Iterative Joint Detection and Decoding of LDPC-Coded V-BLAST Systems

by

Meng-Ying (Brady) Tsai

A thesis submitted to the
Department of Electrical and Computer Engineering
in conformity with the requirements for
the degree of Master of Science (Engineering)

Queen's University
Kingston, Ontario, Canada
July 2008

Abstract

Soft iterative detection and decoding techniques have been shown to be able to achieve near-capacity performance in multiple-antenna systems. To obtain the optimal soft information by marginalization over the entire observation space is intractable; and the current literature is unable to guide us towards the best way to obtain the suboptimal soft information. In this thesis, several existing soft-input soft-output (SISO) detectors, including minimum mean-square error-successive interference cancellation (MMSE-SIC), list sphere decoding (LSD), and Fincke-Pohst maximum-a-posteriori (FPMAP), are examined. Prior research has demonstrated that LSD and FPMAP outperform soft-equalization methods (i.e., MMSE-SIC); however, it is unclear which of the two scheme is superior in terms of performance-complexity trade-off. A comparison is conducted to resolve the matter. In addition, an improved scheme is proposed to modify LSD and FPMAP, providing error performance improvement and a reduction in computational complexity simultaneously. Although list-type detectors such as LSD and FPMAP provide outstanding error performance, issues such as the optimal initial sphere radius, optimal radius update strategy, and their highly variable computational complexity are still unresolved. A new detection scheme is proposed to address the above issues with fixed detection complexity, making the scheme suitable for practical implementation.

Acknowledgments

This dissertation would not have been possible without the support and guidance of many individuals. First, I am forever indebted to Professor Shahram Yousefi, my supervisor, for his continual guidance and invaluable advices over the years. I fully appreciate his encouragement, patience, and continuous support throughout the program. He has been a mentor, a friend and more. He is everything one could ask for in a supervisor.

I would like to thank Professor Brian M. Frank, Professor Saeed Gazor, and Professor Mohammad Zulkernine who served on my thesis committee. I also want to thank the group of High Performance Computing Virtual Laboratory (HPCVL) for providing a great computing environment to run most of the simulations in this thesis. Special thanks goes out to Dr. Hartmut Schmider for his time and assistance. My sincere gratitude to my friends and colleagues at Queen's University, especially to Kitty Wong and Shan Lin for helping me get through the difficult times. Their emotional care and support have made this thesis and many other publications possible.

Lastly, and most importantly, I wish to thank Jung-Kun Tsai and Chi-Yu L. Tsai for being the most wonderful parents. To them, I dedicate this thesis.

Table of Contents

List of Figures

List of Abbreviations

APP	*A-Posteriori* Probability
ARQ	Automatic Repeat-Query
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BF	Bit Flipping
BLAST	Bell Laboratories Layered Space-Time
D-BLAST	Diagonal-Bell Laboratories Layered Space-Time
DLD	Dynamic List Detection
ECC	Error Correcting Code
FPMAP	Fincke-Pohst Maximum-A-Posteriori
i.i.d.	independent and identically distributed
IEEE	Institute of Electrical and Electronics Engineers
IJDD	Iterative Joint Detection and Decoding
IPTV	Internet Protocol Television
ISI	Inter-Symbol Interference
LDPC	Low-Density Parity-Check
LLR	Log-Likelihood Ratio
LOS	Line-of-Sight
LSD	List Sphere Decoding

LTE	Long Term Evolution
MAP	Maximum A-Posteriori
ML	Maximum Likelihood
MLG	Majority-Logic
MMSE	Minimum Mean-Square Error
MMSE-SIC	Minimum Mean-Square Error-Successive Interference Cancellation
OFDM	Orthogonal Frequency-Division Multiplexing
PSK	Phase-Shift Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
SIC	Successive Interference Cancellation
SISO	Soft-Input Soft-Output
SNR	Signal-to-Noise Ratio
SPA	Sum-Product Algorithm
STBC	Space-Time Block Coding
STC	Space-Time Coding
STTC	Space-Time Trellis Coding
V-BLAST	Vertical-Bell Laboratories Layered Space-Time
VoIP	Voice-over-Internet Protocol
WCDMA	Wideband Code Division Multiple Access
WiMAX	Worldwide Interoperability for Microwave Access
WSD	Window-Shifting Detector

Chapter 1

Introduction

In 1844, the first electrical communication system, namely *telegraphy*, was established by Samuel Morse. The telegraph line linked Washington with Baltimore and information was sent using Morse code in which letters of the English alphabet were represented by a sequence of dots and dashes. In 1876, Alexander Graham Bell invented the first telephone system that provided voice-based communication service over hundreds of miles. In 1894, Guglielmo Marconi further developed the first wireless communication system called radiotelegraph and demonstrated that information did not necessary have to be sent through a wire. Over the last century, various wireless communication systems have been developed rapidly and have become a necessary part of our daily life. The demands for high-speed wireless data services have generated a significant amount of research activity in the area of digital communication. Due to limited bandwidth, it is important to increase the communication spectral efficiency to achieve a higher transmission rate with an acceptable probability of error. Multiple-antenna systems, also known as multiple-input multiple-output

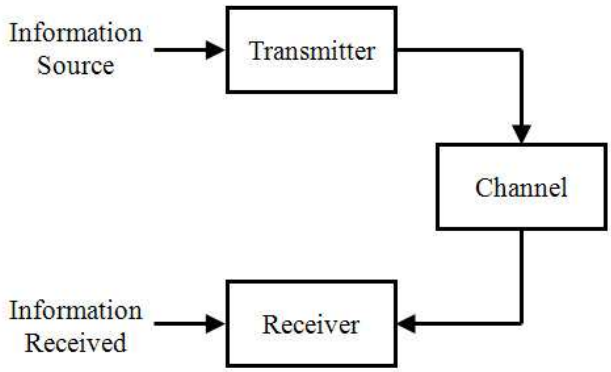


Figure 1.1: Basic communication system.

(MIMO) systems, are currently one of the major research topics in telecommunication as they are capable of transmitting data at high speeds, beyond the scope of any single-antenna system. In this introductory chapter, the basic elements of a communication system will be presented. The discussion will be further extended to wireless communication along with the challenges that arise from the wireless channel. This chapter will conclude with a discussion of a MIMO system and how it utilizes the wireless channel to provide several advantageous properties.

1.1 An Overview of the Digital Communication System

In general, a communication system consists of three basic components, namely, the transmitter, the channel, and the receiver as shown in Figure 1.1. The transmitter converts information signal into a form that is suitable for transmission through the physical channel. The channel is a physical medium used to transport the signal from the transmitter to the receiver such as wirelines, fiber-optic cables, and free

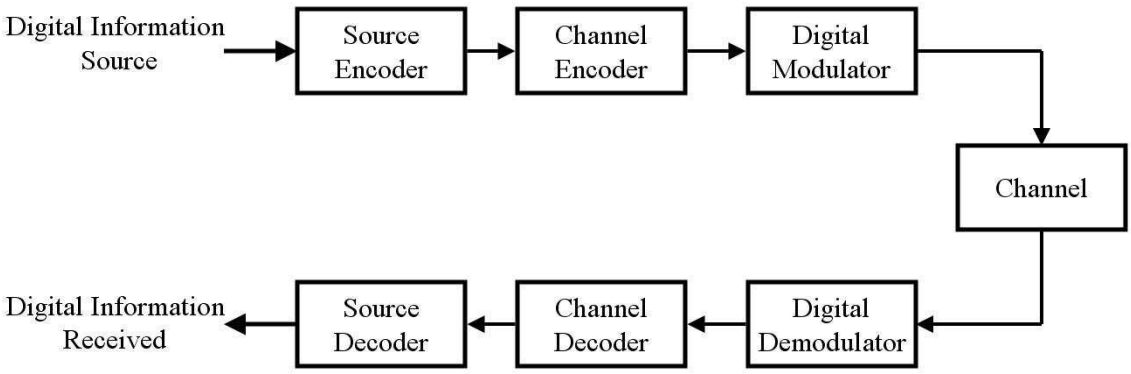


Figure 1.2: The basic elements of a digital communication system.

space. Different channels have different properties and therefore, must be modeled distinctively. The receiver detects and estimates the information signal contained in the received signal. The distortion caused by the channel on the information signal is usually the major obstacle in recovering the information signal.

A typical objective of a communication system is to transmit information signal as efficient as possible while maintaining accuracy. One way of doing so is to transmit the information signal in digital format. Then, the system in Figure 1.1 will be classified as a digital communication system. Digital transmission allows us to remove the redundancy in the information signal and efficiently conserve the channel bandwidth. Also in long-distance transmissions, digital signal is easier to regenerate and helps to eliminate the effect of noise and interference at each regeneration point. Another reason for choosing digital transmission over analog is that digital communication systems are often cheaper to implement.

The basic elements of a single-user digital communication system are shown in Figure 1.2. The source encoder removes the unnecessary redundancy in the information signal, in a process known as data compression. The channel encoder then introduces,

in a controlled manner, some redundancy in the binary information sequence to help the receiver overcome the effects of noise and interference encountered in the channel. Next, the encoded information sequence is passed to the digital modulator in which the binary information sequence is mapped to signal waveforms appropriate for transmission across the channel. After the receiver receives the information sequence, it tries to reconstruct the original information signal by passing the received sequence through the digital demodulator, channel decoder and source decoder in reverse order. The difference between the original information signal and the reconstructed signal is an important performance measure of such system.

1.2 An Overview of the Wireless Channel

In general, when the signal propagates across free space, reflection, diffraction and scattering are the three basic propagation mechanisms impacting the waveform. As a result, signal disperses and travels in multiple paths before being received by the receive antenna. Due to such dispersion and traveling delays, the receive antenna usually receives multiple copies of the transmitted signal with random attenuations, delays and phase shifts. The time difference between the first arriving component and the last one is referred to as the delay spread of the channel. The delay spread might cause the adjacent transmitted symbols to interfere with one another, creating a problem called inter-symbol interference (ISI). A wireless channel in which ISI occurs is referred to as *frequency selective*; otherwise it is referred to as *frequency flat*. In the corresponding frequency domain, frequency selectivity also describes a channel in which the coherence bandwidth, defined as the minimum separation in frequency such that two signals will undergo uncorrelated fading, is smaller than the

bandwidth of the signal. Similarly, frequency flat describes a channel in which the coherence bandwidth is larger than the bandwidth of the signal. Often, equalizers are deployed to compensate for the effects of the ISI. Certain modulation and signaling techniques such as orthogonal frequency-division multiplexing (OFDM) and code division multiple access (CDMA) can also provide robustness to fading.

A channel can also be classified as a *fast* fading or *slow* fading channel. To determine whether a channel is fading fast or slowly, symbol duration is compared with the coherence time of the channel. The coherence time of the channel is defined as a length of time over which the channel stays relatively constant. In a fast fading channel, the symbol duration is greater than the coherence time and in slow fading channel, the symbol duration is less than the coherence time. The coherence time of the channel is related to a quantity known as the doppler spread. When a transmit or receive antenna is moving, the velocity causes a shift in frequency on each multipath signal. This phenomenon is referred to as doppler shift. Since signals traveling along different paths can have different doppler shifts, any transmitted frequency results in a range of received frequencies. This spectral widening is also referred to as doppler spread. In frequency domain, under the condition that the signal bandwidth is much greater than doppler spread, the fading is referred to as slow fading because the effects of doppler spread are negligible. Otherwise, we deal with a fast fading channel.

1.3 The Benefits of Multiple-Antenna Systems

Fading in a wireless channel causes significant performance degradation in a communication system because substantial distortion in the received signal can prevent the receiver from recovering the embedded information properly. The probability of

experiencing a fade in the channel becomes the limiting factor in the system performance. There is also a significant probability that the channel will experience a deep fade and may result in temporary failure of the communication link due to a severe drop in the signal-to-noise ratio (SNR). Diversity techniques such as frequency diversity, temporal diversity, and spatial diversity can be used to improve the system performance by sending replicas of the transmitted signal through different means.

Frequency diversity is a technique that uses different carrier frequencies to achieve diversity. Application of this technique includes orthogonal frequency-division multiplexing (OFDM) and frequency-hopping spread spectrum (FHSS). In order to ensure that different replicas of the signal experience independent fades, the carrier frequencies should be separated by more than the coherence bandwidth of the channel. This will result in inefficient use of bandwidth.

Another method is *temporal diversity* in which multiple versions of the same signal are transmitted at different time instants. One example of this technique is the use of a repetition code. However, more efficient error-correcting coding (ECC), such as low-density parity-check (LDPC) codes and turbo codes, can also be used to reduce the amount of redundancy. This method is especially powerful when these codes utilize sufficiently long codewords. This is because, in such a case, ECC has a good chance of recovering the symbols lost during deep fading. However, if the deep fading lasts for a period of time such that a large fraction of the codeword is affected, error correcting coding might not be able to provide much advantage. In a single-antenna system, deep fading becomes one of the biggest challenges and usually error control methods based on automatic repeat-query (ARQ) are required to maintain the quality of service. Similar to frequency diversity, temporal diversity is not bandwidth efficient

because of the underlying redundancy.

One method of diversity that does not suffer from bandwidth deficiency is *spatial diversity*. It can be achieved by using multiple antennas at the receiver or the transmitter. Each pair of transmit and receive antennas can provide a signal path with independent fading if the antennas are separated far enough (usually more than half of the wavelength). By sending the replicas of the same signal through different paths, multiple independently faded replicas of the signal can be obtained at the receiver end to provide a more reliable detection. For example, in a MIMO system with n_T transmit and n_R receive antennas, there will be $n_T \times n_R$ communication links in total. This is also the maximal *diversity gain* available for such an architecture [1]. Roughly speaking as long as one out of $n_T \times n_R$ links does not undergo deep fade, the communication between the transmitter and the receiver can be sustained. This diversity gain gives the MIMO system a higher probability of surviving through deep fading in the channel. Moreover, a MIMO system has the ability to transmit more than one symbol per channel use; where as in a single antenna system, only one symbol can be transmitted. More precisely, a MIMO system can transmit at most $\min\{n_T, n_R\}$ symbols per channel use asymptotically and therefore, obtains $\min\{n_T, n_R\}$ greater capacity than a single antenna system. This is also known as *spatial multiplexing gain* [1]. In summary, the advantage of a MIMO system can be derived from either, (i) increasing the spatial diversity and/or, (ii) increasing spatial multiplexing gain (rate). Space-time coding refers to modulation and signaling over a MIMO architecture to reap diversity and spatial multiplexing gains simultaneously. Space-time block coding (STBC) [2][3], space-time trellis coding (STTC) [4][5], and Bell labs layered space-time (BLAST) [6] coding are some known examples. More specifically,

STBC such as in Alamouti's code [2] is applied to achieve (i) and BLAST methods such as vertical-BLAST (V-BLAST) [7] are applied to achieve (ii). There are also schemes formulated especially for utilizing the combination of (i) and (ii); however, there is a trade-off relationship limiting the gains [1]. More detailed discussion on these space-time processes will be provided in Section 2.2. All these benefits from a MIMO system are obtained without increasing transmitting power or sacrificing bandwidth efficiency. This is the reason why MIMO is the preferable architecture.

Recently, MIMO technology has been used throughout many advance wireless standards such as Mobile WiMAX (IEEE 802.16e-2005) [8], WiFi (IEEE 802.11n) [9], and the family of High-Speed Packet Access (HSPA) [10] systems. In 2007, Nortel was the first in industry to complete live cellular calls using MIMO technology in each of the major 4G technologies; WiMAX [8], Long Term Evolution (LTE) [11], and Ultra Mobile Broadband (UMB) [12] networks [13]. With recent advances in circuit design and antenna technologies, the circuit design of multiple-antenna is feasible not only on base stations, but also on handheld devices. MIMO technologies can be utilized to provide high-quality services to meet the high demand of real-time applications such as live video streaming, voice-over-Internet protocol (VoIP), and Internet protocol television (IPTV). Given that MIMO technology has been widely used in many 3G standards such as Wideband Code Division Multiple Access (W-CDMA) [11] and HPSA, it is reasonable to suspect that all upcoming 4G systems or beyond will also employ MIMO technology to obtain higher spectral efficiency in the near future.

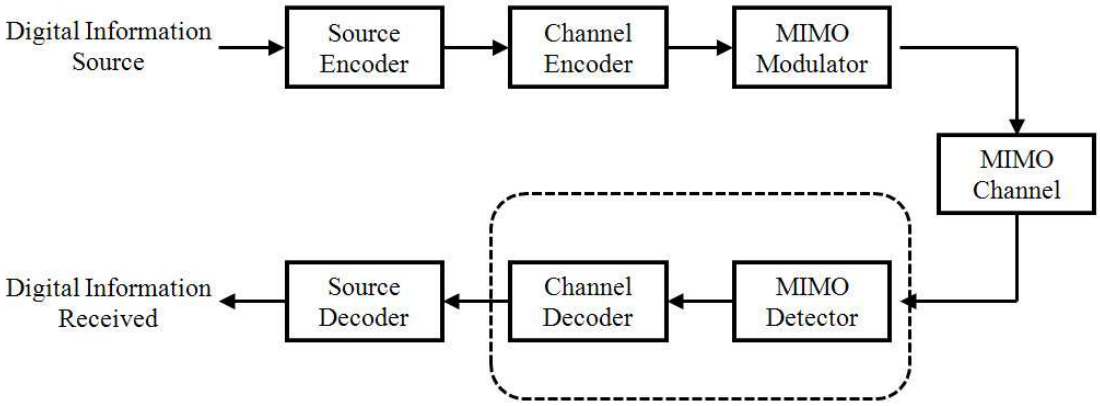


Figure 1.3: A basic MIMO system diagram.

1.4 Incorporating Channel Coding to Multiple - Antenna Systems

It is believed that channel coding is an inseparable part of MIMO communication systems if one targets high-speed applications demanding high quality of service (QoS). Figure 1.3 shows the basic block diagram of a communication system employing channel coding on top of multiple-antenna modulation. In this thesis, our research focuses on the receiver side of the system; in particular, the channel decoder and the MIMO detector. Often, the channel code is referred to as the outer code and the space-time code is referred to as the inner code. The source encoder/decoder is not in the scope of this research and will not be considered.

The discovery of turbo codes [14] and low-density parity-check (LDPC) [15][16] codes for forward error correction made it possible to approach the Shannon capacity limits of many single-antenna systems. Similar to traditional linear block codes, the superior performance of such channel codes relies on large codeword lengths which increases the encoding and decoding complexity. Yet simple decoding methods for both

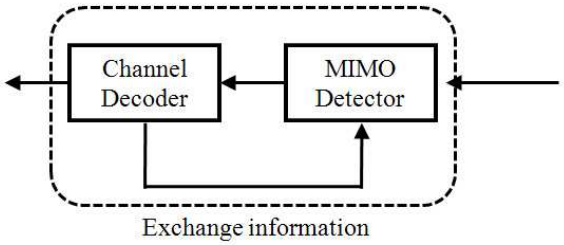


Figure 1.4: Abstract framework of iterative joint detection and decoding in MIMO systems.

turbo and LDPC codes have been investigated using the concept of iterative decoding leading to low-complexity high-performance receivers (i.e., turbo decoding and belief propagation). Roughly speaking, iterative decoding is a generic name for decoding schemes that make use of exchanging soft information between modules, where the soft information is defined as the a-posteriori probability on each information bits in terms of log-likelihood ratio. Therefore, decoders for advance channel codes such as turbo codes and LDPC codes often require soft information as input and output soft information, also known as soft-input soft-output (SISO) decoder. In order to employ such channel codes and accommodate the need of the SISO decoder, MIMO detectors must be capable of producing soft information instead of hard decision output.

An optimal receiver for MIMO systems with an outer channel code will have to perform joint detection and decoding by virtue of interdependencies between consecutive MIMO sessions within each codeword span. However, such a process is prohibitively complex. In traditional receiver design, detection and decoding are done separately as shown in Figure 1.3 to simplify the entire procedure for receiving data. Such simplification suffers from performance degradation because each detection is only capable of detecting a small fraction of a codeword and the correlation between these detected

symbols (bits) and the rest of the codeword is neglected. Fortunately, one can avoid such substantial performance loss by adopting iterative joint detection and decoding (IJDD). This is an iterative process derived directly from the concept of iterative decoding in which soft information is exchanged between the detector and the channel decoder as shown in Fig. 1.4. Through exchanging of information with the channel decoder, the detector can explore the correlation of data bits in a codeword and improve its decision based on the knowledge of the codeword inter-dependencies. This generates a strong motivation to construct detectors that are capable of accepting and producing soft-information for MIMO systems.

Previous research has shown that SISO detectors can be developed using equalization techniques or maximum a-posteriori (MAP) detection approaches. In [17], minimum mean-square error (MMSE) detector is modified to output soft information while taking a-priori (soft) information from the channel decoder into consideration. In [18], MMSE-successive interference cancellation (MMSE-SIC) detector is derived from MMSE detectors with the addition of signal interference cancelation as the pre-process. Similar to the non-iterative MMSE detector, equalization-based detection methods do not provide promising performance although they benefit from low complexity. In [19], a type of modified sphere decoding called list sphere decoding (LSD) is proposed to provide a list of candidates for computing soft information. LSD guarantees that the list of candidates is closer to the received signal than all other candidates among the entire constellation space in terms of Euclidean distance. [20] proposes a new scheme, dubbed Fincke-Pohst Maximum-A-Posteriori (FPMAP), based on sphere decoding to estimate soft information. This new scheme is capable of updating and improving the list of candidates with the consideration of the

a-priori information obtained from the channel decoder. Both LSD and FPMAP are MAP-type receivers.

