



HAL
open science

Application of Relay Coding in Wi-Fi for 802.11s Standardization

Zhipeng Zhao

► **To cite this version:**

Zhipeng Zhao. Application of Relay Coding in Wi-Fi for 802.11s Standardization. Networking and Internet Architecture [cs.NI]. Télécom ParisTech, 2010. English. NNT: . pastel-00559723

HAL Id: pastel-00559723

<https://pastel.hal.science/pastel-00559723>

Submitted on 26 Jan 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Application des Codes Relais au Wi-Fi en vue de la
Standardisation IEEE 802.11s

ZHAO Zhipeng

January 25, 2011

Abstract

Ce mémoire de thèse porte sur la conception, la validation, et l'application du codage canal en mode relais dans le cadre de système WiFi. A cette fin, le modèle de transmission coopératif est introduit dans le chapitre 1. Pour l'application d'une diversité de coopération, il est nécessaire de mettre en oeuvre le codage et le décodage de code spatio-temporel. C'est l'objet du chapitre 2 qui se consacre à l'étude du décodeur MIMO intégré dans le système coopératif.

Le chapitre 3 se consacre à l'étude du protocole baptisé Relais-SISO et sa mise en oeuvre dans le standard IEEE802.11a à 5 GHz. Pour cela de nouvelles structures de trame pour la couche physiques sont proposées en ajoutant des préambules destinés à être exploités par le mode coopératif. On caractérise ainsi une transmission à 3 terminaux source, relais et destinataire. Des résultats de simulation démontrent l'avantage de la technique coopérative. A notre connaissance, ces travaux sont les premiers à proposer une application du codage relais au WiFi.

Le chapitre 4, une stratégie hybride basée sur la solution Relais-SISO est présentée car elle améliore les performances du système Relais-SISO. Le mode hybride proposé s'appuie sur l'évaluation de la qualité du lien coopératif pour activer ce mode.

Dans le chapitre 5, l'optimisation du mode coopératif de la couche physique est réalisé en mettant en oeuvre la chaîne de partitionnement du Golden Code. On met en évidence le fait qu'une simple combinaison de ce dernier et d'un code convolutif, fournit un gain de codage grâce au gain du déterminant du Golden code.

Le travail serait incomplet sans tenir compte du nouveau standard WiFi IEEE802.11n ratifié en Octobre 2009. C'est l'objet du chapitre 6, qui propose un protocole coopératif, baptisé Relais-MIMO, qui exploite d'une manière conjointe la diversité MIMO et la diversité de coopération. Un scénario mettant oeuvre un émetteur à 2 antennes, un relais à 2 antennes, un destinataire à 2 antennes, est proposé en appliquant un schéma de codage parfait de type 4×4 .

CONTENTS

1	Cooperative Diversity	3
1.1	Cooperative system	3
1.2	Cooperative diversity and STBC	4
1.3	Cyclic division algebra based STBC	8
2	MIMO Decoder	13
2.1	MIMO system description	13
2.2	Left processing: MMSE-GDFE	15
2.3	Right processing : column permutation and lattice reduction	16
2.4	Lattice decoding	17
2.4.1	ML decoding algorithm	18
2.4.2	Algorithm Zero-Forcing	20
2.4.3	Heuristic algorithms: DFE and KSE	20
2.4.4	Sequential decoder based algorithm : Fano decoder and stack decoder	23
2.4.5	Generalized Schnorr-Euchner decoder: SEF decoder	26
2.5	MIMO decoder performance	26
2.5.1	Direct decoding	27
2.5.2	Greedy ordering	33
2.5.3	LLL reduction	39
2.5.4	Conclusion	45
3	Relay-SISO	46
3.1	IEEE 802.11a PHY layer	47
3.2	Relay-SISO PHY layer	51
3.3	Relay-SISO preambles and cooperation procedure	55

3.3.1	Data field	55
3.3.2	Carrier frequency offset estimation and correction	56
3.3.3	Amplify-and-forward procedure	59
3.3.4	Channel coefficients estimation	60
3.3.5	Cooperation procedure	64
3.4	Relay-SISO Performance	67
3.4.1	Flat fading channel	67
3.4.2	Multipath channel	68
3.4.3	Discussion on performance	70
4	Hybrid Cooperation	71
4.1	Normalized minimum squared Euclidean distance	71
4.2	Equivalent metric for IEEE 802.11a and Relay-SISO	80
4.3	Hybrid mode and performance	90
5	Golden Code Coset and Its Application in Cooperation System	92
5.1	Golden code and partition	92
5.2	Golden code partition and coset bit mapping	96
5.3	Application and performance	101
5.3.1	New modulation and coding scheme for data rate 12Mbps in Relay-SISO	101
5.3.2	New modulation and coding scheme for data rate 24Mbps in Relay-SISO	102
5.3.3	Application in 2×2 MIMO system	102
6	Relay-MIMO	105
6.1	IEEE 802.11n PHY Layer	105
6.2	Relay-MIMO PHY Layer	110
6.3	Relay-MIMO preambles and cooperation procedure	113
6.3.1	HT data field	114
6.3.2	Cyclic shift processing	114
6.3.3	AF procedure	115
6.3.4	Channel coefficient estimation	118

6.4	Relay-MIMO Performance	123
6.4.1	Flat fading channel	123
6.4.2	Multipath channel	123
6.4.3	Discussion on performance	124
6.5	Algebraic reduction for 4×4 Perfect code	125
6.5.1	Algebraic reduction	125
6.5.2	Algebraic reduction for 4×4 Perfect code	128
6.5.3	Performance	133

LIST OF FIGURES

2.1	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 2x2 MIMO system with QPSK constellation	27
2.2	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 2x2 MIMO system with 16-QAM constellation	27
2.3	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 4x4 MIMO system with QPSK constellation	28
2.4	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 4x4 MIMO system with 16-QAM constellation	28
2.5	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 16x16 MIMO system with QPSK constellation	28
2.6	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 16x16 MIMO system with 16-QAM constellation	29
2.7	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 2x2 MIMO system with QPSK constellation	30
2.8	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 2x2 MIMO system with 16-QAM constellation	30
2.9	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 4x4 MIMO system with QPSK constellation	31
2.10	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 4x4 MIMO system with 16-QAM constellation	31
2.11	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 16x16 MIMO system in QPSK constellation	31

