

國立交通大學

電信工程學系碩士班

碩士論文

IEEE 802.11n 多重輸出輸入最佳化接收器設計

Optimum MIMO Receiver for IEEE 802.11n System



研究生：吳俊穎

指導教授：吳文榕 博士

中華民國九十六年七月

IEEE 802.11n 多重輸出輸入最佳化接收器設計

Optimum MIMO Receiver Design for IEEE 802.11n System

研究生：吳俊穎

Student : Jiun-Ying Wu

指導教授：吳文榕 博士

Advisor : Dr. Wen-Rong Wu

國立交通大學

電信工程學系碩士班

碩士論文

A Thesis

Submitted to Department of Communication Engineering

College of Electrical and Computer Engineering

National Chiao-Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

In

Communication Engineering

July 2007

Hsinchu, Taiwan, Republic of China

中華民國九十六年七月

IEEE 802.11n 多重輸出輸入最佳化接收器設計

研究生：吳俊穎

指導教授：吳文榕 教授

國立交通大學電信工程學系碩士班

摘要

IEEE 802.11n 使用了正交分頻多工 (OFDM) 與多重輸入輸出 (MIMO) 作為其基頻之傳輸模式。與傳統的無限網路系統相比，802.11n 可以提供更為高速的傳輸速度。然而，多重輸入輸出接收器的設計是一個非常困難的工作。最大相似度 (ML) 接收器擁有最好的效能，但是卻要付出極高的複雜度去實現。球型接收器 (SD) 可以有有效的簡化運算複雜度，然而其運算量非常不固定，不利於硬體上之實現。在本論文中，我們將提出一個新的演算法來解決這個問題。我們所提出的演算法，在效能上能接近最佳的 ML 接收器，並且可以大幅降低複雜度；與 SD 相比，我們提出的演算法擁有固定的運算量。另外一個優點是，我們提出的演算法，可以容易的在複雜度與效能中做出取捨。除此之外，我們可以延伸此演算法來計算軟性位元，以提供軟性解碼器之使用。模擬的結果顯示出，在與傳統的方法(最小均方差等化器 (MMSE)) 相比，我們所提出的演算法在效能上能有超過 10 dB 的提升。

Optimum MIMO Receiver Design for IEEE 802.11n System

Student: Jiun-Ying Wu

Advisor: Dr. Wen-Rong Wu

Department of Communication Engineering
National Chiao-Tung University

Abstract

IEEE 802.11n uses the multiple input multiple output (MIMO) with orthogonal frequency division multiplexing (OFDM) as its transmission scheme. Compare with conventional wireless network system, IEEE 802.11n can provide a very high throughput. However, the receiver design for MIMO systems remains a very challenging task. As we known, the maximum-likelihood detector (MLD) is the optimum receiver, but the cost of implementation is very high due to its high complexity. In this thesis, we propose a new algorithm to overcome the problem. The proposed algorithm can achieve the performance of MLD, and reduce the complexity significantly. Meanwhile, the throughput of the proposed algorithm is constant, and it can be easy to change complexity by modifying the parameter in our algorithm. Simulation with 802.11n system shows that our proposed algorithm has significant gain compared with conventional method.

誌謝

本篇論文得以順利完成，首先要特別感謝我的指導教授 吳文榕博士，在課業學習與論文研究上不厭其煩的引導我正確的方向。

另外，我要感謝許兆元學長、謝弘道學長、李俊芳學長與楊華龍學長等在研究上不吝指導，且同時感謝寬頻傳輸與訊號處理實驗室所有同學與學弟妹們的幫忙。



Contents

1	Introduction	1
2	IEEE 802.11n Transmitter Overview	4
2.1	Introduction	4
2.2	Packet Format	4
2.3	Cyclic Shift	5
2.4	High-Throughput Preamble	6
2.4.1	HT SINGAL Field	7
2.4.2	HT-STF training symbol	9
2.4.3	HT-LTF training symbol	9
2.5	Data Field	9
2.5.1	Scrambler	9
2.5.2	FEC	10
2.5.3	Stream Parser	12
2.5.4	Frequency Interleaver	12
2.5.5	QAM Mapping	13
2.5.6	OFDM modulation	15
2.6	Channel Model	15
3	IEEE 802.11n Receiver Design	17
3.1	Packet Detection	17
3.2	Frequency Synchronization	19
3.3	Symbol Timing Offset	21
3.4	Channel Estimation	22

3.5	Phase Tracking	24
3.6	Sampling Frequency Offset	25
4	MIMO Detector	27
4.1	Problem Definition	27
4.2	Maximum Likelihood Detector	28
4.3	Sphere Decoding	28
4.4	Proposed Method	31
4.5	Complexity Analysis	35
4.6	Simulations	37
5	MIMO Soft Bit Demapping	45
5.1	Bit-interleaved Coded Modulation	45
5.2	Log Likelihood Ratio	45
5.3	List Sphere Decoding	47
5.4	Soft Output of Proposed Algorithm	48
5.5	Simulations	50
6	Simulations With IEEE 802.11n System	59
7	Conclusions	66
7.1	Conclusion	66
7.2	Future Work	66



List of Figures

1	802.11n Block Diagram	4
2	802.11n Packet Format	5
3	HT Singal Field	8
4	Constellation for Legacy Signal Field and the HT signal Field	8
5	Data Scrambler	10
6	Convolution Encoder	10
7	Puncture Process	11
8	QAM Modulation	14
9	Singal flow structure of the delay and correlation algorithm . .	17
10	Process of Packet Detection	18
11	Preamble for CFO Estimation	20
12	Preamble for Symbol Timing	22
13	Histogram of Number of Candidates	31
14	Projection in each group	32
15	BER comparison for 2×2 16-QAM(I)	39
16	BER comparison for 2×2 16-QAM(II)	39
17	BER comparison for 3×3 16-QAM(I)	40
18	BER comparison for 3×3 16-QAM(II)	40
19	BER comparison for 4×4 16-QAM(I)	41
20	BER comparison for 4×4 16-QAM(II)	41
21	BER comparison for 2×2 64-QAM(I)	42
22	BER comparison for 2×2 64-QAM(II)	42
23	BER comparison for 3×3 64-QAM(I)	43

24	BER comparison for 3×3 64-QAM(II)	43
25	BER comparison for 4×4 64-QAM(I)	44
26	BER comparison for 4×4 64-QAM(II)	44
27	BER comparison of soft demapping for 2×2 16-QAM system(I)	52
28	BER comparison of soft demapping for 2×2 16-QAM system(II)	52
29	BER comparison of soft demapping for 3×3 16-QAM system(I)	53
30	BER comparison of soft demapping for 3×3 16-QAM system(II)	53
31	BER comparison of soft demapping for 4×4 16-QAM system(I)	54
32	BER comparison of soft demapping for 4×4 16-QAM system(II)	54
33	BER comparison of soft demapping for 2×2 64-QAM system(I)	55
34	BER comparison of soft demapping for 2×2 64-QAM system(II)	55
35	BER comparison of soft demapping for 3×3 64-QAM system(I)	56
36	BER comparison of soft demapping for 3×3 64-QAM system(II)	56
37	BER comparison of soft demapping for 4×4 64-QAM system(I)	57
38	BER comparison of soft demapping for 4×4 64-QAM system(II)	57
39	BER comparison of hard decision and soft demapping for 3×3 64-QAM system	58
40	BER comparison of hard decision and soft demapping for 4×4 64-QAM system	58
41	PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Channel B)	61
42	PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Channel D)	62

43	PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Rayleigh uncorrelated Channel)	63
44	PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 1/2 code rate, Rayleigh uncorrelated Channel)	64
45	PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 2×2 , 16QAM, 1/2 code rate, Rayleigh uncorrelated Channel)	65



List of Tables

1	Cyclic shift for non- HT portion of the packet	6
2	Cyclic shift for HT portion of the packet	6
3	Normalization Factor	13
4	Pilot Value for 20MHz	15
5	Pilot Value for 40MHz	16
6	Number of Multiplications required in proposed and ML algorithm	37
7	Number of Multiplications required in proposed and SD algorithm	37



1 Introduction

In recent years, wireless data communication has been grown extensively. As a result, existing wireless communication systems can not satisfy the fast-growing demand. It is know that increasing the bandwidth can effectively increase the data transmission rate. However, the resource for wireless communication is limited. Thus, how to improve the spectrum efficiency of wireless communication systems becomes an important issue.

One effectively way to improve capacity is to use the multiple transmit and receive antennas, resulting in multi-input multi-output (MIMO) systems. Conventionally, the communication system is implemented as a single-input single-output (SISO) system. In 1998, Telatar and Foschini have shown that the MIMO channel capacity can grow approximately linearly with the number of antennas used [1], [2]. MIMO techniques can basically group into two categories. The first one aims to improve the power efficiency and transmission reliability by maximizing spatial diversity. One popular example of such a system is the space-time block codes (STBC) [3]. The second category uses a layered approach to increase capacity. One popular example is the vertical-Bell Laboratories layered space-time (V-BLAST) architecture [4], [5]. In this case, independent data streams are transmitted over different antennas to increase the data rate.

Another way to enhance the spectrum efficiency is to use the orthogonal frequency division multiplexing (OFDM) technique. This technique, proposed by Salzberg in 1967 [6], evolves to becomes a digital multi-carrier modulation scheme. The main idea of ODEM is to spilt a high rate stream

into a number of low rate streams, and transmit them simultaneously over different sub-carriers. Each sub-carrier is modulated with a conventional modulation scheme, maintaining its data rate similar to conventional single-carrier modulation scheme in the same bandwidth. The primary advantage of OFDM is its ability to convert the multi-path frequency-selective fading channel into a number of flat sub-channels. The receiver can easily adapt to channel conditions with simple equalization. In implementation, OFDM symbols can be generated by the efficient Fast Fourier transform (FFT) algorithm. Nowadays, OFDM has developed into popular schemes for wideband digital communication systems, such as terrestrial digital video broadcasting (DVB-T), the IEEE 802.11a/g wireless local area network (WLAN) and the IEEE 802.16e Worldwide Interoperability for Microwave Access (WiMAX) Wireless MANs. Combining with the MIMO structure, OFDM can further enhance the spectrum efficiency. The MIMO-OFDM scheme has been considered as the main solution for enhancing the data rates in next-generation wireless communication systems, such as IEEE 802.11n High Throughput WLAN and the Fourth-Generation Cellular Communications System (4G).

However, the receiver design for MIMO systems between the cost-performance tradeoff remains a very challenging task. The maximum-likelihood detector (MLD) is the optimum receiver, but the cost of implementation is very high due to its high complexity. Another popular approach is called sphere decoding (SD) [7]; it can achieve the performance of MLD and significantly reduce the computational complexity. But the disadvantage is that the throughput of SD is not constant, it is difficult to have an efficient hardware implementation. In this thesis, we propose a new algorithm to overcome the

problem. The proposed algorithm can reach the performance of MLD, and reduce the complexity significantly. Meanwhile, the throughput of the proposed algorithm is constant. We also extend this algorithm to accommodate the MIMO soft-bit demapping problem. This will be very useful in modern bit-interleaved coded modulation systems (BICM).

The rest of this thesis is organized as follows. In Chapter 2, we review the specification of 802.11n Draft 2.0, and the transceiver structure of 802.11n. In Chapter 3, we present the design of the receiver structure and describe synchronization method. In Chapter 4, we discuss the MIMO detection algorithm including MLD, SD and the proposed algorithm. Also, we will show the simulation results, conduct performance comparison, and analyze computational complexity in this chapter. In Chapter 5, we extend the proposed algorithm to reduce the computational complexity in the MIMO soft-bit demapping problem. Finally, we give some conclusions and outline future works in Chapter 6.

2 IEEE 802.11n Transmitter Overview

2.1 Introduction

IEEE802.11n is the next generation wireless local area networks (WLAN) standard. The transmission technique is similar to the previous IEEE802.11a standard, except for that the MIMO (multiple-input multiple-output) technology is introduced. The MIMO technology uses multiple transmitter and receiver antennas to increase data throughput (with spatial multiplexing), or increase range (with space-time coding).

The publication of the standard is currently expected in September 2008. In this thesis, we will target on the latest specification "IEEE 802.11n Draft 2.00".

Figure 1 shows the transmitter block diagram of the specification.

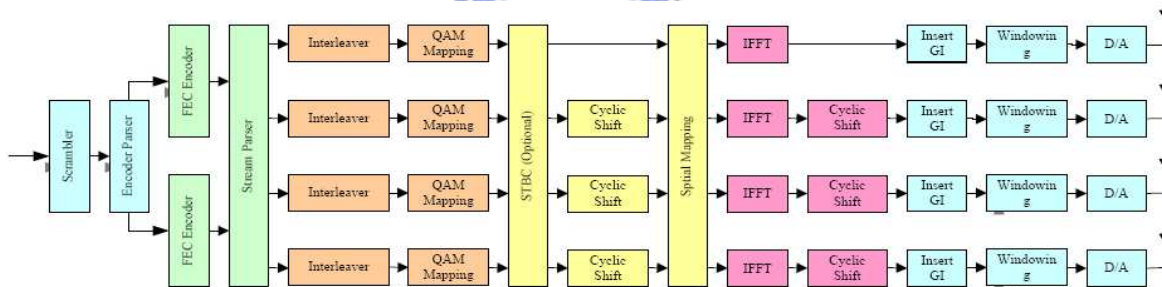


Figure 1: 802.11n Block Diagram

2.2 Packet Format

There are three kinds of packet format shown in fig.2.

The elements of packets are:

L-STF: Legacy Short Training Field

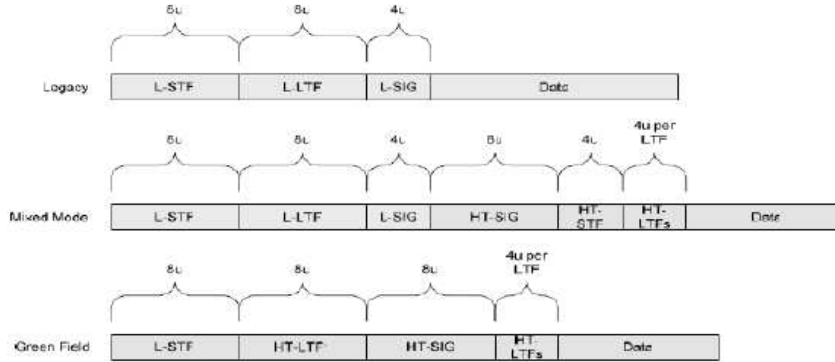


Figure 2: 802.11n Packet Format

L-LTF: Legacy Long Training Field

L-SIG: Legacy Singal Field

HT-SIG: High Throughput Singal Field

HT-STF: High Short Training Field

HT-LTF1: First High Long Training Field

HT-LTF's: Additional High Long Training Field

Data: The Data Field

2.3 Cyclic Shift

Cyclic shifts are used to prevent unintentional beamforming when the same signal or scaled multiples of one signal are transmitted through different spatial streams or transmit chains. A cyclic shift of duration T_{CS} on a signal $s(t)$ on interval $0 \leq t \leq T$ is defined as follows,

$$s_{cs}(t) = \begin{cases} s(t - T_{CS}) & T_{CS} < t \leq T \\ s(t - T_{CS} + T) & 0 \leq t \leq T_{CS} \end{cases} \quad (1)$$

The cyclic shift is applied to each OFDM symbol in the packet separately.

T_{CS}^{tx} values for the non-HT portion of the packet				
Number of Tx Chains	Cyclic shift for Tx chain 1	Cyclic shift for Tx chain 2	Cyclic shift for Tx chain 3	Cyclic shift for Tx chain 4
1	0ns	-	-	-
2	0ns	-200 ns	-	-
3	0ns	-100 ns	-200 ns	-
4	0ns	-50ns	-100 ns	-150ns

NOTE: With more than four TX chains, each cyclic shift on the additional TX chains shall not be less than -200 ns nor greater than 0 ns.

Table 1: Cyclic shift for non- HT portion of the packet

T_{CS}^{sts} values for HT portion of the packet				
Number of space time streams	Cyclic shift for space time stream 1	Cyclic shift for space time stream 2	Cyclic shift for space time stream 3	Cyclic shift for space time stream 4
1	0ns	-	-	-
2	0ns	-400 ns	-	-
3	0ns	-400 ns	-200 ns	-
4	0ns	-400 ns	-200 ns	-600 ns

Table 2: Cyclic shift for HT portion of the packet

Table 1 specifies the values for the cyclic shifts that are applied in the non-HT short training field (in an HT mixed format packet), the non-HT long training field, and non-HT SIGNAL field. It also applies to the HT SIGNAL field in an HT mixed format packet. The values of the cyclic shifts to be used during the HT portion of the HT mixed format preamble and the data portion of the frame are specified in Table 2.

2.4 High-Throughput Preamble

The high throughput (HT) preambles are defined in HT mixed and in green-field formats to carry the required information to operate in a system with multiple transmit and multiple receiver antennas.

In the mixed format, packets in this mode packets are transmitted with a

preamble compatible with the legacy mode. The legacy Short Training Field (L-STF), the legacy Long Training Field (L-LTF) and the legacy signal field are transmitted, so they can be decoded by legacy 802.11a/g devices. The rest of the packet has a new format. In this mode, the receiver shall be able to decode both the Mixed Mode packets and legacy packets.

In the greenfield format, all of the non-HT fields are omitted. The specific HT fields used are:

- HT SIGNAL Field(HT-SIG): provides all the information required to interpret the HT packet format,
- One HT Short Training Field (HT-STF) for automatic gain control (AGC) convergence, timing acquisition, and coarse frequency acquisition,
- One or several HT Long Training Fields (HT-LTF): provided as a way for the receiver to estimate the channel between each spatial mapper input and receive chain. The first HT-LTFs (Data HTLTFs) are necessary for demodulation of the HT-Data portion of the PPDU, and are followed, for sounding packets only, by optional HT-LTFs (Extension HT-LTFs) to sound extra spatial dimensions of the MIMO channel.

