications with gpus and fpgas," in *2008 Symposium on Application Specific Processors*. IEEE, 2008, pp. 101–107.

[74] L. Boher, R. Rabineau, and M. Helard, "Fpga implementation of an iterative receiver for mimo-ofdm systems," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 6, pp. 857–866, 2008.

[75] X.-T. Vu, N. A. Duc, and T. A. Vu, "16-qam transmitter and receiver design based on fpga," in *2010 Fifth IEEE International Symposium on Electronic Design, Test & Applications*. IEEE, 2010, pp. 95–98.

[76] J. S. Park and T. Ogunfunmi, "Efficient fpga-based implementations of mimo-ofdm physical layer," *Circuits, Systems, and Signal Processing*, vol. 31, no. 4, pp. 1487–1511, 2012.

[77] N. C. Karmakar, Y. Yang, and A. Rahim, "Mimo implementation using fpga," in *Microwave Sleep Apnoea Monitoring*. Springer, 2018, pp. 197–211.

[78] M. Kumm, H. Klingbeil, and P. Zipf, "An fpga-based linear all-digital phase-locked loop," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 9, pp. 2487–2497, 2010.

[79] T. Yoshida, D. Nojima, L. Lanante Jr, Y. Nagao, M. Kurosaki, B. Sai, and H. Ochi, "Rtl design of joint cfo and iq-imbalance compensator for narrow-band wireless system," *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 6, no. 1, pp. 94–101, 2012.

[80] J. Mar, C.-C. Kuo, and S.-H. Chou, "Fpga implementation of sdr based cfo estimation and compensation circuit for ofdm system," *Journal of Signal Processing Systems*, vol. 66, no. 2, pp. 141–146, 2012.

[81] T. H. Pham, S. A. Fahmy, and I. V. McLoughlin, "Efficient integer frequency offset estimation architecture for enhanced ofdm synchronization," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1412–1420, 2015.

[82] J. Luo, Q. Huang, S. Chang, X. Song, and Y. Shang, "High throughput cholesky decomposition based on fpga," in *2013 6th International Congress on Image and Signal Processing (CISP)*, vol. 3. IEEE, 2013, pp. 1649–1653.

[83] P. B. Rapajic and B. S. Vucetic, "Linear adaptive transmitter-receiver structures for asynchronous cdma systems," *European Transactions on Telecommunications*, vol. 6, no. 1, pp. 21–27, 1995.

[84] J. Eilert, D. Wu, and D. Liu, "Efficient complex matrix inversion for mimo software defined radio," in *2007 IEEE International Symposium on Circuits and Systems*. IEEE, 2007, pp. 2610–2613.

[85] Z. Dostál, T. Kozubek, A. Markopoulos, and M. Menšík, "Cholesky decomposition of a positive semidefinite matrix with known kernel," *Applied Mathematics and Computation*, vol. 217, no. 13, pp. 6067–6077, 2011.

[86] F. T. Luk and H. Park, "A proof of convergence for two parallel jacobi svd algorithms," *Computers, IEEE Transactions on*, vol. 38, no. 6, pp. 806–811, 1989.

[87] T.-H. Pham, A. Nallanathan, and Y.-C. Liang, "Joint channel and frequency offset estimation in distributed mimo flat-fading channels," *IEEE Transactions on Wireless Communications*, vol. 7, no. 2, pp. 648–656, 2008.

[88] M. A. Woodbury, "Inverting modified matrices," *Memorandum report*, vol. 42, no. 106, p. 336, 1950.

[89] A. N. D'Andrea, U. Mengali, and R. Reggiannini, "The modified cramer-rao bound and its application to synchronization problems," *IEEE Transactions on Communications*, vol. 42, no. 234, pp. 1391–1399, 1994.

[90] O. Besson and P. Stoica, "On parameter estimation of mimo flat-fading channels with frequency offsets," *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 602–613, 2003.

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

[92] N. D. Hemkumar, "A systolic vlsi architecture for complex svd," Ph.D. dissertation, Rice University, 1991.

[93] M. Athi, S. Zekavat, and A. Struthers, "Real time signal processing of massive sensor arrays via a parallel fast converging svd algorithm: Latency, throughput and resource analysis," *IEEE Sensors Journal*, vol. 14, no. 7, pp. 3899–3911, 2015.

[94] A. Ibrahim, M. Valle, L. Noli, and H. Chible, "Assessment of fpga implementations of one sided jacobi algorithm for singular value decomposition," in *VLSI (ISVLSI), 2015 IEEE Computer Society Annual Symposium on*. IEEE, 2015, pp. 56–61.