2.12	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 16x16 MIMO system with 16-QAM constellation	32
2.13	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 2x2 MIMO system with QPSK constellation	33
2.14	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 2x2 MIMO system with 16-QAM constellation	33
2.15	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 4x4 MIMO system with QPSK constellation	34
2.16	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 4x4 MIMO system with 16-QAM constellation	34
2.17	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 16x16 MIMO system with QPSK constellation	34
2.18	MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 16x16 MIMO system with 16-QAM constellation	35
2.19	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 2x2 MIMO system with QPSK constellation	36
2.20	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 2x2 MIMO system with 16-QAM constellation	36
2.21	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 4x4 MIMO system with QPSK constellation	37
2.22	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 4x4 MIMO system with 16-QAM constellation	37
2.23	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 16x16 MIMO system with QPSK constellation	38
2.24	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 16x16 MIMO system with 16-QAM constellation	38

2.25	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 2x2 MIMO system with QPSK constellation	39
2.26	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 2x2 MIMO system with 16-QAM constellation	39
2.27	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 4x4 MIMO system with QPSK constellation	40
2.28	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 4x4 MIMO system with 16-QAM constellation	40
2.29	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 16x16 MIMO system with QPSK constellation	40
2.30	MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 16x16 MIMO system with 16-QAM constellation	41
2.31	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 2x2 MIMO system with QPSK constellation	42
2.32	MIMO decoder performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 2x2 MIMO system with 16-QAM constellation	42
2.33	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 4x4 MIMO system with QPSK constellation	43
2.34	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 4x4 MIMO system with 16-QAM constellation	43
2.35	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 16x16 MIMO system with QPSK modulation	43
2.36	MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 16x16 MIMO system with 16-QAM constellation	44
3.1	802.11a general architecture of processing	47
3.2	Convolutional encoder: $R = 1/2$ with $g_0 = (133)_8$ and $g_1 = (171)_8$	48
3.3	Output of convolutional code at $R = 1/2$	48

3.4	Output of convolutional code at $R = 2/3$	48
3.5	Output of convolutional code at $R = 3/4$	49
3.6	802.11a frequency mapping for the OFDM symbol: 48 subcarriers are used for transmission of data symbol while 4 subcarriers are used for pilots at frequency index -21,-7,+7,+21; the other 12 subcarriers including DC are kept no used which generate protection band.	50
3.7	IEEE 802.11a frame	51
3.8	Relay-SISO PHY layer where an STBC unit is added between mapping unit and IFFT unit	51
3.9	Relay-SISO S frame format	52
3.10	Relay-SISO R frame format	54
3.11	Relay-SISO D frame format	54
3.12	Relay-SISO channel parameters estimation	61
3.13	Relay-SISO PHY layer transmission procedure	65
3.14	Relay-SISO PHY layer reception procedure	66
3.15	Relay-SISO and 802.11a over flat fading channel: each MPDU contains 125 bytes and it is sent at 12Mbps.	68
3.16	Relay-SISO and 802.11a over flat fading channel: each MPDU contains 125 bytes and it is sent at 24Mbps.	68
3.17	Relay-SISO and 802.11a over channel type A: each MPDU contains 125 bytes and it is sent at 12Mbps.	69
3.18	Relay-SISO and 802.11a over channel type A: each MPDU contains 125 bytes and it is sent at 24Mbps.	70
4.1	16-QAM constellation	73
4.2	Word error rate in function of NMSED (D_{\min}^2) with QPSK modulation.	76
4.3	Word error rate in function of NMSED (D_{\min}^2) with 16-QAM modulation.	76
4.4	Non-cooperation, cooperation and Hybrid transmission in Rayleigh-fading channel with QPSK modulation: in cooperation(including Hybrid mode) case, the path gain are configured to $G_{sr} = 0\text{dB}$, $G_{rd} = 0\text{dB}$ and $G_{sr} = 10\text{dB}$, $G_{rd} = 10\text{dB}$	77
4.5	Non-cooperation, cooperation and Hybrid transmission in Rayleigh-fading channel with 16-QAM modulation: in cooperation(including Hybrid mode) case, the path gain are configured to $G_{sr} = 0\text{dB}$, $G_{rd} = 0\text{dB}$ and $G_{sr} = 10\text{dB}$, $G_{rd} = 10\text{dB}$	78
4.6	Percentage of cooperation in Hybrid mode with QPSK modulation	78
4.7	Percentage of cooperation in Hybrid mode with 16-QAM modulation	79

4.8	Received adjacent points having Hamming distance $d_h > 1$	80
4.9	Example of IEEE 802.11a interleaving/deinterleaving for 16-QAM	81
4.10	Frame error rate in function of NMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.	81
4.11	Bit error rate in function of NMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.	81
4.12	Frame error rate in function of ENMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.	89
4.13	Frame error rate in function of ENMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 400 bytes at datarate 24Mbps.	89
4.14	Relay-SISO, IEEE 802.11a and Hybrid over flat fading channel: each MPDU contains 125 bytes and it is sent at 12Mbps.	90
4.15	Relay-SISO, IEEE 802.11a and Hybrid over flat fading channel: each MPDU contains 125 bytes and it is sent at 24Mbps.	91
4.16	Relay-SISO, IEEE 802.11a and Hybrid over channel type A: each MPDU contains 125 bytes and it is sent at 12Mbps.	91
4.17	Relay-SISO, IEEE 802.11a and Hybrid over channel type A: each MPDU contains 125 bytes and it is sent at 24Mbps.	91
5.1	802.11a convolutional code (133,171) with different puncture mode	101
5.2	Partitioned Golden code for Relay-SISO 12Mbps	102
5.3	Partitioned Golden code for Relay-SISO 24Mbps	103
5.4	Partitioned Golden code in MIMO Rayleigh fading channel	104
6.1	802.11n mixed mode frame	106
6.2	802.11n general architecture of processing	106
6.3	Output of convolutional code at $R = 5/6$	107
6.4	IEEE 802.11n 20MHz frequency mapping for the HT portion OFDM sym- bol: 52 subcarriers are used for transmission of data symbol while 4 sub- carriers are used for pilots at frequency index -21,-7,+7,+21; the other 8 subcarriers including DC are kept no used which generate protection band.	108
6.5	Relay-MIMO PHY layer for data A: the 4×4 Perfect code is used where the inputs are 2 parallel QAM symbol streams. The coded signal is partially sent on 2 Tx antennas as in IEEE 802.11n configuration while the rests are kept in a buffer.	110
6.6	Relay-MIMO PHY layer for data B: the memoried codewords are transmitted	111
6.7	Relay-MIMO baseband equivalent transmission viewed at the receiver's side	112