2.4.1 HT SINGAL Field

The high-throughput signal field is used to carry information required to interpret the HT packet formats. The fields of the HT SIGNAL field are described in table

The HT-SIG is composed of two parts, HT-SIG1 and HT-SIG2, each containing 24 bits, as shown in Figure 3 (Format of HT-SIG1 and HT-SIG2). All the fields in the HT-SIG are transmitted LSB first, and HT-SIG1 is

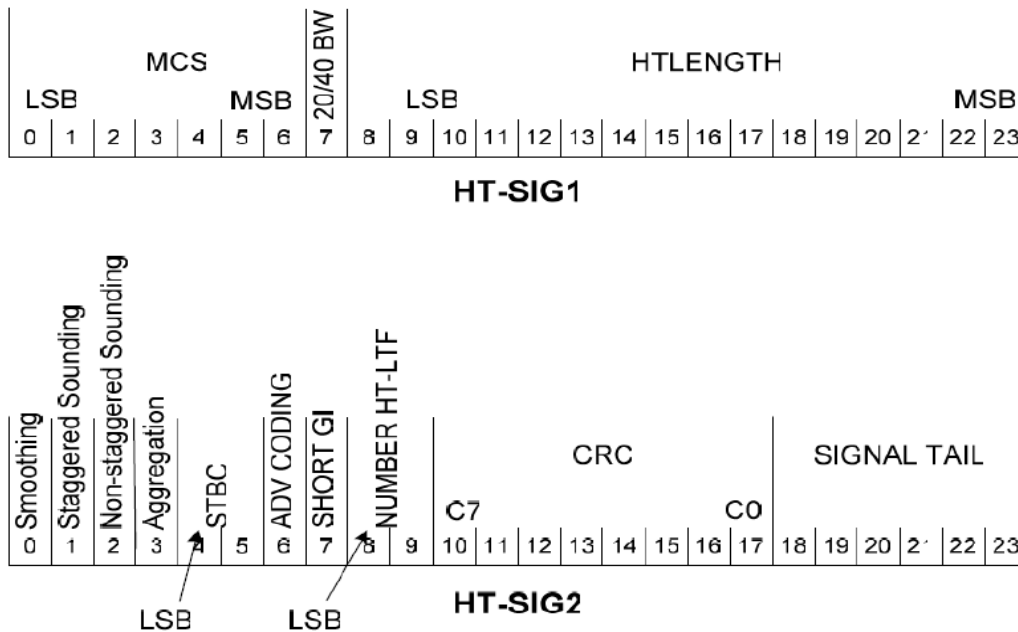


Figure 3: HT Singal Field

transmitted before HT-SIG2.

The HT-SIG parts are encoded, interleaved, BPSK mapped, and have pilots inserted following the steps described in IEEE 802.11a standard. The result of BPSK modulation constellation is rotated by 90 degree relative to the traditional constellation shown in Figure 4

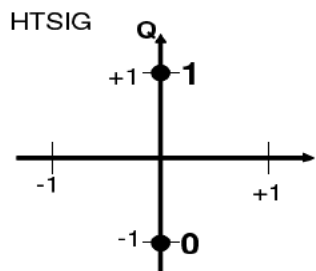


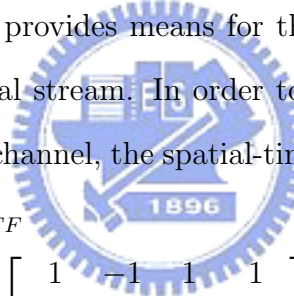
Figure 4: Constellation for Legacy Signal Field and the HT signal Field

2.4.2 HT-STF training symbol

The purpose of the HT STF training field is to improve AGC training in a multi transmit and multi-receive system. The duration of the HT-STF is $4\mu\text{sec}$. The frequency sequence used to construct the HT-STF in 20MHz transmission is identical to the legacy STF; in 40MHz transmission the HT-STF is constructed from the 20MHz version by frequency shifting and duplicating, and rotating the upper sub-carriers by 90° .

2.4.3 HT-LTF training symbol

The HT long training field provides means for the receiver to estimate the channel between each spatial stream. In order to estimate all the response for each pass of the MIMO channel, the spatial-time mapping matrix is used. The mapping matrix P_{HTLTF}


$$\mathbf{H}_k = \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ -1 & 1 & 1 & 1 \end{bmatrix} \quad (2)$$

2.5 Data Field

2.5.1 Scrambler

The DATA field shall be scrambled with a length-127 frame-synchronous scrambler. The octets of the PSDU are placed in the transmit serial bit stream, bit 0 first and bit 7 last. The frame synchronous scrambler uses the generator polynomial $S(x)$ as follows, and is illustrated in Figure ??.

The 127-bit sequence generated repeatedly by the scrambler shall be (left-most used first), 00001110 11110010 11001001 00000010 00100110 00101110 10110110 00001100 11010100 11100111 10110100 00101010 11111010 01010001

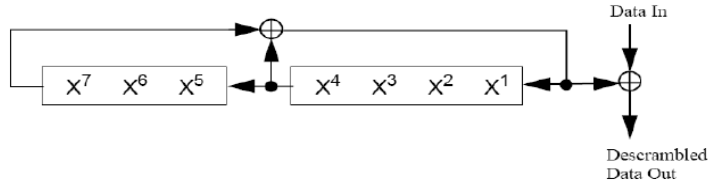


Figure 5: Data Scrambler

10111000 1111111, when the all ones initial state is used. The same scrambler is used to scramble transmit data and to descramble receive data. When transmitting, the initial state of the scrambler will be set to a pseudo random non-zero state. The seven LSBs of the SERVICE field will be set to all zeros prior to scrambling to enable estimation of the initial state of the scrambler in the receiver.

2.5.2 FEC

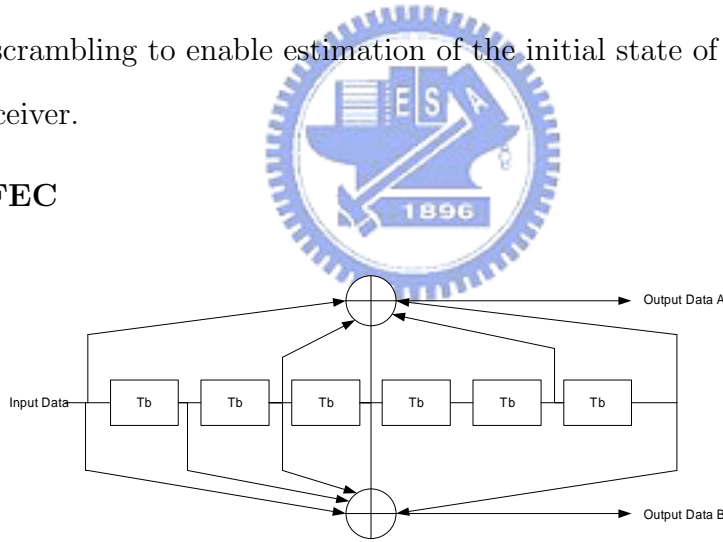


Figure 6: Convolution Encoder

The mandatory coder is the convolutional code (CC) encoder and the optional one is a low-density parity-check (LDPC) encoder. We only focus our research in CC encoder. The CC encoder is defined by the Figure 6. A single CC encoder is used when $N_{ss} = 1$ or $N_{ss} = 2$, two CC encoders are used when $N_{ss} = 3$ or $N_{ss} = 4$.

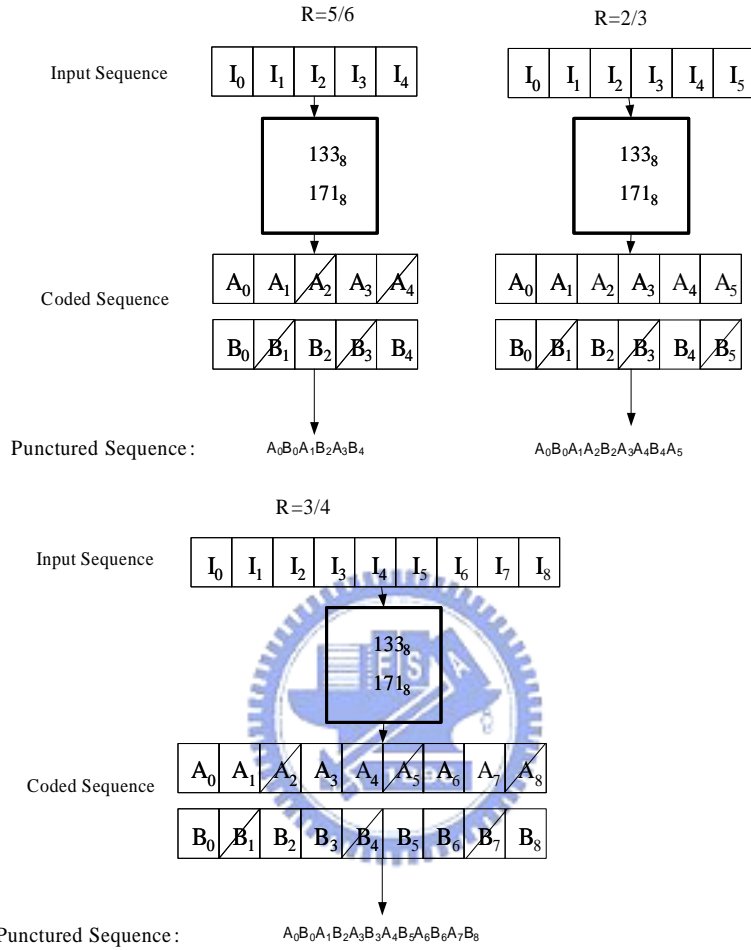


Figure 7: Puncture Process

If two encoders are used, the data scrambled bits are divided between the encoders by sending alternating bits to different encoders.

The CC encoder output only provides the encoding output with $\frac{1}{2}$ code rate. In order to support a variable code rate, we can use puncture operation after encoding. The procedure of puncturing is shown in Figure 7. There are three kinds of code rates ($\frac{2}{3}, \frac{3}{4}, \frac{5}{6}$) defined.

2.5.3 Stream Parser

After coding and puncturing, the data bit streams at the output of the CC encoders are parsed to multiple spatial streams. Define $s = \max\{\frac{N_{BPSS}(i_{ss})}{2}, 1\}$, which is the number of bits assigned to a single axis (real or imaginary) in a constellation point in spatial stream i_{ss} .

Consecutive blocks of $s(i_{ss})$ bits are assigned to different spatial streams in a round robin fashion.

If two encoders are present, the output of each encoder is used alternately for each round robin cycle, i.e., at the beginning bits from the output of first encoder are fed into all spatial streams, and then bits from the output of second encoder are used and so on.

2.5.4 Frequency Interleaver

In order to have frequency diversity gain and reduce the spatial correlation of the MIMO channel, the frequency interleaver is used. The interleaving is defined with three permutations, and the interleaving block size (N_{CBPSS} bits). Parameters are defined using a table (Table ??). The first permutation is defined by the rule:

$$\begin{aligned} i &= N_{ROW}(k \bmod N_{COL}) + \text{floor}(k/N_{COL}) \\ k &= 0, 1, \dots, N_{CBPSS} - 1 \end{aligned} \quad (3)$$

The second permutation is defined by the rule:

$$\begin{aligned} j &= s(i_{ss}) \times \text{floor}(i/s(i_{ss})) \\ &+ (i + N_{CBPSS}(i_{ss}) - \text{floor}(N_{COL} \times i/N_{CBPSS}(i_{ss}))) \bmod s(i_{ss}) \\ i &= 0, 1, \dots, N_{CBPSS} - 1 \end{aligned} \quad (4)$$

The third permutation is defined by the rule:

$$r = (j - ((i_{ss} - 1) \times 2) \bmod 3 + 3 \times ((i_{ss} - 1/3) \times N_{ROT} \times N_{BPSCS}(i_{ss})))$$

$$j = 0, 1, \dots, N_{CBPSS} - 1$$

2.5.5 QAM Mapping

The encoded and interleaved binary serial output data shall be divided into groups of N_{BPSC} (1, 2, 4, or 6) bits for BPSK, QPSK, 16QAM and 64QAM. The conversion shall be performed according to Gray-coded constellation mappings, illustrated in Figure 8, with the input bit, b_0 , being the earliest in the stream.

The average power for different modulation size should be normalized. The output values, d , are formed by multiplying the resulting $(I+jQ)$ value by a normalization factor K_{mod} , as described in Equation 5.

$$d = (I + jQ) \times K_{mod} \quad (5)$$

The normalization factor, K_{mod} , depends on the base modulation mode, as prescribed in Table 8

Modulation	K_{mod}
BPSK	1
QPSK	$\sqrt{2}$
16QAM	$\sqrt{10}$
64QAM	$\sqrt{42}$

Table 3: Normalization Factor

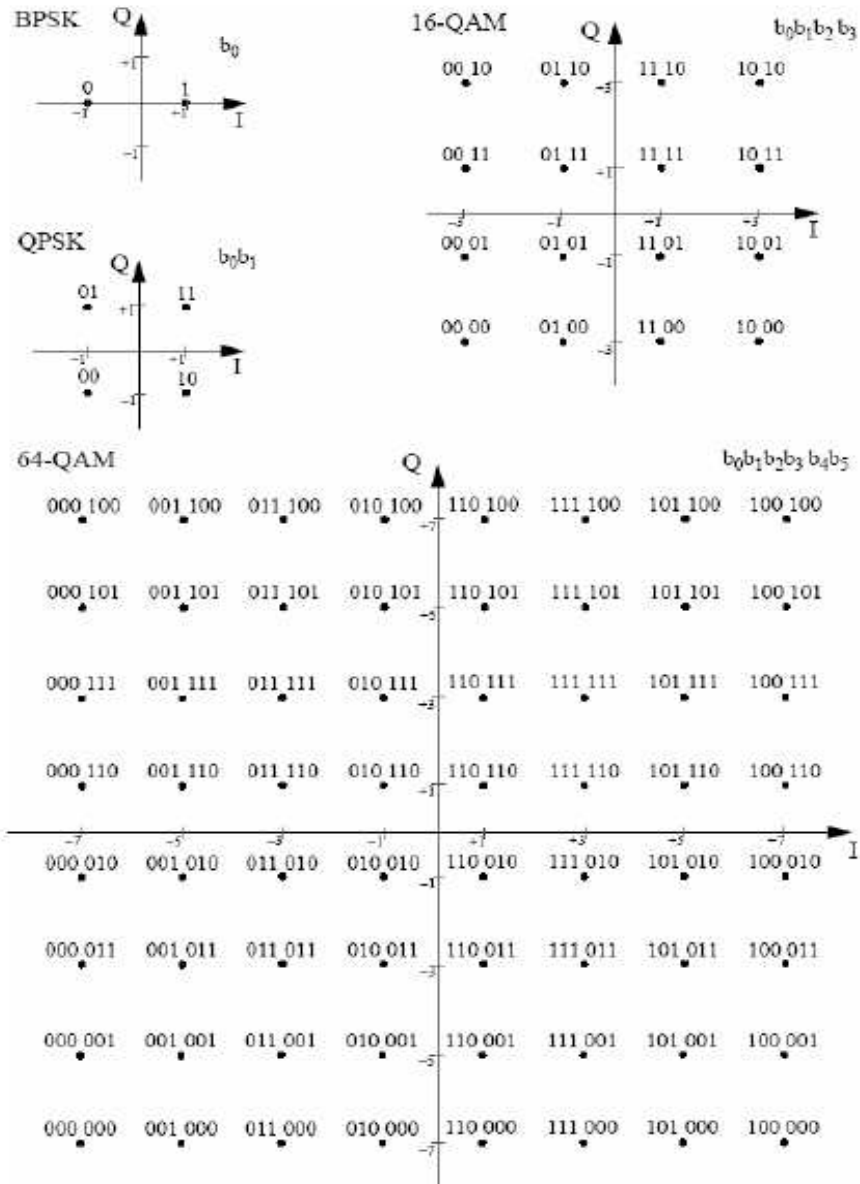


Figure 8: QAM Modulation

2.5.6 OFDM modulation

In 20MHz mode, each OFDM symbol contains 64 subcarrier. There are 56 tones being used(52 data tones and 4 pilot tones). The frequency index of the data tones are -28 -22, -20 -8, -6 -1, 1 6, 8 20, 22 28, and the pilot tones are -21,-7,7,21. The 0th subcarrier is DC, the subcarrier is null. The pilot values are shown in Table 4

N_{STS}	i_{STS}	$\Psi_{i_{STS},0}^{(N_{STS})}$	$\Psi_{i_{STS},1}^{(N_{STS})}$	$\Psi_{i_{STS},2}^{(N_{STS})}$	$\Psi_{i_{STS},3}^{(N_{STS})}$
1	1	1	1	1	-1
2	1	1	1	-1	-1
2	2	1	-1	-1	1
3	1	1	1	-1	-1
3	2	1	-1	1	-1
3	3	-1	1	1	-1
4	1	1	1	1	-1
4	2	1	1	-1	1
4	3	1	-1	1	1
4	4	-1	1	1	1

Table 4: Pilot Value for 20MHz

In 40MHz mode, each OFDM symbol contains 128 subcarrier. There are 114 tones is used (108 data tones and 6 pilot tones). The frequency index of the data tones are -58 -54, -52 -26, -24 -12, -10 -2, 2 10, 12 24, 26 52, 54 58, and the pilot tones are -53,-25,-11,11,25,53. The pilot values show in Table 5

2.6 Channel Model

There are six channel models, A, B, C, D, E and F, provided by TGn Sync [15] for 802.11n. The environments for these channel models can be described

N_{STS}	i_{STS}	$\Psi_{i_{STS},0}^{(N_{STS})}$	$\Psi_{i_{STS},1}^{(N_{STS})}$	$\Psi_{i_{STS},2}^{(N_{STS})}$	$\Psi_{i_{STS},3}^{(N_{STS})}$	$\Psi_{i_{STS},4}^{(N_{STS})}$	$\Psi_{i_{STS},5}^{(N_{STS})}$
1	1	1	1	1	-1	-1	1
2	1	1	1	-1	-1	-1	-1
2	2	1	1	1	-1	1	1
3	1	1	1	-1	-1	-1	-1
3	2	1	1	1	-1	1	1
3	3	1	-1	1	-1	-1	1
4	1	1	1	-1	-1	-1	-1
4	2	1	1	1	-1	1	1
4	3	1	-1	1	-1	-1	1
4	4	-1	1	1	1	-1	1

Table 5: Pilot Value for 40MHz

as follows.