In this thesis, we choose LDPC codes as the underlying channel code due to their superior error performance and relatively simple decoder. LDPC codes have been recommended for a myriad of communication standards and technologies such as Mobile WiMAX [8] (IEEE 802.16e-2005) and 10GBase-T Ethernet [21] (IEEE 802.3an).

1.5 Contributions of the Thesis

This thesis investigates SISO detectors for MIMO systems. More specifically, we will study their application to iterative joint detection and decoding systems. There are three main contributions:

1) We examine the important existing soft-input soft-output MIMO detectors based on MMSE-SIC, LSD and FPMAP. Prior research has demonstrated that LSD and FPMAP outperform soft-equalization methods [20]; however, it is not clear which scheme is superior in terms of performance-complexity trade-off. This leads us to provide a detailed examination of these two schemes. In particular, we elaborate on how to choose a proper initial radius, an important parameter for both schemes, through simulations based on a rough guideline proposed in [19]. Most importantly, simulations are also carried out to examine their bit-error rate (BER) performance and complexity for LDPC-coded systems.

2) The second contribution is to propose an improved scheme, referred to as Dynamic-List Detector (DLD), that can be adapted to any SISO detector such as LSD and FPMAP where soft-information is obtained via marginalization over a *candidate*

list. This proposed scheme is developed based on dynamically adjusting the detector's parameters according to statistical properties of the channel noise and fading resulting in a better performance-complexity trade-off. Despite the improved performance given by sphere-decoding-based detection schemes (i.e., LSD and FPMAP), an inherent problem of sphere decoding still exists: detection complexity can fluctuate significantly depending on the system parameters and the statistics of the channel. This is a great disadvantage in the system implementation process leading to our third contribution.

3) The third contribution of the thesis is to provide a novel list-type detector referred to as Window-Shifting Detector (WSD) with a reduction in complexity to resolve the problem mentioned earlier. This new list-type detector integrates an entirely different structure from a variety of modified sphere decoding detectors and is very flexible in terms of performance-complexity adjustment, making it suitable for different system requirements. Most importantly, this proposed scheme has fixed detection complexity regardless of system parameters and channel statistics. Lastly, simulation results will be provided to demonstrate the BER performance and computational complexity of this particular scheme relative to those of the existing schemes.

1.6 Outline of the Thesis

The remainder of this thesis is organized as follows. Chapter 2 presents the channel-coded MIMO system model as well as the optimal and sub-optimal receivers for such a system. In particular, we discuss an iterative joint detection and decoding receiver in detail followed by relevant background material on space-time coding and low-density parity-check codes. In Chapter 3, some important existing soft-input soft-output

MIMO detectors are investigated. To the best of our knowledge, a comparative study of these receivers in terms of error performance and complexity has not been provided in the literature previously. Simulations are carried out to compare the schemes and parameters affecting the behavior of the detector are discussed. In Chapter 4, an improved scheme, referred to as the Dynamic-List Detector (DLD), is proposed to modify LSD and FPMAP, providing error performance improvement and a reduction in computational complexity simultaneously. In addition, an new detection scheme referred to as the Window-Shifting Detector (WSD) with fixed detection complexity is presented along with the corresponding simulation results. Chapter 5 concludes the thesis followed by directions for future research.

Chapter 2

Fundamentals

In this chapter, starting from the general block diagram of a channel-coded MIMO system, we discuss the components of the system used in this thesis in details. In particular, space-time coding and low-density parity-check (LDPC) codes will be discussed following discussions on optimal and suboptimal receivers.

2.1 MIMO System Model

Let us consider a single-user point-to-point coded MIMO system with n_T transmit and n_R receive antennas. The system block diagram is shown in Figure 2.1. In this section, we will examine the transmitter, the MIMO channel, and the receiver respectively to provide the general framework of a coded MIMO system. We will focus our discussion on the receiver and how iterative joint detection and decoding technique fits into this system as it is the focal point of this thesis.

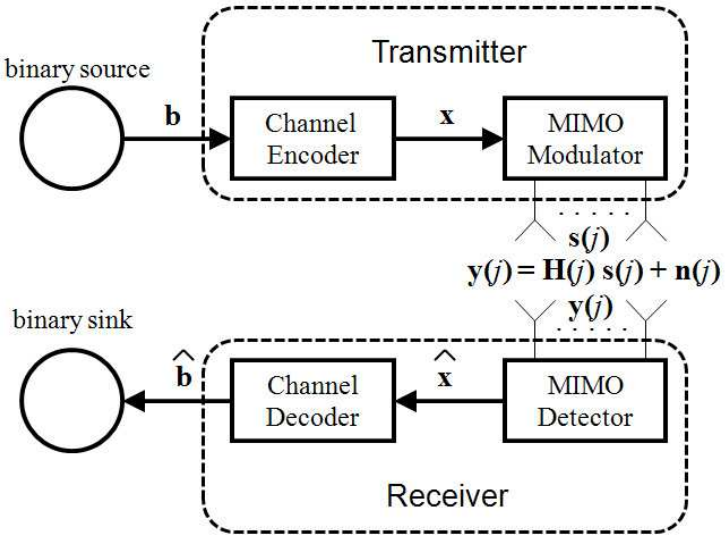


Figure 2.1: MIMO system model with a serial concatenated receiver.

2.1.1 The Transmitter

The transmitter is responsible for performing the necessary processes to ensure the accuracy and efficiency of information delivery. Before the input binary vector $\mathbf{b} \in \{0,1\}^l$ is transmitted, it is encoded by an outer binary channel code of rate $R = l/m$ to produce codeword $\mathbf{x}$. Note that in Figure 2.1 above, the transmission of a single codeword $\mathbf{x}$ of the channel code is depicted. Later on, interleavers might be used to scramble multiple consecutive codewords of the channel code before MIMO modulation. The encoded bits $\mathbf{x}$ of size m are modulated using digital modulation techniques (i.e., phase-shift keying (PSK) or quadrature amplitude modulation (QAM)) and distributed to multiple transmission sessions. Within the jth MIMO transmission session, the symbols are uniquely mapped to a space-time transmission matrix $\mathbf{s}(j)$ from the space-time codebook $\mathcal{C}_{ST}$. Now assuming that a Q-ary constellation S is used; each resulting constellation symbol represents $p_m = \log_2 Q$ bits. Note that Q is chosen to be a power of 2. In general, we assume that n_s information symbols

$\{s_i\}_{i=1}^{n_s}$ are embedded in each transmission matrix $\mathbf{s}(j)$ communicated from n_T transmit antennas over T time slots. As a result, each transmission matrix is valued at $n_s\, p_m$ information bits over T time slots.

2.1.2 The MIMO Channel

The MIMO channel during the jth transmission session is represented by a channel matrix $\mathbf{H}(j)$. Depending on the fading properties of the channel, the channel can be modeled using various statistical distributions. Two most common fading distributions are Rayleigh and Ricean. Rayleigh distribution is suitable for the environments without a line-of-sight (LOS) component; otherwise, Ricean is preferred [22]. In this thesis, we consider a discrete-time Rayleigh block-fading MIMO channel model, where the complete channel state information (CSI) is known to the receiver but not to the transmitter. Assuming that fading coefficients do not change during the course of each MIMO transmission session, $\mathbf{H}(j)$ is an $n_R \times n_T$ matrix whose elements are independent and identically distributed (i.i.d.) complex circular-symmetric Gaussian random variables with zero mean and variance σ_n^2. In general, the multiple paths may be spatially correlated but we assume the channel is spatially white. Thus, the components of $\mathbf{H}(j)$ are independent of one another. During the jth transmission session, the input-output relation for the channel can be represented as

$$\mathbf{y}(j) = \mathbf{H}(j)\mathbf{s}(j) + \mathbf{n}(j) \tag{2.1}$$

where $\mathbf{y}(j)$ is the $n_R \times T$ received signal matrix for the jth MIMO transmission session and $\mathbf{n}(j)$ is the corresponding $n_R \times T$ additive noise vector composed of i.i.d. circular-symmetric complex Gaussian entries with mean of zero and unit variance.

2.1.3 The Optimal Receiver

The objective of the receiver is to estimate the transmitted information $\mathbf{b}$ while minimizing the corresponding probability of error. It is important to note that $m > n_s\, p_m$, that is, due to the use of large codeword length of the outer channel code as discussed previously, each MIMO transmission only delivers a small fraction of the entire codeword. As a result, an optimal receiver will have to estimate the transmitted information $\mathbf{b}$ based on the observation of several MIMO transmissions. In general, this estimation can be evaluated based on minimizing either frame, symbol, or bit error rate. If we assume that the system requires $n_b = \frac{m}{n_s\, p_m}$ MIMO channel uses in order to complete transmitting an information sequence $\mathbf{b}$, the best estimate the receiver can make to minimize the probability of bit error based on Bayesian detection theory is to maximize the bit a-posteriori probabilities (APPs):

$$\hat{b}_i = \arg \max_{b_i \in \{0,1\}} p(b_i | \{\mathbf{y}(j)\}_{j=1}^{n_b}, \{\mathbf{H}(j)\}_{j=1}^{n_b}), \qquad 1 \leq i \leq m. \tag{2.2}$$

Similarly, the receiver can minimize the probability of symbol error by maximizing the symbol APPs:

$$\hat{s}_i = \arg \max_{s_i \in S} p(s_i | \{\mathbf{y}(j)\}_{j=1}^{n_b}, \{\mathbf{H}(j)\}_{j=1}^{n_b}), \qquad 1 \leq i \leq \frac{m}{p_m} = n_s\, n_b. \tag{2.3}$$

Then the estimate of the information sequence $\mathbf{b}$ is obtained by applying the inverse mapping of channel encoder and MIMO modulator combined. To minimize the probability of frame (codeword) error one is to maximize the frame APPs:

$$\hat{\mathbf{b}} = \arg \max_{\mathbf{b} \in \{0,1\}^l} p(\mathbf{b} | \{\mathbf{y}(j)\}_{j=1}^{n_b}, \{\mathbf{H}(j)\}_{j=1}^{n_b}). \tag{2.4}$$

To perform the above, joint detection and decoding is implemented: a prohibitively complex marginalization process which can not be considered for systems under study in this thesis.

2.1.4 The Sub-optimal Receivers

Given that implementing the optimal receiver is infeasible, sub-optimal receivers are the only practical solution for the coded MIMO system studied here. Let us consider two different approaches in detail.

2.1.4.1 Serial Concatenated Receivers

One obvious sub-optimal solution is to consider the MIMO detector and the channel decoder independently to simplify the receiver's complexity. This is referred to as the serial concatenated scheme and is shown in Figure 2.1. As discussed in Section 1.4, such simplification suffers from performance degradation because each detection is only capable of detecting a small fraction of a codeword and the correlation between these detected symbols and the rest of the codeword is neglected. When the MIMO detector is considered independently, optimal detection is equivalent to picking the $\mathbf{s}(j)$ from the equiprobable space-time codebook $\mathcal{C}_{ST}$ for every jth transmission session that maximizes the a-posteriori probability:

$$\hat{\mathbf{s}}(j) = \arg \max_{\mathbf{s}(j) \in \mathcal{C}_{ST}} p(\mathbf{s}(j)|\mathbf{y}(j)) = \arg \max_{\mathbf{s}(j) \in \mathcal{C}_{ST}} p(\mathbf{y}(j)|\mathbf{s}(j)). \tag{2.5}$$

This is also known as maximum likelihood (ML) detection which is locally optimal for equiprobable space-time codebooks. However, due to the separation of the detector and the decoder, even if both perform their optimal task locally, the overall performance of the receiver is still far from the optimal receiver described in Section 2.1.3. Next, we introduce a similar scheme in which the detection and decoding are done separately just like in this system, but they are repeated in an iterative fashion to give us a better performance-complexity trade-off.

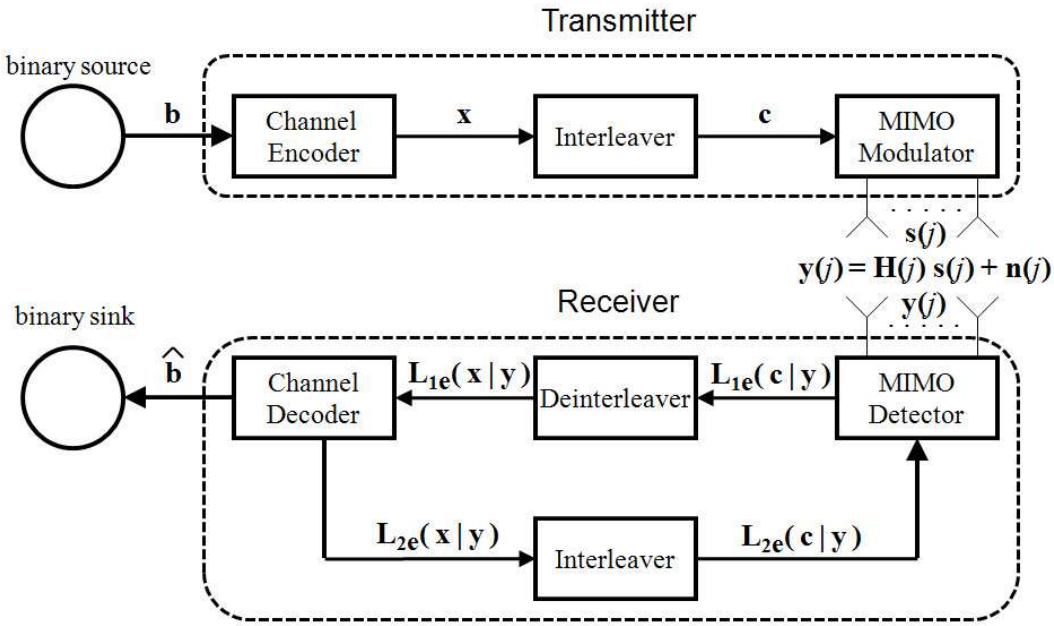


Figure 2.2: Iterative Joint Detection and Decoding System Model.

2.1.4.2 Iterative Joint Detection and Decoding Receivers

Iterative joint detection and decoding scheme works in an iterative fashion to approximate the optimal joint detection and decoding. A detailed block diagram of a coded MIMO system model utilizing such scheme is presented in Figure 2.2. This scheme allows the MIMO detector and the channel decoder to exchange soft reliability information in an iterative fashion until desired performance is achieved. Interleaver/deinterleaver are also employed to ensure the inputs to SISO decoder or detector are as decorrelated as possible. This changes the MIMO transmitter slightly such that q consecutive codewords $\mathbf{x}$ are first scrambled by the random interleaver before being sent to the MIMO modulator. The resulting interleaving depth $q \cdot m$ is a very important parameter affecting the overall system performance. More discussion on this parameter will be provided in Section 2.1.6.

As discussed previously in Section 2.1.3, estimation of the transmitted information vector $\mathbf{b}$ can be done based on minimizing frame, symbol, or bit error rates. For the rest of the thesis, we will focus on minimizing the probability of bit error. On the receiver side, MIMO detector performs detections on each individual transmission session with the consideration of the a-priori probabilities $L_{2e}(\mathbf{c}|\mathbf{y})$ from channel decoder. After $n_k = \frac{q \cdot m}{n_s \cdot p_m}$ sessions are processed, their corresponding bit-level soft information $L_{1e}(\mathbf{c}|\mathbf{y})$ is sent to the deinterleaver and then to the channel decoder. Note that it is possible for the MIMO detector to send out partial soft information before completing n_k detection sessions to improve efficiency as soon as it obtains enough soft information for any codeword $\mathbf{x}$. For simplicity, we will, however, not consider this scenario. In addition, the transmission session index of the received signal matrix $\mathbf{y}$ will be dropped after being processed by the MIMO detector. In general, the soft information is expressed in the form of a log-likelihood ratio (LLR) as

$$L_1(c_i|\mathbf{y}) = \log \frac{p[c_i = 1|\mathbf{y}]}{p[c_i = 0|\mathbf{y}]}, \qquad i = 1, 2, ..., n_s\, p_m \tag{2.6}$$

where $p[c_i = 1|\mathbf{y}]$ and $p[c_i = 0|\mathbf{y}]$ are the a-posteriori probabilities (APPs) of the ith bit of $\mathbf{c}$. We later show that the log-likelihood ratio can be decomposed into two parts: the a-priori information and the extrinsic information, where the a-priori information is referring to the information coming from the channel decoder. In order to prevent the channel decoder from absorbing information originated from itself, MIMO detector only sends extrinsic part of the soft information to the channel decoder [23]. There are several ways to estimate the bit-level soft information depending on the MIMO detector employed in the system. The most accurate method is to estimate the bit-level soft information based on maximizing the a-posteriori probability (APP); more disccussion on this will be provided in Section 2.1.5. Note that if no a-priori

probabilities $L_{2e}(c_i|\mathbf{y})$ is available in the first iteration, $L_{2e}(c_i|\mathbf{y})$ can simply be set to zero.

The structure of channel decoder varies depending on the choice of error-correcting code employed. The channel decoder takes in soft estimation $L_{1e}(x_n|\mathbf{y})$, $n = \{1, 2, \cdots, m\}$, from the MIMO detector and produces another set of soft information $L_{2e}(x_n|\mathbf{y})$ based on the structure of the error-correcting code. These extrinsic information are interleaved and fed back to the MIMO detector for the next round of detection. The MIMO detector will again use the same received signal matrix $\mathbf{y}$ and the new a-priori information from the channel decoder to update its soft estimation. The term iterative comes from the fact that both the MIMO detector and channel decoder update their estimation repeatedly in a *turbo* manner until a stopping criterion is met. Then, the estimation of the transmitted information vector $\hat{\mathbf{b}}$ can be generated.

2.1.5 Bitwise Maximum A-Posteriori Detection

Maximum likelihood detection minimizes the block or sequence error probability; however it does not necessarily minimize the bit or symbol error probability. Maximizing the a-posteriori probability (APP) for a given bit minimizes the probability of making an error on that bit. In many iterative decoding algorithms, the a-posteriori probability is expressed in a "soft" notation called LLR. This notation provides a more convenient way to capture the hard decision using the sign of LLR and the reliability of the decision using the magnitude of LLR in one single value. As mentioned in Section 2.1.4.2, soft information calculated by MAP detectors are expressed as follows

$$L_1(c_i|\mathbf{y}) = \log \frac{p[c_i = 1|\mathbf{y}]}{p[c_i = 0|\mathbf{y}]}, \qquad i = 1, 2, ..., n_s\, p_m. \tag{2.7}$$

Applying Bayes' rule,

$$\begin{aligned} L_1(c_i|\mathbf{y}) &= \log \frac{\sum_{\mathbf{c}:c_i=1} p(\mathbf{y}|\mathbf{c})p(\mathbf{c})}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c})p(\mathbf{c})} \\ &= \underbrace{\log \frac{p(c_i=1)}{p(c_i=0)}}_{L_{1a}(c_i|\mathbf{y})} + \underbrace{\log \frac{\sum_{\mathbf{c}:c_i=1} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} p(c_j)}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} p(c_j)}}_{L_{1e}(c_i|\mathbf{y})} \end{aligned} \tag{2.8}$$

where $L_{1a}(c_i|\mathbf{y})$ and $L_{1e}(c_i|\mathbf{y})$ denote the a-priori and the extrinsic parts of the soft information on bit c_i, respectively. Again, the a-priori information is referring to the information coming from the channel decoder. In any iterative decoding process including the iterative joint detection and decoding system described in Section 2.1.4.2, only extrinsic part of the soft information should be exchanged between modules (i.e., the MIMO detector and the channel decoder). This is because we want to ensure that each module makes decision (estimation) without considering its previous result to prevent error amplification. As a result, each module should only be fed with information which does not originated from itself.