6.8	Relay-MIMO S frame: the S frame is sent by the source terminal	112
6.9	Relay-MIMO R frame: the R frame is sent by the relay terminal	112
6.10	Relay-MIMO D frame: the D frame is received by the destination terminal	112
6.11	Relay-MIMO R frame's preambles	113
6.12	Relay-MIMO compatibility with IEEE 802.11a, IEEE 802.11n and Relay-SISO PHY layer	113
6.13	Relay-MIMO channel parameters estimation	119
6.14	Relay-MIMO and IEEE 802.11n over 2×2 MIMO flat fading channel: each MPDU contains 125 bytes and it is sent at MCS=9(26Mbps).	123
6.15	Relay-MIMO and IEEE 802.11n over 2×2 MIMO flat fading channel: each MPDU contains 125 bytes and it is sent at MCS=11(52Mbps).	124
6.16	Relay-MIMO and 802.11n over 2×2 MIMO channel type A: each MPDU contains 125 bytes and it is sent at MCS=9(26Mbps).	124
6.17	Relay-MIMO and 802.11n over 2×2 MIMO channel type A: each MPDU contains 125 bytes and it is sent at MCS=11(52Mbps).	125
6.18	ZF-DFE decoder with algebraic reduction for 4x4 Perfect code in single-relay two-antenna NAF system	133

LIST OF TABLES

2.1	Lattice QAM symbols set \mathcal{S}	14
3.1	IEEE 802.11a rate-dependent parameters	47
3.2	Multipath channel type A	69
4.1	Simplified multiplication table of $\mathcal{D}^2(\mathcal{D}^2, d_H)$	88
4.2	Simplified addition table of $\mathcal{D}^2(\mathcal{D}_1^2, \mathcal{D}_2^2)$	88
6.1	802.11n cyclic shift value definition for non-HT portion	107
6.2	802.11n cyclic shift value definition for HT portion	107
6.3	IEEE 802.11n MCS parameters for 20MHz, $N_{SS} = 2$, GI=800ns, EQM . .	109

ACRONYMS

A-LTF Amplified Long Training Field

AF Amplify-and-Forward

AHT-LTF Amplified High-Throughput Long Training Field

AGC Automatic Gain Control

AR Algebraic Reduction

AWGN Additive White Gaussian Noise

BER Bit Error Rate

BPSK Binary Phase-Shift Keying

CDA Cyclic Division Algebra

CFO Carrier Frequency Offset

CB Cooperative Block

CLPS Closest Lattice Point Search

CP Cyclic Prefix

CS Cyclic Shift

CSD Cyclic Shift Delay

CSI Channel State Information

DC Direct Current

DF Decode-and-Forward

DFE Decision Feedback Equalizer

DFT Discrete Fourier Transform

DMT Diversity-Multiplexing gain Tradeoff

ENMSED Equivalent Normalized Minimum Squared Euclidean Distance

EQM Equal Modulation

FEC Forward Error Correction
FER Frame Error Rate
FFT Fast Fourier Transform
FSVL Finding Small Vectors in Lattices
GF Green Field
GI Guard Interval
GI2 Guard Interval for long training symbols
GO Greedy Ordering
HPG High Path Gain
HT High-Throughput
HT-DATA High-Throughput Data Field
HT-LTF High-Throughput Long Training Field
HT-SIG High-Throughput SIGNAL Field
HT-STF High-Throughput Short Training Field
i.i.d independant and identically-distributed
IFFT Inverse Fast Fourier Transform
KSE K-best Schnorr-Euchner
LDPC Low-Density Parity-Check
LLL Lenstra, Lenstra and Lovasz
LLR Log-Likelihood Ratio
LPF Low-Pass Filter
LPG Low Path Gain
L-LTF Legacy Long Training Field
L-SIG Legacy SIGNAL Field
L-STF Legacy Short Training Field
LTF Long Training Field
MAC Medium Access Control
MCS Modulation and Coding Scheme
MIMO Multiple-Input Multiple-Output

ML Maximum Likelihood

MM Mixed Mode

MMSE Minimum Mean Square Error

MMSE-GDFE Minimum Mean Square Error Generalized Decision-Feedback Equalizer

MPDU MAC Protocol Data Unit

MSED Minimum Squared Euclidean Distance

NAF Non-orthogonal AF

NAV Network Allocation Vector

NMSED Normalized Minimum Squared Euclidean Distance

OAF Orthogonal AF

OFDM Orthogonal Frequency-Division Multiplexing

PA Power Amplifier

PHY Physical

PSDU Physical Service Data Unit

PLCP Physical Layer Convergence Procedure

Q-BPSK Quadrature BPSK

QAM Quadrature Amplitude Modulation

QoS Quality of Service

QPSK Quadrature Phase-Shift Keying

R-LTF Relayed Long Training Field

RF Radio Frequency

RHT-LTF Relayed High-Throughput Long Training Field

SD Sphere Decoder

SE Schnorr-Euchner

SEF SE decoder with Fano-like metric

SFO Sampling Frequency Offset

SNR Signal-to-Noise Ratio

STF Short Training Field

STBC Space-Time Block Code

TCM Trellis Coded Modulation

WER Word Error Rate

WLAN Wireless Local Area Network

ZF Zero-Forcing

ACKNOWLEDGEMENTS

First, I thank my advisor Professor Jean-Claude BELFIORE, for his kind guidance during my research work. He shows me the way towards the wireless communications and the method to solve the encountered problems.

I would like to take this opportunity to thank the members of COMELEC department in ENST for their enlightening lessons during my studies: Ghaya REKAYA who teaches the space-time coding and Philippe CIBLAT who teaches the signal processing.

I appreciate also the help of Sheng YANG for his advices in cooperative system.

Special thanks to my colleague David CHAMPION for his great helps in Wi-Fi system development and STBC coding/decoding solution, without his assistance this thesis would not have been possible.

I also give gratitude to Bruno ROUVIO for his kind assistances in computer and network techniques.

Let me say “thank you” to my dear colleagues in COMSIS: Philippe LECLAIR and Roxana OJEDA, who gave me great supports to complete this thesis; Olivier NALPAS and Farid ZAOUIA, who dedicated their precious time to explain the hardware architecture.

This work is partly supported by RNRT in the project RNRT/RADIC-SF/COMSIS with reference number ANR-05-RNRT-014-01.

Last but not least, I thank my parents and my wife for their continuous encouragements since always.

INTRODUCTION

The Wi-Fi system has been largely deployed for indoor or outdoor applications. This technology builds on IEEE 802.11 standards and operates on both 2.4GHz and 5.2GHz. Like many other wireless communication systems, this system suffers from the fading problem that the transmissions experience severe signal attenuation. The Multiple-Input Multiple-Output (MIMO) technology, which has been introduced in the new Wi-Fi generation IEEE 802.11n, can be an efficient method to combat the fading problem. However, multi-antennas solution will be expensive and sometimes MIMO technology can not be implemented due the device's size.