(Channel-A) a typical office environment, non-line-of-sight (NLOS) conditions, and 50 ns rms delay spread,

(Channel-B) a typical large open space and office environments, NLOS conditions, and 100 ns rms delay spread

(Channel-C) a large open space (indoor and outdoor), NLOS conditions, and 150 ns rms delay spread

(Channel-D) the same as model C, line-of-sight (LOS) conditions, and 140 ns rms delay spread (10 dB Ricean K-factor at the first delay

(Channel-E) a typical large open space (indoor and outdoor), NLOS conditions, and 250 ns rms delay spread.

The time resolution of the channel model is 10ns, which is one-fifth of the sampling period (3.2ms/64=50ns). We oversample the transmitted signal by a factor of 5 and interpolate it linearly in order to convolve with the response generated by the channel model.

3 IEEE 802.11n Receiver Design

3.1 Packet Detection

Packet detection is the task of finding an approximate estimate of the start of the preamble for an incoming packet. The structure of L-LTF preamble enables the receivers to use a very simple and efficient algorithm to detect the packet. The algorithm we used is called the delay and correlate algorithm. The method is illustrated in Figure 9, which shows the signal flow structure of the delay and correlation algorithm. We define three parameter:

$$c_n = \sum_{k=0}^{L-1} r_{n+k} r_{n+k+D}^* \tag{6}$$

$$p_n = \sum_{k=0}^{L-1} r_{n+k+D} r_{n+k+D}^* \tag{7}$$

$$m_n = \frac{|c_n|^2}{(p_n)^2} \tag{8}$$

The figure shows two sliding windows C and P. The window C is used to calculate the crosscorrelation between the received signal and a delay version of received signal. The delay z^{-D} is equal to the period of the preamble. In IEEE 802.11n, the delay D is 16. The window P is used to calculate the received signal energy. The value of the P window is then used to normalize

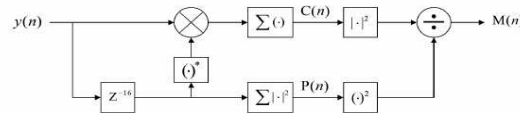


Figure 9: Singal flow structure of the delay and correlation algorithm

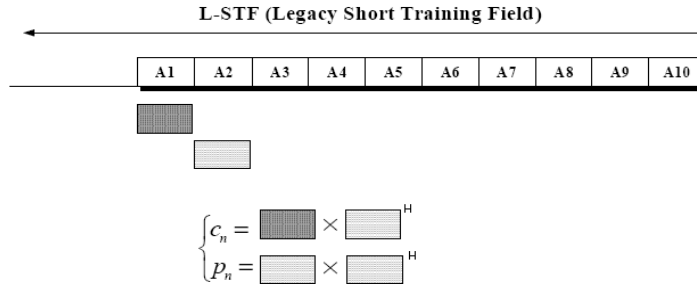


Figure 10: Process of Packet Detection

the decision statistic, so that it is not dependent on absolute received power level.

The variable m_n is restricted between the range of $[0,1]$. If the received signal only consists of noise, the output c_n of the delayed crosscorrelation is a zero-mean random variable. Once a packet is received, c_n corresponds to a crosscorrelation of two identical preambles. Thus, m_n jumps up quickly to its maximum value. Figure 10 shows the operation of the algorithm.

The approach outlined above is useful for high to normal SNR. Unfortunately, for low SNR, the decision statistic m_n may exceed the threshold even no packet is present. Thus, the probability of the false alarm will be increased. We can increase the threshold level; however, the probability of missing will be increased also. Here, we propose a modification to solve this problem. We add an extra window to accumulate the decision statistic. If most decision statistics (may be three forth) in the window are higher than the threshold, we then declare the detection of a packet. The rationale for this approach is that if a packet actually arrives, the decision statistics m_n will remain high until the end of preamble. Simulations show that the modified approach can have more robust detection performance.

3.2 Frequency Synchronization

Both the transmitter and the receiver use their own oscillators to generate carrier and sampling signals. Unfortunately, the signal generated from the oscillator in the receiver will never have the same frequency as that from the transmitter. The difference between the transmitter and receiver oscillator frequencies is called frequency offset. The frequency offset can be divided into two types: The carrier frequency and sampling frequency offsets.

One of the main drawbacks of OFDM is its sensitivity to carrier frequency offset. The degradation is caused by two main phenomena: reduction of amplitude of desired subcarrier and ICI caused by neighboring carriers. The amplitude loss occurs because the desired subcarrier is no longer sampled at the peak of the sinc-function of DFT.

Defined the baseband transmitted signal is s_n . The complex bandpass signal y_n can be expressed as:

$$y_n = s_n e^{j2\pi f_{tx} n T_s} \quad (9)$$

where f_{tx} is the transmitter carrier frequency, T_s is sampling period. The receiver downconverts the signal with the carrier frequency f_{rx} . If we ignore the noise for the moment, the received complex baseband signal r_n will be

$$\begin{aligned} y_n &= s_n e^{j2\pi f_{tx} n T_s} e^{j2\pi f_{rx} n T_s} \\ &= s_n e^{j2\pi (f_{tx} - f_{rx}) n T_s} \\ &= s_n e^{j2\pi (\Delta f) n T_s} \end{aligned} \quad (10)$$

Many algorithms have been developed to estimate the CFO in OFDM systems. We use a data-aided scheme operating on the received time domain

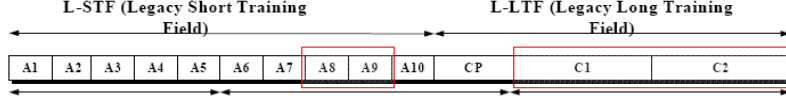


Figure 3: Preamble for CFO Estimation

Figure 11: Preamble for CFO Estimation

signal. The preamble is at least two consecutive repeated symbols. Let D be the delay between the identical samples of the two repeated symbols and define an intermediate variable z as

$$\begin{aligned}
 z &= \sum_{n=0}^{L-1} r_n r_{n+D}^* \\
 &= \sum_{n=0}^{L-1} s_n e^{j2\pi f_{tx} n T_s} (s_{n+D} e^{j2\pi f_{tx} (n+D) T_s})^* \\
 &= \sum_{n=0}^{L-1} (s_n s_{n+D}^*) (e^{j2\pi f_{tx} n T_s} e^{-j2\pi f_{tx} (n+D) T_s}) \\
 &= e^{-j2\pi (\Delta f) D T_s} \sum_{n=0}^{L-1} |s_n|^2
 \end{aligned} \tag{11}$$

Equation (11) is a sum of complex variables with an angle proportional to the frequency offset. Then, the frequency offset can be estimated as:

$$\Delta f = -\frac{1}{2\pi D T_s} \angle z \tag{12}$$

where the $\angle z$ takes the angle of its argument.

In 802.11n system, we use short preambles for coarse CFO estimate, and long preambles for fine CFO estimate. Figure 11 shows this approach.

A limitation of the CFO estimation method is its operation range, which defines how large the CFO can be estimated. The range is directly related

to the length of the repeated symbols(D). The angle of z is of the form $-2\pi(\Delta f)DT_s$, which is unambiguously defined only in the range $[-\pi, \pi)$. Thus the absolute value of the CFO must be larger than the limit shown below.

$$|\Delta f| \leq \frac{\pi}{2\pi DT_s} = \frac{1}{2DT_s} \quad (13)$$

For short training symbol, the sample time T_s is 50ns, and the length of the repeated symbols is 16. Thus, the maximum frequency error that can be estimated is 625kHz. This should be compared with the maximum possible frequency error in the 802.11n system. The transmitter center frequency tolerance shall be 20 ppm maximum for the 5GHz band and 25 ppm maximum for the 2.4GHz band. If the transmitter and receiver clocks have the maximum allowed error, but with opposite signs. The total amounts of frequency error is 200kHz for the 5GHz band and 120kHz for the 2.4GHz band. The maximum frequency offset is well within the range of the estimation.

3.3 Symbol Timing Offset

The objective of symbol timing offset (STO) estimation is to locate the edge of an OFDM symbol. The result is used to define the DFT window, a set of samples for the DFT operation. Since the preamble is available to the receiver, it enables the receiver to use the simple correlation-based STO algorithm. After the packet is detected, the start of the packet is roughly located. The STO refines the precision to the sample-level. Thus, the packet detection can be regarded as a coarse symbol timing synchronization, and

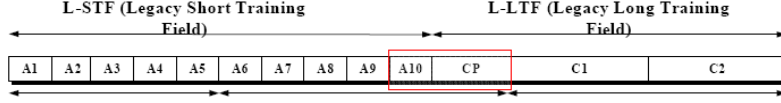


Figure 12: Preamble for Symbol Timing

the symbol timing correction as a fine symbol timing synchronization. The refinement is performed by calculating the crosscorrelation of the received signal r_n and a known reference t_k .

$$\hat{t}_s = \arg \max_n \left| \sum_{k=0}^{L-1} r_{n+k} t_k^* \right|^2 \quad (14)$$

Where L is the length of reference. We select the end of short training symbol and the cyclic prefix of long training symbol as the reference signal, shown in Figure 14. The value of n that corresponds to maximum absolute value of the crosscorrelation is the symbol timing estimate. Thus, L equals 48 here.

3.4 Channel Estimation

The channel estimation is the task of estimating the frequency response of the channel. Since we generally assume that the channel response in wireless LAN system is quasistationary, the channel response does not change during one packet. Therefore, we only need to estimate the channel once in one packet. We use the HT-LTF training symbol to estimate the channel response.

After DFT processing, the time-domain receive signal will be transformed to the frequency domain. The frequency domain received preamble for the

k-th subcarrier can be expressed (without noise) as

$$\mathbf{y}_k = \mathbf{H}_k \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N_t} \end{bmatrix} \quad (15)$$

Where \mathbf{H}_k is a $N_r \times N_t$ MIMO channel response matrix we want to estimate, and $x_1, x_2 \cdots x_{N_t}$ are corresponding preamble in k-th subcarrier. The channel matrix contains $N_r \times N_t$ elements, but we only have N_r equations. ?? Therefore, we need to collect more equations in different time. Review the time-space mapping matrix and depend the number of transmitted antenna, we can generate (14) to the equations below:

When two transmitted antennas are used, the equations can be increased as

$$[\mathbf{y}_k(n), \mathbf{y}_k(n+1)] = \mathbf{H}_k \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix} \quad (16)$$

$$[\mathbf{y}_k(n), \mathbf{y}_k(n+1)] \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix}^T = \mathbf{H}_k \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix} \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix}^T \quad (17)$$

Since $x_1, x_2 = \pm 1$, we then have

$$[\mathbf{y}_k(n), \mathbf{y}_k(n+1)] \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix}^T = \mathbf{H}_k \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = 2\mathbf{H}_k \quad (18)$$

Thus,

$$\mathbf{H}_k = 1/2 \times [\mathbf{y}_k(n), \mathbf{y}_k(n+1)] \begin{bmatrix} x_1 & -x_1 \\ x_2 & x_2 \end{bmatrix}^T \quad (19)$$

Use the same method in three and four transmitted antennas. We can have the channel estimation result as

$$\mathbf{H}_k = 1/4 \times [\mathbf{y}_k(n), \mathbf{y}_k(n+1), \mathbf{y}_k(n+2), \mathbf{y}_k(n+3)] \begin{bmatrix} x_1 & -x_1 & x_1 & x_1 \\ x_2 & x_2 & -x_2 & x_2 \\ x_3 & x_3 & x_3 & -x_3 \end{bmatrix}^T \quad (20)$$

and

$$\mathbf{H}_k = 1/4 \times [\mathbf{y}_k(n), \mathbf{y}_k(n+1), \mathbf{y}_k(n+2), \mathbf{y}_k(n+3)] \begin{bmatrix} x_1 & -x_1 & x_1 & x_1 \\ x_2 & x_2 & -x_2 & x_2 \\ x_3 & x_3 & x_3 & -x_3 \\ -x_4 & x_4 & x_4 & x_4 \end{bmatrix}^T \quad (21)$$

3.5 Phase Tracking

Frequency synchronization is not a perfect process, so there is always some residual frequency error. The residual frequency offset will cause constellation rotation, affecting signal detection. Here, we use the data-aided method to eliminate the residual frequency error. In 802.11n, there are four pilot subcarriers in 20 MHz bandwidth mode, and eight pilot subcarriers in 40 MHz bandwidth mode. Using the known pilot subcarrier and the estimated channel, we can reestablish the received data of n-th received antenna in k-th subcarrier y_n^k without phase shift as

$$y_n^k = \mathbf{H}_{n,m}^{\hat{k}} \Psi_m^k \quad (22)$$

where $\mathbf{H}_{n,m}^{\hat{k}}$ is the estimated channel matrix in k-th subcarrier for the path from m-th transmit antenna to n-th receive antenna, and Ψ_m^k is the pilot value of m-th transmit antenna in k-th subcarrier.

Then, we can find the phase error by the difference of the phase between received data R_n^k and Y_n^k

$$\hat{\phi} = \angle[R_n^k(Y_n^k)^*] \quad (23)$$

After calculating the phase error of all pilot tones in one OFDM symbol, we take average of these phase, and then subtract it from the phase of the receive signal in each tone.

3.6 Sampling Frequency Offset

The sampling frequency offset (SFO) results from the mismatch between the frequency of the digital to analog converter (DAC) at the transmitter, and that of the analog to digital converter (ADC) at the receiver. The SFO will have two main effects: (1) a slow drift of the symbol timing boundary, which rotates the phase of the subcarriers similar to the timing offset, and (2) a loss of the orthogonality of the subcarriers. This is due to the subcarriers are not sampled at the optimum positions and ICI occurs. This will also result in a loss of signal to noise ratio (SNR).

The sampling error is defined as

$$t_{\Delta} = \frac{T_{rx} - T_{tx}}{T_{tx}} \quad (24)$$

where T_{rx} and T_{tx} are the transmitter and receiver sampling periods. The overall effect, after DFT, on the received subcarrier $R_{l,k}$ is shown as

$$R_{l,k} = e^{j2\pi kt_{\Delta} l \frac{T_s}{T_u}} x_{l,k} \text{sinc}(\pi kt_{\Delta}) H_{l,k} + W_{l,k} + C_{t_{\Delta}} \quad (25)$$

where l is the OFDM symbol index, k is the subcarrier index, T_s is the duration of the total symbol, T_u is the duration of the useful data portion, $W_{l,k}$ is additive white noise and $C_{t_{\Delta}}$ is the induced ICI due to the SFO.

Since the sampling error t_{Δ} is very small, the ICI term can be ignored. The remained item is $e^{j2\pi kt_{\Delta} l \frac{T_s}{T_u}}$. This term shows the amount of rotation angle experienced by the different subcarrier. Even though the term t_{Δ} is quite small, as l increases, the rotation may eventually become so large that correct demodulation is no longer possible. Therefore, it is necessary to compensate for the error.

The receiver only uses one oscillator to generate the carrier frequency and sampling frequency. Thus, we can find sampling frequency error using the estimated carrier frequency offset.

$$\frac{SFO}{CFO} = \frac{f_s}{f_c} \quad (26)$$

$$t_{\Delta} = \frac{1}{f_s} - \frac{1}{f_s + SFO} \quad (27)$$



4 MIMO Detector

4.1 Problem Definition

We first consider a flat-fading MIMO system with N_T transmit and N_R receive antennas. The complex baseband received vector, defined by $\mathbf{y}_c = [y_1, y_2, \dots, y_{N_R}]^T$, is given by:

$$\mathbf{y}_c = \mathbf{H}_c \mathbf{x}_c + \mathbf{w} \quad (28)$$

where $\mathbf{x}_c = [x_1, x_2, \dots, x_{N_T}]^T$ is the x_i transmitted signal vector, \mathbf{H}_c is a $N_T \times N_R$ complex channel matrix, and $\mathbf{w} = [w_1, w_2, \dots, w_{N_R}]^T$ is a i.i.d. complex Gaussian noise vector. Here, we assume that each element in \mathbf{x}_c is a quadrature amplitude modulation (QAM) signal.

The complex system model can be converted to an equivalent real-valued model by separating a complex signal to the real and image parts. Thus, (28) can be rewritten as

$$\mathbf{y} = \mathbf{H} \mathbf{x} + \mathbf{w} \quad (29)$$

where

$$\mathbf{y} = [\text{Re}\{\mathbf{y}_c\} \quad \text{Im}\{\mathbf{y}_c\}]^T \quad (30)$$

$$\mathbf{x} = [\text{Re}\{\mathbf{x}_c\} \quad \text{Im}\{\mathbf{x}_c\}]^T \quad (31)$$

$$\mathbf{H} = \begin{bmatrix} \text{Re}\{\mathbf{H}_c\} & -\text{Im}\{\mathbf{H}_c\} \\ \text{Im}\{\mathbf{H}_c\} & \text{Re}\{\mathbf{H}_c\} \end{bmatrix} \quad (32)$$

We will use the representation shown (29) in the following derivation. For simplicity, we define $2N_T = 2N_R = N$.

4.2 Maximum Likelihood Detector

The optimum detector for a MIMO system is Maximum Likelihood (ML) detector. For a received vector \mathbf{y} , and the known channel matrix \mathbf{H} , the ML detector finds the transmitted vector \mathbf{x} minimizing the distance from \mathbf{y} to $\mathbf{H}\mathbf{x}$, i.e.,:

$$\hat{\mathbf{x}}_{ML} = \arg \min_{\mathbf{x} \in \Psi} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \quad (33)$$

where the set Ψ indicates all possible transmit vectors. The ML detector needs an exhaustive search over the entire set of Ψ . Let the QAM constellation size be M . For the number of transmitted bit streams being N , the computational complexity of ML detector is on the order of $O(M^N)$. When M is large, the computational complexity of the ML detector becomes prohibited. Thus, many alternatives have been proposed to reduce the computational complexity. The most well-known one may be the Sphere Decoding(SD) algorithm.