Up to this point, we have assumed that $\mathbf{s}(j)$ and $\mathbf{y}(j)$ are general space-time transmitting and receiving matrices, respectively. However, we will only consider spatial multiplexing in the form of vertical-BLAST (V-BLAST) for the rest of this thesis. This allows us to drop all the transmission session indices. Therefore, from this point on, we assume that $\mathbf{s}$ is an $n_T \times 1$ transmitted signal vector, $\mathbf{y}$ is an $n_R \times 1$ received signal vector, and $\mathbf{H}$ is an $n_R \times n_T$ channel gain matrix. If the encoded bit vector $\mathbf{c}$ is uniquely mapped to symbol vector $\mathbf{s}$ and symbols are transmitted through AWGN, the extrinsic part of Equation (2.8) can be expressed as

$$L_{1e}(c_i|\mathbf{y}) = \log \left(\frac{\sum_{\mathbf{s}:c_i=1} e^{-\frac{1}{2\sigma_n^2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|^2 + \sum_{j,j\neq i} \log p(s_j)}}{\sum_{\mathbf{s}:c_i=0} e^{-\frac{1}{2\sigma_n^2}\|\mathbf{y}-\mathbf{H}\mathbf{s}\|^2 + \sum_{j,j\neq i} \log p(s_j)}} \right) \tag{2.9}$$

where σ_n^2 is the additive noise variance. Furthermore, Equation (2.9) can be efficiently computed using Max-log approximation and becomes [19]

$$\begin{aligned} L_{1e}(c_i|\mathbf{y}) \;\cong\; & \frac{1}{2}\max_{\mathbf{s}:c_i=1}\left\{-\frac{1}{\sigma_n^2}\|\mathbf{y}-\mathbf{Hs}\|^2+\mathbf{c}_{[i]}^T\cdot\mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right\}- \\ & \frac{1}{2}\max_{\mathbf{s}:c_i=0}\left\{-\frac{1}{\sigma_n^2}\|\mathbf{y}-\mathbf{Hs}\|^2+\mathbf{c}_{[i]}^T\cdot\mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right\} \end{aligned} \tag{2.10}$$

where $\mathbf{c}_{[i]}$ denotes the subvector of $\mathbf{c}$ obtained by omitting its ith element c_i, and $\mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})$ denotes the subvector of $\mathbf{L}_{2e}(\mathbf{c}|\mathbf{y})$ omitting its ith element $\mathbf{L}_{2e}(c_i|\mathbf{y})$. Note that $\mathbf{L}_{2e}(\mathbf{c}|\mathbf{y})$ is the soft information provided by the channel decoder and the computation of this information depends on the decoder structure. More information on the channel decoder will be discussed in Section 2.3. Detailed derivation of (2.10) can be found in the Appendix A.

Optimal MAP detection can be achieved by examining every possible codeword $\mathbf{c}$ combinations that could be transmitted. In another word, for every possible codeword $\mathbf{c}$, those with $c_i = 1$ will contribute to the numerator of Equation (2.9) and those with $c_i = 0$ will contribute to the denominator. Such computation can be prohibitively complex; however, simplification can be accomplished by reducing the search space. This is exactly where LSD and FPMAP come into place providing a small but reliable search space.

2.1.6 Interleaving Performance Gain

An interleaver can be thought of as a pseudo-random block scrambler in this thesis. A pseudo-random block interleaver is a variation of a block interleaver in which data is written in memory in sequential order and read in a pseudo-random order. In iterative decoding, it decorrelates the inputs to the two SISO modules so that an

iterative algorithm based on information exchange between the two modules can be applied. The main purpose of decorrelating the inputs is to ensure that after correction of some of the errors in one module, some of the remaining errors could still be correctable in the second module. In a sense, this minimizes burst error effect. In particular, interleaving is performed between the channel decoder and the SISO MIMO detector with similar purpose in this thesis. It is shown in [24] that the interleaver size is inversely proportional to the bit error probability and is crucial for the overall performance gain. In this thesis, however, we assume that interleaver size is fixed as our main focus is on the performance of the SISO MIMO detectors. It is also important to note that we assume the interleaver depth is greater than a codeword in our system; therefore, higher delay is expected.

2.2 MIMO Space-Time Processing

As discussed previously, the benefits of employing multiple antennas in wireless communication systems come from the use of spatial diversity and a larger channel capacity. These benefits are obtained through space-time processing which are usually designed with a trade-off among error performance, transmission rate, and decoding complexity. In general, there are two approaches designed for multiple-antenna systems to improve the system performance.

2.2.1 Space-Time Coding (STC)

This approach is performed by adding controlled redundancies in both spatial and temporal domains. These redundancies provide correlations in the transmitted signals

to increase the reliability of data transmission. In a multipath environment, MIMO systems employing space-time coding can be used to combat the effect of multipath fading and obtain better error performance. A simple example of space-time code is the Alamouti's Code [2]. This code uses two transmit and two receive antennas to send two symbols over two timeslots. This results in an average transmission rate of one symbol per timeslot. While this approach is not the focus of this thesis, more details on it can be found in [25].

2.2.2 Bell Labs Spatial Multiplexing Systems

This approach, also known as Bell Laboratories layered space-time (BLAST), is performed by sending independent symbols from each transmit antenna. There are two main BLAST architectures: diagonal-BLAST (D-BLAST) and vertical-BLAST (V-BLAST).

2.2.2.1 Diagonal-BLAST (D-BLAST)

In this architecture, the input data stream is split into n_T substreams by a serial-to-parallel (s/p) converter as shown in Figure 2.3 (b). Then, these substream, also known as layers, are cyclically shifted before transmission. Figure 2.4 (b) illustrates an example of a four-transmit-antenna system with each substream layered diagonally across antennas and time. Note that each symbol transmitted is denoted by $s_i(j)$, where i ($i = A, B, C, D$) refers to layer index and j ($j = 1, 2, \cdots$) refers to the symbol index in each layer. The main reason for transmitting layers from different antennas is to reduce the effect of deep fade by introducing transmit diversity. For instance, if the path gains from one of the transmit antennas are in deep fade, every symbol

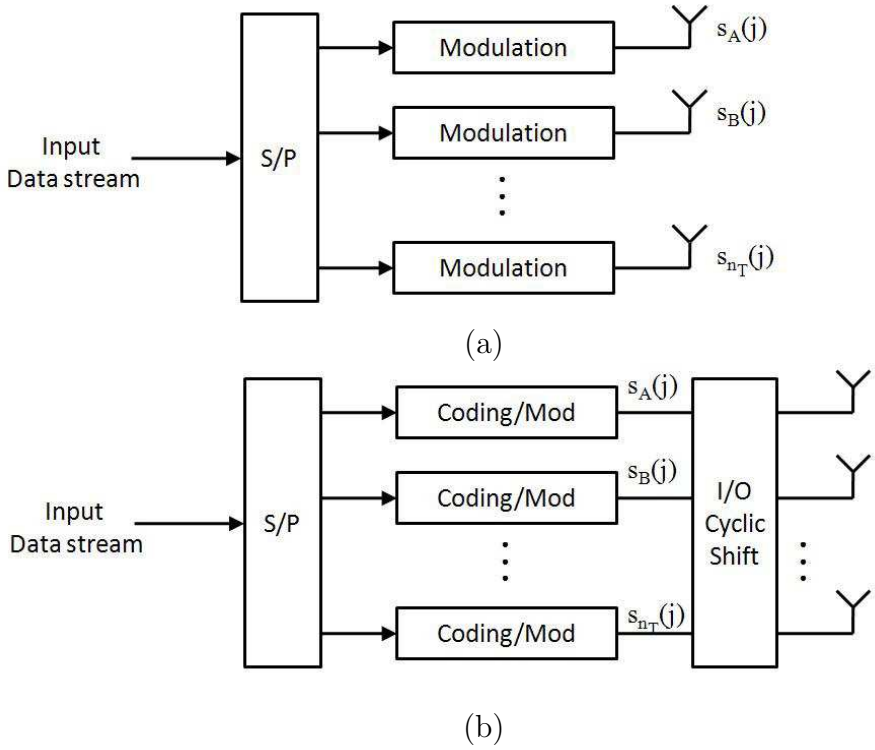
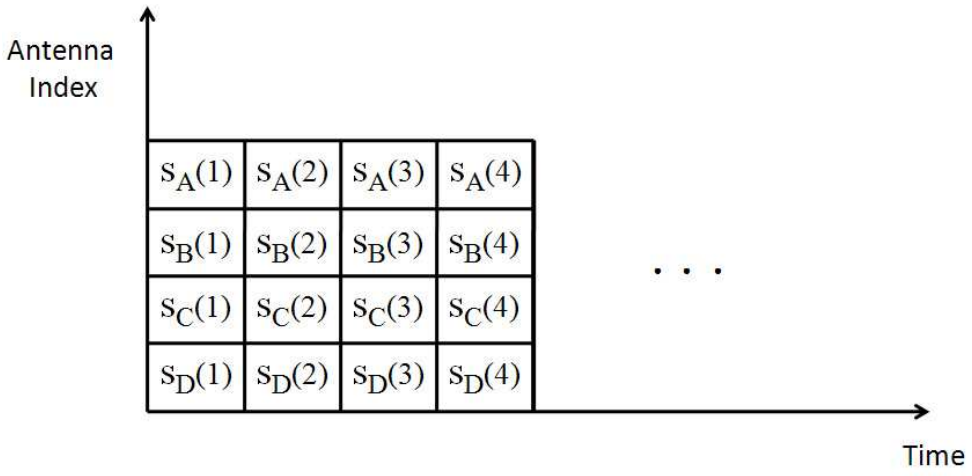
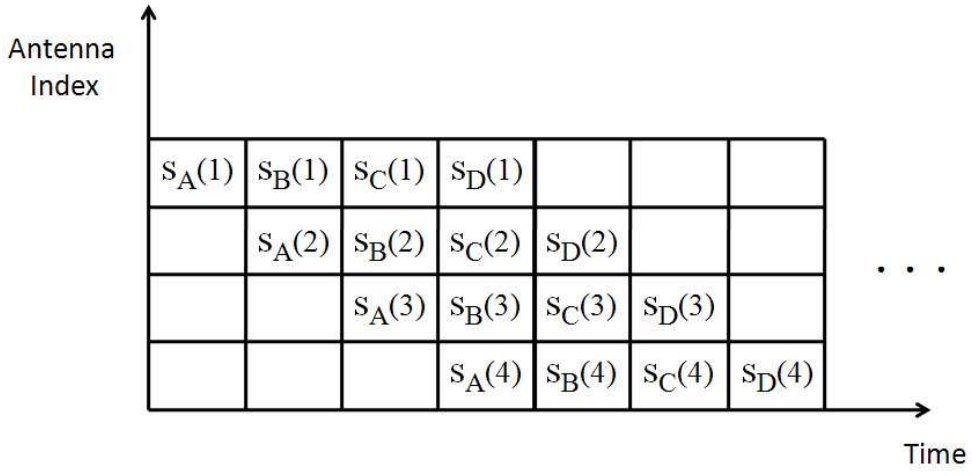


Figure 2.3: (a) V-BLAST encoder (b) D-BLAST encoder.



(a)



(b)

Figure 2.4: (a) V-BLAST signal transmission assignment (b) D-BLAST signal transmission assignment.

sent from that particular antenna will be affected and the performance of the system will be significantly degraded. With the cyclic shift feature to distribute the symbols to different antennas, the effect of deep fade is spread out among each layer and it is easier to recover layers with the additional help of an error correcting code. In fact, the role of this cyclic shifting in combating deep fading is similar to the job of interleaving to overcome burst errors. However, this additional feature does come with a price. D-BLAST suffers from high encoding/decoding complexity and also a reduction in the throughput of the system. This reduction is caused by the inefficient use of the first $(n_T - 1)$ and last $(n_T - 1)$ time instants as shown in Figure 2.4 (b).

2.2.2.2 Vertical-BLAST (V-BLAST)

As shown in Figure 2.3 (b), the encoder of D-BLAST is very similar to that of V-BLAST as depicted in Figure 2.3 (a). Due to the simple vertical layering in V-BLAST, the encoding and decoding of this scheme is easier; making it the most common form of spatial multiplexing: signals are transmitted from different antenna. In this architecture, the input data stream is split into n_T substreams by a serial-to-parallel (s/p) converter as shown in Figure 2.3 (a). Then, each substream is modulated and transmitted from the corresponding transmit antenna. It is also possible to use error correcting coding in each substream to improve the performance, but uncoded substreams are often assumed in V-BLAST. Figure 2.4 (a) illustrates an example of a four transmit antenna system with vertical layering of the symbols, where each row of symbols is aligned with its corresponding transmit antenna and each column represents the transmitting time instants. As demonstrated, V-BLAST can transmit n_T symbols per channel use. In this thesis, we chose V-BLAST as our space-time

scheme to fully exploit the spatial multiplexing gain in the MIMO channel. In order to compensate for the loss in error performance, a powerful error correcting code is needed. Given the superior performance of LDPC codes [15][16], it makes V-BLAST and LDPC the perfect combination for this thesis. Moreover, we will also observe how this combination enhances the error floor of the system performance to lower BER.

2.3 Low-Density Parity-Check Codes

Low-density parity-check (LDPC) codes are a class of linear block codes which provide channel-capacity-approaching performance. They were first discovered by Gallager in the early 1960's but unfortunately were neglected for almost 20 years due to the lack of practical and feasible decoding techniques. In 1981, Tanner generalized LDPC codes and introduced a graphical representation of LDPC codes. For the next following 14 years, coding theorists again ignored this remarkable discovery until the late 1990's, when they realized the significance of LDPC codes and how the graphical representation of LDPC codes plays a major role in the decoding process.

Similar to ordinary parity-check codes, an LDPC code is defined by its parity-check matrix $\mathbf{A}$. For the binary cases within this thesis, codeword $\mathbf{x} = (x_1, x_2, x_3, \cdots, x_m) \in F_2^m$ [1] belongs the null-space of $\mathbf{A}$ (i.e., $\mathbf{x}$ is a valid codeword if and only if $\mathbf{x} \cdot \mathbf{A}^T = 0$). A *regular* binary LDPC code is a linear block code with 2^k codewords and each codeword has a length of m with the following properties [26]:

1. each row of $\mathbf{A}$ contains exactly w_r 1's,

[1] F_2 is the binary field

2. each column of $\mathbf{A}$ contains exactly w_c 1's,

3. the number of 1's in common between any two columns is no greater than 1,

4. both w_r and w_c are small in comparison with the length of the code and the number of rows in $\mathbf{A}$.

The key difference between LDPC codes and ordinary parity-check codes is that the $\mathbf{A}$ of LDPC codes has a low density of 1's. If $\mathbf{A}$ is low density and the number of 1's in each column and row is not constant, the code is referred to as an *irregular* LDPC code. It is easy to determinate whether a LDPC code is *regular* or *irregular* through its graphical representation. More discussion on graphical representations will be given in Section 2.3.1.

The performance of an LDPC code is highly influenced by the codeword length; the larger the codeword length, the better the performance in general. However, as the codeword length increases, the encoding and decoding processes get more complex as well.

The goal of channel coding has always been to bring the performance close to the Shannon limit with moderate encoding and decoding complexity. Turbo codes were discussed in 1993 by Claude Berrou [14]. However, LDPC codes are the preferred error correcting scheme in this thesis and many standards such as Mobile WiMAX [8] (IEEE 802.16e-2005) and 10GBase-T Ethernet [21] (IEEE 802.3an). Due to efficient use of iterative decoding on appropriate graphical representation, LDPC codes are decodable at moderate costs. Yet, unlike traditional algebraic codes, these codes are hard to encode in general; unless specific combinational or algebraic structure is embedded in the code to allow for easy generation of parity bits. In addition, the preferred method for practical implementation is the one that provides systematic

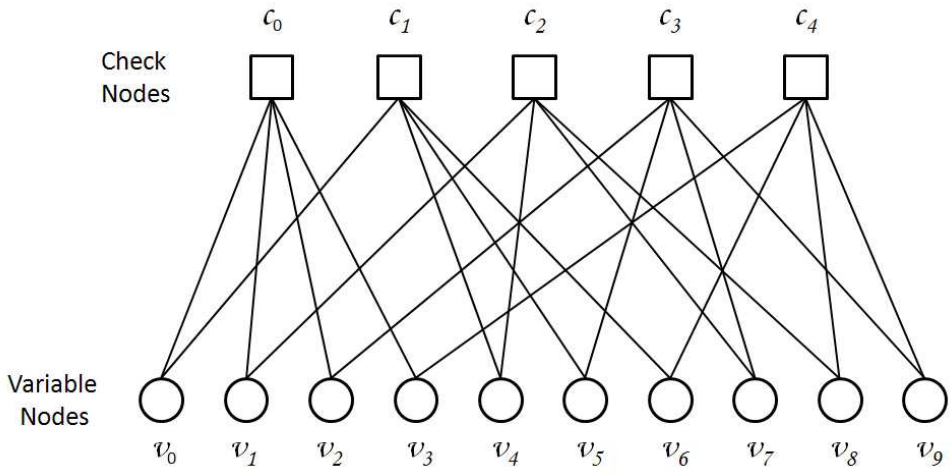


Figure 2.5: A Tanner Graph.

encoding. In other words, the up-front of the resulting codeword coordinates are exactly the uncoded data bits and the rest of the codeword coordinates are the parity bits. In this thesis, we will be considering LDPC codes proposed in Mobile WiMAX standard [8] (IEEE 802.16e-2005). More discussion on this particular LDPC will be provided in Section 2.3.2-2.3.3.

2.3.1 Graphical Representation

Tanner showed that LDPC codes can be represented by a bipartite graph: a Tanner graph. A bipartite graph consists of nodes and edges where nodes are categorized into two types: for instance the variable nodes and the check nodes and edges are only allowed to connect two nodes of different type. The Tanner graph of an LDPC code is constructed according to its parity-check matrix such that check node j is connected to variable node i when the ith row and jth column element a_{ij} in $\mathbf{A}$ is a 1. Thus for a *regular* LDPC code, w_r represents the number of edges connected to each check node and w_c represents the number of edges connected to each variable node: these

are the check and variable degrees, respectively. For *irregular* LDPC codes, the check and variable degrees are not fixed. As an example, consider an $\mathbf{A}$ matrix of a $(10, 5)$ linear block code with $w_c = 2$ and $w_r = 4$ as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}. \tag{2.11}$$

The corresponding Tanner graph is shown in Figure 2.5. Note that variable nodes v_0,v_1,v_2 and v_3 are connected to check node c_0 because $a_{00} = a_{01} = a_{02} = a_{03} = 1$. The Tanner graph of an LDPC code is similar to the trellis of a convolutional code. Not only does it provide a complete representation of the code, but it also helps to describe the iterative decoding algorithm. An important feature of a Tanner graph is its *girth*, i.e., the length of its shortest cycle. This parameter directly controls the error performance of the codes once decoded with an iterative MAP decoder. The girth can be seen as the minimum number of edges that it takes to go from one particular node back to itself. For example, the girth of the Tanner graph shown in Figure 2.5 is six. In general, the larger the girth, the better the error performance will be.

2.3.2 IEEE 802.16e Standard LDPC Code

LDPC codes are hard to encode in general unless specific combinational or algebraic structure is embedded in the code to allow for easy generation of parity bits. The

IEEE802.16e-2005 standard, also known as Mobile Worldwide Interoperability for Microwave Access (Mobile WiMAX), provides a set of LDPC codes with such property. Not only the generation of parity bits is easy, the encoding is also systematic. These LDPC codes accommodate various code rates (i.e., $R = 1/2, 2/3, 3/4$ and $5/6$) as well as different block sizes. In general, the $n \times m$ parity check matrix $\mathbf{A}$ is characterized by several smaller $z \times z$ matrices $\mathbf{P}_{i,j}$ with $m = z \times 24$ and $n = z \times m_b$. The matrix $\mathbf{A}$ is defined as:

$$\mathbf{A} = \begin{bmatrix} \mathbf{P}_{0,0} & \mathbf{P}_{0,1} & \mathbf{P}_{0,2} & \cdots & \mathbf{P}_{0,23} \\ \mathbf{P}_{1,0} & \mathbf{P}_{1,1} & \mathbf{P}_{1,2} & \cdots & \mathbf{P}_{1,23} \\ \mathbf{P}_{2,0} & \mathbf{P}_{2,1} & \mathbf{P}_{2,2} & \cdots & \mathbf{P}_{2,23} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{m_b-1,0} & \mathbf{P}_{m_b-1,1} & \mathbf{P}_{m_b-1,2} & \cdots & \mathbf{P}_{m_b-1,23} \end{bmatrix} \tag{2.12}$$

where $m_b = (1 - R) \times 24$ and $\mathbf{P}_{i,j}$ is one of a set of $z \times z$ permutation matrices or a zero matrix. In order to further describe every $\mathbf{P}_{i,j}$, the $m_b \times 24$ model matrix $\mathbf{A}_{bm}$ is defined as:

$$\mathbf{A}_{bm} = \begin{bmatrix} p_{0,0} & p_{0,1} & p_{0,2} & \cdots & p_{0,23} \\ p_{1,0} & p_{1,1} & p_{1,2} & \cdots & p_{1,23} \\ p_{2,0} & p_{2,1} & p_{2,2} & \cdots & p_{2,23} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{m_b-1,0} & p_{m_b-1,1} & p_{m_b-1,2} & \cdots & p_{m_b-1,23} \end{bmatrix} \tag{2.13}$$

where $p_{i,j}$ provides detailed information on each corresponding $\mathbf{P}_{i,j}$; whether it is an identity matrix, a circular right-shifted version of identity matrix, or a zero matrix. More specifically, if $p_{i,j} = 0$, $\mathbf{P}_{i,j}$ is an identity matrix and if $p_{i,j} < 0$, $\mathbf{P}_{i,j}$ is a zero matrix. If $p_{i,j} > 0$, $\mathbf{P}_{i,j}$ is a circular right-shifted identity matrix with the circular shift size being the value of $p_{i,j}$. As a result, LDPC codes with different code rates will

have different model matrix; but, for any code with a particular rate, the codeword size can simply be adjusted by varying the value of z and produce codeword sizes ranging from $m = 576$ to $m = 2304$. In this thesis, we will be using two LDPC codes proposed from this standard for our simulations; one with rate 1/2 and length 2304 and the other one with rate 1/2 and length 1152. For simplicity, we will call these codes WiMAX A Code and WiMAX B Code, respectively. They are derived from the same model matrix and more detail on the model matrix can found in the Appendix C.