Recently, a new diversity technique, names *cooperative diversity*, suggests an evolutive solution. In the works of Laneman and Wornell [1, 2], the authors introduced a new family of transmission protocols that exploit the distributed antennas network. These transmission protocols propose a users' cooperation scheme in which the diversity gain is achieved by applying the *relaying* technique. Research works in [3, 4] show the advantages of different cooperation protocols in terms of the diversity and multiplexing gain tradeoff[5].

Generally, the relaying techniques is divided into two categories: decode-and-forward and amplify-and-forward. Compared with decode-and-forward cooperation, the amplify-and-forward mode is easier to be applied in the current Wi-Fi system. Considering the spectral efficiency, the non-orthogonal amplify-and-forward scenario is more attractive than the orthogonal amplify-and-forward scenario. In this thesis, we focus on the application of the cooperative codes for the non-orthogonal amplify-and-forward cooperation in the context of Wi-Fi networks. As far as we know today, this is the first application of the cooperative diversity in the Wi-Fi systems.

This thesis is organized as follows: in chapter 1, the NAF cooperation scenario is presented as well as the coding schemes for the single-relay single-antenna and single-relay two-antenna cooperation systems. In chapter 2, we will focus on the MIMO decoders to evaluate the performance and the complexity of different decoder algorithms. Based on IEEE 802.11a, the cooperative system Relay-SISO is proposed in chapter 3 and in chapter 4 we will show a hybrid cooperative strategy which operates in 802.11a mode and Relay-SISO mode alternatively. A new coding scheme will be introduced in chapter 5 to improve the performance of Relay-SISO system. We present also the Relay-MIMO system in chapter 6 as a cooperative solution for the new arriving Wi-Fi standard 802.11n.

CHAPTER 1

COOPERATIVE DIVERSITY

It is well-known that the MIMO technology provides a spatial diversity to protect the wireless communication from fading phenomenon. However in the real environment, the MIMO technology has always the physical constraints for example the number of antennas is often limited by the equipment's size. In [6, 7], the authors proposed in the cellular system a new diversity strategy which requires the cooperation of different terminals (users). Unlike the classic communication system which involves only the transmitter and the receiver, the cooperative communication system includes one or several *relay* terminals which can provide extra spatial diversity gain. In this chapter, we will present the cooperative system, especially the Non-orthogonal AF system as well as the proposed cooperative space-time code.

1.1 Cooperative system

The cooperative system can be found in the early works of Cover and Gamal in [8] where the capacity for the cooperative system using relay protocol is studied. According to the processing at the relay's side, the cooperation system can be generally classified into 2 categories [9, 10]: the amplify-and-forward (AF) and the decode-and-forward (DF). In DF mode the relay terminal decodes the received information and resends it to the destination terminal while in AF mode the relay terminal forwards directly the received signal. Due to the decoding latency at the relay's side, the application of DF cooperation is more difficult than AF cooperation in the real-time environment ¹.

For the AF cooperation system, the half-duplex constraint is always imposed. In [11], Laneman et al. propose an AF scheme which consists of 2 periods: in the first period the source transmits the signals to both the relays and the destination; in the second period, the relays send the received signals to the destination. The destination will combine the received signals to decode the information. In this AF protocol, the transmissions of the source and the relays are configured in orthogonal subchannels and there is a loss of spectral efficiency because of the retransmission. This AF protocol is named Orthogo-

¹The decoding processing will produce certainly the delay at the relay terminal's side that we should consider the time synchronization issue for the source, relay and destination terminals. The cooperation with quick amplify-and-forward strategy is more adapted to solve the synchronization problem in the Wi-Fi network

nal AF (OAF) protocol. In [3], Azarian et al. propose a new AF scheme where in the second period the source and the relays transmit simultaneously. Since this transmission is no longer orthogonal, it is named Non-orthogonal AF (NAF) protocol. This scheme makes full use of the radio resource and it outperforms the original OAF scheme in terms of Diversity-Multiplexing gain Tradeoff (DMT)[5]. The single-relay single-antenna NAF scheme is studied in [3] and the MIMO NAF is discussed in [4], where Yang et al. generalize the DMT analyse and propose the family of Space-Time Block Code (STBC) which can achieve the optimal DMT for different NAF cooperation systems.

1.2 Cooperative diversity and STBC

The single-relay single antenna NAF cooperative system consists of 3 terminals: the source terminal, the destination terminal and the relay terminal. For the convenience of notation, the terminals will be denoted by: s for the source terminal, d for the destination terminal and r for the relay terminal. Let us consider the one antenna case where h_{ij} represents the channel coefficient from terminal i to terminal j . We suppose the quasi-static Rayleigh channel that the channel coefficients are independent Gaussian variables that $h_{ij} \sim \mathcal{CN}(0, \sigma_{ij}^2)$.

We divide the cooperation into 2 timeslots. Let $x_t^{(i)}$ denote the transmit symbol by terminal i and $y_t^{(j)}$ denote the received symbol by terminal j in timeslot t . Let w_t denote the Additive White Gaussian Noise (AWGN) term at the destination and v_t denote the AWGN term at the relay that we have $v_t \sim \mathcal{CN}(0, N_0)$, $w_t \sim \mathcal{CN}(0, N_0)$.

In the first timeslot, the source transmits with power π_{s1} . The received signals at relay terminal and destination terminal are given by:

$$\begin{cases} y_1^{(r)} = \sqrt{\pi_{s1}}h_{sr}x_1^{(s)} + v_1 \\ y_1^{(d)} = \sqrt{\pi_{s1}}h_{sd}x_1^{(s)} + w_1 \end{cases} \quad (1.1)$$

In the second timeslot, the relay retransmits the received symbol $y_1^{(r)}$ which is normalized by a scalar factor b :

$$x_2^{(r)} = by_1^{(r)} \quad (1.2)$$

The normalization factor b is calculated in order to meet the power constraint $E\{|x_2^{(r)}|^2\} \leq 1$ that this parameter is given by:

$$b = \sqrt{\frac{1}{\pi_{s1}|h_{sr}|^2 + N_0}} \quad (1.3)$$

The source terminal transmits $x_2^{(s)}$ with power π_{s2} while the relay terminal transmits $x_2^{(r)}$ with power π_{r2} . The destination receives in the second timeslot:

$$y_2^{(d)} = \sqrt{\pi_{s2}}h_{sd}x_2^{(s)} + \sqrt{\pi_{r2}}h_{rd}x_2^{(r)} + w_2 \quad (1.4)$$

In order to keep the same transmitting power, the power coefficients are subjected to:

$$\pi_{s1} + \pi_{s2} + \pi_{r2} = 2 \quad (1.5)$$

This transmission can be described by:

$$\begin{bmatrix} y_1^{(d)} \\ y_2^{(d)} \end{bmatrix} = \begin{bmatrix} \sqrt{\pi_{s1}}h_{sd} & 0 \\ \sqrt{\pi_{s1}\pi_{r2}bh_{rd}h_{sr}} & \sqrt{\pi_{s2}}h_{sd} \end{bmatrix} \begin{bmatrix} x_1^{(s)} \\ x_2^{(s)} \end{bmatrix} + \begin{bmatrix} 0 \\ \sqrt{\pi_{r2}bh_{rd}v_1} \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad (1.6)$$

For the power allocation issue, we use the configuration $\pi_{s1} = 1$, $\pi_{s2} = 0.5$ and $\pi_{r2} = 0.5$ that the power for each timeslot is normalized to 1.