4.3 Sphere Decoding

The main idea of sphere decoding is to search a subset of Ψ , which satisfies

$$\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 < r^2 \quad (34)$$

If r is small, the required computational complexity will be effectively reduced. Using QR-decomposition, we can have $\mathbf{H} = \mathbf{Q}\mathbf{R}$, where \mathbf{Q} is orthonormal matrix and \mathbf{R} is a upper triangle matrix. Also,

$$\mathbf{R} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,N} \\ 0 & r_{2,2} & \cdots & r_{2,N} \\ \vdots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & r_{N,N} \end{bmatrix} \quad (35)$$

Using the result, we can rewrite (34) as

$$\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \|\mathbf{y} - \mathbf{Q}\mathbf{R}\mathbf{x}\|^2 = \|\mathbf{Q}^T\mathbf{y} - \mathbf{R}\mathbf{x}\|^2 = \|\mathbf{y}' - \mathbf{R}\mathbf{x}\|^2 < r^2 \quad (36)$$

Where $\mathbf{y}' = \mathbf{Q}^T\mathbf{y}$. Let the i th component of \mathbf{y}' be y'_i and that of \mathbf{x} be x_i .

Using the upper triangle structure of R , we can further decompose (36) as

$$\begin{aligned} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 &= \sum_{p=1}^n (y'_p - \sum_{m=p}^n r_{m,p}x_m)^2 \\ &= (y'_N - r_{N,N}x_N)^2 \\ &\quad + (y'_{N-1} - r_{N-1,N}x_N - r_{N-1,N-1}x_{N-1})^2 + \dots \\ &< r^2 \end{aligned} \quad (37)$$

As we can see, the first summation term in (37) only depends on x_N , the second on x_N and x_{N-1} , and so on. Thus, (37) can form a tree search structure of x_i , $i = N, N-1, \dots, 1$. The first-level of (34) can then be written as

$$(y'_N - r_{N,N}x_N)^2 < r^2 \quad (38)$$

We can find all possible x_N 's satisfying (38). Then, for each candidate of x_N , we can find all x_{N-1} 's satisfying the constraint (see (37)). Continue this process, we can find all candidates of $x_N \dots x_1$. Thus, we only search possible transmitted vectors, \mathbf{x} , inside the sphere and output the one minimizing the distance between \mathbf{y} and $\mathbf{H}\mathbf{x}$.

In the representation (36), we can see that the radius r is a very critical parameter. If the radius is too large, there will be many candidates, and the computational complexity will be high. On the other hand, if the radius is too small, there may be no candidates can be found in certain level. In [9],

a method to find a proper radius is suggested. The radius can be chosen as:

$$r^2 = C \times |\det(\mathbf{H})|^{(1/N)} \quad (39)$$

Where C is a constant, \mathbf{H} is channel matrix.

Equation (36) can not guarantee that the number of candidates inside the sphere is non-zero. When no candidates can be found, we can either enlarge the radius and conduct the tree structure again (until at least one candidate is found), or randomly choose a candidate as the output. Apparently, the former approach will have better result. However, the computational complexity will be higher.

Although the SD algorithm requires lower computational complexity than the ML detector, it has other problems. From the algorithm described above, it is easy to see that the number of candidates inside the sphere can have large variation. It depends on the channel, the noise level, and the radius r . To have a better idea of this problem, we conduct simulations for the LSD algorithm using a 4×4 system with 64-QAM transmission. We set the radius as $r^2 = 8 \times |\det(\mathbf{H})|^{(1/4)}$, and have 10000 runs. Figure 13 shows the histogram of the number of candidates in the SD algorithm. From the histogram, we can see that the number of candidates inside the sphere has a large variation, though the averaged number is not particularly high. This is property is not desirable in hardware implementation. Other problems include the large latency (due to the tree structure), and the difficulty in determination of the radius (even with (39)). Here, we propose a new algorithm to overcome the problems.

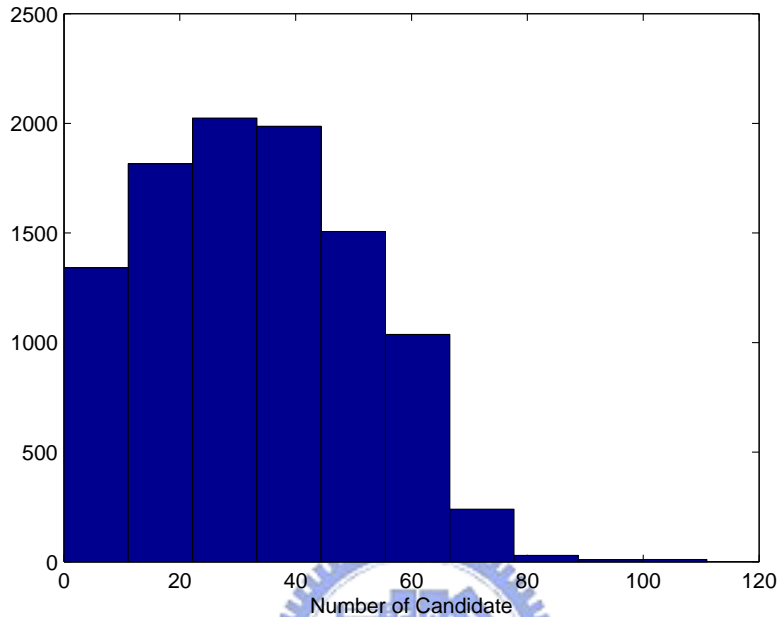


Figure 13: Histogram of Number of Candidates

4.4 Proposed Method

Recall that the ML detection uses an exhaustive search for all dimensions, the complexity is then $O(M^N)$. The main idea of the proposed algorithm is that we only conduct exhaustive search in a small number of dimensions (maybe one or two dimension), and conduct optimum estimation in rest of the dimensions. As a result, the required computational complexity is only $O(M)$ or $O(M^2)$.

For purpose of illustration, we first use a 2×2 4-PAM (real) system as an example. Let the channel matrix be \mathbf{H} and the receive signal vector be $\mathbf{x} = [x_1 \ x_2]^T$. The constellation of the receive signal, excluding noise, is $\mathbf{H}\mathbf{x}$, $x_1, x_2 \in \{-3, -1, 1, 3\}$. If we let x_2 be a continuous variable, there will be four

parallel lines in the constellation plot, corresponding to $x_1 = \{-3, -1, 1, 3\}$. Thus, to estimate x_2 for a given x_1 , we can project the received vector \mathbf{y} onto the corresponding line. In this case, we then have four candidate sets of $\{x_1, x_2\}$. We can then choose the one with the smallest $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$ as the output. Figure 14 shows the idea of the proposed algorithm.

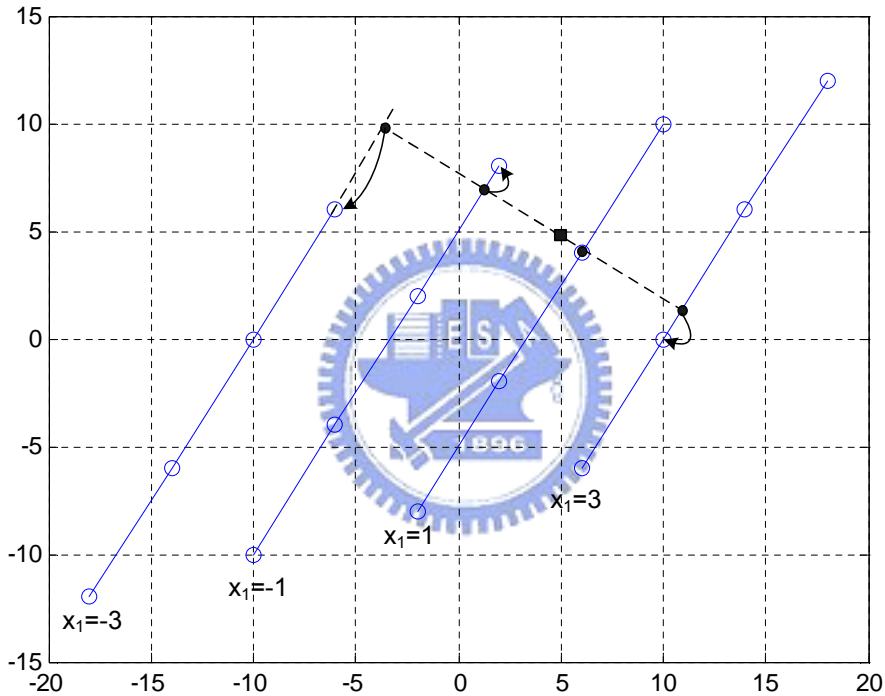


Figure 14: Projection in each group

We now generalize the idea to a $N \times N$ MIMO system. Recall our system model in (29), the channel matrix is $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_N]$. The transmitted vector is $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$, where x_i is a PAM symbol. The possible values of a PAM symbol is $S_{PAM} = \{-(M-1), -(M-3), \dots, M-1\}$, where M is the size of PAM signal constellation. We can rewrite the system model as

$$\mathbf{y} = \mathbf{h}_1 x_1 + \mathbf{h}_2 x_2 + \dots + \mathbf{h}_N x_N + \mathbf{w} \quad (40)$$

Let $x_p = s_k$, where $s_k = -(M - 1) + 2(k - 1)$ and $k = 1, 2, \dots, M$. We can rewrite (46) as

$$\begin{aligned}
\mathbf{y} &= \underbrace{\mathbf{h}_1 x_1 + \dots + \mathbf{h}_p s_k + \dots + \mathbf{h}_N x_N}_{N \text{ terms}} + \mathbf{w} \\
\mathbf{y} - \mathbf{h}_p s_k &= \underbrace{\mathbf{h}_1 x_1 + \dots + \mathbf{h}_N x_N}_{N-1 \text{ terms}} + \mathbf{w} \\
\tilde{\mathbf{y}}_p^k &= \tilde{\mathbf{H}} \tilde{\mathbf{x}}_p^k + \mathbf{w}
\end{aligned} \tag{41}$$

where $\tilde{\mathbf{y}}_p^k = \mathbf{y} - \mathbf{h}_p s_k$, $\tilde{\mathbf{H}}_p = [\mathbf{h}_1, \dots, \mathbf{h}_{p-1}, \mathbf{h}_{p+1}, \dots, \mathbf{h}_N]$, and $\tilde{\mathbf{x}}_p = [x_1, \dots, x_{p-1}, x_{p+1}, \dots, x_N]^T$.

Now, we can project $\tilde{\mathbf{y}}_p^k$ onto the subspace spanned by $\tilde{\mathbf{H}}_p$, and then quantize the result to obtain an estimated symbol in the constellation (corresponding to $x_p = s_k$). Note that the project can be conducted with the least-squares (LS) method. Let $\check{\mathbf{x}}_p^k = [\check{x}_1^k, \dots, \check{x}_{p-1}^k, \check{x}_{p+1}^k, \dots, \check{x}_N^k]^T$ be the quantized vector of the projection of $\tilde{\mathbf{y}}_p^k$. Then, we have

$$\check{\mathbf{x}}_p^k = \text{Quan}\{(\tilde{\mathbf{H}}_p^T \tilde{\mathbf{H}}_p)^{-1} \tilde{\mathbf{H}}_p^T \tilde{\mathbf{y}}_p^k\} \tag{42}$$

where the operation $\text{Quan}\{\cdot\}$ indicates the quantization operations, assigning a vector to the nearest constellation point. Note that the vector in (42) is of dimension $N - 1$. Denote the complete N -dimensional vector as \mathbf{v}_p^k . Then,

$$\mathbf{v}_p^k = [\check{x}_1^k, \dots, \check{x}_{p-1}^k, s_k, \check{x}_{p+1}^k, \dots, \check{x}_N^k] \tag{43}$$

Thus, \mathbf{v}_p^k is a candidate for the ML solution. We can then repeat the process for all k 's and obtain a set of candidates $\{\mathbf{v}_p^{s_1}, \dots, \mathbf{v}_p^{s_M}\}$. Choose the one with the minimum distance and obtain the best exhausted-search solution in the direction of x_p . Now, we can now search L different dimensions to

have better result where $L \leq N$. The computational complexity will then be $O(LM)$. The higher L indicates the higher computational complexity, but the better performance. Thus, L can be chosen as a compromise between the performance and the computational complexity.

The result derived above can now be expressed as:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}=\mathbf{v}_p^k} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2, (k = 1, 2, \dots, M); \quad (44)$$

$$(p = p_1, p_2, \dots, p_L), p_i \in \{1, 2, \dots, N\}, p_q \neq p_j) \quad (45)$$

The scheme proposed above can be generalized such that the exhausted search can be conducted in D directions (dimensions) where $D = 1, 2, \dots, N$. In other words, M^D projections have to be conducted in the chosen directions. If $D = 1$, this is the algorithm we described above. If $D = N$, it will be equivalent to the ML detector. For the systems we considered here, we found that $D = 2$ will approach the ML solution very well. For each projection, we can choose fixed values from any two dimensions $\{x_p = s_k, x_q = s_j\}$ where $p \neq q$. Then,

$$\begin{aligned} \mathbf{y} &= \underbrace{\mathbf{h}_1 x_1 + \dots + \mathbf{h}_p s_k + \dots + \mathbf{h}_q s_j + \dots + \mathbf{h}_N x_N}_{N \text{ terms}} + \mathbf{w} \\ \mathbf{y} - \mathbf{h}_p s_k - \mathbf{h}_q s_j &= \underbrace{\mathbf{h}_1 x_1 + \dots + \mathbf{h}_N x_N}_{N-2 \text{ terms}} + \mathbf{w} \\ \tilde{\mathbf{y}}_{p,q}^{k,j} &= \tilde{\mathbf{H}}_{p,q} \mathbf{x}_{p,q} + \mathbf{w} \end{aligned} \quad (46)$$

where $\tilde{\mathbf{y}}_{p,q}^{k,j} = \mathbf{y} - \mathbf{h}_p s_k - \mathbf{h}_q s_j$, and $\tilde{\mathbf{H}}_{p,q}$ is a matrix consisting of columns of $\tilde{\mathbf{H}}$ with $\tilde{\mathbf{h}}_p$ and $\tilde{\mathbf{h}}_q$ removed, and $\mathbf{x}_{p,q}$ is a vector consisting of elements of \mathbf{x} with x_p and x_q removed. Let $\check{\mathbf{x}}_{p,q}^{k,j}$ be the quantized vector of the projection of $\mathbf{x}_{p,q}$. Using the similar method in (42), we can find $\check{\mathbf{x}}_{p,q}^{k,j}$ as

$$\check{\mathbf{x}}_{p,q}^{k,j} = \text{Quan}\{(\tilde{\mathbf{H}}_{p,q}^T \tilde{\mathbf{H}}_{p,q})^{-1} \tilde{\mathbf{H}}_{p,q}^T \tilde{\mathbf{y}}_{p,q}^{k,j}\} \quad (47)$$

Then, let $\tilde{\mathbf{x}}_{p,q}^{k,j} = [\tilde{x}_1^{k,j}, \dots, \tilde{x}_N^{k,j}]$ and

$$\mathbf{v}_{p,q}^{k,j} = [\tilde{x}_1^{k,j}, \dots, s_k, \dots, s_j, \dots, \tilde{x}_N^{k,j}]. \quad (48)$$

We can see that $\mathbf{v}_{p,q}^{k,j}$ is a candidate for the ML solution. Now, we can choose L combinations of (p, q) to conduct the exhausted search. In other words, we conduct L exhausted search in D dimensions. The computational complexity will be $O(LM^2)$. Finally, we can choose one vector minimizing the distance from \mathbf{y} to $\mathbf{H}\mathbf{x}$, i.e,

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}=\mathbf{v}_{p,q}^{k,j}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2, (k, j = 1, 2, \dots, M; (p, q) = L \text{ combinations}) \quad (49)$$

Note that (p, q) has $N(N-1)/2$ combinations. As we can see, we only have to use a small L and the result can approach the ML solution very well. The choice of the combinations does not have a specific rule. From simulations, we found that if (p, q) can be chosen such that x_p and x_q correspond to the real and complex component of a symbol, good performance can be achieved.

4.5 Complexity Analysis

In this paragraph, we will compare the computational complexity for the algorithms considered. Here, we only consider the number of multiplications used. For a $N_t \times N_t$ MIMO system with M^2 -QAM transmission, the ML detector requires $(2N_t^2 + 3N_t) \times M^{2N_t}$ multiplications in the evaluation of (33). The proposed algorithm requires $(4N_t^2 - 2N_t \cdot D) \times 2^D L$ multiplications in (42), and $(2N_t^2 + 3N_t) \times M^D L$ multiplications in (44). There is a matrix inverse operation in (42). However note that the matrix does not have to be recomputed when a new exhausted search is initiated. The

matrix differs from its previous one only by D columns, and some fast algorithm has been proposed to reduce the computational complexity of the inversion. We can use the fast algorithm proposed in [10], which requires $24N_t^3 + 12N_t^2 \times L$ multiplications. Thus, the proposed algorithm requires $(6N_t^2 + 3N_t - 2N_t D)M^D L + 4N_t^3 + 12N_t^2 \times L$ multiplications in total. Complexity comparison for the 2×2 , 3×3 and 4×4 with the 64-QAM is shown in the Table 6. As we can see, the proposed algorithm has a remarkable reduction in computational complexity.

Since the throughput of the SD algorithm is variable, it is very difficult to evaluate its computational complexity analytically. Thus, we use the computer simulation to actually count how many multiplications are conducted for each run. We simulate SD for 10000 runs, and find the averaged and top one percent averaged results.

Table 7 shows the complexity comparison of the SD and the proposed algorithm. The computational complexities of the proposed algorithm shown are maximum ones in (6). From the table, we can see that the computational complexity of the proposed algorithm is still much lower than the SD algorithm.