2.3.3 The LDPC Encoding Algorithm

The encoding process at the transmitter generates parity-check bits $\mathbf{p} = (p_0, \cdots, p_{m-1})$ based on an information block $\mathbf{b} = (b_0, \cdots, b_{k-1})$. These parity-check bits $\mathbf{p}$ are then combined with the information block $\mathbf{b}$ to form a systematic codeword $\mathbf{x} = [\mathbf{b}\ \ \mathbf{p}] = [b_0, b_1, \cdots, b_{k-1}, p_0, p_1, \cdots, p_{m-1}]$. One method of encoding is to determine a $(k \times m)$ generator matrix $\mathbf{G}$ from $\mathbf{A}$ such that $\mathbf{G}\mathbf{A}^T = 0$. Codewords can be generated by multiplying the information block $\mathbf{b}$ by the generator matrix $\mathbf{G}$ (i.e., $\mathbf{x} = \mathbf{b}\mathbf{G}$). However, this method can be very complex especially when the codeword size is large. In [8], a very efficient encoding scheme is presented such that encoding can be performed directly from $\mathbf{A}$ avoiding the matrix multiplication via the generator matrix. First, thanks to the specific structure of the parity-check matrix, $\mathbf{A}$ is decomposed as:

$$\mathbf{A} = \begin{bmatrix} A_{l\times k} & B_{l\times z} & T_{l\times l} \\ C_{z\times k} & D_{z\times z} & E_{z\times l} \end{bmatrix} \tag{2.14}$$

where $l = n - z$. Let $\mathbf{x} = [\mathbf{b}\ \ \mathbf{p}] = [\mathbf{b}\ \ p_1\ \ p_2]$ in which the parity-check part $\mathbf{p}$ is further broken down to p_1 and p_2 with sizes equal to z and l, respectively. They can

be calculated as [8]

$$p_1^T = (ET^{-1}A + C)\mathbf{b}^T \tag{2.15}$$

and

$$p_2^T = T^{-1}(A\mathbf{b}^T + Bp_1^T). \tag{2.16}$$

The n-bits codewords can be formed simply by appending their corresponding p_1 and p_2 to the information block $\mathbf{b}$.

2.3.4 The LDPC Decoding Algorithms

LDPC codes can be decoded in many ways; such as majority-logic (MLG) decoding, bit-flipping (BF) decoding, weighted BF decoding, maximize a-posteriori probability (MAP) decoding, and iterative decoding based on belief or probability propagation. The first two are hard-decision decoding algorithm, while the last two are soft-decision decoding schemes. In weighted BF algorithm, some soft information assists the decoding process to achieve a BER performance in between that of the other two classes. Among these various decoding schemes, the iterative belief propagation provides the best error performance at feasible levels of complexity.

2.3.5 The Sum-Product Algorithm in the Log-Likelihood Domain

Iterative decoding based on belief propagation, also known as the sum-product algorithm (SPA), iteratively calculates the reliability measures of each codeword coordinate. These reliability measures are exchanged between variable nodes and check

nodes in the form of messages (hence a message-passing algorithm). In general, messages that are being passed around consist of two probabilistic values and the calculation of these messages requires large numbers of multiplications and additions. One way to simplify SPA is to express the messages in terms of LLRs. More specifically, LLR of a binary random variable u is defined as

$$L(u) = \ln \frac{P(u=1)}{P(u=0)}. \tag{2.17}$$

The sign of $L(u)$ is the hard decision of the binary random variable and the magnitude of $L(u)$ is the reliability of this decision. From this modification, each message not only captures the information provided by the two probabilistic values but also can be reduced to a single value. However, basic operations in $\mathbb{R}$ domain, such as multiplication and addition, must be adopted in the log-likelihood domain as well. Using the property of natural logarithm, multiplication simply transforms to additive operation; but, addition transforms into a more complicated log-likelihood domain operation. This operation in the log-likelihood domain can be generalized as a special algebra operation of LLR-values called *box-plus* operation. Another special algebra operation that acts as a complement to a *box-plus* operator is the *box-minus* operator which can be used to further simplify the calculations in a log-likelihood domain SPA. For more on these special algebra operations of LLR-values, please refer to Appendix D.

As discussed previously, LDPC codes are characterized by their parity-check matrix A. The number of rows m represents the number of parity checks that each codeword has to satisfy. Each parity-check equation verifies w_r codeword coordinates of each codeword. A codeword passes the parity check if the modulo-2 sum of the

corresponding bits is zero. For example, let us consider the parity-check matrix provided by Equation (2.11) and we want to check whether a codeword $\mathbf{x} = (x_0, \cdots, x_9)$ is valid. The matrix consists of five parity checks:

$$\begin{aligned} x_0 \oplus x_1 \oplus x_2 \oplus x_3 &= 0 \bmod 2 \\ x_0 \oplus x_5 \oplus x_6 \oplus x_7 &= 0 \bmod 2 \\ x_1 \oplus x_4 \oplus x_7 \oplus x_8 &= 0 \bmod 2 \\ x_2 \oplus x_5 \oplus x_7 \oplus x_9 &= 0 \bmod 2 \\ x_3 \oplus x_6 \oplus x_8 \oplus x_9 &= 0 \bmod 2. \end{aligned}$$

If the word passes these five conditions, it is a valid codeword. The SPA is an iterative algorithm that uses the correlation of the parity checks in the parity-check matrix to detect and correct erroneous bits. In the log-likelihood domain, distinct messages are generated in every node and passed along every edge in the form of LLRs. In general, there are two types of messages. Messages generated in check nodes provide the estimated reliability values to their neighboring variable nodes according to the information gathered from every other neighboring nodes except for the node that the message is sent to. Messages generated in variable nodes indicate the nodes' own reliability values to their neighboring check nodes, and these reliability values are estimated according to the information gathered from neighboring check nodes other than the node that the message is sent to.

Before getting into the algorithm, some notations must be defined. Let $\mathcal{N}(w)$ denote the set of neighbors of a vertex w in a factor graph. When $k \in \mathcal{N}(w)$, this implies vertex k and w shares an edge. Let $\mathcal{N}(w)/k$ denotes all the neighbors of w other than k. In addition, let $q_{w \to k}(x)$, $x \in \{0, 1\}$, denote the message that

variable node w sends to check node k indicating the probability of variable node w being 0 or 1, based on all the check nodes connecting to w excluding k. Similarly, let $r_{k\to w}(x)$, $x \in \{0,1\}$ denotes the message that check node k sends to variable node w indicating the probability of variable node w being 0 or 1, based on all the variable nodes connecting to k excluding w. If the encoded codeword $\mathbf{x}$ has the form $\mathbf{x}$=$[x_1, x_2, ..., x_m]$, then messages in LLRs sent from variable nodes to check nodes and from check nodes to variable nodes is defined respectively as follows.

$$\lambda_{w\to k}(x_n) = \log\left(\frac{q_{w\to k}(1)}{q_{w\to k}(0)}\right) \tag{2.18}$$

$$\Lambda_{k\to w}(x_n) = \log\left(\frac{r_{k\to w}(1)}{r_{k\to w}(0)}\right) \tag{2.19}$$

where $n = \{1, 2, \cdots, m\}$. During the initialization stage, each variable node is assigned with the corresponding LLR values. Since we are considering the IJDD system described in Section 2.1.4.2, the LLRs are given by the MIMO detector and denoted as $L_{1e}(x_n|\mathbf{y})$. Without loss of generality, in this section, we will replace $L_{1e}(x_n|\mathbf{y})$ by a more general LLR-variable $L(x_n)$. Then we let $\lambda_{w\to k}(x_n) = L(x_n)$ and $\Lambda_{k\to w}(x_n) = 0$ for all the edges connecting check nodes and variable nodes. SPA can be summarized as follows.

- Check node update: For each check node k and for each $w \in \mathcal{N}(k)$, compute

$$\Lambda_{k\to w}(x_n) = 2\tanh^{-1}\left(\prod_{w'\in\mathcal{N}(k)/w} \tanh\left(\frac{\lambda_{w'\to k}(x_n)}{2}\right)\right). \tag{2.20}$$

- Variable node update: For each variable node w and for each $k \in \mathcal{N}(\mathrm{w})$, compute

$$\lambda_{w\to k}(x_n) = L(x_n) + \sum_{k'\in\mathcal{N}(w)/k} \Lambda_{k'\to w}(x_n). \tag{2.21}$$

- Termination: All variable nodes add all the messages that they received from all their neighbors as follows.

$$\lambda_w(x_n) = L(x_n) + \sum_{k \in \mathcal{N}(\mathrm{w})} \Lambda_{k \to w}(x_n). \tag{2.22}$$

Once $\lambda_w(x_n)$ for every x_n is computed, decision can be made such that $x_n = 0$ if $\lambda_w(x_n) \geq 0$ and $x_n = 1$ if $\lambda_w(x_n) < 0$. If $\mathbf{x}$ passes the membership test (i.e., $\mathbf{x} \cdot \mathbf{A^T} = 0$), $\mathbf{x}$ is the decoded codeword; otherwise, the algorithm will be repeated until the estimated codeword passes the membership test. If the maximum number of iteration is reached and $\mathbf{x}$ still does not pass the membership test, a failure will be declared. In particular, for the IJDD system described in Section 2.1.4.2, failing the membership test leads to the next round of turbo iteration if the system has not reached the stopping criterion (i.e., the maximum number of turbo iteration). Extrinsic soft information of each codeword coordinate can be generated as:

$$L_{2e}(x_n|\mathbf{y}) = \lambda_w(x_n) - L(x_n), \qquad n = \{1, 2, \cdots, m\}. \tag{2.23}$$

Note that the extrinsic information is obtained by subtracting the a-priori information from the complete soft information. Here, the a-priori information is referred to the information coming from the MIMO detector.

Performing hyperbolic tangent and inverse hyperbolic tangent during the check node update will significantly increase the computational complexity of the algorithm. However, it has been shown that special algebra operations such as *box-plus* and *box-minus* can be used to efficiently calculate messages during the check node update. If the *box-minus* operator is incorporated into the algorithm, instead of computing messages for every k and every $w \in \mathcal{N}(\mathrm{k})$ during the check node update, we can

compute universal messages for every k as follows:

$$\Lambda_k = 2\tanh^{-1}\left(\prod_{w'\in\mathcal{N}(k)}\tanh\left(\frac{\lambda_{w'\to k}(x_n)}{2}\right)\right). \tag{2.24}$$

Then, we use the box-minus operator to generate specific messages $\Lambda_{k\to w}(x_n)$ by subtracting the message received from the targeting variable node $\lambda_{w\to k}(x_n)$ from the universal message Λ_k. Furthermore, many approximations are developed to simplify *box-plus* and *box-minus* operations, making the sum-product algorithm practically feasible. More discussion on these approximations is provided in Appendix D.

Chapter 3

Soft-Input Soft-Output Detectors

Iterative joint detection and decoding (IJDD) is a technique that allows the receiver to approach near-optimal performance in a MIMO system with an outer channel code. It is essential that the detector module in this system is capable of processing soft-decision inputs and produces soft-decision outputs. Optimal soft-output detection involves marginalization over the entire constellation space. Unfortunately, performing such process for a MIMO system is infeasible; especially when large number of antennas are used in the system. In this chapter, various existing sub-optimal soft-input soft-output (SISO) detectors including minimum mean-square error-successive interference cancellation (MMSE-SIC), list sphere decoding (LSD) and Fincke-Pohst maximum-a-posteriori (FPMAP) are investigated. We start with the study of the performance and complexity trade-offs in LSD and FPMAP; providing detailed comparison of these two schemes. Also, a method to obtain a proper initial sphere decoding radius, an influential system parameter, is presented.

3.1 Minimum Mean-Square Error Equalizer with Successive Interference Cancellation

Minimum mean-square error (MMSE) is a common linear equalization technique that has been used for simple detection. The ability to detect symbols with linear complexity is why MMSE-type receivers have been widely used in practice. In MMSE equalization, the goal of the equalizer is to minimize the the average mean-square error (MSE) between the transmitted vector $\mathbf{s}$ and its estimate $\tilde{\mathbf{s}}$,

$$\min E[\|\mathbf{s} - \tilde{\mathbf{s}}\|^2], \tag{3.1}$$

where the MMSE symbol estimate $\tilde{\mathbf{s}}$ is obtained by passing the received signal $\mathbf{y}$ through an MMSE filter $\mathbf{G}_{MMSE}$, that is,

$$\tilde{\mathbf{s}} = \mathbf{G}_{MMSE}\mathbf{y}. \tag{3.2}$$

The coefficients of the MMSE filter are chosen to minimize (3.1) and are given by [27]

$$\mathbf{G}_{MMSE} = \mathbf{H}^H(\mathbf{H}\mathbf{H}^H + \sigma_n^2\mathbf{I}_{n_R})^{-1} \tag{3.3}$$

where $(\cdot)^H$ is the Hermitian operator, σ_n^2 is the variance of the channel noise, and $\mathbf{I}_{n_R}$ is the $n_R \times n_R$ identity matrix.

In [18], MMSE-SIC, a SISO version of MMSE equalizer, has been developed to suit the need of IJDD systems. It is a two step process in which successive interference cancellation (SIC) is performed prior to the equalization process to reduce the inter-symbol interference (ISI). SIC utilizes the a-priori information provided by the channel decoder module to obtain an updated $n_R \times 1$ received signal $\tilde{\mathbf{y}}^j$ in which the effects of ISI from jth ($j = 1, 2, ..., n_T$) transmitted symbol are minimized. In the first step,

soft estimates of the symbols transmitted from the jth antenna are evaluated as

$$\tilde{x}_j = E[s_j] = \sum_{\hat{s} \in S} \hat{s}\, p(s_j = \hat{s}). \tag{3.4}$$

Denote

$$\tilde{\mathbf{x}}_j \triangleq [\tilde{x}_1, ..., \tilde{x}_{j-1}, 0, \tilde{x}_{j+1}, ..., \tilde{x}_{n_T}]^T \tag{3.5}$$

as a vector of soft estimates of the transmitted symbols with jth index set to zero. Soft interference cancellation can then be performed to obtain

$$\tilde{\mathbf{y}}^j \triangleq \mathbf{y} - \mathbf{H}\tilde{\mathbf{x}}_j = \mathbf{H}(\mathbf{s} - \tilde{\mathbf{x}}_j) + \mathbf{n}. \tag{3.6}$$

Note that in the first iteration when a-priori symbol information is not available, the initial a-priori symbol probability can be set to $1/Q$. In the second step, an instantaneous linear MMSE filter is applied to $\tilde{\mathbf{y}}^j$ to obtain the MMSE estimate of s_j

$$\tilde{s}_j = \mathbf{w}_{j,MMSE}^H \tilde{\mathbf{y}}^j \tag{3.7}$$

and the filter $\mathbf{w}_{j,MMSE}^H$ is chosen to minimize the mean square error between the transmitted symbol s_j and the filter output $\tilde{s}_j$.

$$\mathbf{w}_{j,MMSE} = E_s(\mathbf{H}\Delta_j\mathbf{H}^H + \sigma_n^2\mathbf{I}_{n_R})^{-1}\mathbf{H}\mathbf{e}_j \tag{3.8}$$

where Δ_j is a diagonal matrix and can be expressed as

$$\begin{aligned} \Delta_j &\triangleq \mathrm{COV}\{\mathbf{s}_j - \tilde{\mathbf{x}}_j\} = E[(\mathbf{s}_j - \tilde{\mathbf{x}}_j)(\mathbf{s}_j - \tilde{\mathbf{x}}_j)^H] \\ &= \mathrm{diag}\{1 - |\tilde{x}_1|^2, ..., 1 - |\tilde{x}_{j-1}|^2, 1, 1 - |\tilde{x}_{j+1}|^2, ..., 1 - |\tilde{x}_{n_T}|^2\}. \end{aligned} \tag{3.9}$$

E_s is the average signal energy and $\mathbf{e}_j$ denotes an $n_R \times 1$ vector with all zero entries, except for the jth entry being 1. The detailed derivation of (3.8) can be found in the Appendix B. In [28], the distribution of the residual interference-plus-noise at the output of a linear MMSE equalizer is concluded to be well approximated by a Gaussian distribution. According to [18], equivalent representation of (3.7) can be

expressed as

$$\tilde{s}_j = u_j s_j + \eta_j \tag{3.10}$$

where

$$u_j = \mathbf{w}_{j,MMSE}^H \mathbf{H}\mathbf{e}_j \tag{3.11}$$

and $\eta_j \backsim \mathcal{N}_c(0, z_j^2)$ with parameter

$$z_j^2 = u_j - u_j^2. \tag{3.12}$$

From (3.10), (3.11), and (3.12), the likelihood function of s_j can be approximated as follows

$$p(\tilde{s}_j | s_j = \hat{s}) = \frac{1}{\pi z_j^2} \exp\left[\frac{- \parallel \tilde{s}_j - u_j \hat{s} \parallel^2}{\eta_j^2}\right]. \tag{3.13}$$

There is a major difference between the MMSE-SIC detection and the optimal approach based on bitwise maximum a-posteriori (MAP) described in Section 2.1.5 in terms of generating bitwise soft information. An MMSE-SIC detector extracts the extrinsic LLR of coded bit c_i from $\tilde{s}_j$, whereas the bitwise maximum a-posteriori (MAP) detector extracts the extrinsic LLR based on the entire received signal $\mathbf{y}$. Thus, the soft information $L_1(c_i|\mathbf{y})$ estimated by the MIMO detector according Figure 2.2 should be replaced by $L_{1e}(c_i|\tilde{s}_j)$ if an MMSE-SIC detector is used in such system. This is exactly one reason why soft-equalization methods do not perform as well as MAP detection approaches; but they are simpler in terms of computational complexity. The extrinsic information of the corresponding ith binary bit c_i in symbol

s_j delivered by the MMSE equalizer is calculated as

$$\begin{aligned} L_{1e}(c_i|\tilde{s}_j) &= \log \frac{\sum_{s^+ \in S_{j,i}^+} p(\tilde{s}_j|s_j = s^+)p(s_j = s^+)}{\sum_{s^- \in S_{j,i}^-} p(\tilde{s}_j|s_j = s^-)p(s_j = s^-)} - \log \frac{p(c_i = +1)}{p(c_i = -1)} \\ &= \log \frac{\sum_{s^+ \in S_{j,i}^+} \exp\left[\frac{- \parallel \tilde{s}_j - u_j s^+ \parallel^2}{\eta_j^2} + \sum_{k=1,k\neq i}^{p_m} \{s^+\}_k \cdot \frac{L_{2e}^*(c_i|\tilde{s}_j)}{2}\right]}{\sum_{s^- \in S_{j,i}^-} \exp\left[\frac{- \parallel \tilde{s}_j - u_j s^- \parallel^2}{\eta_j^2} + \sum_{k=1,k\neq i}^{p_m} \{s^-\}_k \cdot \frac{L_{2e}^*(c_i|\tilde{s}_j)}{2}\right]} \end{aligned} \tag{3.14}$$

where $S_{j,i}^+$ is the set of all possible values of s_j for which the rth LDPC-coded bit is $+1$, and $S_{j,i}^-$ is similarly defined. $\{s^+\}_k$ denotes the corresponding kth binary bit in symbol s^+ and $\{s^-\}_k$ is similarly defined. $L_{2e}^*(c_i|\tilde{s}_j)$ denotes the extrinsic information received from the channel decoder in the previous iteration.

3.2 List Sphere Detection

In IJDD systems, optimal detector obtains exact APP on each bit by marginalizing over the entire constellation space; however, the required computational complexity of such process is intractable. List sphere decoding (LSD) is a suboptimal detection method that constrains its marginalization in a subset of the entire space. This subset, presented as a candidate list $\mathcal{L}$, is then used to approximate APP in its corresponding LLR format as shown in (2.9). Since LSD is a technique derived directly from sphere decoding algorithm (SDA) proposed in [29], we will briefly described SDA before we discuss the core of LSD.