The Golden code [12] can be applied in single-relay single-antenna scenario and it is proven that the Golden code achieves the optimal DMT [4]. Let $\mathbf{X} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}$ denote the codeword. The cooperation is realized as follows: in the first 2 timeslots, the source terminal send x_{11} and x_{12} then x_{21} and x_{22} in the following 2 timeslots.

From 1.6, the system with Golden code can be described as:

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{V} + \mathbf{W} \quad (1.7)$$

where,

$$\left\{ \begin{array}{l} \mathbf{Y} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} \\ \mathbf{H} = \begin{bmatrix} \sqrt{\pi_{s1}}h_{sd} & 0 \\ \sqrt{\pi_{s1}\pi_{r2}bh_{rd}h_{sr}} & \sqrt{\pi_{s2}}h_{sd} \end{bmatrix} \\ \mathbf{X} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} \\ \mathbf{V} = \begin{bmatrix} 0 & 0 \\ \sqrt{\pi_{r2}bh_{rd}v_{21}} & \sqrt{\pi_{r2}bh_{rd}v_{22}} \end{bmatrix} \\ \mathbf{W} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \end{array} \right. \quad (1.8)$$

The NAF cooperation system turns into a MIMO-like system and the MIMO decoding technique can be applied on this cooperation system.

We notice that the noise power in first period is not the same as the one in the second period. Thus, it is necessary to perform the noise normalization before decoding \mathbf{X} . This normalization is achieved by left-multiplying a diagonal matrix $\begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix}$:

$$\begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} \mathbf{Y} = \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} \mathbf{H}\mathbf{X} + \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} (\mathbf{V} + \mathbf{W}) \quad (1.9)$$

where,

$$\rho = \sqrt{\frac{1}{1 + \pi_{r2}b^2|h_{rd}|^2}} \quad (1.10)$$

Let us define:

$$\left\{ \begin{array}{l} \tilde{\mathbf{Y}} = \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} \mathbf{Y} \\ \tilde{\mathbf{H}} = \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} \mathbf{H} \\ \tilde{\mathbf{W}} = \begin{bmatrix} 1 & 0 \\ 0 & \rho \end{bmatrix} (\mathbf{V} + \mathbf{W}) \end{array} \right. \quad (1.11)$$

The noise terms $\widetilde{\mathbf{W}}$ are i.i.d AWGN that $\text{E}\{\widetilde{\mathbf{W}}\widetilde{\mathbf{W}}^H\} = 2N_0\mathbf{I}$.

The Signal-to-Noise Ratio (SNR) is defined by:

$$\text{SNR} = \frac{\sigma_{sd}^2}{N_0} \quad (1.12)$$

Moreover, the relay terminal may have a geographical advantage that the quality of link $s \rightarrow r$ or $r \rightarrow d$ is better than the direct link $s \rightarrow d$. We define G_{sr} the gain of link $s \rightarrow r$ to link $s \rightarrow d$ in SNR and G_{rd} the one of link $r \rightarrow d$, respectively, that:

$$G_{sr} = \frac{\sigma_{sr}^2}{\sigma_{sd}^2}, G_{rd} = \frac{\sigma_{rd}^2}{\sigma_{sd}^2} \quad (1.13)$$

For the single-relay multiple antennas NAF cooperation system, let us suppose M antennas at the source, N antennas at the destination and K antennas at the relay. Like the single-relay single-antenna system, the cooperation is accomplished in 2 periods and we suppose that each period contains T timeslots.

Let $x_{p,m,t}$ denote the transmitted signal on antenna m and $y_{p,n,t}$ is the received signal on antenna n in period p at timeslot t , respectively. $\mathbf{X}_p^{(i)} = [x_{p,m,t}]$ denotes the signal transmitted by terminal i and $\mathbf{Y}_p^{(j)} = [y_{p,n,t}]$ denotes the signal received by terminal j in period p . Let $\mathbf{H}_{ij} = [h_{n,m}^{(ij)}]$ denote the coefficient matrix of channel $j \rightarrow i$ where $h_{n,m}^{(ij)}$ is the channel response from transmit antenna m to receive antenna n . The i.i.d AWGN terms are denoted by $\mathbf{W}_p = [w_{p,n,t}]$ at destination and by $\mathbf{V}_p = [v_{p,k,t}]$ at the relay. We have $w_{p,n,t} \sim \mathcal{CN}(0, N_0)$, $v_{p,k,t} \sim \mathcal{CN}(0, N_0)$ and $\text{E}\{\mathbf{W}_p\mathbf{W}_p^H\} = \text{E}\{\mathbf{V}_p\mathbf{V}_p^H\} = N_0T\mathbf{I}$. The source transmits with power π_{s1} in the first period and the signals received at the relay and at the destination are given by:

$$\begin{cases} \mathbf{Y}_1^{(d)} = \sqrt{\pi_{s1}}\mathbf{H}_{sd}\mathbf{X}_1^{(s)} + \mathbf{W}_1 \\ \mathbf{Y}_1^{(r)} = \sqrt{\pi_{s1}}\mathbf{H}_{sr}\mathbf{X}_1^{(s)} + \mathbf{V}_1 \end{cases} \quad (1.14)$$

In the second period, the relay amplifies the received signals by multiplying the normalization matrix \mathbf{B} :

$$\mathbf{X}_2^{(r)} = \mathbf{B}\mathbf{Y}_1^{(r)} \quad (1.15)$$

The relay transmits $\mathbf{X}_2^{(r)}$ with power π_{r2} while the source transmits the second part of signal with power π_{s2} . At the destination, the received signal is expressed by:

$$\mathbf{Y}_2^{(d)} = \sqrt{\pi_{s2}}\mathbf{H}_{sd}\mathbf{X}_2^{(s)} + \sqrt{\pi_{r2}}\mathbf{H}_{rd}\mathbf{X}_2^{(r)} + \mathbf{W}_2 \quad (1.16)$$

The matrix \mathbf{B} is a $K \times K$ matrix which is calculated to meet the transmit power constraint:

$$P_{\text{relay}} = \frac{1}{T}\text{Tr}\left(\text{E}\left\{\mathbf{X}_2^{(r)}\left(\mathbf{X}_2^{(r)}\right)^H\right\}\right) \leq K \quad (1.17)$$