Note that the result show in Table 7 for the SD algorithm is an averaged result. It is highly possible that in some cases, the candidates in the radius is very large, and in others is very small. This property makes efficient hardware implementation difficult. The other advantage of the proposed algorithm is that it allows a tradeoff between the complexity and performance. When the complexity is the main concern, we can use smaller D and L . On the contrary, when the performance is the main concern, those parameters can

$N_T \times N_T$ ($N = 2N_T$)	D=1 L=1	D=1 L=N/2	D=1 L=N	D=2 L=1	D=2 L=2	D=2 L=4	ML
2×2	248	432	800	1,040	-	-	57,344
3×3	600	1,368	2,520	2,148	4,080	-	7.07×10^6
4×4	1,168	3,136	5,760	3,712	6,912	13,312	7.38×10^8

Table 6: Number of Multiplications required in proposed and ML algorithm

$N_T \times N_T$ ($N = 2N_T$)	Proposed Method (maximum)	SD (averaged)	SD (top 1 % averaged)
2×2	1,040	1,015	2,853
3×3	4,080	3,602	9,599
4×4	13,312	13,683	33,340

Table 7: Number of Multiplications required in proposed and SD algorithm

be enlarged.

4.6 Simulations



In the section, we report simulation results demonstrating the performance of the proposed algorithm. We consider the 2×2 , 3×3 and 4×4 MIMO systems with 16-QAM and 64-QAM transmission. The channel is a Rayleigh flat-fading channel. The bit error rate (BER) is used as the performance measure. Three algorithms are compared, namely, the ML, SD and proposed algorithms. The radius of the SD method is set as $8 \times |\det(\mathbf{H})|^{(1/N)}$.

Figure 15-20 show the results for 2×2 , 3×3 and 4×4 MIMO systems with 16-QAM modulation. From Figure 15 and 16, we can see that for 2×2 systems, the performance of the proposed algorithm with $D=1$ and $L=4$ is worse than that of the ML detector. When $D=2$ and $L=1$, the performance of the proposed algorithm is almost as good as that of the ML algorithm. From Figure 17 and 18, we can see that for 3×3 systems, the performance of

proposed algorithm with $D=1$ and $L=6$ is worse than that of the ML detector. However, it can achieve the ML bound when $D=2$ and $L=2$. Figure 19 and 20 shows the result for 4×4 systems. As we can see, the proposed algorithm can achieve the ML performance when $D=2$ and $L=4$.

We then evaluate the performance of the proposed algorithm in 64-QAM transmission. Figure 21 and 22 show the simulation results for 2×2 systems. As we can see, the performance of the proposed algorithm with $D=2$ and $L=1$ can achieve the same performance as the ML detector. Also, with $D=1$ and $L=4$, the performance is also very close to that of the ML detector. Figure 23 and 24 show the simulation results for the 3×3 systems. From Figure 23, we can see that the performance of our algorithm with $D=1$ and $L=6$ has a small deviation from that of the ML detector. From Figure 24, we see that when $D=2$ and $L=2$, the proposed algorithm performs the same as that of the ML detector. Finally, Figure 25 and 26 show the result for 4×4 systems. As we can see, when $D=1$ and $L=8$, the proposed algorithm is worse than that of the ML detector. However, it can achieve ML performance when $D=2$ and $L=4$. Note that in all cases, the SD algorithm can have almost the same performance as that of the ML detector. Note that the price to pay is a higher computational complexity since a larger radius has to be used.

Compare the simulation results for 16-QAM and 64-QAM transmission. We can find that the proposed algorithm can have more advantages when a high QAM size is used.

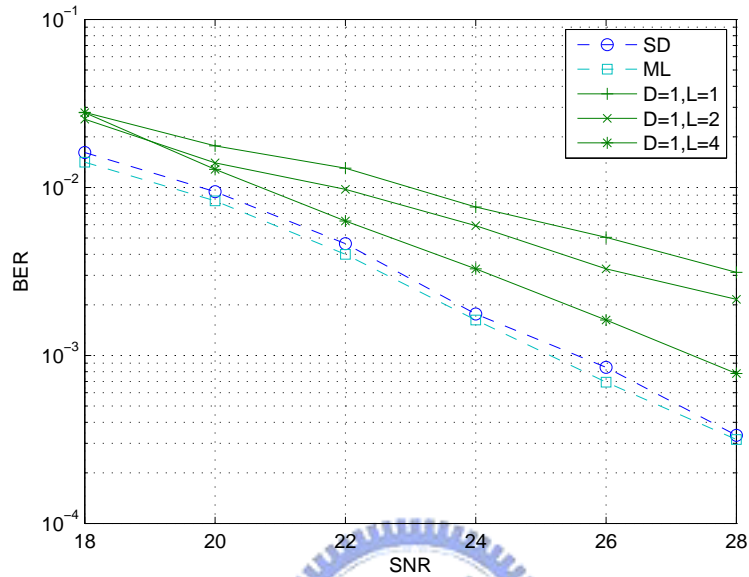


Figure 15: BER comparison for 2×2 16-QAM(I)

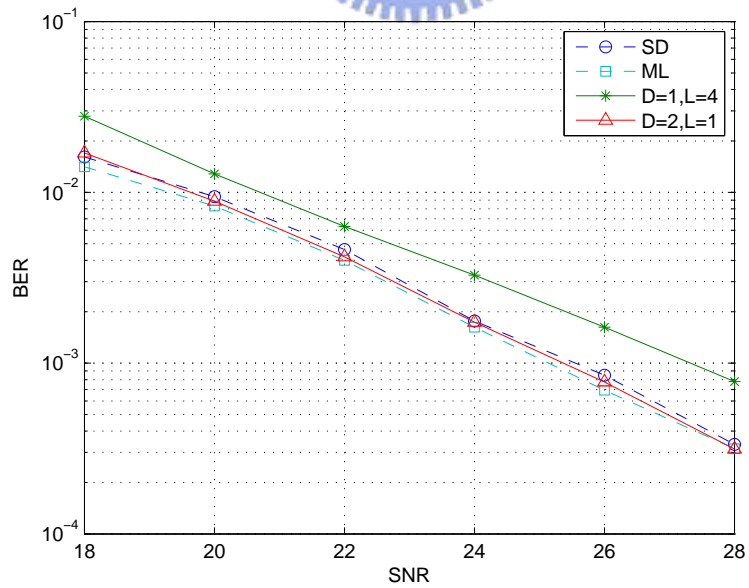


Figure 16: BER comparison for 2×2 16-QAM(II)

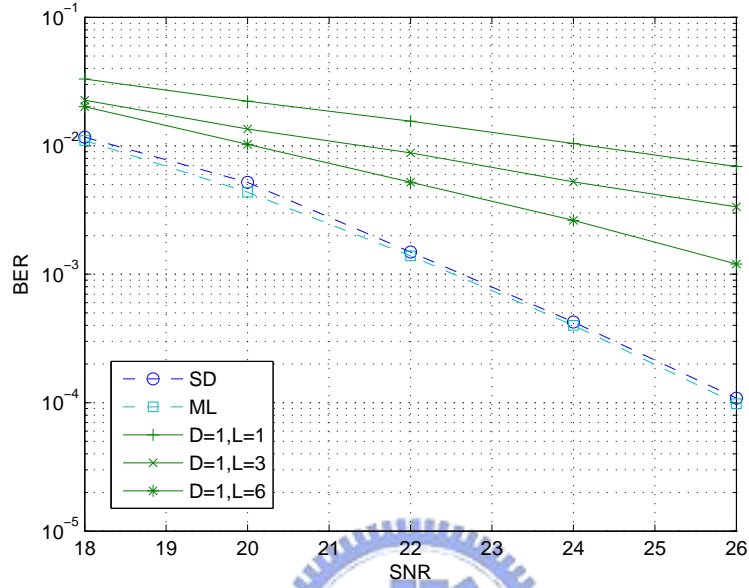


Figure 17: BER comparison for 3×3 16-QAM(I)

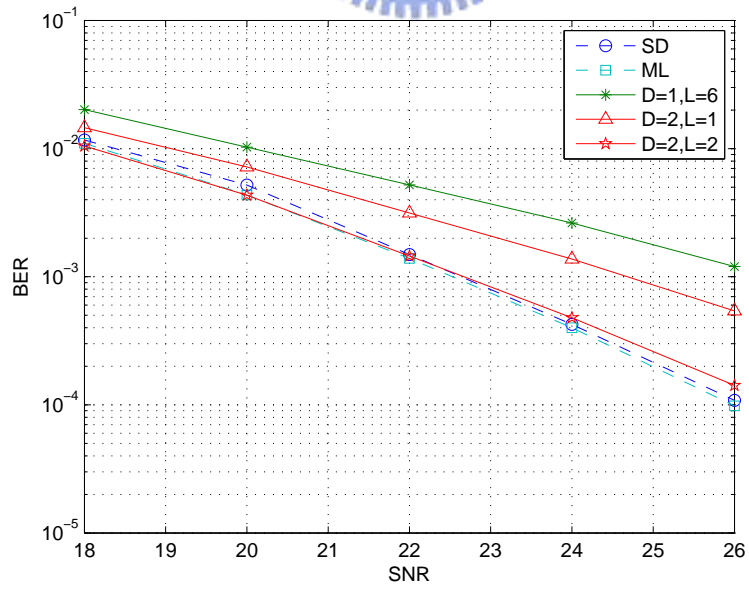


Figure 18: BER comparison for 3×3 16-QAM(II)

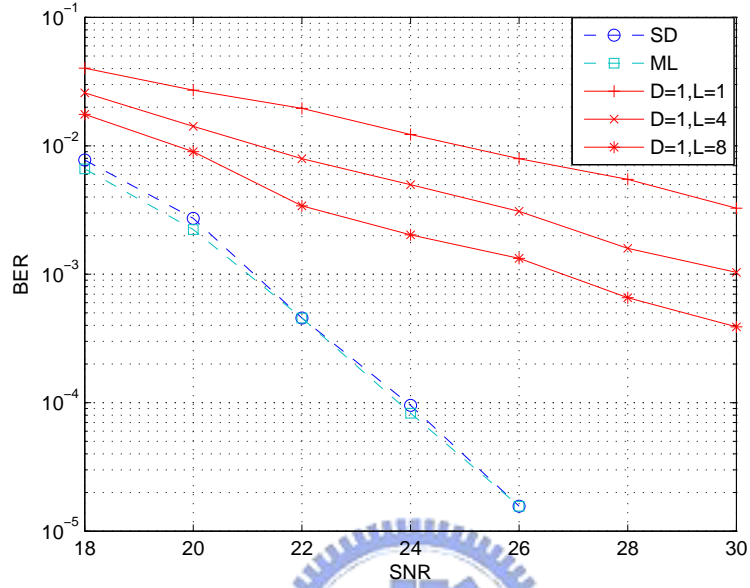


Figure 19: BER comparison for 4×4 16-QAM(I)

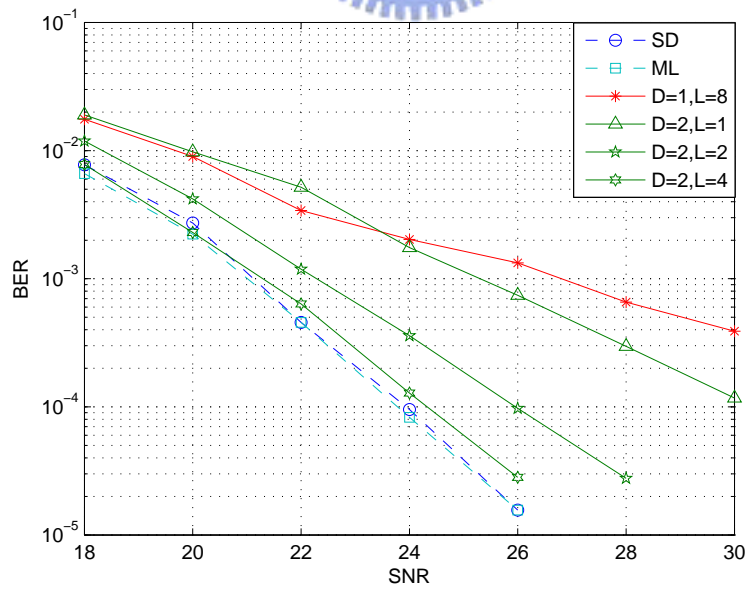


Figure 20: BER comparison for 4×4 16-QAM(II)

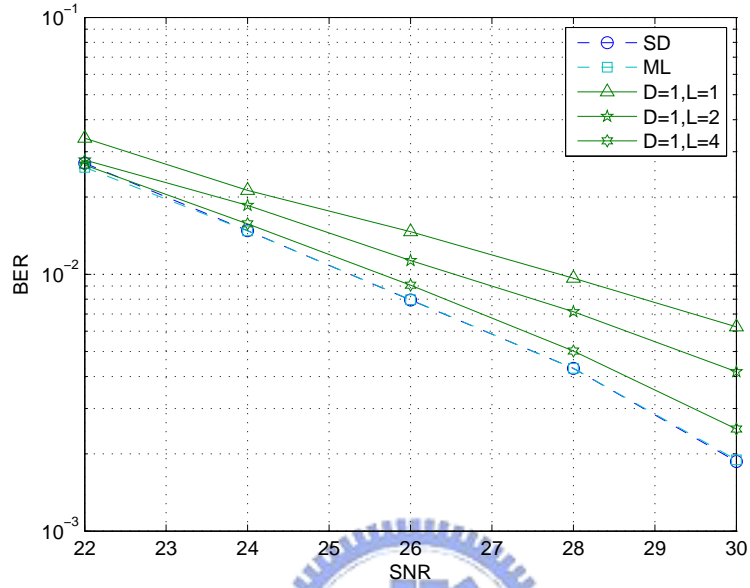


Figure 21: BER comparison for 2×2 64-QAM(I)

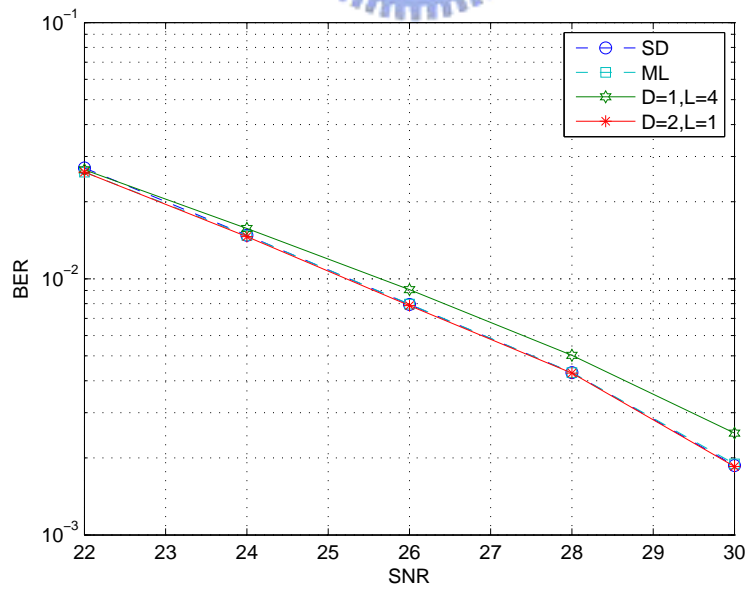


Figure 22: BER comparison for 2×2 64-QAM(II)

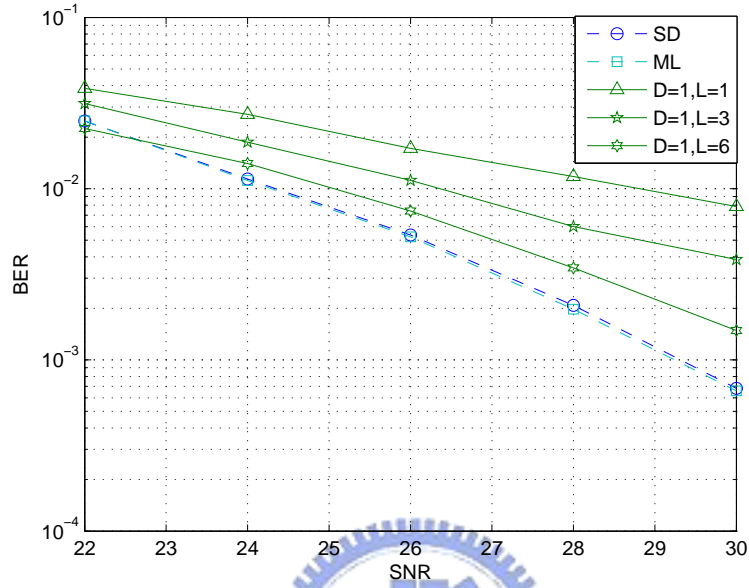


Figure 23: BER comparison for 3×3 64-QAM(I)

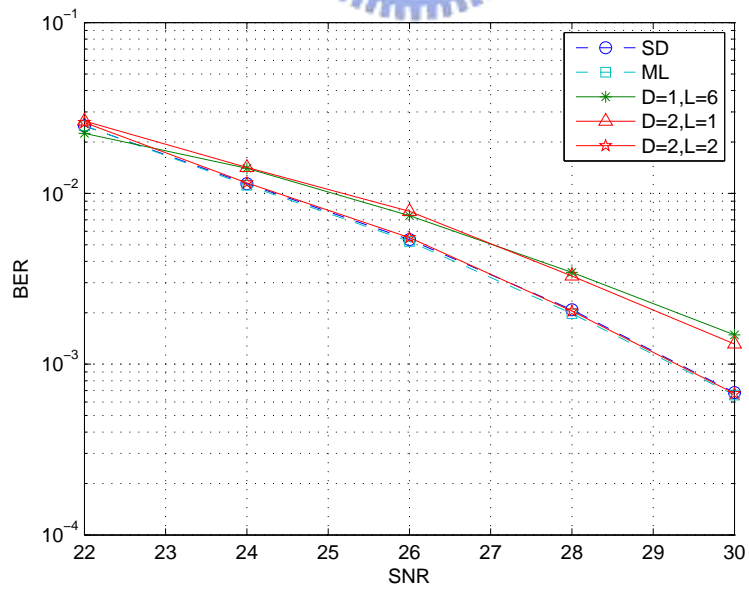


Figure 24: BER comparison for 3×3 64-QAM(II)