In a V-BLAST MIMO system without error correcting codes, optimal maximum-likelihood (ML) detection requires finding the signal point $\hat{\mathbf{s}}_{ml}$ of the transmitter vector signal set $\mathcal{S}^{n_T}$ in a transformed vector space (i.e., $\mathbf{Hs}$) that minimizes the Euclidean distance with respect to the received signal vector $\mathbf{y}$:

$$\hat{\mathbf{s}}_{ml} = \arg \min_{\mathbf{s} \in \mathcal{S}^{n_T}} \| \mathbf{y} - \mathbf{Hs} \|^2 . \tag{3.15}$$

One obvious way of obtaining $\hat{\mathbf{s}}_{ml}$ is to calculate $\| \mathbf{y} - \mathbf{Hs} \|^2$ for every possible $\mathbf{s}$ that could have been sent and pick the $\mathbf{s}$ that yields the smallest metric value. This is referred to as an exhaustive search. The performance of SDA is equivalent to an exhaustive search (optimal detection) without actually searching every possible signal point, also known as lattice point, in the entire space. The premise behind SDA is to generate a hypersphere with radius d in which only a portion of lattice points $\mathbf{s}$ are contained and satisfy the condition

$$d^2 \geq \| \mathbf{y} - \mathbf{Hs} \|^2 . \tag{3.16}$$

For simplicity, real-value realization of the system is assumed such that $\mathbf{H}$ becomes an $n \times m$ matrix and $n_R \geq n_T$ is also assumed. More discussion on underdetermined MIMO systems such that $n_R < n_T$ can be found in [30]. The novelty of SDA is to express $\mathbf{H}$ in terms of its $\mathbf{QR}$ decomposition form and utilize the upper triangle matrix $\mathbf{R}$ to eliminate all coordinates that do not satisfy (3.16). By substituting $\mathbf{H}$

in terms of its **QR** decomposition, (3.16) can be equivalently expressed as follows:

$$\begin{aligned} d^2 \geq \| \mathbf{y} - \mathbf{H}\mathbf{s} \|^2 &= \left\| \mathbf{y} - \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ 0 \end{bmatrix} \mathbf{s} \right\|^2 \\ &= \left\| \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \end{bmatrix} \mathbf{y} - \begin{bmatrix} \mathbf{R} \\ 0 \end{bmatrix} \mathbf{s} \right\|^2 \\ &= \| \mathbf{Q}_1^T\mathbf{y} - \mathbf{R}\mathbf{s} \|^2 + \| \mathbf{Q}_2^T\mathbf{y} \|^2 \end{aligned} \tag{3.17}$$

where $\mathbf{R}$ is an $m \times m$ upper triangular matrix and $\mathbf{Q}_1$ and $\mathbf{Q}_2$ represent the first m and last $(n-m)$ columns of Q, respectively. Denoting $\hat{\mathbf{s}} = \mathbf{R}^{-1}\mathbf{Q}_1^T\mathbf{y}$, it follows that $\| \mathbf{Q}_2^T\mathbf{y} \|^2 = \| \mathbf{y} \|^2 - \| \mathbf{H}\hat{\mathbf{s}} \|^2$ and (3.17) can now be re-written as

$$\acute{d}^2 \geq \| \mathbf{R}(\hat{\mathbf{s}} - \mathbf{s}) \|^2 \tag{3.18}$$

where $\acute{d}^2 = d^2 - \| \mathbf{y} \|^2 - \| \mathbf{H}\hat{\mathbf{s}} \|^2$. Until now, the variables on the right hand side of the inequality are expressed in matrix and vector format. After the expansion, the above inequality will have the following form,

$$\acute{d}^2 \geq r_{m,m}^2(s_m - \hat{s}_m)^2 + r_{m-1,m-1}^2 \left(s_{m-1} - \hat{s}_{m-1} + \frac{r_{m-1,m}}{r_{m-1,m-1}}(s_m - \hat{s}_m) \right)^2 + \cdots \tag{3.19}$$

where $r_{i,j}$ denotes the (i,j)th entry of $\mathbf{R}$. The right hand side of the inequality becomes a summation of different terms. By adding more terms, the inequality is forming a stronger necessary condition to eliminate the lattice points outside the hypersphere. If every term on the right hand side of the inequality is considered as a necessary condition, the first condition to satisfy for $\hat{\mathbf{s}}$ to be in the hypersphere is

$$\acute{d}^2 \geq r_{m,m}^2(s_m - \hat{s}_m)^2, \tag{3.20}$$

which implies

$$\left\lceil \hat{s}_m - \frac{\acute{d}}{r_{m,m}} \right\rceil \leq s_m \leq \left\lfloor \hat{s}_m + \frac{\acute{d}}{r_{m,m}} \right\rfloor \tag{3.21}$$

where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote rounding to the nearest larger and smaller element in the set of numbers which spans the lattice, respectively. Given a coordinate s_m that satisfies (3.22), the set of coordinates $\{s_{m-1}\}$, that satisfy

$$\left\lceil \hat{s}_{m-1|m} - \frac{\acute{d}_{m-1}}{r_{m-1,m-1}} \right\rceil \leq s_{m-1} \leq \left\lfloor \hat{s}_{m-1|m} + \frac{\acute{d}_{m-1}}{r_{m-1,m-1}} \right\rfloor \tag{3.22}$$

where $\hat{s}_{m-1|m} = \hat{s}_{m-1} - \frac{r_{m-1,m}}{r_{m-1,m-1}}(s_m - \hat{s}_m)$ and $\acute{d}^2_{m-1} = \acute{d}^2 - (y_m - r_{m,m}s_m)$, are then determined. The algorithm continues in a similar fashion for s_{m-2} and so on until s_1. Once it reaches s_1, all lattice points within the hypersphere are obtained. The above algorithm can be summarized in six steps and the input of the algorithm includes $\mathbf{R}$, $\mathbf{y}$, $\hat{\mathbf{s}}$, and d. $\mathbf{R}$ is the upper triangular matrix obtained from $\mathbf{QR}$ decomposition of the $\mathbf{H}$ matrix, $\mathbf{y}$ is the received vector from the channel, $\hat{\mathbf{s}}$ is the Babai estimate [31] (i.e., $\hat{\mathbf{s}} = \mathbf{H}^{\dagger}\mathbf{y}$, $\mathbf{H}^{\dagger}$ is the Penrose-Moore pseudo-inverse: $\mathbf{H}^{\dagger} = (\mathbf{H}^{*}\mathbf{H})^{-1}\mathbf{H}^{*}$, where $\mathbf{H}^{*}$ is the conjugate transpose), and d is the radius of the hypersphere. The algorithm is formalized as follow:

Input : $\mathbf{R}$, $\mathbf{y}$, $\hat{\mathbf{s}}$, and d.

1) Set $k = m$, $\acute{d}^2_m = d^2 - \| \mathbf{y} \|^2 - \| \mathbf{H}\hat{\mathbf{s}} \|^2$, $\hat{s}_{m|m+1} = \hat{s}_m$.

2) (Bound for s_k) Set $z = \frac{\acute{d}_k}{r_{k,k}}$, UB($s_k$)=$\lfloor z + \hat{s}_{k|k+1} \rfloor$, $s_k = \lceil -z + \hat{s}_{k|k+1} \rceil - 1$.

3) (Increase s_k) $s_k = s_k + 1$, If $s_k \leq$UB(s_k), go to step 5, else go to step 4.

4) (Increase k) $k = k + 1$; if $k = m + 1$ terminate algorithm, else go to step 3.

5) (Decrease k) If $k = 1$ go to step 6. Else $k = k - 1$, $\hat{s}_{k|k+1} = \hat{s}_k - \sum_{j=k+1}^{m} \frac{r_{k,j}}{r_{k,k}}(s_j - \hat{s}_j)$, $\acute{d}^2_k = \acute{d}^2_{k+1} - r^2_{k+1,k+1}(s_{k+1} - \hat{s}_{k+1|k+2})^2$, and go to step 2.

6) Solution found. Save $\mathbf{s}$ and go to step 3.

Once all points in the hypersphere are found, exhaustive search is then performed within the hypersphere. This way, detection complexity is reduced as the search space becomes smaller, while the detection process still guarantees to find the closest lattice point to the received signal $\mathbf{y}$ in terms of Euclidean distance. Thus, SDA is a good optimal detection candidate when no error correction codes are employed in a MIMO system. The SDA explained above is the basic Fincke and Pohst implementation [29]. There are also other improved schemes such as the Viterbo and Boutros implementation [32] and the Schnorr-Euchner implementation [33]. In the Viterbo and Boutros implementation, the radius is changed adaptively during the search; specifically, the radius is updated each time a point is found. In the Schnorr-Euchner implementation, the interval of possible coordinates inside the sphere are spanned in a zig-zag order, starting from the midpoint of the interval. It has been shown that the Schnorr-Euchner implementation is more efficient than Viterbo and Boutros implementation [34]. Based on Schnorr-Euchner enumeration, we implemented all our sphere-decoding-based simulations based on [30] (Algorithm II, Smart Implementation) given that it is the most efficient SDA in the literature.

In [19], LSD is developed to be the SISO version of SDA to accommodate soft output. Instead of finding the closest lattice point, LSD intends to find $\mathcal{L}$ that contains $\mathcal{N}$ lattice points that are closer to the received signal $\mathbf{y}$ than any other lattice points in terms of Euclidean distance. Lattice points that are closer to the received point $\mathbf{y}$ are more important in the LLR calculations and therefore points that are far away can be ignored. Based on this requirement, LSD is developed by modifying SDA in two ways. Every time a lattice point within the radius d is found: 1) radius is kept fixed; 2) the point is added to $\mathcal{L}$ if the list is not already full. If the list is full and

the new point considered is closer to the received point than the lattice point with furthest distance in the existing list, the new point will replace that point in the list; otherwise, the list will not be updated. Once the list $\mathcal{L}$ is generated, the same $\mathcal{L}$ is used in the next few consecutive iterations to generate soft information based on MAP criterion described in Section 2.1.5. Extrinsic part of the soft information for each bit can then be estimated using (2.9) or (2.10) by only examining $\mathcal{L}$.

3.3 Fincke-Pohst MAP

Although LSD is a suboptimal detection technique using MAP criterion, the underlying $\mathcal{L}$ search structure is developed based on ML criterion shown in (3.15). In other words, LSD uses ML criterion to generate $\mathcal{L}$ and then uses MAP criterion to evaluate soft information $L_{1e}(c_i|\mathbf{y})$. Fincke-Pohst MAP (FPMAP) is a true MAP detector that uses MAP criterion to generate $\mathcal{L}$ as well as evaluate soft information.

$$\hat{\mathbf{s}}_{map} = \arg \max_{\mathbf{s} \in \mathcal{S}^{n_T}} p(\mathbf{s}|\mathbf{y}). \tag{3.23}$$

By using Bayes' rule, MAP can be rewritten as

$$\begin{aligned} \arg \max_{\mathbf{s} \in \mathcal{S}^{n_T}} p(\mathbf{s}|\mathbf{y}) &= \arg \max_{\mathbf{s} \in \mathcal{S}^{n_T}} \frac{p(\mathbf{y}|\mathbf{s})p(\mathbf{s})}{p(y)} \\ &= \arg \max_{\mathbf{s} \in \mathcal{S}^{n_T}} p(\mathbf{y}|\mathbf{s})p(\mathbf{s}) \end{aligned} \tag{3.24}$$

in which if all the transmitted symbols s_i $(i = 1, 2, \cdots, n_T)$ are independently transmitted, we can further write

$$p(s) = \prod_{k=1}^{n_T} p(s_k) = \exp\left[\sum_{k=1}^{n_T} \log p(s_k)\right]. \tag{3.25}$$

For an AWGN channel, MAP criterion in (3.23) can be rewritten to the following optimization problem

$$\hat{\mathbf{s}}_{map} = \arg \min_{\mathbf{s} \epsilon S^{n_T}} \left[\|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 - \sum_{k=1}^{n_T} \log p(s_k) \right]. \tag{3.26}$$

However, the goal of FPMAP is to find an $\mathcal{L}$ that is capable of providing accurate soft information approximation in (2.9). Instead of finding the $\hat{\mathbf{s}}_{map}$, the above criterion is used to generate a hypersphere with radius $\sqrt{d^2 + \sum_{k=1}^{n_T} \log p(s_k)}$ such that

$$d^2 + \sum_{k=1}^{n_T} \log p(s_k) \geq \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2. \tag{3.27}$$

Same procedures that apply to LSD can be applied to (3.27). By expressing $\mathbf{H}$ in terms of its $\mathbf{QR}$ decomposition form and exploiting the upper triangular matrix $\mathbf{R}$ to eliminate coordinates that do not satisfy the necessary condition, (3.27) can be written and expanded, similar to (3.18) and (3.19), as

$$\begin{aligned} \acute{d}^2 &\geq \| \mathbf{R}(\hat{\mathbf{s}} - \mathbf{s}) \|^2 - \sum_{k=1}^{n_T} \log p(s_k) \\ &= r_{m,m}^2 (s_m - \hat{s}_m)^2 - \log p(s_m) \\ &\quad + r_{m-1,m-1}^2 \left(s_{m-1} - \hat{s}_{m-1} + \frac{r_{m-1,m}}{r_{m-1,m-1}} (s_m - \hat{s}_m) \right)^2 - \log p(s_{m-1}) + \cdots \end{aligned} \tag{3.28}$$

where $\acute{d}^2 = d^2 - \| \mathbf{y} \|^2 - \| \mathbf{H}\hat{\mathbf{s}} \|^2$ and $r_{i,j}$ denotes the (i,j)th entry of $\mathbf{R}$. Restrictions on each coordinate are examinated dimension by dimension and these steps can be formulated as follow:

Input : $\mathbf{R}$, $\mathbf{y}$, $\hat{\mathbf{s}}$, d, and $p(\mathbf{s})$.

1) Set $k = m$, $\acute{d}_m^2 = d^2 - \| \mathbf{y} \|^2 - \| \mathbf{H}\hat{\mathbf{s}} \|^2$, $\hat{s}_{m|m+1} = \hat{s}_m$.

2) (Bound for s_k) Set $z = \frac{\acute{d}_k}{r_{k,k}}$, UB($s_k$)=$\lfloor z + \hat{s}_{k|k+1} \rfloor$, $s_k = \lceil -z + \hat{s}_{k|k+1} \rceil - 1$.

3) (Increase s_k) $s_k = s_k + 1$, If $r_{k,k}^2(s_k - \hat{s}_{k|k+1})^2 > \acute{d}_k^2 + \log p(s_k)$ and $s_k \leq UB(s_k)$, go to step 3, else proceed. If $s_k \leq$UB(s_k), go to step 5, else go to step 4.

4) (Increase k) $k = k + 1$; if $k = m + 1$ terminate algorithm, else go to step 3.

5) (Decrease k) If $k = 1$ go to step 6. Else $k = k - 1$, $\hat{s}_{k|k+1} = \hat{s}_k - \sum_{j=k+1}^{m} \frac{r_{k,j}}{r_{k,k}}(s_j - \hat{s}_j)$, $\acute{d}_k^2 = \acute{d}_{k+1}^2 - r_{k+1,k+1}^2(s_{k+1} - \hat{s}_{k+1|k+2})^2 + \log p(s_{k+1})$, and go to step 2.

6) Solution found. Save $\mathbf{s}$ and go to step 3.

Assume that the search yields the set $\mathcal{L}$. The extrinsic part of the soft information for each bits can then be estimated using (2.9) or (2.10) by only using $\mathbf{s} \in \mathcal{L}$. Although different criterion is used, FPMAP still inherits its searching method through sphere decoding. However, the advantage of this scheme is that the searching algorithm takes the a-priori information from the channel decoder into consideration. Thus, FPMAP is repeated for each consecutive iteration and usually each consecutive iteration of FPMAP results in a fewer number of points that are used to generate soft information.

3.4 The Choice of Initial Radius

The initial radius is an influential parameter for any detector derived from sphere decoding algorithm. Detectors such as LSD and FPMAP cannot ignore the issue of initial radius; and yet an accurate estimation method for this parameter still eludes us. The initial choice of radius affects FPMAP's performance as well as complexity. However, the initial radius only affects the complexity of LSD as list size is guaranteed to be fixed. In LSD, it makes sense that choosing an unnecessary large radius will result in too many lattice points in the sphere and finding the $\mathcal{N}$ best $\mathbf{s}$ to construct

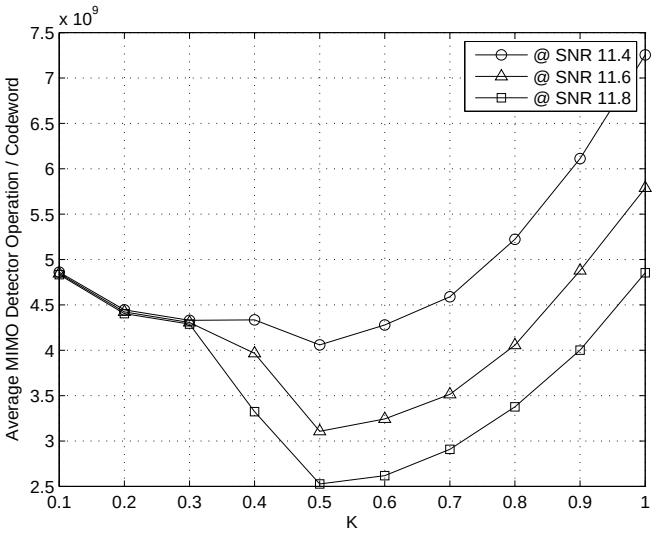


Figure 3.1: Complexity analysis of various choices of radius in 4×4 V-BLAST LSD IJDD MIMO system employing WiMAX A Code, 16-QAM, and interleaver size of 6912.

$\mathcal{L}$ will increase the detection complexity. In contrast, choosing a small radius will result in not having enough lattice point and the search will have to start all over again with a more appropriate radius until at least $\mathcal{N}$ closest lattice points are in the sphere. This repetitive process results in redundant computation and also increases the detection complexity. Thus, it is important that we pick a good initial radius to make sure that LSD is in its optimal condition. One possible choice of radius is proposed in [19] as:

$$r^2 = 2\sigma^2 K n_R - \mathbf{y}^*(\mathbf{I} - \mathbf{H}(\mathbf{H}^*\mathbf{H})^{-1}\mathbf{H}^*)\mathbf{y} \tag{3.29}$$

where $0 < K < 1$ is a constant to be chosen by the system designer. Equation (3.29) provides a rough guideline to estimate the initial radius with an unknown parameter K which has to be manually adjusted to suit different system environments. In other words, a good initial radius can be found by testing a series of K's between 0 and

1. As an example, consider an IJDD MIMO system that uses 4 transmit, 4 received antennas, 16-QAM signal constellation, and WiMAX A Code. The coded sequence is interleaved by a random block interleaver of size 6912. The system performs maximum of 5 turbo iterations and 8 belief propagation iterations. Figure 3.1 shows that when $K = 0.5$, the system requires minimum amount of computational complexity between the range of practical SNR from 11.4 to 11.8. In general, a good choice of K varies with system parameters such as number of transmit/receive antennas, type of constellation, and type of error correcting code. For any set of system parameters, the same process is required to obtain satisfactory choice of K during the design process. In our work, the initial radii of all LSD simulations are selected based on such method in order to ensure that LSD is using least amount of complexity to obtain the performance given by a fixed list size $\mathcal{N}$.

3.5 Performance and Complexity Analysis of LSD and FPMAP

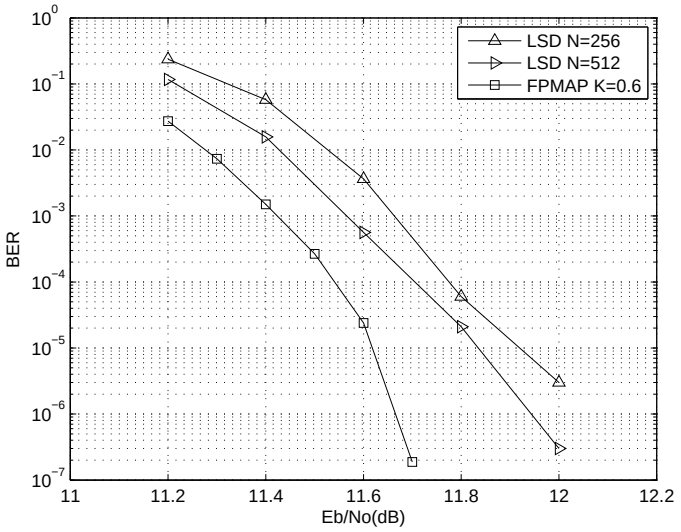
After exploring some existing SISO detectors, performance and complexity comparison between schemes is needed to summarize the pros and cons of each scheme. In general, equalization methods provide low complexity but their error performance is far from optimal. This has again been shown in [20]. LSD and FPMAP are two best SISO detectors in the literatures; however, it is still unclear which list-type detectors is better. In our study, we will only focus on analyzing the performance and complexity of list-type detectors.

LSD and FPMAP differ from each other in terms of the method used to generate

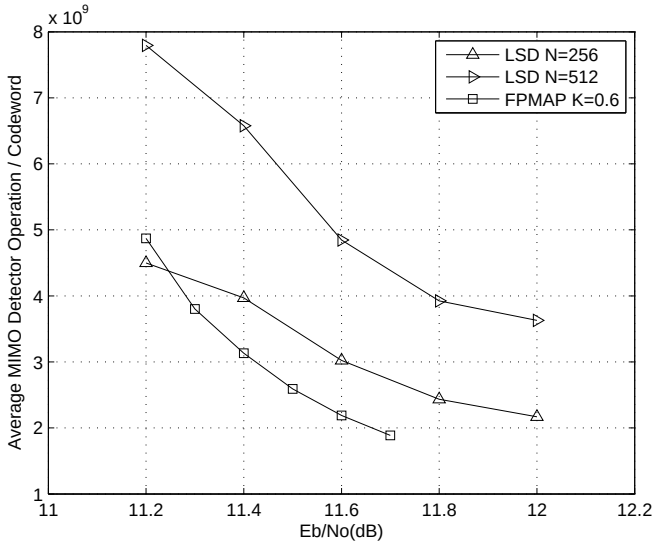
and update the list. LSD generates a candidate list $\mathcal{L}$ that concentrates on minimizing (3.15) with fixed cardinality $\mathcal{N}$ for each consecutive iteration. A larger candidate list will result in a better approximation; thus, the performance of LSD can be easily adjusted through different $\mathcal{N}$. FPMAP generates $\mathcal{L}$ that focuses on minimizing (3.26) and is capable of obtaining $\mathcal{L}$ with the consideration of the a-priori information from the channel decoder. Unlike in LSD, the size of $\mathcal{L}$ is not fixed and new set of $\mathcal{L}$ is generated in every iteration. As the iteration proceeds, channel decoder provides a-priori information that converges to the actual codeword. By adopting new a-priori information, the list generated by FPMAP gets smaller and smaller after each consecutive iteration. Due to this fact, although the list is repeatedly updated, the overall system complexity may not necessary be higher than LSD despite previous claims in [35]. When FPMAP updates its list, two important things happen. First, the updated list provides more accurate soft information. Second, each iteration reduces list size and therefore, complexity is lower than in an identical system where the list size is not decreasing. This means that theoretically, it is difficult to conclude which system outperforms the other in terms of performance and complexity. Practically, we can compare the two detectors through simulations.