Let us define:

$$\begin{cases} \Sigma_{\mathbf{X}_1^{(r)}} = \text{E}\left\{\mathbf{X}_1^{(r)}\left(\mathbf{X}_1^{(r)}\right)^H\right\} \\ \Sigma_{\mathbf{V}_1} = \text{E}\left\{\mathbf{V}_1\mathbf{V}_1^H\right\} \end{cases} \quad (1.18)$$

By (1.15), we have:

$$\text{Tr} \left(\mathbf{B} \left(\pi_{s1} \mathbf{H}_{sr} \boldsymbol{\Sigma}_{\mathbf{X}_1^{(s)}} \mathbf{H}_{sr}^H + \boldsymbol{\Sigma}_{\mathbf{V}} \right) \mathbf{B}^H \right) \leq K \quad (1.19)$$

The transmit power condition can alternatively be given by:

$$\text{Tr} \left(\left(\pi_{s1} \mathbf{H}_{sr} \boldsymbol{\Sigma}_{\mathbf{X}_1^{(s)}} \mathbf{H}_{sr}^H + \boldsymbol{\Sigma}_{\mathbf{V}} \right) \mathbf{B}^H \mathbf{B} \right) \leq K \quad (1.20)$$

By combining (1.14) (1.15) and (1.16), the single-relay multiple-antennas cooperation system is described by:

$$\begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \end{bmatrix} = \begin{bmatrix} \sqrt{\pi_{s1}} \mathbf{H}_{sd} & \mathbf{0} \\ \sqrt{\pi_{r2} \pi_{s1}} \mathbf{H}_{rd} \mathbf{B} \mathbf{H}_{sr} & \sqrt{\pi_{s2}} \mathbf{H}_{sd} \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \sqrt{\pi_{r2}} \mathbf{H}_{rd} \mathbf{B} \mathbf{V} \end{bmatrix} + \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{bmatrix} \quad (1.21)$$

We can simplify the above expression by:

$$\mathbf{Y} = \mathbf{H} \mathbf{X} + \mathbf{U} + \mathbf{W} \quad (1.22)$$

where

$$\left\{ \begin{array}{l} \mathbf{Y} = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \end{bmatrix} \\ \mathbf{H} = \begin{bmatrix} \sqrt{\pi_{s1}} \mathbf{H}_{sd} & \mathbf{0} \\ \sqrt{\pi_{r2} \pi_{s1}} \mathbf{H}_{rd} \mathbf{B} \mathbf{H}_{sr} & \sqrt{\pi_{s2}} \mathbf{H}_{sd} \end{bmatrix} \\ \mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} \\ \mathbf{U} = \begin{bmatrix} \mathbf{0} \\ \sqrt{\pi_{r2}} \mathbf{H}_{rd} \mathbf{B} \mathbf{V} \end{bmatrix} \\ \mathbf{W} = \begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{bmatrix} \end{array} \right. \quad (1.23)$$

In order to achieve the AWGN condition, we normalize the system by left-multiplying $\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma} \end{bmatrix}$:

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma} \end{bmatrix} \mathbf{Y} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma} \end{bmatrix} \mathbf{H} \mathbf{X} + \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Gamma} \end{bmatrix} (\mathbf{U} + \mathbf{W}) \quad (1.24)$$

For the noise terms, the AWGN condition is equivalent to:

$$\mathbb{E} \left\{ \boldsymbol{\Gamma} (\sqrt{\pi_{r2}} \mathbf{H}_{rd} \mathbf{B} \mathbf{V} + \mathbf{W}_2) (\sqrt{\pi_{r2}} \mathbf{H}_{rd} \mathbf{B} \mathbf{V} + \mathbf{W}_2)^H \boldsymbol{\Gamma}^H \right\} = \mathbb{E} \left\{ \mathbf{W}_1 \mathbf{W}_1^H \right\} \quad (1.25)$$

which gives:

$$\boldsymbol{\Gamma} \left(\pi_{r2} \mathbf{H}_{rd} \mathbf{B} \mathbf{B}^H \mathbf{H}_{rd}^H + \mathbf{I} \right) \boldsymbol{\Gamma}^H = \mathbf{I} \quad (1.26)$$

Let us define:

$$\left\{ \begin{array}{l} \boldsymbol{\Sigma} = (\pi_{r2} \mathbf{P} \mathbf{P}^H + \mathbf{I})^{-1} \\ \mathbf{P} = \mathbf{H}_{rd} \mathbf{B} \end{array} \right. \quad (1.27)$$

The matrix $\boldsymbol{\Gamma}$ is then obtained by performing Cholesky decomposition on $\boldsymbol{\Sigma}$:

$$\boldsymbol{\Gamma}^H \boldsymbol{\Gamma} = \boldsymbol{\Sigma} \quad (1.28)$$

The 4×4 Perfect code [13] is proposed for the single-relay two-antenna NAF scenario, in [4] where we configure $M = N = K = 2$ and $T = 4$. Let \mathbf{X} denote a codeword that:

$$\mathbf{X} = \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \underline{x}_3 \\ \underline{x}_4 \end{bmatrix} \quad (1.29)$$

where \underline{x}_i is the i^{th} row.

For the cooperation, the source terminal transmits in the first period $\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \end{bmatrix}$ and in the second period $\begin{bmatrix} \underline{x}_3 \\ \underline{x}_4 \end{bmatrix}$ in the second period.

The 4×4 Perfect code and the Golden code belong to a full-diversity full-rate STBC family with non-vanishing determinant and this STBC family will be presented in section 1.3.

1.3 Cyclic division algebra based STBC

The perfect codes family, including the Golden code and the 4×4 Perfect code, is a type of STBC derived from the Cyclic Division Algebra (CDA). Let a CDA be denoted by $\mathcal{A} = \mathcal{A}(\mathbb{K}/\mathbb{F}, \sigma, \gamma)$, where $\mathbb{F} = \mathbb{Q}(i)$ is the center and \mathbb{K} is the algebraic extension over \mathbb{F} of degree n with σ the generator of its Galois group $\text{Gal}(\mathbb{K}/\mathbb{F})$.

This CDA is generated by:

$$\mathcal{A} = \left\{ \sum_{i=0}^{n-1} z_i \cdot e^i, z_i \in \mathbb{K} \right\} \quad (1.30)$$

where $z \cdot e = e \cdot \sigma$, $e^n = \gamma$, $\gamma \in \mathbb{F}$ and γ is not a norm in \mathbb{K} .