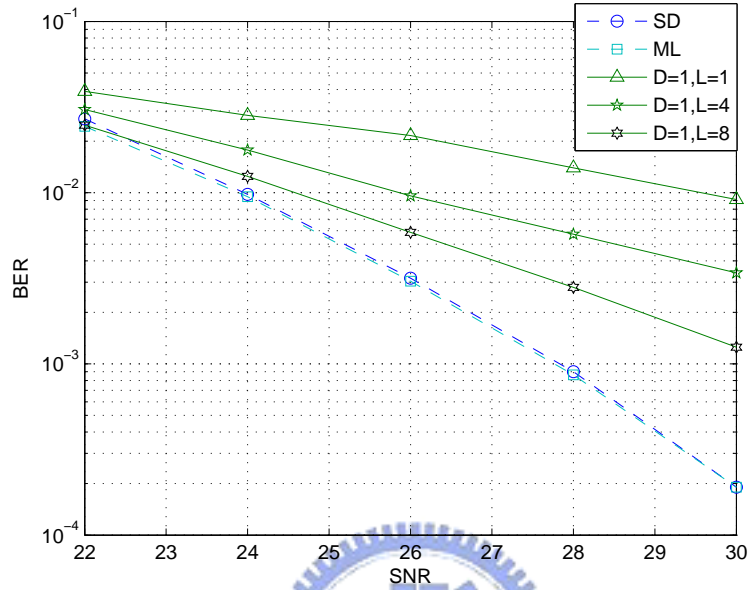


Figure 25: BER comparison for 4×4 64-QAM(I)

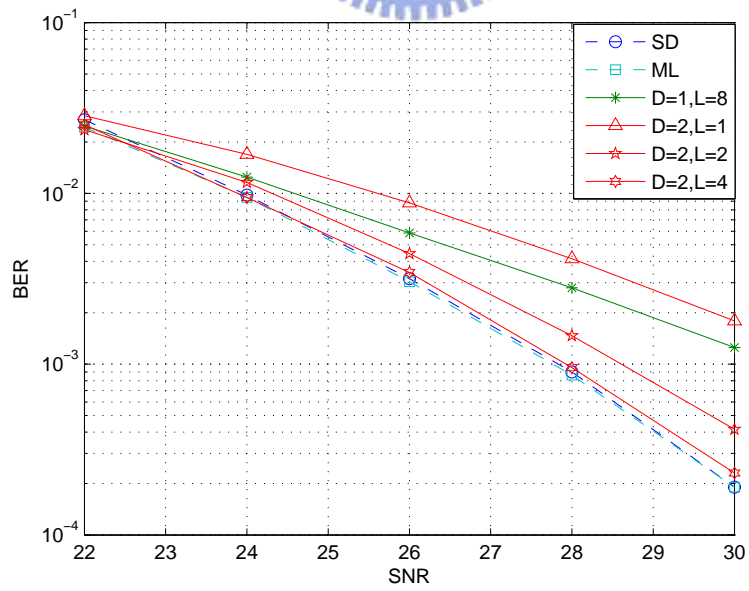


Figure 26: BER comparison for 4×4 64-QAM(II)

5 MIMO Soft Bit Demapping

5.1 Bit-interleaved Coded Modulation

Originally proposed by Zehavi [8], BICM inserts a bit-level interleaver between the encoder and modulator. With this operation, bursty bit errors can be reduced, and the correction capability of the decoder can be enhanced. As a result, BICM is suited to the fast fading channel environment. For non-fading frequency-selective channels, BICM can be cooperated with OFDM to have frequency diversity gain. Besides, when the MIMO structure is added, spatial diversity can be achieved as well. Conventional, it uses a MMSE equalizer to suppress the interference arising in the MIMO system, and then transfer the MIMO system into multiple SISO systems. Then, it applies one-dimensional soft-bit demappers obtaining soft-bit information for the received signal.

5.2 Log Likelihood Ratio

Due to use of BICM system, we have to demap a QAM symbol to its corresponding bits before decoding. The operation is called bit demapping. There are two kind of bits can be demapped: hard and soft bits. Hard bits are the detected bits being equals to “0” or “1”. Soft bits corresponds to probabilities of being equal to “0” or “1”. It is well know that soft bits in the BICM scheme can have much better performance. The computation of soft bit demapping in the SISO systems is low. In the MIMO systems; however, its computational complexity is very high, comparable to the ML detector. In this section, we will extend the algorithm proposed in the previous section

to overcome the problem. In soft demapping, Log-Likelihood Ratio(LLR) is generally used instead of probability to represent the soft information. Let \mathbf{y} be the received signal vector, $b_{n,k}$ be the k th bit of its n th component, and \mathbf{x} be the transmit signal vector. The LLR of bit $b_{n,k}$ is defined as

$$\begin{aligned} LLR(b_{n,k}) &= \ln \frac{P[b_{n,k} = 1|\mathbf{y}]}{P[b_{n,k} = 0|\mathbf{y}]} \\ &= \ln \frac{\sum_{\mathbf{c}_1 \in S_{n,k}^{(1)}} P[\mathbf{x} = \mathbf{c}_1|\mathbf{y}]}{\sum_{\mathbf{c}_0 \in S_{n,k}^{(0)}} P[\mathbf{x} = \mathbf{c}_0|\mathbf{y}]} \end{aligned} \quad (50)$$

where $S_{n,k}^{(1)}$ denotes the set comprising all the symbols with $b_{n,k} = 1$, and so does $S_{n,k}^{(0)}$.

Applying Bayes rule and assuming the transmit symbols being equally probable, we can have (50) as

$$LLR(b_{n,k}) = \ln \frac{\sum_{\mathbf{c}_1 \in S_{n,k}^{(1)}} P[\mathbf{y}|\mathbf{x} = \mathbf{c}_1]}{\sum_{\mathbf{c}_0 \in S_{n,k}^{(0)}} P[\mathbf{y}|\mathbf{x} = \mathbf{c}_0]} \quad (51)$$

Since the noise is Gaussian, each term in the summation of (51) has an exponential form. This can be simplified by the log-sum approximation. Thus,

$$LLR(b_{n,k}) = \ln \frac{\max_{\mathbf{c}_1 \in S_{n,k}^{(1)}} P[\mathbf{y}|\mathbf{x} = \mathbf{c}_1]}{\max_{\mathbf{c}_0 \in S_{n,k}^{(0)}} P[\mathbf{y}|\mathbf{x} = \mathbf{c}_0]} \quad (52)$$

From Gaussian distribution, we have the conditional PDF $P(\mathbf{x}|\mathbf{y})$ as

$$P(X|Y) \propto \exp\left(-\frac{1}{2\sigma^2} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2\right) \quad (53)$$

Using (53) in (52), we can obtain

$$LLR(b_{n,k}) = \left\{ \min_{\mathbf{x} \in S_{n,k}^{(1)}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 - \min_{\mathbf{x} \in S_{n,k}^{(0)}} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \right\} \times \frac{1}{2\sigma^2} \quad (54)$$

With the QAM mapper, we can derive two sets, $S_{n,k}^{(1)}$ and $S_{n,k}^{(0)}$, and find one symbol minimizing the distance from \mathbf{y} to $\mathbf{H}\mathbf{x}$ in each set. The difference of the two distances is the LLR we desire. The term $\frac{1}{2\sigma^2}$ can be considered as a weighting factor. If the variance of noise is low, the weighting of the LLR is high. That means this LLR is reliable, and vice versa.

5.3 List Sphere Decoding

In previous section, we review the sphere decoding algorithm. Sphere decoding is a detection algorithm used to reduce the computational complexity of the ML detector. We can use similar idea of sphere decoding and extend it to conduct soft bit demapping. The algorithm is called list sphere decoding (LSD), and described below. First, reserve all the symbols lie inside the sphere $\|\mathbf{y}-\mathbf{H}\mathbf{x}\|^2 < r^2$, yielding a candidate list. Then, separate these candidates into two sets, one with $b_{n,k} = 1$ and the other $b_{n,k} = 0$. Finally, find the one minimizing the distance from \mathbf{y} to $\mathbf{H}\mathbf{x}$ in each set, and take the difference of the two distances to obtain the LLR. This LSD does not increase the computational complexity of the original SD algorithm too much. The LSD inherent the variable throughput problem associated with the SD. In addition, the LSD induce another problem that it cannot guarantee at least one symbol in each set. That is to say, it is possible that one of the sets is empty. A suboptimum approach is to give an extreme value to represent the LLR of the bit. If we want to avoid the case, we must enlarge the radius and raise the number of candidate. The computational complexity will be increased also. This is a performance-complexity trade-off.

5.4 Soft Output of Proposed Algorithm

The proposed in the previous section can also be extended to conduct soft bit demapping. As we can see, the parameter L is used to define the number of exhausted searches. When L is chosen as $\frac{N}{D}$, the search can cover all dimensions (thought separately, not jointly). As we can see, with this setting, the proposed algorithm can achieve the ML performance for 4×4 systems. For 2×2 and 3×3 systems, L can be chosen as less than $\frac{N}{D}$. Thus, we consider these two cases.

For the first case where $L = \frac{N}{D}$, all dimensions are searched. Thus, it guarantee at least one point can be found in two sets, $S_{n,k}^{(1)}$ and $S_{n,k}^{(0)}$. With the proposed algorithm, we can have an efficient method to find LLRs. For the ease of description, we consider a $N_t \times N_t$ system (where $N = 2N_t$ with 16-QAM transmission in a noise-free environment. The received signal can be expressed as (real-value model)

$$\mathbf{y} = \mathbf{H}\mathbf{x} \quad (55)$$

where $\mathbf{y} = [y_1 \ y_2 \ \dots \ y_N]^T$, and $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$. For each element of \mathbf{x} , the possible amplitude is $-3, -1, 1, 3$, and the corresponding mapped bits are $\{00\}, \{01\}, \{11\}, \{10\}$. Recalling (43), we can have projected vectors as \mathbf{v}_p^k 's, where $k = 1, 2, 3, 4$, for x_p . It is simple to see that for $k = 1, 2$, the first bit of x_p is 0, and for $k = 3, 4$, the first bit of x_p is 1. It is equivalent to say that two candidates are also available for $b_{p,1} = 0$ and $b_{p,1} = 1$. Define $L_p^k = \|\mathbf{y} - \mathbf{H}\mathbf{v}_p^k\|^2$. From (54), we can then approximate the LLR of $b_{p,1}$ as

$$LLR(b_{p,1}) = \min\{L_p^{-3}, L_p^{-1}\} - \min\{L_p^1, L_p^3\} \quad (56)$$

Similarly, for $k = 1, 4$, the second bit of x_p is 0, and for $k = 2, 3$, the second bit of x_p is 1. Using the similar method, we can have the LLR of $b_{p,2}$ as

$$LLR(b_{p,2}) = \min\{L_p^{-3}, L_p^3\} - \min\{L_p^{-1}, L_p^1\} \quad (57)$$

Similarly, we can derive the LLR corresponding to the 64-QAM system of x_p :

$$\begin{aligned} LLR(b_{p,1}) &= \min\{L_n^{-7}, L_n^{-5}, L_n^{-3}, L_n^{-1}\} \\ &\quad - \min\{L_n^1, L_n^3, L_n^5, L_n^7\} \end{aligned} \quad (58)$$

$$\begin{aligned} LLR(b_{p,2}) &= \min\{L_n^{-7}, L_n^{-5}, L_n^5, L_n^7\} \\ &\quad - \min\{L_n^{-3}, L_n^{-1}, L_n^1, L_n^3\} \end{aligned} \quad (59)$$

$$\begin{aligned} LLR(b_{p,3}) &= \min\{L_n^{-7}, L_n^{-1}, L_n^1, L_n^7\} \\ &\quad - \min\{L_n^{-5}, L_n^{-3}, L_n^3, L_n^5\} \end{aligned} \quad (60)$$

As shown above, the soft demapping is simple in the proposed algorithm. The above approach for the LLR calculation is for $D = 1$. It is straightforward to generalize the result for case of $D = 2$.

For the second case is where $L < \frac{N}{D}$, there is no symmetric property can be explored and we can not guarantee at least one candidate can be found for $S_{n,k}^{(1)}$ and $S_{n,k}^{(0)}$. In this case, we can use the similar idea in the LSD algorithm. We can construct a candidate list with the vectors we select in (43) or (48), correspond to the case of $D = 1$ or $D = 2$. Then, we also partition these candidates into two sets, one with $b_{n,k} = 1$ and the other with $b_{n,k} = 0$. Then, we can find the one minimizing the distance from \mathbf{y} to $\mathbf{H}\mathbf{x}$ in each set, and take the difference of the two distances to obtain the LLR. Note that the size

of candidates is LM or LM^2 , a function of L . This is contrast to the LSD algorithm in which the number of candidates cannot be controlled.

5.5 Simulations

In the section, we will report simulation results demonstrating the performance of the proposed algorithm. Similar to the last section, We consider 2×2 , 3×3 and 4×4 MIMO systems with 16-QAM and 64-QAM transmission. The channel is again a Rayleigh flat-fading channel. A convolutional code with a constraint length of 7 is used in the transmitter, and soft decoding is conducted.

Two algorithms are compared, namely, the LSD and proposed algorithms. The radius of the LSD method is set as $8 \times |\det(\mathbf{H})|^{(1/N)}$.

Figure 27-32 show the results for 16-QAM transmission. Figure 27 and 28 show the result of 2×2 systems. Let the target BER be 10^{-3} . From the figures, we can see the performance of the proposed algorithm with $D=1$ and $L=4$ is 0.5dB better than that of the LSD algorithm, and 2.5dB better than that of the LSD algorithm when $D=2$ and $L=1$. Figure 29 and 30 show the simulation result of 3×3 systems. When $D=2$ and $L=1$, the proposed algorithm can achieve the performance of that of the LSD algorithm. When $D=2$ and $L=2$, The performance of the proposed algorithm is 1.5 dB better than that of LSD. The result of 4×4 16-QAM system is shown in Figure 31 and 32. Performance of proposed algorithm with $D=2$ and $L=1$ is approximately the same as the of the LSD algorithm. When $D=2$ and $L=2$, the proposed algorithm outperform the LSD algorithm about 0.5 dB.

Figure 33-38 show the result of 64-QAM transmission. From Figure 33

and Figure 34, we can see the result of 2×2 systems. When $D=1$ and $L=2$, the proposed algorithm can have the same performance as that of the LSD algorithm, and when $D=2$ and $L=1$, the proposed algorithm can outperform the LSD algorithm about 3 dB. Figure 35 and 36 show the results for 3×3 systems. From Figure 35, we can see when $D=1$ the proposed algorithm cannot better than the LSD algorithm. From Figure 36, we found that to outperform the LSD algorithm, at least the setting of $D=2$ and $L=2$ must be used. Finally, Figure 37 and 38 show the result for 4×4 systems. When $D=2$ and $L=2$, the proposed and LSD algorithm have the same performance. With $D=2$ and $L=4$, the proposed method outperform the LSD algorithm by 1 dB.

To realize how much performance gain we can obtain with the soft decoding, we conduct another set of simulations. Figure 39 and 40 show the results of hard and soft decoding, Figure 39 for a $3 \times 3/64$ -QAM, and Figure 40 for a $4 \times 4/64$ -QAM system. Note that in soft decoding, MIMO soft demapping is required. As we can see, in 3×3 systems, the gap between the LSD and SD algorithm is approximately 2.5 dB. For the proposed algorithm with $D=2$ and $L=2$, the gap is about 3.5 dB. In 4×4 systems, the gap of the LSD and SD algorithms is 1.5 dB, and 2.2 dB of the proposed algorithm when $D=2$ and $L=4$.