The two criteria used in evaluating systems are error performance and computational complexity, but due to the nature of the relationship, we must consider them simultaneously to be able to compare systems. Using the same system that we discussed in Section 3.4, we now compare the two receivers in terms of performance and complexity. Figure 3.2 (a) shows the BER performance of systems utilizing LSD with list size $\mathcal{N}$=256 and 512 as well as FPMAP with choice of radius according to (3.29)

with K=0.6. Figure 3.2 (b) shows the computational complexity of the corresponding systems. More specifically, this graph shows the average arithmetic operations (i.e., sum of additions, multiplications, and comparisons) each detector required to detect a single codeword. We observe that by increasing the list size of LSD, the BER performance can be improved due to more accurate soft information, but at the same time, the average of arithmetic operations per codeword increases. Another observation is that FPMAP not only has better BER performance but also requires less computational complexity than both LSD schemes. At BER=10^{-6}, FPMAP has 0.28 dB improvement over LSD $\mathcal{N}$=512 scheme with 2.25 times complexity reduction. Another simulation is performed in a similar environment except with a different channel code, WiMAX B Code. Again, Figure 3.3 (a) shows the bit-error-rate (BER) performance of systems utilizing LSD with list size $\mathcal{N}$=256, 512, and 1024 as well as FPMAP with choice of radius according to (3.29) with K=0.6. Figure 3.3 (b) shows the computational complexity of the corresponding systems. Similar result can be observed. In addition, we can see that the BER improvements from LSD $\mathcal{N}$=512 to $\mathcal{N}$=1024 is smaller than the improvements from $\mathcal{N}$=256 to $\mathcal{N}$=512. However, the average arithmetic operations per codeword increases exponentially. Other simulations employing different LDPC channel codes, interleaver sizes, and various combinations of maximum number of LDPC decoding and turbo iterations are also performed; similar trade-offs in performance and complexity are observed. From these simulation results, we can conclude that in a V-BLAST LDPC-coded MIMO system, FPMAP is a preferable scheme in terms of error performance and computational complexity.

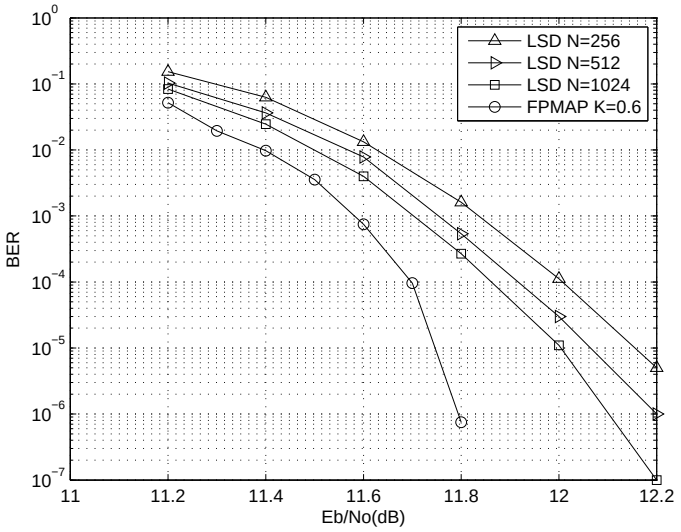


(a)

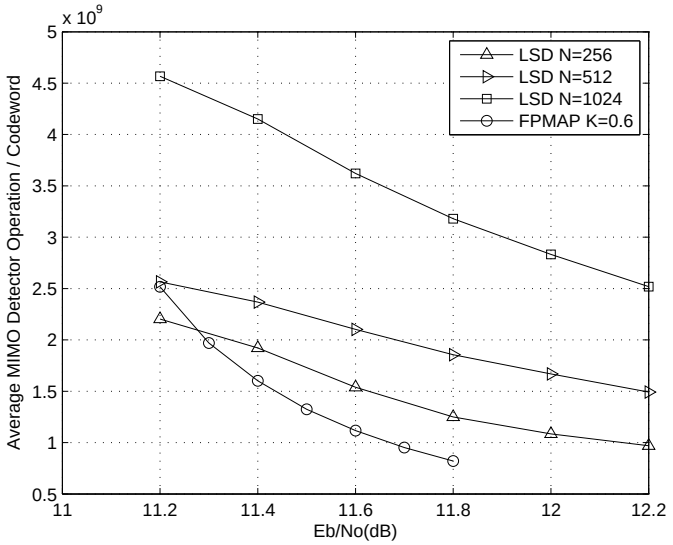


(b)

Figure 3.2: (a) BER performance of 4×4 V-BLAST system employing WiMAX A Code with 8 maximum number of decoding iterations and 5 maximum number of turbo iterations, 16-QAM, and interleaver size of 6912. (b) Complexity Analysis of systems in (a).



(a)



(b)

Figure 3.3: (a) BER performance of 4×4 V-BLAST system employing WiMAX B Code with 8 maximum number of decoding iterations and 5 maximum number of turbo iterations, 16-QAM, and interleaver size of 3456. (b) Complexity Analysis of systems in (a).

3.6 Conclusion

In this chapter, the issue of picking a proper initial sphere decoding radius is elaborated based on the rough guideline provided in [19]. We showed that a satisfactory choice of radius can be obtained through performing Monte-Carlo simulations during the design process. Priori research has demonstrated that LSD and FPMAP outperform soft-equalization methods; however, it is unclear which of the two scheme is superior in terms of performance-complexity trade-off. A comparison is conducted to resolve the matter. Our simulation results show that FPMAP can obtain up to 0.3 dB in error performance at error rate of 10^{-6} and simultaneously, obtain up to a 70% reduction in computational complexity. From these simulation results, we can conclude that in a V-BLAST LDPC-coded MIMO system, FPMAP is a preferable scheme in terms of error performance and computational complexity.

Chapter 4

New Detectors for Coded V-BLAST Systems

4.1 Introduction

Sub-optimal soft-input soft-output (SISO) detectors such as list-sphere decoding (LSD) and Fincke-Pohst Maximum-A-Posteriori (FPMAP) approximate soft information via marginalization over a candidate list, where the accuracy of the soft information estimation is directly influenced by the list size $\mathcal{N}$ as well as the quality of the list. A good quality list is the one with list candidates closer to the received signal in terms of Euclidean distance. A larger list will have better approximation but it is computationally more expensive. Depending on the statistical properties of the channel noise and fading, some received signals can be detected using a small $\mathcal{L}$ while some received signals require a larger $\mathcal{L}$. In LSD, by setting the list size $\mathcal{N}$ to a constant as well as the list staying the same and not being updated for every turbo detection iteration, the system might waste a lot of unnecessary computations.

It will be beneficial if the system can dynamically adjust its list size $\mathcal{N}$ according to the statistical properties of the channel noise and fading (i.e., use larger list size for received signals that are harder to detect and smaller list size for received signals that are easier to detect). Based on this idea, an improved scheme referred to as Dynamic-List Detector (DLD) [35] is presented in this chapter. This proposed scheme can be adapted to any list-type SISO detector such as LSD and FPMAP, resulting in a better performance-complexity trade-off. Furthermore, common issues with list-type detectors such as those founded upon the sphere decoding algorithm, including the optimal initial sphere radius, optimal radius update strategy, and their highly variable computational complexity are still unresolved. To address those issues, another detection scheme referred to as Window-Shifting Detector (WSD) [36] is presented. This new detection scheme has fixed detection complexity, making the scheme suitable for practical implementation.

4.2 Dynamic-List Detectors

The motivation behind Dynamic-List Detector (DLD) is to dynamically adjust the detector's list size $\mathcal{N}$ according to the channel noise and fading to provide a better performance-complexity trade-off in an iterative joint detection and decoding (IJDD) system. Since both the MIMO detector and the channel decoder exchange soft information repeatedly in a turbo manner, we will call each detection and decoding instance as one turbo iteration as apposed to possible iterations in the decoder alone. Codeword test is usually performed after each turbo iteration in order to determine whether more iterations are needed. Detection error occurs when the maximum number of turbo iterations is reached and the codeword test still shows an invalid

codeword. Given the channel conditions, some received signals require more turbo iterations while other received signals can be detected easier and faster. However, when the receiver first obtains a received signal, the number of turbo iterations required to successfully estimate the transmitted codeword is unknown. If the detector starts off with a smaller list size to evaluate the a-posteriori probabilities (APPs), the computation process in the first iteration can then be shortened. If the channel decoder does not successfully decode the codeword in the first iteration, the list size $\mathcal{N}$ could be increased to ensure that the detector will provide more reliable soft information in the next iteration. This motivates DLD to increase list size $\mathcal{N}$ as turbo iteration proceeds in an IJDD MIMO system. This allows the system to dynamically adjust its computation complexity for every detection instance according to the statistical properties of the noise and fading while maintaining good overall performance. Once the list $\mathcal{L}$ is generated, the extrinsic part of the soft information for each bits can be estimated using (2.9) or (2.10) by only examining $\mathbf{s} \in \mathcal{L}$. Next, we will show how this scheme can be adopted in LSD and FPMAP described in Section 3.2 and Section 3.3, respectively.

4.2.1 Dynamic List Sphere Decoding

The underlying technique in Dynamic List Sphere Decoding (DLSD) is list sphere decoding (LSD). In other words, DLSD can be seen as a modified version of LSD developed based on the dynamic-list idea described in the previous section. This allows DLSD to start with a smaller list size and if needed gradually increase the list size as turbo iteration proceeds. The concept of dynamic-list can be easily adapted

to LSD by applying the following rule,

$$\mathcal{N}_1 < \mathcal{N}_2 < \ldots < \mathcal{N}_{max}, \tag{4.1}$$

where $\mathcal{N}_i$ is the list size for the ith turbo iteration. When implementing such a scheme, it is easier to find the $\mathcal{L}$ with $\mathcal{N}_{max}$ in the beginning and sort the $\mathcal{L}$ in an ascending order so that the search only needs to be performed once for detection of each codeword. Although sorting the $\mathcal{L}$ might require high computational complexity, especially when $\mathcal{L}$ is large, it is still worth the effort since this operation is only performed in the first turbo iteration. In the next couple of consecutive turbo iterations, $\mathcal{L}$ with various list size $\mathcal{N}$ can be generated instantly without additional complicated operations. This extra effort of generating sorted list might mislead us to believe that higher computational complexity is required in comparison with LSD; however, we have shown through simulation results that DLSD is capable of improving overall system performance and simultaneously, reduce the computational complexity especially in higher signal-to-noise ratios (SNRs). This is because erroneous bits embedded in the received signals are more likely to be corrected in the first couple of turbo iterations where relative small list sizes ($\mathcal{N}$'s) are used to estimate soft information. This results in a low detection complexity. At the same time, more distorted received signals are concerned, DLSD increases the probability of successful detections by using larger list sizes in the final few turbo iterations to provide more accurate soft information. This is how improved performance is achieved by DLSD.

4.2.2 Dynamic Fincke-Pohst MAP

Similarly, Dynamic Fincke-Pohst Maximum-A-Posteriori (DFPMAP) can be seen as a modified version of Fincke-Pohst Maximum-A-Posteriori (FPMAP) developed based

on the dynamic-list idea. DFPMAP can use the same dynamic-list concept but the algorithm does not allow us to control the resulting list size $\mathcal{N}$ directly. Instead, radius is increased to ensure that more lattice points are covered within every FPMAP search after each turbo iteration. If the initial radius is evaluated based on (3.29), adjusting the constant K would be a good choice to vary the radius. In this case, DFPMAP can be achieved by applying the following rule,

$$K_1 < K_2 < \ldots < K_{max}, \tag{4.2}$$

where K_i is the constant applied to (3.29) for the ith turbo iteration. In [20], it is shown that FPMAP stays competitive to LSD in terms of computational complexity by obtaining fewer lattice points to generate soft information as turbo iteration proceeds. Then, this begs the question why does DFPMAP try to change the radius such that more lattice points are used in each consecutive turbo iteration? Similar to the reason why LSD can be modified to DLSD to obtain better performance-complexity trade-off in high SNRs, DFPMAP allows the detector to use fewer lattice points to calculate soft information for received signals that are easier to detect resulting in low detection complexity. At the same time, it adaptively provides more accurate soft information for highly distorted received signals increasing the probability of successful detection. In almost all cases studied, the improved scheme will have either performance gain or complexity gain or both over conventional schemes.

4.2.3 Simulation Results

Simulations are performed to compare the improved schemes with conventional schemes. For simplicity, we consider the same systems discussed in Section 3.5 where the

IJDD MIMO system uses 4 transmit, 4 receive antennas and 16-QAM signal constellation. Simulations are conducted with two different LDPC codes, WiMAX A Code and WiMAX B Code. If WiMAX A Code is used, the coded sequence is interleaved by a random block interleaver of size 6912; otherwise, it is interleaved with size 3456. The system performs 8 maximum number of LDPC decoding iterations and 5 maximum number of turbo iterations (i_{max}). We assume a discrete-time Rayleigh block-fading multiantenna channel model, where the complete channel state information (CSI) is know to the receiver but not to the transmitter. In the first system incorporating WiMAX A Code, Figure 4.1(a) illustrates that DLSD with $\{\mathcal{N}_i\}_{i=1}^{5} = \{128, 256, 512, 1024, 2048\}$ outperforms the conventional LSD with $\mathcal{N} = 512$ especially at high SNR. Furthermore, Figure 4.1(b) shows that although at low SNR DLSD suffers from higher complexity, in SNR values of practical interest, its complexity drops significantly below that of LSD's. Note that the complexity here refers to the average number of MIMO detector operations including additions, multiplications, and comparisons. This example perfectly demonstrates how the improved scheme can obtain performance and complexity improvements simultaneously.

The dynamic list sizes assigned for each turbo iteration are obtained heuristically through extensive simulations. Different sets of list size will result in different performance-complexity trade-offs in the system. In our study, the list size increases exponentially as turbo iteration proceeds. This is chosen so that the first couple of turbo iterations use relative small list size for soft information evaluation to compensate for the higher computational complexity required for more accurate soft information. At the same time, we do not want to assign an enormous list size to the last couple turbo iterations which would overwhelm the overall system complexity. As a

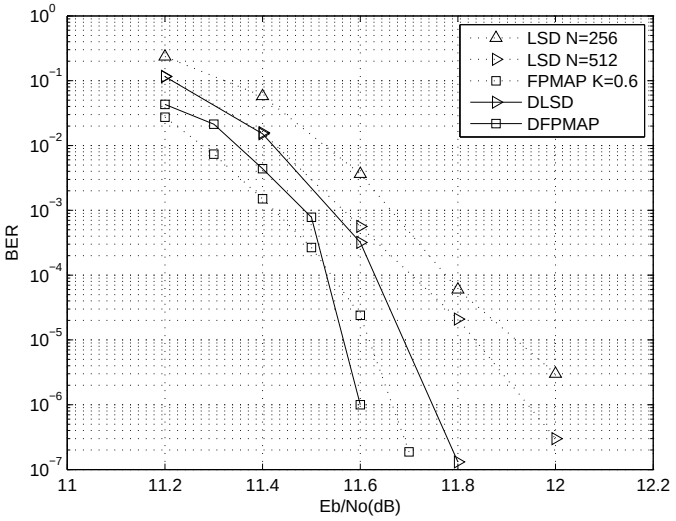
result, the increment of the list size will depend on the system's maximum number of turbo iterations (i_{max}). Larger i_{max} implies the increment of the list size can be smaller and vice versa. For example, if $i_{max} = 3$ in the previous simulation, DLSD will have to increase the increment of the list size (i.e., $\{\mathcal{N}_i\}_{i=1}^{3} = \{128, 512, 2048\}$) in order to outperform, or stay competitive to, LSD with $\mathcal{N} = 512$.

Given that the radius size can be varied by adjusting the constant K in Equation (3.29) as discussed in Section 3.4. Another comparison is illustrated in Figure 4.1(a) between DFPMAP with $\{K_i\}_{i=1}^{5} = \{0.4, 0.5, 0.6, 0.7, 0.8\}$ and conventional FPMAP with $K = 0.6$. Although in this scenario, DFPMAP performs similar to the traditional FPMAP, Figure 4.1(b) shows that the complexity of DFPMAP is reduced by almost half compared to FPMAP. Again, various K's assigned for each turbo iteration are obtained heuristically through extensive simulations. Different sets of K's will result in different performance-complexity trade-offs. From our simulations, we found that by linearly increasing K, the resulting list size will increase exponentially. Picking a too-large of a K will result in an enormous list which would overwhelm the overall system complexity. Therefore, it has to be chosen carefully.

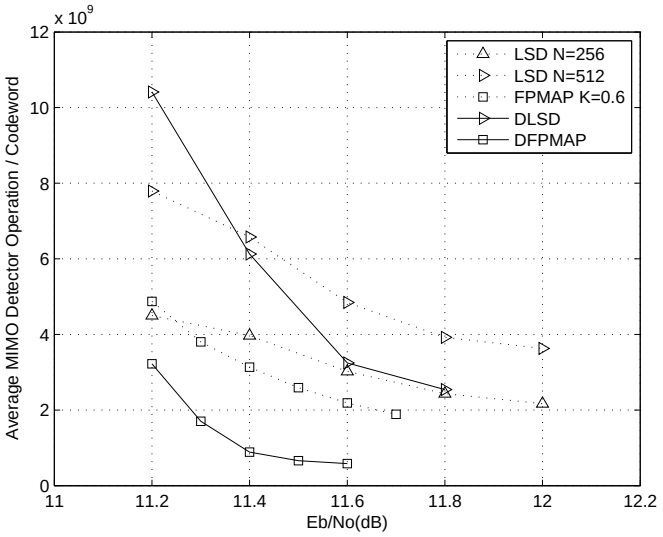
In the second system, incorporating WiMAX B Code, Figure 4.2 again shows that DLSD with $\{\mathcal{N}_i\}_{i=1}^{5} = \{128, 256, 512, 1024, 2048\}$ not only outperforms the conventional LSD with $\mathcal{N} = 512$, but also requires less computational complexity in the range of practical SNR values. Although DFPMAP with $\{K_i\}_{i=1}^{5} = \{0.4, 0.5, 0.6, 0.7, 0.8\}$ suffers from 0.1 dB loss at BER of 10^{-6} compared with FPMAP with $K = 0.6$, Figure 4.2(b) again shows that the complexity of DFPMAP is reduced by almost half compared to FPMAP. Both of these examples demonstrate the advantage of employing the improved schemes over the conventional schemes in either LSD

or FPMAP scenarios.

Note that there is no rigorous way to assign $\{\mathcal{N}_i\}_{i=1}^{i_{max}}$ and $\{K_i\}_{i=1}^{i_{max}}$, where i_{max} is the maximum number of turbo iterations, to achieve the best trade-off between performance and complexity; it is up to the system designers to adjust these values according to the rough guidelines provided in this section in order to meet their requirement.