For the Golden code, the \mathbb{K} is selected as $\mathbb{K} = \mathbb{F}(\theta) = \mathbb{Q}(i, \theta)$ that:

$$\mathbb{K} = \{s_1 + s_2\theta, \quad s_1, s_2 \in \mathbb{Q}(i)\} \quad (1.31)$$

with

$$\begin{cases} \theta = \frac{1+\sqrt{5}}{2} \\ \bar{\theta} = \sigma(\theta) = \frac{1-\sqrt{5}}{2} \end{cases} \quad (1.32)$$

The construction of \mathcal{A} can be described by:

$$\mathcal{A} = \{x + y \cdot e, \quad x, y \in \mathbb{K}/\mathbb{F}\} \quad (1.33)$$

There exists a matrix representation of the element in \mathcal{A} . Let \mathcal{M} denote the mapping from an element in \mathcal{A} to its matrix representation. The element e is mapped to:

$$\mathcal{M}(e) = \begin{bmatrix} 0 & 1 \\ \gamma & 0 \end{bmatrix} \quad (1.34)$$

For $x \in \mathbb{K}$, its matrix representation is given by:

$$\mathcal{M}(x) = \text{diag}(x, \sigma(x)) = \begin{bmatrix} x & 0 \\ 0 & \sigma(x) \end{bmatrix} \quad (1.35)$$

where $[x, \sigma(x)]$ is the canonic embedding of $x \in \mathbb{K}$.

The element $(x + y \cdot e) \in \mathcal{A}$ can be expressed by:

$$\mathcal{M}(x + y \cdot e) = \mathcal{M}(x) + \mathcal{M}(y)\mathcal{M}(e) = \begin{bmatrix} x & y \\ \gamma\sigma(y) & \sigma(x) \end{bmatrix} \quad (1.36)$$

To give the construction of Golden code, we start from the integer ring of \mathbb{K} , denoted by $\mathcal{O}_{\mathbb{K}}$.

The integer ring $\mathcal{O}_{\mathbb{K}}$ is generated by:

$$\mathcal{O}_{\mathbb{K}} = \{s_1 b_1 + s_2 b_2, \quad s_1, s_2 \in \mathbb{Z}(i)\} \quad (1.37)$$

with $b_1 = 1, b_2 = \theta$ are the integer basis of \mathbb{K} .

This interger ring is isomorphic to the lattice whose basis are generated by canonical embedding of the integer basis of \mathbb{K}/\mathbb{F} .

Let $\Lambda(\mathcal{O}_{\mathbb{K}})$ denote this lattice then its generator is given by:

$$\Phi_{\Lambda(\mathcal{O}_{\mathbb{K}})} = \begin{bmatrix} 1 & \theta \\ 1 & \bar{\theta} \end{bmatrix} \quad (1.38)$$

However, the basis are not orthonormal which means the lattice $\Lambda(\mathcal{O}_{\mathbb{K}})$ is not a rotated version of $\mathbb{Z}^2(i)$. It is worthy to notice that a lattice is a rotation version of $\mathbb{Z}^n(i)$ implies this lattice keeps the same performance as $\mathbb{Z}^n(i)$ in AWGN channel. This is a necessary condition for STBC code design. For this purpose, we take the ideal $\mathcal{I} = \mathcal{I}(\alpha)$ in $\mathcal{O}_{\mathbb{K}}$ where $\alpha = 1 + i + i\theta^2$:

$$\mathcal{I}(\alpha) = \alpha \mathcal{O}_{\mathbb{K}} \quad (1.39)$$

The generator of the lattice $\Lambda(\mathcal{I})$ is:

$$\Phi_{\Lambda(\mathcal{I})} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha & \alpha\theta \\ \bar{\alpha} & \bar{\alpha}\bar{\theta} \end{bmatrix} \quad (1.40)$$

where the factor $\sqrt{5}$ is for normalization. We can verify that the generated lattice is a rotated version of $\mathbb{Z}^2(i)$ since $\Phi_{\Lambda(\mathcal{I})}$ is unitary.

The Golden code is generated by taking $x, y \in \alpha \mathcal{O}_{\mathbb{K}}$ in (1.33):

$$\mathcal{G} = \left\{ x + y \cdot e, \quad x, y \in \alpha \mathcal{O}_{\mathbb{K}} \right\} \quad (1.41)$$

By using the matrix representation, the Golden code can be described as:

$$\mathcal{G} = \left\{ \mathbf{X} = \text{diag}\left(\Phi_{\Lambda(\mathcal{I})}\bar{x}\right) + \text{diag}\left(\Phi_{\Lambda(\mathcal{I})}\bar{y}\right)\mathcal{M}(e), \quad \bar{x}, \bar{y} \in \mathbb{Z}^2(i) \right\} \quad (1.42)$$

² $\mathcal{I}(\alpha)$ denotes the principal ideal generated by α .

From (1.40), the Golden code is given by:

$$\mathcal{G} = \left\{ \mathbf{X} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha(s_1 + \theta s_2) & \alpha(s_3 + \theta s_4) \\ \gamma \bar{\alpha}(s_3 + \bar{\theta} s_4) & \bar{\alpha}(s_1 + \bar{\theta} s_2) \end{bmatrix}, \quad s_i \in \mathbb{Z}(i) \right\} \quad (1.43)$$

The element γ is selected as $\gamma = i \in \mathcal{O}_{\mathbb{K}}$ which is not a norm of any element in $\mathcal{O}_{\mathbb{K}}$. The condition $|\gamma| = 1$ is necessary such that we transmit with the same power on each antenna for a good spectral efficiency.

Let x_{ij} denote the i^{th} row j^{th} column entry of a codeword \mathbf{X} then the coding procedure is:

$$\begin{bmatrix} x_{11} \\ x_{12} \\ x_{21} \\ x_{22} \end{bmatrix} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha & \alpha\theta & 0 & 0 \\ 0 & 0 & \alpha & \alpha\theta \\ 0 & 0 & \gamma\bar{\alpha} & \gamma\bar{\alpha}\bar{\theta} \\ \bar{\alpha} & \bar{\alpha}\bar{\theta} & 0 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} \quad (1.44)$$

The Golden code's generation matrix $\mathbf{M}_{\mathcal{G}}$ is given by:

$$\mathbf{M}_{\mathcal{G}} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha & \alpha\theta & 0 & 0 \\ 0 & 0 & \alpha & \alpha\theta \\ 0 & 0 & \gamma\bar{\alpha} & \gamma\bar{\alpha}\bar{\theta} \\ \bar{\alpha} & \bar{\alpha}\bar{\theta} & 0 & 0 \end{bmatrix} \quad (1.45)$$

For the 4×4 Perfect code, we keep $\mathbb{Q}(i)$ as the center \mathbb{F} and \mathbb{K} is an algebraic extension of degree 4 over \mathbb{F} that:

$$\mathbb{K} = \mathbb{F}(\theta) = \mathbb{Q}(i, \theta) \quad (1.46)$$

Here, θ is a 4-degree primary element over $\mathbb{Q}(i)$:

$$\theta = \xi_{15} + \xi_{15}^{-1} = 2 \cos\left(\frac{2\pi}{15}\right) \quad (1.47)$$

where $\xi = e^{i\frac{2\pi}{15}}$ with the minimal polynomial $\mu(x) = x^4 - x^3 - 4x^2 + 4x + 1$.