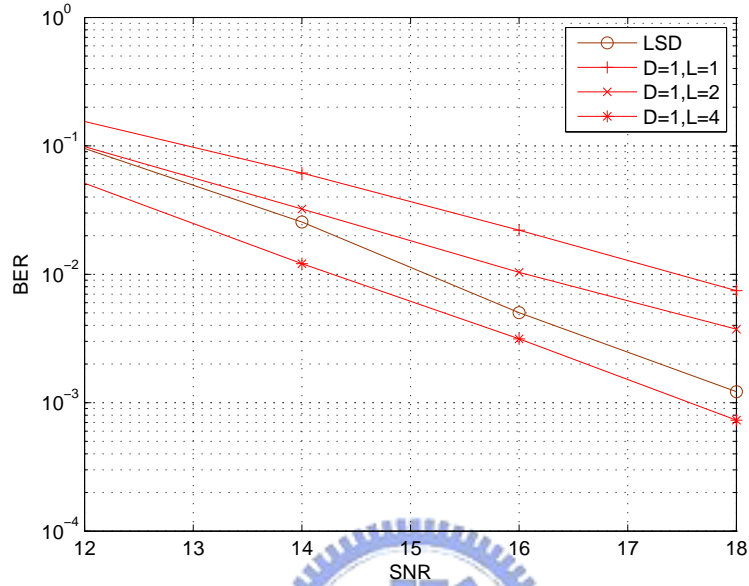


Figure 27: BER comparison of soft demapping for 2×2 16-QAM system(I)

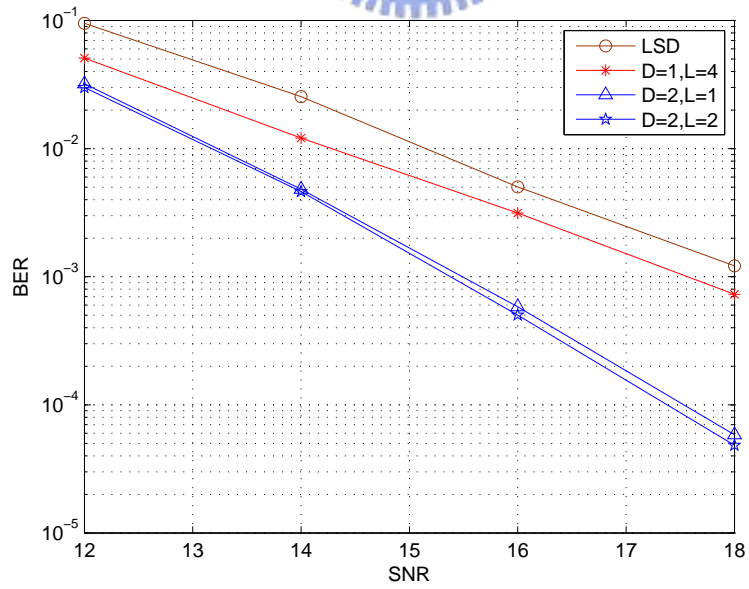


Figure 28: BER comparison of soft demapping for 2×2 16-QAM system(II)

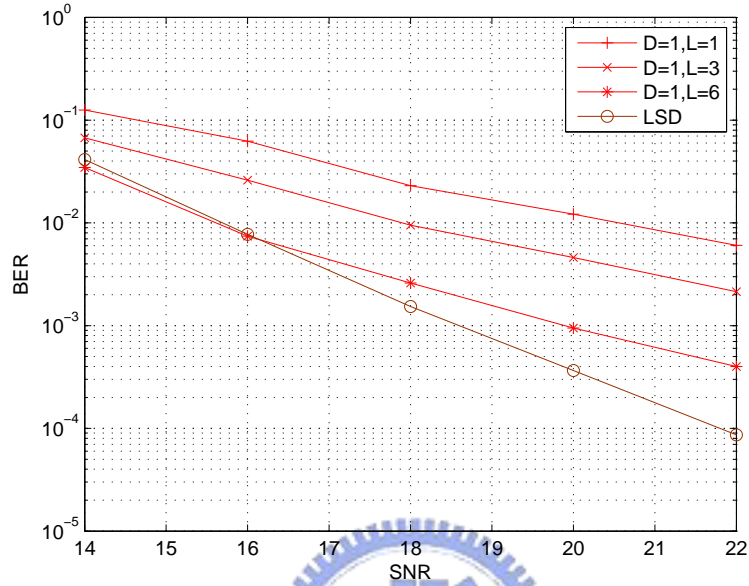


Figure 29: BER comparison of soft demapping for 3×3 16-QAM system(I)

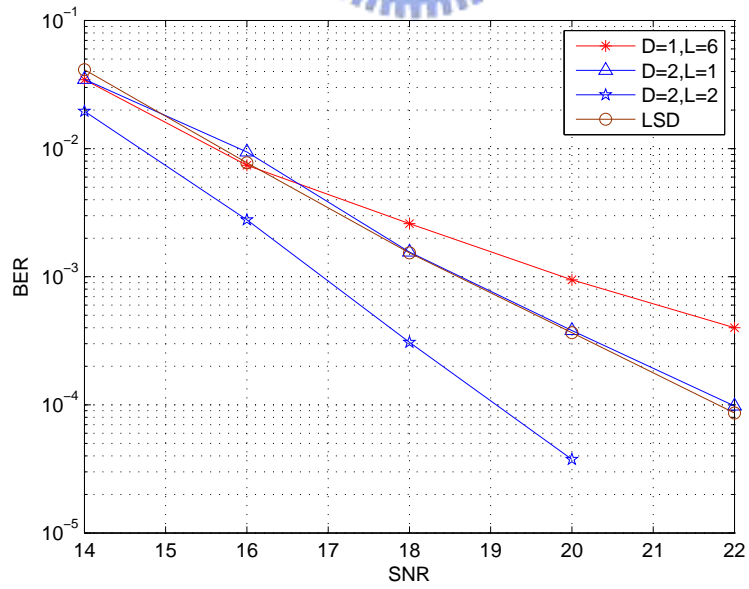


Figure 30: BER comparison of soft demapping for 3×3 16-QAM system(II)

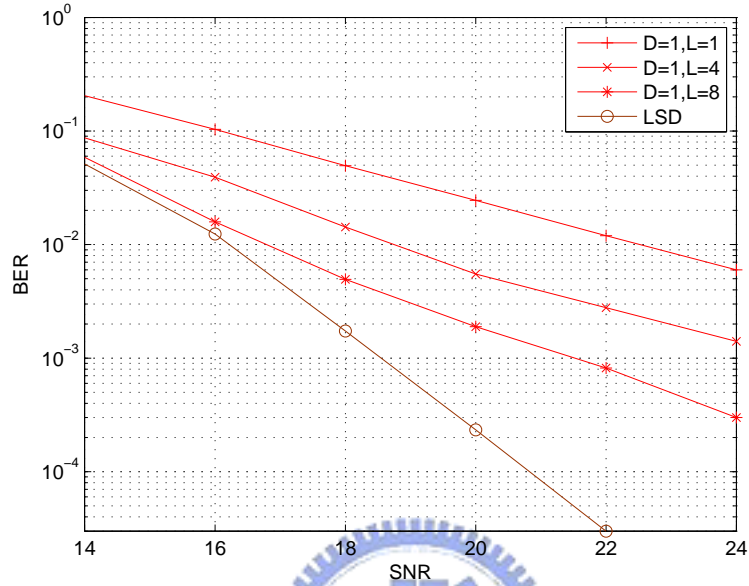


Figure 31: BER comparison of soft demapping for 4×4 16-QAM system(I)

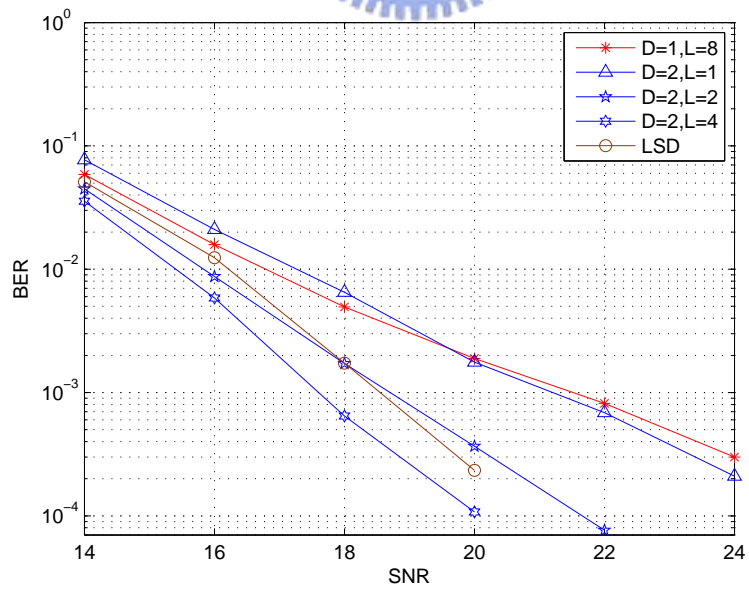


Figure 32: BER comparison of soft demapping for 4×4 16-QAM system(II)

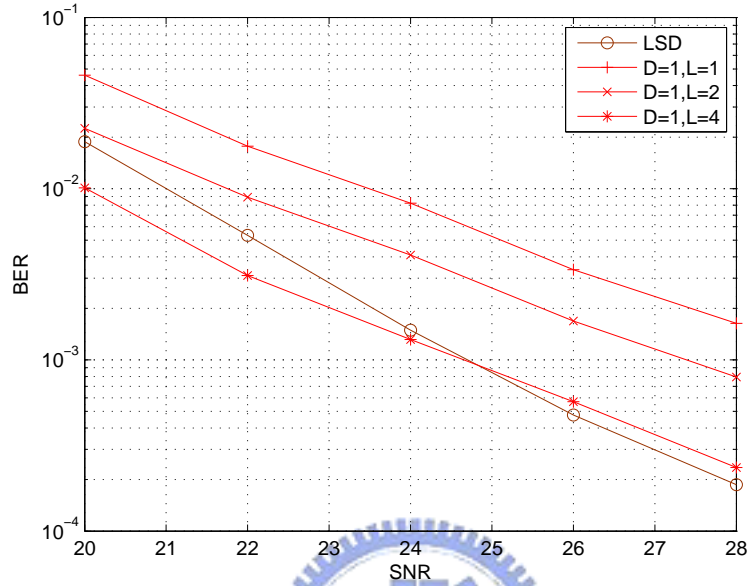


Figure 33: BER comparison of soft demapping for 2×2 64-QAM system(I)

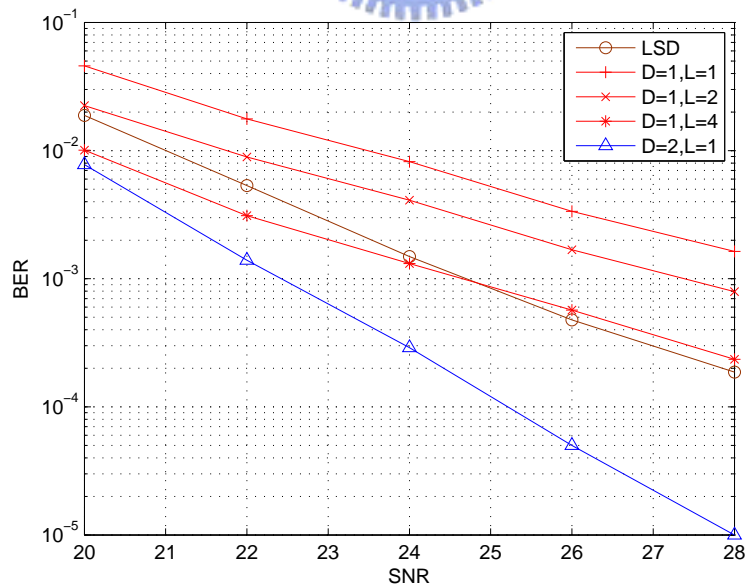


Figure 34: BER comparison of soft demapping for 2×2 64-QAM system(II)

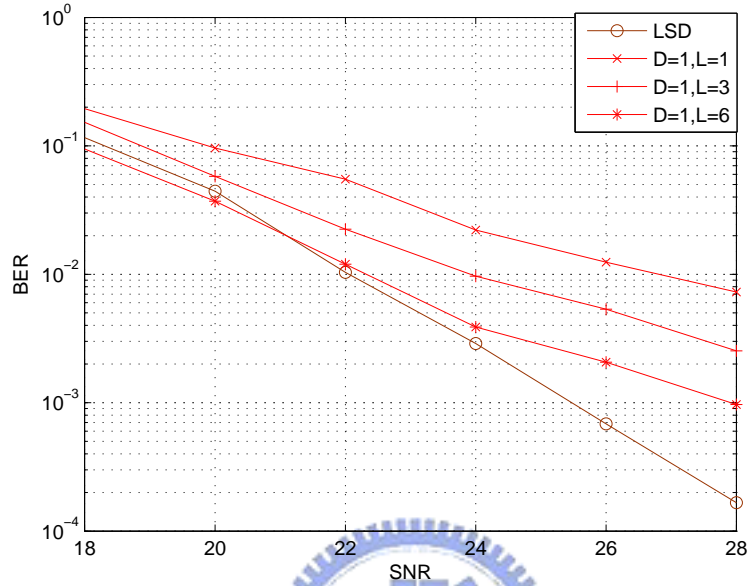


Figure 35: BER comparison of soft demapping for 3×3 64-QAM system(I)

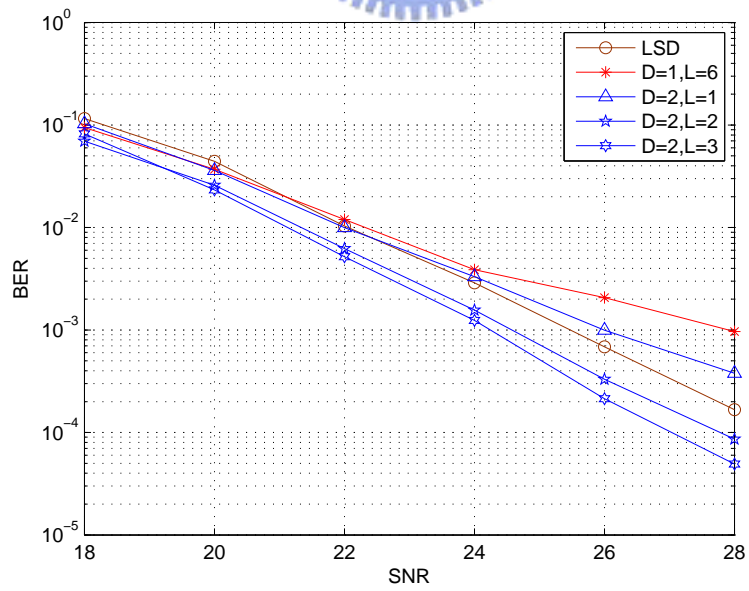


Figure 36: BER comparison of soft demapping for 3×3 64-QAM system(II)

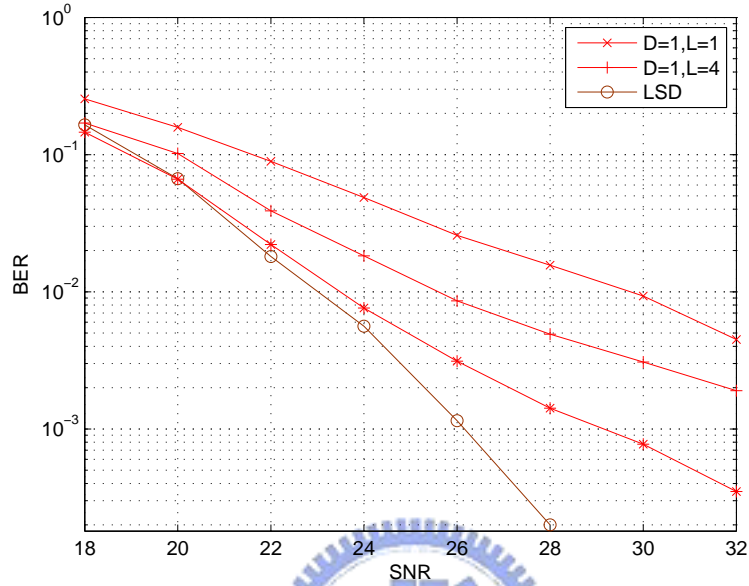


Figure 37: BER comparison of soft demapping for 4×4 64-QAM system(I)

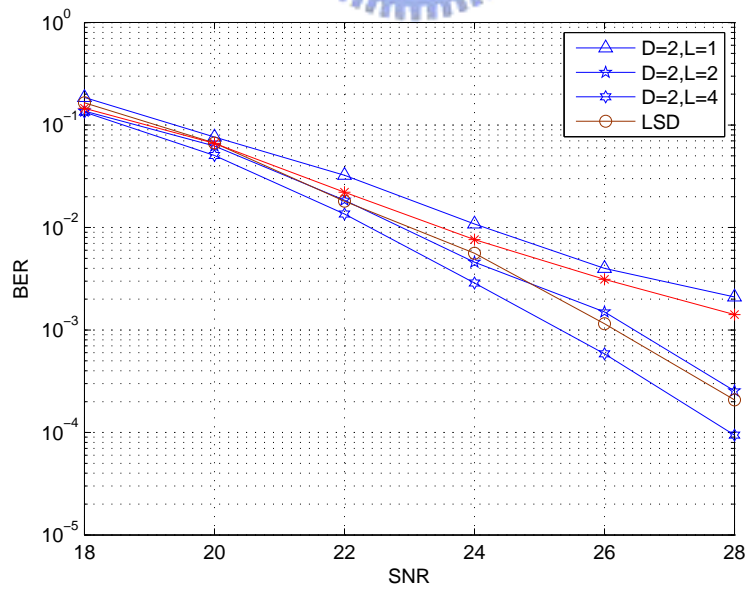


Figure 38: BER comparison of soft demapping for 4×4 64-QAM system(II)

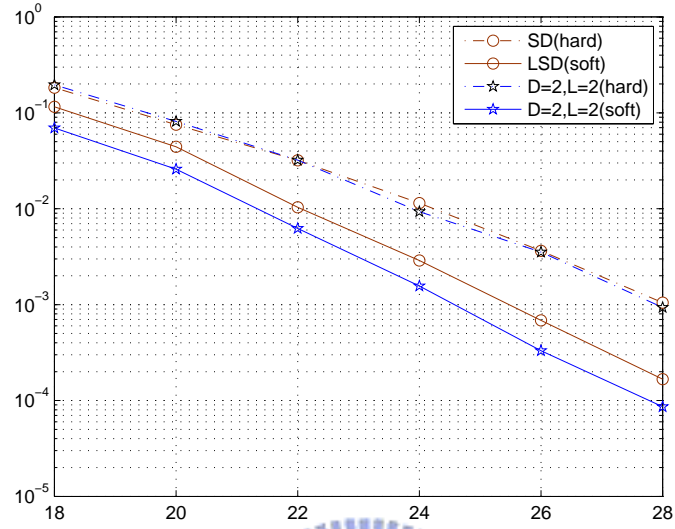


Figure 39: BER comparison of hard decision and soft demapping for 3 × 3 64-QAM system

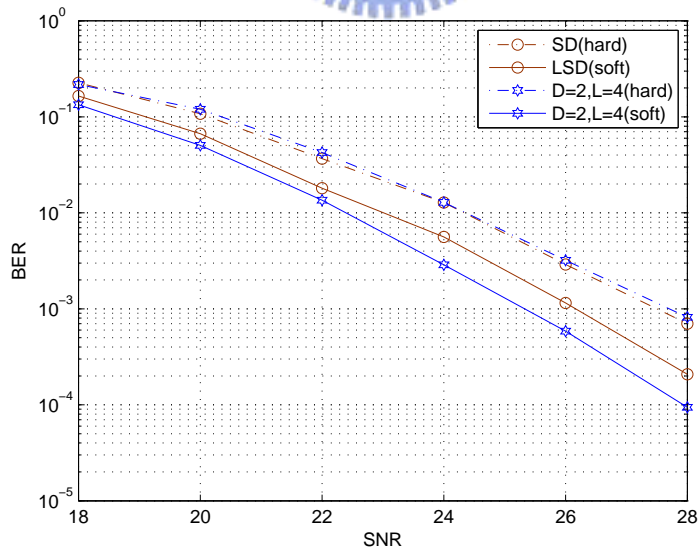
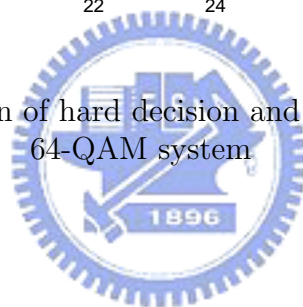


Figure 40: BER comparison of hard decision and soft demapping for 4 × 4 64-QAM system

6 Simulations With IEEE 802.11n System

In this chapter, we apply the proposed MIMO detection algorithm in the IEEE 802.11n system, and evaluate its performance. For simplicity, we only consider the channelization bandwidth of 20MHz (56 occupied sub-channels). To achieve the highest throughput, we set the number of transmitted antennas as 4 ($N_T = 4$), the number of the received antennas is also 4 ($N_R = 4$), the code rate of the convolution coder is 5/6, and the QAM size is 64. Note that this is the most challenging scenario in the IEEE 802.11n system. In order to compare the effect of different code rate, we simulate the system 4×4 64-QAM with 1/2 code rate. Besides, we also consider the common use mode 2×2 16-QAM with 1/2 code rate.