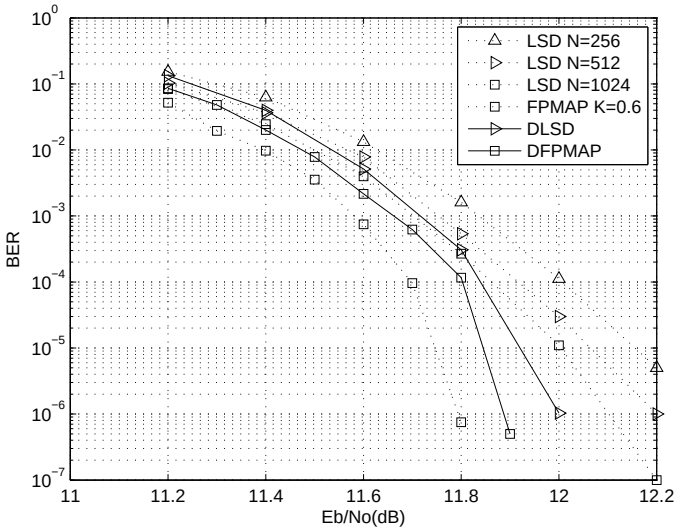


(a)

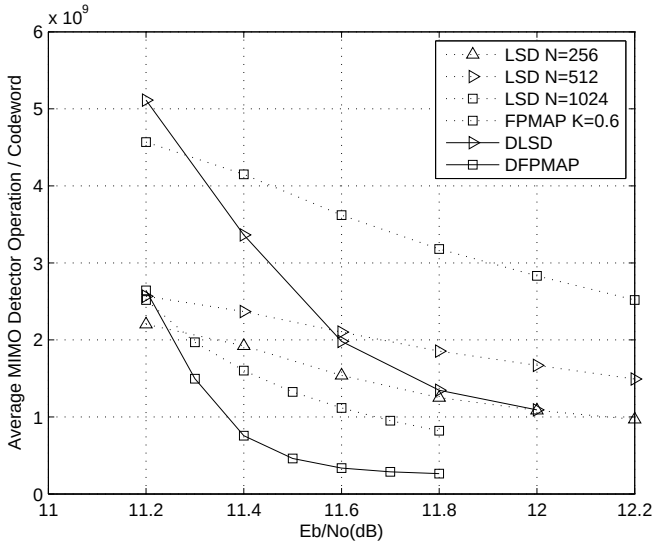


(b)

Figure 4.1: (a) BER performance of 4×4 V-BLAST system employing WiMAX A Code with 8 maximum number of decoding iterations and 5 maximum number of turbo iterations, 16-QAM, and interleaver size of 6912 (b) Complexity Analysis of systems in (a).



(a)



(b)

Figure 4.2: (a) BER performance of 4×4 V-BLAST system employing WiMAX B Code with 8 maximum number of decoding iterations and 5 maximum number of turbo iterations, 16-QAM, and interleaver size of 3456 (b) Complexity Analysis of systems in (a).

4.3 Windows-Shifting Detectors

The premise behind Windows-Shifting Detector (WSD) is to obtain candidate lists that are *somewhat* close to the received point and is motivated by the chase-type decoding technique [37] used in traditional block code decoding. In chase-decoding-based algorithms, the candidate list is generated by making an initial guess and then searching through the neighbors of the initial guess. The initial guess can be obtained through low-complexity equalization method such as the Babai estimate [31]. Similarly, in WSD, an initial guess is first obtained and neighbors ($\check{\mathbf{s}}$'s) are searched among coordinates of the initial guess which locally minimize the $\|\mathbf{y} - \mathbf{H}\check{\mathbf{s}}\|$. The entire signal constellation is naturally broken down into subsets. In this section, we consider the IJDD MIMO system model described in Section 2.1.4.2 using V-BLAST. Elements are represented in complex values; however, a standard vectorization can be applied to obtain the real-value realization of the system [38]:

$$\begin{bmatrix} \mathcal{R}(\mathbf{y}) \\ \mathcal{I}(\mathbf{y}) \end{bmatrix} = \begin{bmatrix} \mathcal{R}(\mathbf{H}) & -\mathcal{I}(\mathbf{H}) \\ \mathcal{I}(\mathbf{H}) & \mathcal{R}(\mathbf{H}) \end{bmatrix} \begin{bmatrix} \mathcal{R}(\mathbf{s}) \\ \mathcal{I}(\mathbf{s}) \end{bmatrix} + \begin{bmatrix} \mathcal{R}(\mathbf{n}) \\ \mathcal{I}(\mathbf{n}) \end{bmatrix}$$

where $\mathcal{R}(x)$ and $\mathcal{I}(x)$ are the real and imaginary parts of x, respectively. This technique will be used throughout this section to simplify the description of the algorithm. For example, in a 4x4 MIMO system using the 16-QAM constellation along with vectorization, each coordinate v_i in the symbol vector $\mathbf{v} = [v_1\, v_2 \cdots v_8]$ are in alphabet (coordinate alphabet) $\mathcal{A} = \{\pm 1, \pm 3\}$. Note that the symbol vector $\mathbf{v}$ is equivalent to the transmitted signal $\mathbf{s}$ after vectorization. The entire resulting lattice constellation consists of $4^8 = 65536$ possibilities. In the same system, if $\mathbf{v}$ is treated partially such that only the first 4 coordinates are considered and assuming that the rest of the coordinates are fixed to certain values, then the constellation reduces to $4^4 = 256$

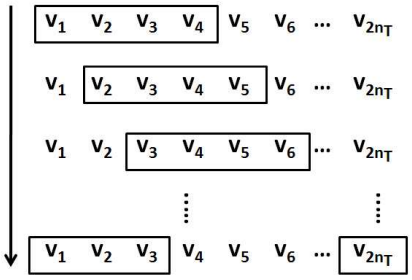


Figure 4.3: Illustration of Window Shifting Detection.

possibilities. Now, we define a new term called a *window* as follows: coordinates that vary are considered to be *inside* a window and coordinates that are fixed are considered to be *outside* a window. Therefore, in the previous example where only 4 coordinates are considered, the window size $\mathcal{W} = 4$. The concept of finding the neighbors of the initial guess partially is used throughout WSD to help generate a candidate list. Assume that vectorization is applied to obtain the real-value realization such that $\mathbf{v} = [v_1\, v_2 \cdots v_{2n_T}]$. The system uses Q-QAM constellation and $\mathcal{W}$ is a preset value. Generation of $\mathcal{L}$ is illustrated in Fig. 4.3 and can be summarized in the following.

WSD Algorithm: Construct the candidate list $\mathcal{L}$ as follows:

$$\mathcal{L} = \bigcup_{z=1}^{2n_T} \Phi_z \,,$$

where

$$\Phi_z = \{[v_1\, v_2 \cdots v_{2n_T}] |\, v_i \in \mathcal{A} \quad \text{for} \quad i \in \mathcal{I}_z \text{ and } \; v_i = \tilde{s}_i \; \text{ elsewhere}\}$$

with $\tilde{s}_i$ being the Babai estimate [31] of the ith symbol s_i and $\mathcal{I}_z$ is the set of integers $\{z, (z+1) \bmod 2n_T, ..., (z+\mathcal{W}) \bmod 2n_T\}$. This results in $|\Phi_z| = |\mathcal{A}|^{\mathcal{W}}$.

For example, using the system discussed previously with $\mathcal{W} = 4$:

$$\Phi_0 = \{[v_1\, v_2 \cdots v_{2n_T}] |\, v_i|_{i=1}^{4} \in \mathcal{A} \text{ and } \; v_i = \tilde{s}_i \; \text{ elsewhere}\}.$$

The resulting candidate list $\mathcal{L}$ can be further reduced to a preferable list size $\mathcal{N}$ by doing an exhaustive trimming within the list. Note that WSD serves the purpose of creating a subset of the entire space. Operating an exhaustive search on the subset is affordable as long as the subset is kept at a reasonable size. Window size $\mathcal{W}$ is a parameter that will directly influence the size of the resulting subset. As a result, $\mathcal{W}$ is an important factor in the performance and complexity of the system. Furthermore, a new set of $\mathcal{L}$ is generated in every iteration by substituting the Babai estimate with the estimate updated by the channel decoder. This new estimated codeword is obtained from the channel decoder's output soft information $\mathbf{L}_{2e}(\mathbf{c}|\mathbf{y})$. It can be calculated by multiplying corresponding bitwise reliability to generate the reliability information on each symbol. The new estimated codeword is picked based on the symbol that has the highest reliability value for every coordinate. Once the list $\mathcal{L}$ is generated, the extrinsic part of the soft information for each bits can be estimated using (2.9) or (2.10) by only examining $\mathbf{s} \in \mathcal{L}$.

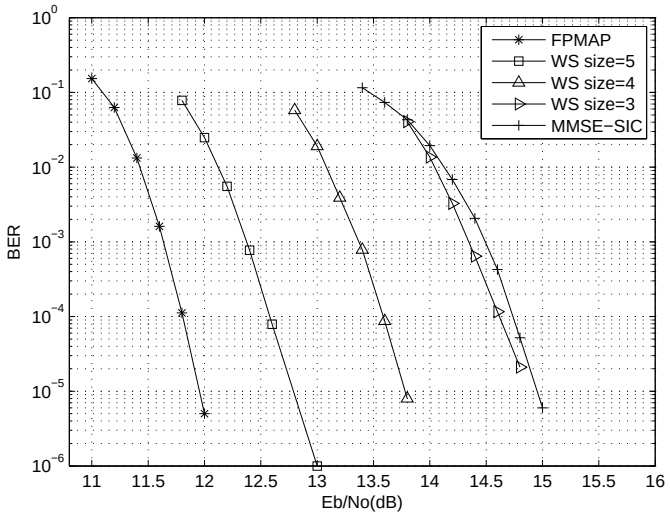
4.3.1 Simulation Results

Monte-Carlo simulations are performed to compare the performance and computational complexity of WSD to both FPMAP and MMSE-SIC detection schemes. According to [35], FPMAP performs slightly better than LSD in terms of BER performance and complexity; thus, only FPMAP is considered in our simulation. In this section, we restrict the size of $\mathcal{L}$ for WSD to 256. Information sequences are encoded by WiMAX B Code. The coded sequence is interleaved by a random block interleaver of size 6912 and then modulated onto 16-QAM using Gray mapping. Signals

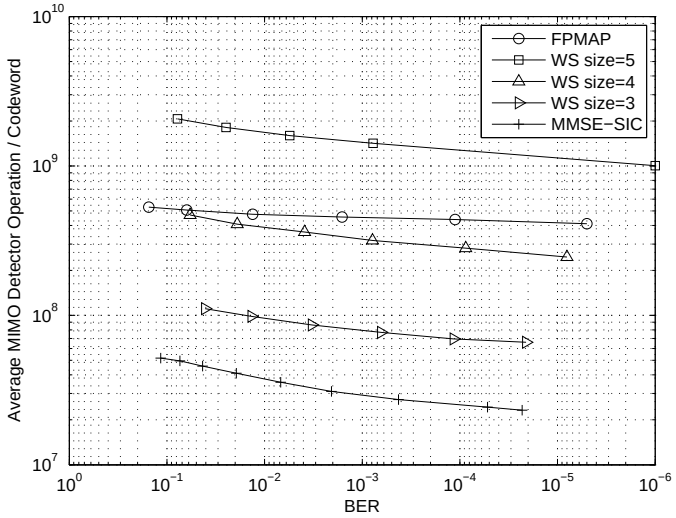
are transmitted over a 4x4 V-BLAST MIMO channel. In LDPC decoder, a maximum of 8 iterations of belief propagation algorithm is performed. Simulation results show that WSD with a larger $\mathcal{W}$ will have better performance but too large of a $\mathcal{W}$ will cause high computational complexity. Figure 4.4 shows that a system employing WSD with $\mathcal{W} = 4$ is 1.75 dB away from FPMAP; however, its computational complexity is reduced by 38.6%. Furthermore, a system employing WSD with $\mathcal{W} = 3$ is 2.8 dB away from FPMAP and its computational complexity is significantly reduced by 84.8%. Other simulations employing different LDPC channel codes, interleaver sizes and list sizes are also performed; similar trade-offs in performance and complexity are observed. WSD suffers from performance degradation compared to the sphere-decoding-based detection schemes because the quality of the generated list is not as good as the one from LSD and FPMAP. In other words, the codewords in the list are not as close to the received point in terms of Euclidean distance. However, the simple generation method of the candidate list $\mathcal{L}$ allows WSD to perform with a reduction in computational complexity and fixed number of flops per detection, making it a perfect candidate for practical implementation. This detection scheme is, especially, suitable for high SNRs where received signals are not as corrupted as the ones in lower SNRs because it gives WSD the advantage to provide list quality similar to the one from LSD and FPMAP without performing complex searching algorithm.

4.4 Conclusion

In this chapter, we have presented an improved scheme for list-type decoder that can be adapted to either LSD or FPMAP scenarios. It has the potential to improve



(a)



(b)

Figure 4.4: (a) BER performance of various detection schemes in a 4×4 V-BLAST system employing WiMAX A Code with 8 maximum number of decoding iterations and 5 maximum number of turbo iterations, 16-QAM, and interleaver size of 6912 (b) Complexity Analysis of schemes in (a).

performance, or complexity, or both simultaneously. Our simulations show that by adopting this improved scheme to LSD, the detector can achieve a complexity reduction of up to 61% at error rate of 10^{-6} and simultaneously, obtain a small BER performance improvement. Similarly, this improved scheme can also be adopted to FPMAP to provide up to 81% reduction in complex with a performance loss less than 0.1 dB at error rate of 10^{-6}, making this the best soft-input soft-output detectors for MIMO system according to our knowledge. We also presented an alternative detection method called WSD for IJDD MIMO systems. Simulation results show that WSD suffers from performance degradation compared to FPMAP, but it obtained a reduction in computational complexity and enjoys the benefit of fixed detection complexity regardless of system parameters and channel statistics. In addition to the above advantages, it also provides a wide range of performance-complexity trade-offs to accommodate different system requirements, making it the perfect candidate for practical implementation.

Chapter 5

Conclusions and Future Work

The advantages offered by a multiple-antenna system are widely recognized in many recent telecommunication standards. In particular, the popularity of MIMO stems from its use of spatial diversity and multiplexing to support high data rate wireless communication. This thesis is motivated by the belief that by incorporating channel coding to MIMO system, we can further improve the overall system error performance. V-BLAST and LDPC pair well together in a system due to the fact that V-BLAST sends information at high speeds and LDPC is a powerful error correcting code. One focuses on achieving high spatial multiplexing gain to maintain high data rates and the other one focuses on utilizing temporal diversity for error correction, making them a perfect combination in the system. Unfortunately, an optimal receiver for the coded MIMO system described above is practically infeasible; therefore, efforts are put towards designing sub-optimal receivers. Because the serial concatenated scheme suffers from performance degradation, iterative joint detection and decoding scheme is used instead. The use of iterative joint detection and detecting is preferred, but requires the development of more efficient soft-input soft-output (SISO) detectors.

Much of the focus in this thesis is aimed at comparing some of the existing SISO detectors and presenting improved schemes for more efficient detection. The following summarized what this thesis has accomplished:

1. We have investigated several existing soft-input soft-output (SISO) detectors; namely, MMSE-SIC, LSD, and FPMAP for LDPC-coded MIMO systems. More specifically, we study their application to iterative joint detection and decoding systems.

2. The issue of picking a proper initial radius for list-type detectors such as those founded upon the sphere decoding algorithm is elaborated based on the rough guideline discussed in Section 3.4. Although the choice of a proper radius can be influenced by system parameters such as number of transmit/receive antennas, type of constellation, and type of error correcting code, we showed that a satisfactory choice of radius can be obtained for different system parameters.

3. Given that LSD and FPMAP both outperform soft-equalization methods (i.e., MMSE-SIC), the performance-complexity comparison of LSD and FPMAP is presented and has been shown that FPMAP is better than LSD in both aspects. Our simulation results show that FPMAP can obtain up to 0.3 dB improvement in error performance at error rate of 10^{-6} and simultaneously, obtain up to a 70% reduction in computational complexity.

4. An improved scheme referred to as Dynamic-List Detector (DLD) is proposed that can be adopted to any SISO detector such as LSD and FPMAP where soft information is obtained via marginalization over a *candidate* list. This proposed scheme is capable of dynamically adjusting the list size according

to statistical properties of the channel noise and fading resulting in a better performance-complexity trade-off. Our simulation results show that by adopting this improved scheme to LSD, the detector can achieve a complexity reduction of up to 61% at error rate of 10^{-6}. What is more impressive is that BER performance did not deteriorate as one would expect, but instead performance is improved simultaneously. Similarly, this improved scheme can also be adopted to FPMAP to provide up to 81% reduction in complex with a performance loss less than 0.1 dB at error rate of 10^{-6}, making this the best soft-input soft-output detectors for MIMO system according to our knowledge.

5. We also presented an alternative detection method called the Window-Shifting Detector (WSD). This new list-type detector integrates an entirely different structure from a variety of modified sphere decoding detectors and is very flexible in terms of performance-complexity adjustment. Simulation results show that WSD suffers from performance degradation compared to FPMAP, but it obtained a reduction in computational complexity and enjoys the benefit of fixed detection complexity regardless of system parameters and channel statistics. In addition to the above advantages, it also provides a wide range of performance-complexity trade-offs to accommodate different system requirements, making it the perfect candidate for practical implementation.

The contributions listed above have resulted in the following published works: [35] and [36]. In the future, iterative joint detection and decoding will definitely be the trend in receiver design to improve the error performance in MIMO systems. More research on soft-input soft-output detectors must be conducted. In [38] [39], J.

Leuschner proposed a new generic maximum likelihood metric expression for space-time block codes. Later, the new metric expression is utilized to produce a new Chase-type sub-optimal decoder [40] for quasi-orthogonal space-time block codes based on reliability measures of the received signals. It is believed that ways of obtaining the reliability measures of the received signals can be developed for spatial-multiplexing detectors. These reliability measures will be beneficial to the development of more efficient detectors. In particular, they would become a great asset in providing a better guideline to the generation of the list $\mathcal{L}$ in Window-Shifting Detector (WSD).

Bibliography

[1] L. Zheng and N. C. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inform. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.

[2] S. M. Alamouti, "A simple transmit diversity techniques for wireless communications," *IEEE J. Select Areas Commun.*, vol. 16, pp. 1451–1458, Oct. 1998.

[3] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block coding for wireless communications: performance results," *IEEE J. Select Areas Commun.*, vol. 17, no. 3, pp. 451–460, Mar. 1999.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.

[5] V. Tarokh, A. Naguib, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criteria in the presence of channel estimation errors, mobility, and multiple paths," *IEEE Trans. Commun.*, vol. 47, no. 2, pp. 199–207, Feb. 1999.

[6] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. Journal*, pp. 41–59, Autumn 1996.

[7] P. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "VBLAST: an architecture for realizing very high data rates over rich scattering wireless channel," *Proc. URSI Int. Symp. Signals, Syst., Electron.*, pp. 295–300, 1998.

[8] *IEEE standard for local and metropolitan area networks - part 16: air interface for fixed and mobile broadband wireless access systems*, 802.16e-2005 and IEEE Std 802.16-2004/Cor1-2005, 2006.

[9] *IEEE standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements - part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications*, 802.11-2007, 2007.

[10] 3rd Generation Partnership Project, "High speed downlink packet access (HSDPA): overall description," *3GPP TS25.308 V7.3.0*, Jun. 2007.

[11] 3rd Generation Partnership Project, "Technical specification group radio access network, physical layer aspects for evolved UTRA (release 7)," *3GPP TS25.814 V1.0.1*, Nov. 2005.

[12] *CDMA2000 hight rate broadcast-multicast packet data air interface specification*, 3GPP2 Standard C.S0054 Rev.1.0, 2004.

[13] "Nortel completes industry's first call over ultra mobile broadband network [online]," 2007, Available : `http://findarticles.com/p/articles/mi_pwwi/is_200703/ai_n18735903`.

[14] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near shannon limit error-correcting coding and decoding : turbo codes," *ICC*, pp. 1064–1070, 1993.

[15] R. G. Gallager, "Low-density parity-check codes," *Cambridge, MA MIT Press*, 1963.

[16] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inform. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[17] M. Tuchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 673–683, Mar. 2002.

[18] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, July 1999.

[19] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[20] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2299–2309, Nov. 2004.

[21] *IEEE standard 802.3 -2005/Cor 1-2006 IEEE standard for information technology - telecommuincations and information exchange between systems - local and metropolitan area networks.*, 802.3 -2005/Cor 1-2006, 2006.

[22] J. G. Proakis and M. Salehi, *Communication systems engineering*, Prentice Hall, 2nd edition, 2002.

[23] J. Hagenauer, E. Offer, and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[24] B. Vucetic and J. Yuan, *Turbo Codes Principles and Applications*, Kluwer Academic Publishers, 2nd edition, 2001.

[25] E. G. Larsson and P. Stoica, "Space-time block coding for wireless communication," *Cambridge*, 2003.

[26] S. Lin and D. J. Costello Jr., *Error Control Coding*, Person Prentice Hall, 2nd edition, 2004.

[27] S. Verdu, *Multiuser detection*, Cambridge University Press, 2nd edition, 1998.

[28] H. V. Poor and S. Verdu, "Probability of error in MMSE multiuser detection," *IEEE Trans. Inform. Theory*, vol. 43, pp. 858–871, May 1997.

[29] U. Fincke and M. Pohst, "Improved methods for calculatin vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, pp. 463–471, Apr. 1985.

[30] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2389–2402, Oct. 2003.

[31] M. Grotschel, L. Lovász, and A. Schriver, *Geometric algorithms and combinatorial optimization*, New York: Springer-Verlag, 2nd edition, 1993.

[32] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1639–1642, May. 1999.

[33] C. P. Schnorr and M. Euchner, "Lattice basis reduction: improved practical algorithms and solving subset sum problems," *Math. Programming.*, vol. 66, pp. 181–191, 1994.

[34] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2201–2214, 2002.

[35] M. Tsai and S. Yousefi, "Dynamic-list joint detection and decoding of LDPC-coded V-BLAST systems," *21st Canadian Conf. on Elec. and Comp. Eng., Niagara Falls, ON.*, May 2008.

[36] M. Tsai and S. Yousefi, "Fixed-complexity list-type iterative joint detection and decoding of LDPC-coded V-BLAST systems," *24st Biennial Symp. on Commun., Kingston, ON.*, June 2008.

[37] D. Chase, "Class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol. IT-18, no. 1, pp. 170–182, Jan. 1972.

[38] J. Leuschner and S. Yousefi, "A new generic maximum-likelihood metric expression for space-time block codes with applications to decoding," *IEEE Trans. Inform. Theory*, vol. 54, no. 2, pp. 888–894, Feb. 2008.

[39] J. Leuschner and S. Yousefi, "On the ML decoding of quasi-orthogonal space-time block codes via sphere decoding and exhaustive search," *IEEE Trans. Wireless Commun, in print.*

[40] J. Leuschner and S. Yousefi, "A new sub-optimal decoder for quasi-orthogonal space-time block codes," *IEEE Commun. Letters, in print.*

[41] T. Clevorn and P. Vary, "The box-minus operator and its application to low-complexity belief propagation decoding," *61st Vehicular Tech. Conf.*, vol. 1, pp. 687–691, 2005.

Appendix A

Proof of Equation (2.10) in Section 2.1.5

According to (2.8),

$$L_{1e}(c_i|\mathbf{y}) = \log \frac{\sum_{\mathbf{c}:c_i=1} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} p(c_j)}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} p(c_j)}$$

in which $p(c_j)$ can be represented in terms of its corresponding LLR values $L(c_j)$ as

$$p(c_j = 0, 1) = \frac{e^{(\frac{1+c_j}{2})L(c_j)}}{1 + e^{L(c_j)}} = \frac{e^{\frac{L(c_j)}{2} + \frac{c_j \cdot L(c_j)}{2}}}{1 + e^{L(c_j)}} = \frac{e^{\frac{c_j \cdot L(c_j)}{2}}}{e^{-\frac{L(c_j)}{2}} + e^{\frac{L(c_j)}{2}}}.$$

Then, $L_{1e}(c_i|\mathbf{y})$ can be further expressed as

$$\begin{aligned} L_{1e}(c_i|\mathbf{y}) &= \log \frac{\sum_{\mathbf{c}:c_i=1} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} \frac{e^{\frac{c_j \cdot L(c_j)}{2}}}{e^{-\frac{L(c_j)}{2}}+e^{\frac{L(c_j)}{2}}}}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c}) \prod_{j,j\neq i} \frac{e^{\frac{c_j \cdot L(c_j)}{2}}}{e^{-\frac{L(c_j)}{2}}+e^{\frac{L(c_j)}{2}}}} \\ &= \log \frac{\sum_{\mathbf{c}:c_i=1} p(\mathbf{y}|\mathbf{c}) \exp(\frac{1}{2}\mathbf{c}_{[i]}^T \cdot \mathbf{L}_{[i]}(\mathbf{c}))}{\sum_{\mathbf{c}:c_i=0} p(\mathbf{y}|\mathbf{c}) \exp(\frac{1}{2}\mathbf{c}_{[i]}^T \cdot \mathbf{L}_{[i]}(\mathbf{c}))} \end{aligned}$$

where $\mathbf{c}_{[i]}$ denotes the subvector of $\mathbf{c}$ obtained by omitting its ith element c_i, and $\mathbf{L}_{[i]}(\mathbf{c})$ denotes the subvector of $\mathbf{L}(\mathbf{c})$ omitting its ith element $L(c_i)$. If the encoded bit vector $\mathbf{c}$ is uniquely mapped to symbol vector $\mathbf{s}$, symbols are transmitted in AWGN channel and $\mathbf{L}(\mathbf{c})$ is obtained from channel decoder, $L_{1e}(c_i|\mathbf{y})$ can be rewritten as

$$L_{1e}(c_i|\mathbf{y}) = \log \frac{\sum_{\mathbf{s}:c_i=1} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{Hs}\|^2 + \frac{1}{2}\mathbf{c}_{[i]}^T \cdot \mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right)}{\sum_{\mathbf{s}:c_i=0} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{Hs}\|^2 + \frac{1}{2}\mathbf{c}_{[i]}^T \cdot \mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right)}$$

Finally, Max-log approximation can be applied to the equation above and $L_{1e}(c_i|\mathbf{y})$ becomes

$$\begin{aligned} L_{1e}(c_i|\mathbf{y}) &= \frac{1}{2}\max_{\mathbf{s}:c_i=1}\left\{-\frac{1}{\sigma^2}\|\mathbf{y}-\mathbf{Hs}\|^2 + \mathbf{c}_{[i]}^T \cdot \mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right\} \\ &- \frac{1}{2}\max_{\mathbf{s}:c_i=0}\left\{-\frac{1}{\sigma^2}\|\mathbf{y}-\mathbf{Hs}\|^2 + \mathbf{c}_{[i]}^T \cdot \mathbf{L}_{2e,[i]}(\mathbf{c}|\mathbf{y})\right\}. \end{aligned}$$

Appendix B

Proof of Equation (3.8) in Section 3.1

Linear MMSE filter is applied to $\tilde{\mathbf{y}}_j$, defined in Section 3.1, to obtain the new MMSE estimate $\tilde{s}_j$

$$\tilde{s}_j = \mathbf{w}_{j,MMSE}^H \tilde{\mathbf{y}}_j$$

such that $E[\|\mathbf{s} - \tilde{\mathbf{s}}\|^2]$ is minimized. We use the principle of orthogonality to find $\mathbf{w}_{j,MMSE}^H$:

$$\begin{aligned} &E[(\mathbf{w}_{j,MMSE}^H \tilde{\mathbf{y}}_j - s_i) \cdot \tilde{\mathbf{y}}_j^H] = 0 \quad \Longrightarrow \\ &\mathbf{w}_{j,MMSE}^H E[\tilde{\mathbf{y}}_j \tilde{\mathbf{y}}_j^H] - E[s_j \tilde{\mathbf{y}}_j^H] = 0 \quad \Longrightarrow \\ &\mathbf{w}_{j,MMSE}^H = E[\tilde{\mathbf{y}}_j \tilde{\mathbf{y}}_j^H]^{-1} E[s_j \tilde{\mathbf{y}}_j^H]. \end{aligned} \tag{B.1}$$

where the first term evaluates to:

$$\begin{aligned} E[\tilde{\mathbf{y}}_j\tilde{\mathbf{y}}_j^H] &= E\left[(\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)+\mathbf{n})(\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)+\mathbf{n})^H\right] \\ &= E\left[(\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)+\mathbf{n})((\mathbf{s}-\tilde{\mathbf{x}}_j)^H\mathbf{H}^H+\mathbf{n}^H)\right] \\ &= E\left[\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)(\mathbf{s}-\tilde{\mathbf{x}}_j)^H\mathbf{H}^H\right]+E\left[\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)\mathbf{n}^H\right] \\ &\quad +E\left[\mathbf{n}(\mathbf{s}-\tilde{\mathbf{x}}_j)^H\mathbf{H}^H\right]+E\left[\mathbf{n}\mathbf{n}^H\right]. \end{aligned}$$

As the second and third term above equal to zero, we have:

$$\begin{aligned} E[\tilde{\mathbf{y}}_j\tilde{\mathbf{y}}_j^H] &= E\left[\mathbf{H}(\mathbf{s}-\tilde{\mathbf{x}}_j)(\mathbf{s}-\tilde{\mathbf{x}}_j)^H\mathbf{H}^H\right]+E\left[\mathbf{n}\mathbf{n}^H\right] \\ &= \mathbf{H}\mathrm{COV}\{\mathbf{s}-\tilde{\mathbf{x}}_j\}\mathbf{H}^H+\sigma_n^2\mathbf{I}_{n_R} \\ &= \mathbf{H}\Delta_j\mathbf{H}^H+\sigma_n^2\mathbf{I}_{n_R} \end{aligned}$$

where $\Delta_j = \mathrm{COV}\{\mathbf{s}-\tilde{\mathbf{x}}_j\}$ is defined in Section 3.1. The second term from Equation (B.1) can be rewritten as

$$\begin{aligned} E\left[s_j\tilde{\mathbf{y}}_j^H\right] &= E\left[s_j((\mathbf{s}-\tilde{\mathbf{x}}_j)^H\mathbf{H}^H+\mathbf{n}^H)\right] \\ &= E\left[s_j s_j^H\mathbf{h}_j\right]+E[s_j(\mathbf{s}-\tilde{\mathbf{x}}_j)_{-j}^H\mathbf{H}_{-j}^H]+E[s_j\mathbf{n}^H] \end{aligned}$$

where $\mathbf{h}_j$ represents the jth column of $\mathbf{H}$, $\mathbf{x}_{-j}$ represent the subvector of vector $\mathbf{x}$ after excluding the jth element and $\mathbf{H}_{-j}$ represents the matrix after excluding the jth column of $\mathbf{H}$. As the second and third term above equal to zero, the right hand side of the equation can be further derived as

$$E\left[s_j\tilde{\mathbf{y}}_j^H\right] = E\left[s_j s_j^H\right]\mathbf{h}_j = E_s\mathbf{H}\mathbf{e}_j$$

where E_s is the average signal energy and $\mathbf{e}_j$ denotes an $n_R \times 1$ vector with all zero entries, except for the jth entry being 1. Therefore,

$$\mathbf{w}_{j,MMSE} = E_s(\mathbf{H}\Delta_j\mathbf{H}^H + \sigma_n^2\mathbf{I}_{n_R})^{-1}\mathbf{H}\mathbf{e}_j.$$

Appendix C

LDPC Model Matrices

Model matrix $\mathbf{A}_{bm}$ for rate 1/2 WiMAX A Code and WiMAX B Code:

$$\left[\begin{array}{cccccccccccccccccccccccc}
-1 & 94 & 73 & -1 & -1 & -1 & -1 & -1 & 55 & 83 & -1 & -1 & 7 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & 27 & -1 & -1 & -1 & 22 & 79 & 9 & -1 & -1 & -1 & 12 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & 24 & 22 & 81 & -1 & 33 & -1 & -1 & -1 & 0 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
61 & -1 & 47 & -1 & -1 & -1 & -1 & -1 & 65 & 25 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 39 & -1 & -1 & -1 & 84 & -1 & -1 & 41 & 72 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & -1 & -1 & 46 & 40 & -1 & 82 & -1 & -1 & -1 & 79 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 95 & 53 & -1 & -1 & -1 & -1 & -1 & 14 & 18 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 \\
-1 & 11 & 73 & -1 & -1 & -1 & 2 & -1 & -1 & 47 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 \\
12 & -1 & -1 & -1 & 83 & 24 & -1 & 43 & -1 & -1 & -1 & 51 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 \\
-1 & -1 & -1 & -1 & -1 & 94 & -1 & 59 & -1 & -1 & 70 & 72 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 \\
-1 & -1 & 7 & 65 & -1 & -1 & -1 & -1 & 39 & 49 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 \\
43 & -1 & -1 & -1 & -1 & 66 & -1 & 41 & -1 & -1 & -1 & 26 & 7 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0
\end{array}\right]$$

Appendix D

The Box-Plus and Box-Minus Operators

Let us consider a special algebra operation of LLR-value of modulo-2 addition of two binary variables u and v:

$$L(u \oplus v) = \log \frac{P(u \oplus v = 1)}{P(u \oplus v = 0)} \tag{D.1}$$

$$= \log \frac{P(u = 1)P(v = 1) + P(u = 0)P(v = 0)}{P(u = 0)P(v = 1) + P(u = 1)P(v = 0)} \tag{D.2}$$

where

$$P(u = 1) = \frac{e^{L(u)}}{1 + e^{L(u)}} \tag{D.3}$$

and

$$P(u = 0) = \frac{1}{1 + e^{L(u)}}. \tag{D.4}$$

Using the Equation (D.3) and (D.4), Equation (D.2) can be presented in terms of individual LLR-value of u and v.

$$L(u \oplus v) = \log \frac{1 + e^{L(u)} e^{L(v)}}{e^{L(u)} + e^{L(v)}}. \tag{D.5}$$

Using the definition of hypertangent

$$\tanh\left(\frac{u}{2}\right) = \frac{e^u - 1}{e^u + 1} \tag{D.6}$$

and inverse hypertangent

$$\tanh^{-1}(x) = \frac{1}{2} \log\left(\frac{1+x}{1-x}\right), \tag{D.7}$$

we can simplify the expression to

$$L(u \oplus v) = 2 \tanh^{-1}\Big(\tanh(u/2) \cdot \tanh(v/2)\Big). \tag{D.8}$$

This special algebraic operation in the log-likelihood domain between two random variables explains the fundamental idea of the box-plus operation. Similar to regular addition in $\mathbb{R}$, we need an addition operator in the log-likelihood domain. Therefore, [23] defines a box-plus operator and use the symbol $\boxplus$ as the notation. This special arithmetic operation allows us to perform addition in the log-likelihood domain and is shown in Equation(D.9) where addition between two LLR-values is defined as the LLR-value of modulo-2 addition of the underlying random variables:

$$L(u_1) \boxplus L(u_2) \triangleq L(u_1 \oplus u_2). \tag{D.9}$$

From now on, we will use L_1 to denote $L(u_1)$ and L_2 to denote $L(u_2)$ to simplify our discussion. As previously derived, a box-plus operator can be expressed in a similar form of Equation (D.5) and Equation (D.8).

$$\begin{aligned} L_1 \boxplus L_2 &= \log \frac{1 + e^{L_1} e^{L_2}}{e^{L_1} + e^{L_2}} && \text{(D.10)} \\ &= 2 \tanh^{-1}\Big(\tanh(L_1/2) \cdot \tanh(L_2/2)\Big). && \text{(D.11)} \end{aligned}$$

We also have:

$$L_1 \boxplus \infty = L_1, \tag{D.12}$$

$$L_1 \boxplus -\infty = -L_1, \tag{D.13}$$

$$L_1 \boxplus 0 = 0. \tag{D.14}$$

This special algebraic operation is defined according to the nature of the SPA. Not only does it perform addition in the log-likelihood domain, it also takes care of the marginalization of the probabilistic values. In general, when more than two random variables are involved, the box-up operation can be written in the form shown in Equation (D.15).

$$\sum_{i=1}^{n_A} \boxplus L_i = 2 \tanh^{-1} \left(\prod_{i=1}^{n_A} \tanh(L_i/2) \right). \tag{D.15}$$

It's clear that computing hypertangent or inverse-hypertangent requires much higher computational complexity than addition or multiplication in $\mathbb{R}$ domain. With such high computational complexity, it would defeat our purpose of using LLR. To overcome this problem, we can use the Jacobian logarithm to come up with approximations for the box-plus operator. Assuming $\delta_1 > \delta_2$, the Jacobian logarithm is derived

as follows:

$$\begin{aligned}
\log(e^{\delta_1} + e^{\delta_2}) &= \log\left(e^{\delta_1}\left(1 + \frac{e^{\delta_2}}{e^{\delta_1}}\right)\right) && \text{(D.16)}\\
&= \log\Big(e^{\delta_1}(1 + e^{\delta_2 - \delta_1})\Big) && \text{(D.17)}\\
&= \log\Big(e^{\delta_1}\left(1 + e^{-(\delta_1 - \delta_2)}\right)\Big) && \text{(D.18)}\\
&= \log(e^{\delta_1}) + \log\Big(1 + e^{-(\delta_1 - \delta_2)}\Big) && \text{(D.19)}\\
&= \delta_1 + \log\Big(1 + e^{-|\delta_1 - \delta_2|)}\Big). && \text{(D.20)}
\end{aligned}$$

In general, we have

$$\begin{aligned}
\log\left(e^{\delta_1} + e^{\delta_2}\right) &= \max(\delta_1, \delta_2) + \log\left(1 + e^{-|\delta_1 - \delta_2|}\right) && \text{(D.21)}\\
&= \max(\delta_1, \delta_2) + f_c(|\delta_1 - \delta_2|) && \text{(D.22)}\\
&\simeq \max(\delta_1, \delta_2). && \text{(D.23)}
\end{aligned}$$

If we ignore the second term on the right hand side of the Equation (D.22), we will end up with the maximum logarithm (max-log) approximation as shown in Equation (D.23). The second term $f_c(|\delta_1 - \delta_2|)$ in Equation (D.22) is called the *correction term* or *correction function*, and it can be stored in a small look-up table as a function of $|\delta_1 - \delta_2|$. Now, we can manipulate the box-plus operator as follows:

$$\begin{aligned}
L_1 \boxplus L_2 &= \log\left(1 + e^{L_1}e^{L_2}\right) - \log\left(e^{L_1} + e^{L_2}\right) && \text{(D.24)}\\
&\cong \operatorname{sgn}(L_1)\operatorname{sgn}(L_2)\min(|L_1|, |L_2|) + f_+(|L_1 + L_2|)\\
&\quad -f_+(|L_1 - L_2|) && \text{(D.25)}\\
&\cong \operatorname{sgn}(L_1)\operatorname{sgn}(L_2)\min(|L_1|, |L_2|) && \text{(D.26)}
\end{aligned}$$

where $f_+(x) = \log(1 + e^{-x})$ is a *correction function*. Instead of taking the maximum of L_1 and L_2, we take the minimum because the term $\log\left(e^{L_1} + e^{L_2}\right)$ is in the denominator. The approximation, shown in Equation (D.25), is called the lookup-sum because it requires a look-up table to approximate the correction term. Normally, if we do not use any approximation to calculate LLRs, the algorithm is called the MAP algorithm in which decision rule is based on maximizing APP. If lookup-sum approximation is used to compute the LLRs, we call the algorithm the log-MAP algorithm. This algorithm results in negligible performance loss in comparison with the MAP algorithm but is easier to implement. The log-MAP algorithm requires only additions and comparisons to compute the LLRs. The approximation shown in Equation (D.26) is called min-sum because it is simply calculated by using minimization. This algorithm ignores the correction functions and uses the concept of max-log approximation. Therefore, it is referred to as the max-log-MAP algorithm.

A box-minus operator is another special algebraic operator that acts as a complement to a box-plus operator. It is a subtraction operator in the log-likelihood domain. [41] defines a box-minus operator and use the symbol $\boxminus$ to denote it.

$$L_3 \boxminus L_4 = \log \frac{1 - e^{L_3} e^{L_4}}{e^{L_3} - e^{L_4}} \tag{D.27}$$

$$= 2 \tanh^{-1} \left(\frac{\tanh(L_3/2)}{\tanh(L_4/2)} \right). \tag{D.28}$$

We also requires three additional constrains:

$$L_3 \boxminus \infty = L_3, \tag{D.29}$$

$$L_3 \boxminus -\infty = -L_3, \tag{D.30}$$

$$0 \boxminus L_4 = 0. \tag{D.31}$$

Note that this operator is defined only when

$$|L_3| < |L_4| \tag{D.32}$$

and also it is not commutative (i.e., $L_3 \boxminus L_4 \neq L_4 \boxminus L_3$). This operator is somewhat similar to the regular subtraction in $\mathbb{R}$.

Approximations for the box-minus operator also exists. Similar to box-plus operators, these approximations are derived from the Jacobian logarithm as well as max-log approximation. However, we can simplify the equations even more because there is an extra constraint (i.e., $L_3 \boxminus L_4$ defined only when $|L_3| < |L_4|$) for the box-minus operator:

$$\begin{aligned} L_3 \boxminus L_4 &\cong \operatorname{sgn}(L_3)\operatorname{sgn}(L_4)\min(|L_3|,|L_4|) + f_-(|L_3+L_4|) \\ &\quad -f_-(|L_3-L_4|) && \text{(D.33)} \\ &\cong \operatorname{sgn}(L_4)L_3 + f_-(|L_3+L_4|) - f_-(|L_3-L_4|) && \text{(D.34)} \\ &\cong \operatorname{sgn}(L_3)\operatorname{sgn}(L_4)\min(|L_3|,|L_4|) && \text{(D.35)} \\ &\cong \operatorname{sgn}(L_4)L_3. && \text{(D.36)} \end{aligned}$$

Again, $f_-(x) = \log(1-e^{-x})$ is a correction function and can be stored in a look-up table. Similar to the box-plus operator, the log-MAP algorithm has almost the same error performance as the MAP algorithm and therefore is practically optimal.

When box-plus and box-minus operators work together in a single equation, the order of elements in the equation is very important since the condition in Equation

(D.32) has to be satisfied when a box-minus operator is involved. A general equation can be obtained as follows:

$$L_1 \boxplus L_2 \boxplus L_3 ... \boxplus L_{n_A} \boxminus L_{n_A+1} \boxminus L_{n_A+2} ... \boxminus L_{n_B} \tag{D.37}$$

$$= \quad 2\tanh^{-1}\left(\frac{\prod_{i=1}^{n_A}\tanh(L_i/2)}{\prod_{i=n_A+1}^{n_B}\tanh(L_i/2)}\right) \tag{D.38}$$

$$= \quad \left(\sum_{i=1}^{n_A}\boxplus L_i\right) \boxminus \left(\sum_{i=n_A+1}^{n_B}\boxplus L_i\right) \tag{D.39}$$

where condition (D.32) in this case will become:

$$\left|\sum_{i=1}^{n_A}\boxplus L_i\right| < \left|\sum_{i=n_A+1}^{n_B}\boxplus L_i\right|. \tag{D.40}$$

If Equation (D.40) is not satisfied, Equation (D.37) will become undefined.