Packet error rate (PER) is used as the performance measure. The payload in a packet is fixed as 1000 bits. According to the specification, a PER of 0.1 is required. We consider three receiver structures. The first one is the conventional approach that a MMSE equalizer is first used to decouple the MIMO channel into multiple SISO channels, and then a SISO soft-demapper is applied for each SISO channels. The second and the third use MIMO soft-bit demapping directly, one with the LSD algorithm, and the other with the proposed algorithm.

Three different MIMO channel models, defined in the specification, are used in our simulations (TGn Sync Channel B, TGn Sync Channel D, and Rayleigh uncorrelated Channel). We also assume that the frequency offset and timing offset have been perfectly compensated in the receiver. The channel characteristics introduced in Chapter 2.7.

From Figure 41 to Figure 43 show the simulation results in different channel. We can see that our proposed algorithm with $D=2$ and $L=4$ significantly outperforms the conventional approach about 14dB in these three channel. When $D=1$ and $L=8$, the performance of proposed algorithm is still 10 dB better than the conventional approach in these three channel. Compared with LSD, the performance of our proposed algorithm when $D=2$ and $L=4$ is near that of LSD in channel B, 0.5dB better in channel D and 1.5dB better in uncorrelated channel. As we can see, our proposed algorithm have better performance in uncorrelated channel.

Figure 44 shows the result of 4×4 64-QAM with $1/2$ code rate in uncorrected channel. We can see that performance of our proposed algorithm with $D=2$ and $L=4$ is only 7dB better than that of MMSE equalizer. When $D=1$ and $L=8$, our proposed algorithm is 6dB better than that of MMSE equalizer, and close to that of LSD. When the coding rate is low, the gap of our proposed algorithm and conventional MMSE equalizer become smaller.

The result of the common use mode 2×2 16-QAM with $1/2$ code rate is shown in Figure 45. We can see that our proposed algorithm performs 6dB better than MMSE equalizer when $D=2$ and $L=2$, 5 dB when $D=1$ and $L=4$.

In conclusion, compared with conventional method, our proposed algorithm can have better performance in higher throughput mode.

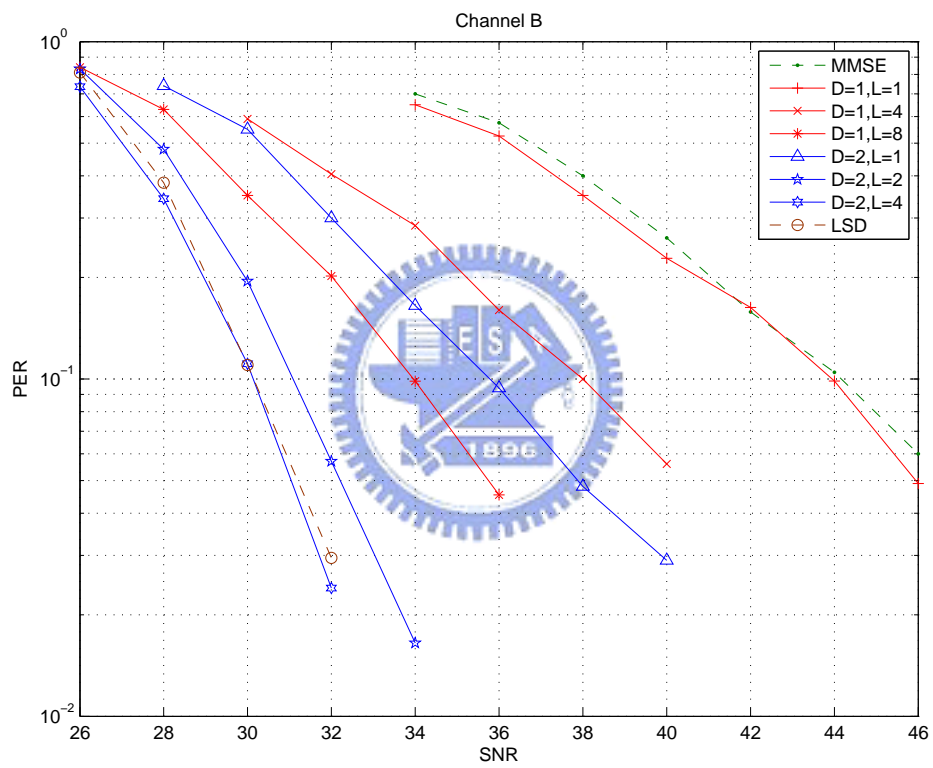


Figure 41: PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Channel B)

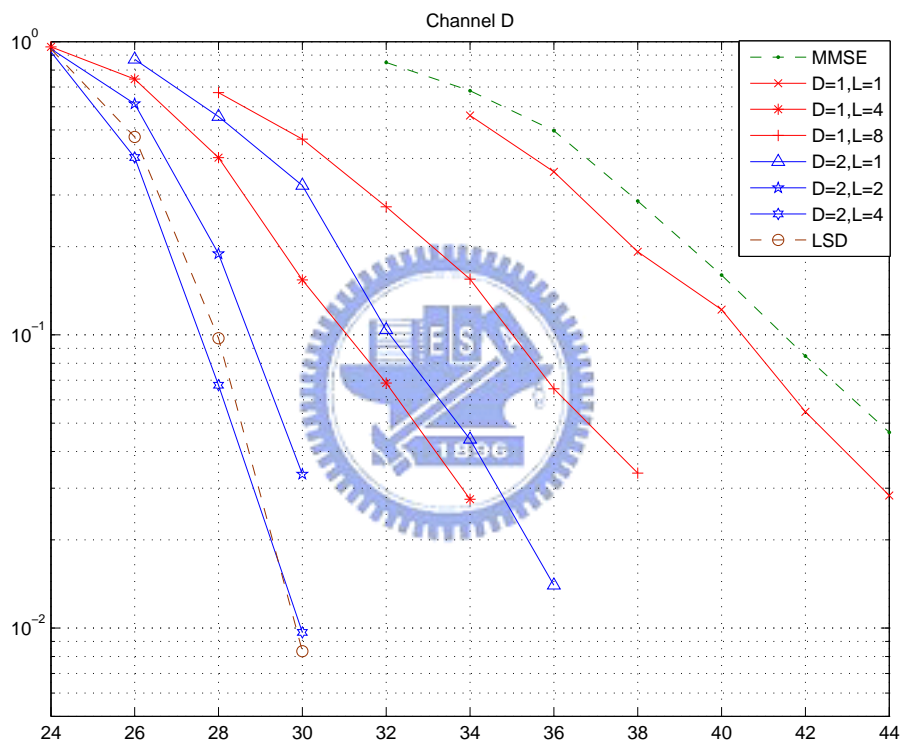


Figure 42: PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Channel D)

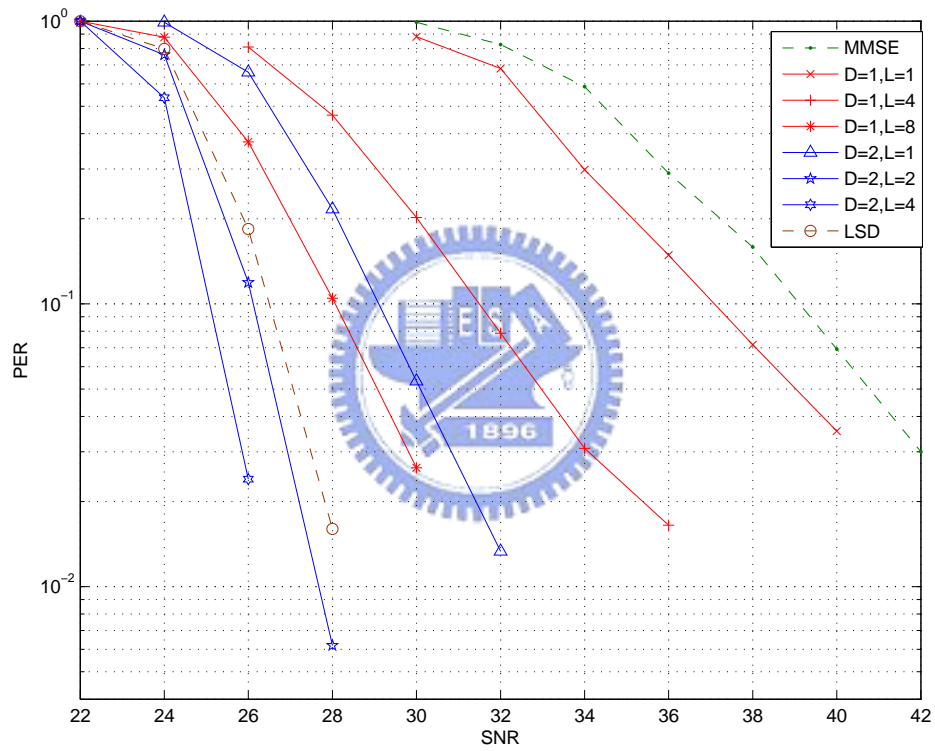


Figure 43: PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 5/6 code rate, Rayleigh uncorrelated Channel)

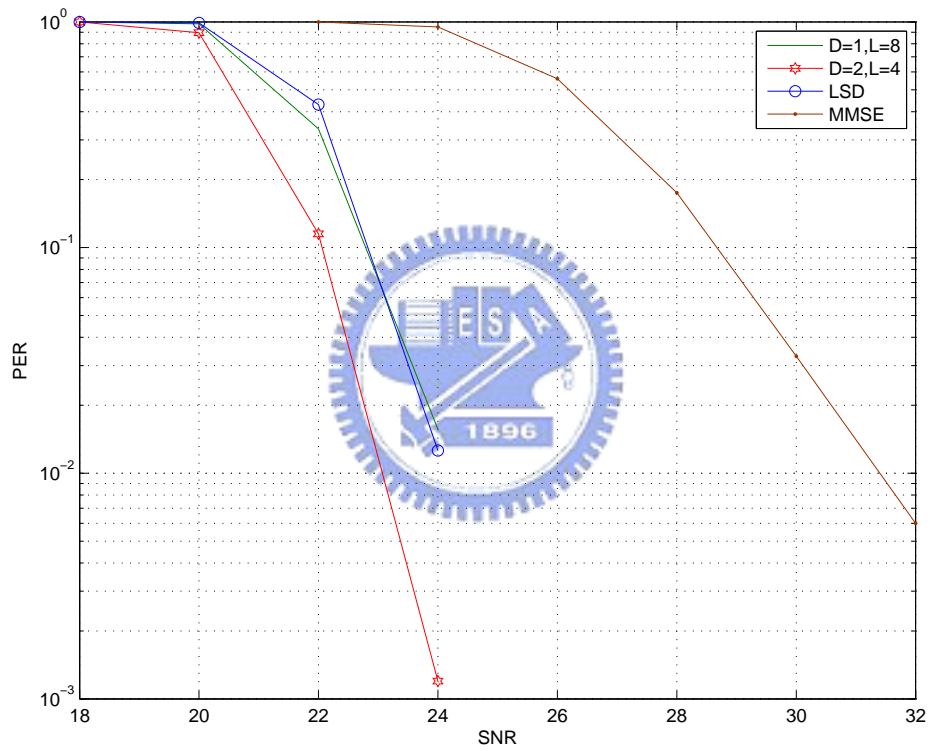


Figure 44: PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 4×4 , 64QAM, 1/2 code rate, Rayleigh uncorrelated Channel)

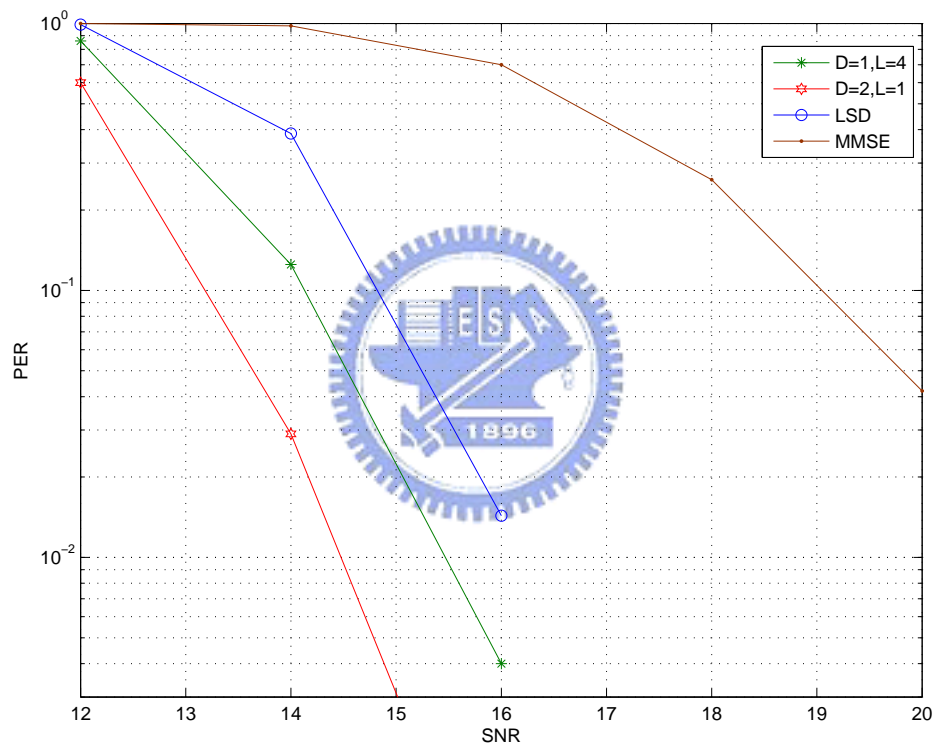


Figure 45: PER comparisons of Proposed Algorithm, LSD and MMSE (20MHz, 2×2 , 16QAM, 1/2 code rate, Rayleigh uncorrelated Channel)

7 Conclusions

7.1 Conclusion

In this thesis, we consider the optimum receiver design in the IEEE 802.11n system. The system uses the MIMO-OFDM scheme to provide a high throughput wireless transmission. The optimum hard-decision MIMO detector, known as the ML detector, can provide superior performance; however, its computational complexity is known to be very high. We then propose a new MIMO detection algorithm to overcome the problem. The proposed algorithm can significantly reduce the computational complexity of the ML algorithm, while the performance remains the same. The proposed algorithm is extended to include soft-decision scenarios. Similar to the hard-decision case, the proposed can greatly reduce the computational complexity of optimum MIMO demapping. Although the SD/LSD algorithm can also reduce the computational complexity of optimum detectors, their throughputs are not constant. This is a serious drawback in real-world implementation. Simulation results with IEEE 802.11n system show that (with different QAM sizes and different spatial streams) the proposed algorithm is significantly better than the conventional approach, a MMSE equalizer combined with a SISO demapper. The more spatial streams and the larger QAM size a system has, the more computations the proposed algorithms can save.

7.2 Future Work

Although the complexity of the proposed algorithms is much lower than that of the ML/optimum MIMO demapper, it is still higher than the MMSE algo-

rithm. How to further reduce the computational complexity of the proposed algorithms worth further studying. In this thesis, we only consider the case where the number of transmit antenna equals to that of the received antenna. With minor modification, the proposed algorithm can be applied in the case where the number of transmit antenna is large than the number of received antenna. Finally, we may also consider the hardware structure that can efficiently implement proposed algorithms.



References

- [1] G.J. Foschini and M.J. Gans, On limits of wireless communications in a fading environment when using multiple antennas, *Wireless Personal Communications*, vol. 6, pp. 311-335, 1998.
- [2] E. Telatar, Capacity of multi-antenna Gaussian channels, *European Trans. on Telecommunications*, vol. 10, no. 6, pp. 585-595, Nov./Dec. 1999.
- [3] S. M. Alamouti, A simple transmit diversity technique for wireless communication, *IEEE JSAC*, vol. 16, no. 8, pp. 1451V1458, Oct. 1998.
- [4] G. J. Foschini, Layered space-time architecture for wireless communication in a fading environment when using multiple antennas, *Bell Labs Syst. Tech. J.*, vol. 1, pp. 41-59, Autumn 1996.
- [5] P. W. Wolnainsky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electronics Letters*, vol. 35, pp. 14-16, Jan. 1999.
- [6] B. R. Salzberg, Performance of an efficient parallel data transmission system, *IEEE Trans. on Commun.*, vol. COM-15, pp. 805-811, Dec. 1967.
- [7] Bertrand M. Hochwald and Stephan ten Brink, Achieving near-capacity on a multiple-antenna channel, *IEEE Trans. on Commun.*, vol. 51, no. 3, pp. 389-399, Mar. 2003.

- [8] E. Zehavi. "8-PSK trellis codes for a rayleigh channel," IEEE Trans. on Commun., vol. 40, pp.873-884, May 1992
- [9] W Zhao, GB Giannakis, Sphere decoding algorithms with improved radius search, IEEE Trans. on Commun., vol. 53, no. 7, pp. 1104- 1109, Jul. 2005.
- [10] Jacob Benesty, Yiteng Huang and Jingdong Chen. "A Fast Recursive Algorithm for Optimum Sequential Signal Detection in a BLAST System." IEEE Trans. On Signal Processing, vol. 51, no. 7, pp. 1722- 1730, Jul. 2003.