Let us write \mathbb{K} by:

$$\mathbb{K} = \{s_0 + s_1\theta + s_2\theta^2 + s_3\theta^3, s_i \in \mathbb{F}\} \quad (1.48)$$

with its Galois group $\text{Gal}(\mathbb{K}/\mathbb{F}) = \{1, \sigma, \sigma^2, \sigma^3\}$:

$$\begin{cases} \sigma : \xi_{15} + \xi_{15}^{-1} \mapsto \xi_{15}^2 + \xi_{15}^{-2} \\ \sigma^2 : \xi_{15} + \xi_{15}^{-1} \mapsto \xi_{15}^4 + \xi_{15}^{-4} \\ \sigma^3 : \xi_{15} + \xi_{15}^{-1} \mapsto \xi_{15}^8 + \xi_{15}^{-8} \end{cases} \quad (1.49)$$

\mathcal{A} is generated by:

$$\mathcal{A}(\mathbb{K}/\mathbb{F}, \sigma, \gamma) = \left\{ x_0 + x_1 \cdot e + x_2 \cdot e^2 + x_3 \cdot e^3, x_i \in \mathbb{K}/\mathbb{F} \right\} \quad (1.50)$$

with $e^4 = \gamma = i$.

The matrix representation mapping \mathcal{M} is given by:

$$\begin{cases} \mathcal{M}(x) = \text{diag}\left(x, \sigma(x), \sigma^2(x), \sigma^3(x)\right), x \in \mathbb{K} \\ \mathcal{M}(e) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \gamma & 0 & 0 & 0 \end{bmatrix} \end{cases} \quad (1.51)$$

where $[x, \sigma(x), \sigma^2(x), \sigma^3(x)]$ is the canonical embedding of $x \in \mathbb{K}$.

Let $X = x_0 + x_1 \cdot e + x_2 \cdot e^2 + x_3 \cdot e^3$, $x_i \in \mathbb{K}$ denote an element in $\mathcal{A}(\mathbb{K}/\mathbb{F}, \sigma, \gamma)$. Its matrix representation is given by:

$$\begin{aligned} \mathcal{M}(X) &= \mathcal{M}(x_0) + \mathcal{M}(x_1 \cdot e) + \mathcal{M}(x_2 \cdot e^2) + \mathcal{M}(x_3 \cdot e^3) \\ &= \begin{bmatrix} x_0 & x_1 & x_2 & x_3 \\ \gamma\sigma(x_3) & \sigma(x_0) & \sigma(x_1) & \sigma(x_2) \\ \gamma\sigma^2(x_2) & \gamma\sigma^2(x_3) & \sigma^2(x_0) & \sigma^2(x_1) \\ \gamma\sigma^3(x_1) & \gamma\sigma^3(x_2) & \gamma\sigma^3(x_3) & \gamma\sigma^3(x_0) \end{bmatrix} \end{aligned} \quad (1.52)$$

Like the Golden code, the 4×4 Perfect code is constructed by taking the elements x_i in an ideal of $\mathcal{O}_{\mathbb{K}}$ which is a rotated version $\mathbb{Z}^4(i)$. This ideal $\mathcal{I}_{\mathbb{K}}$ is constructed by:

$$\mathcal{I}_{\mathbb{K}} = \{s_0 b_0 + s_1 b_1 + s_2 b_2 + s_3 b_3, s_i \in \mathbb{Z}(i)\} \quad (1.53)$$

where $[b_0, b_1, b_2, b_3]$ are the basis:

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} (1 - 3i) + i\theta^2 \\ (1 - 3i)\theta + i\theta^3 \\ -i + (-3 + 4i)\theta + (1 - i)\theta^3 \\ (-1 + i) - 3\theta + \theta^2 + \theta^3 \end{bmatrix} \quad (1.54)$$

By using the canonical embedding of the basis, the generator of the lattice $\Lambda(\mathcal{I}_{\mathbb{K}})$ is given by:

$$\Phi_{\Lambda(\mathcal{I})} = \frac{1}{\sqrt{15}} \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \\ \sigma(b_0) & \sigma(b_1) & \sigma(b_2) & \sigma(b_3) \\ \sigma^2(b_0) & \sigma^2(b_1) & \sigma^2(b_2) & \sigma^2(b_3) \\ \sigma^3(b_0) & \sigma^3(b_1) & \sigma^3(b_2) & \sigma^3(b_3) \end{bmatrix} \quad (1.55)$$

where $\sqrt{15}$ is a normalization factor and $\Phi_{\Lambda(\mathcal{I})}$ is a unitary matrix. The lattice $\Lambda(\mathcal{I})$ is then a rotated version of $\mathbb{Z}^4(i)$.

The 4×4 Perfect code is given by:

$$\mathcal{X} = \left\{ \mathbf{X} = \sum_{k=0}^3 \text{diag}(\Phi_{\Lambda(\mathcal{I})} \bar{s}_k) \mathcal{M}(e^k), \bar{s}_k \in \mathbb{Z}^4(i) \right\} \quad (1.56)$$

A codeword is generated from $s_{j,k} \in \mathbb{Z}(i)$ by:

$$\mathbf{X} = \frac{1}{\sqrt{15}} \begin{bmatrix} \sum_{k=0}^3 s_{0,k} b_k & \sum_{k=0}^3 s_{1,k} b_k & \sum_{k=0}^3 s_{2,k} b_k & \sum_{k=0}^3 s_{3,k} b_k \\ \gamma \sum_{k=0}^3 s_{3,k} \sigma(b_k) & \sum_{k=0}^3 s_{0,k} \sigma(b_k) & \sum_{k=0}^3 s_{1,k} \sigma(b_k) & \sum_{k=0}^3 s_{2,k} \sigma(b_k) \\ \gamma \sum_{k=0}^3 s_{2,k} \sigma^2(b_k) & \gamma \sum_{k=0}^3 s_{3,k} \sigma^2(b_k) & \sum_{k=0}^3 s_{0,k} \sigma^2(b_k) & \sum_{k=0}^3 s_{1,k} \sigma^2(b_k) \\ \gamma \sum_{k=0}^3 s_{1,k} \sigma^3(b_k) & \gamma \sum_{k=0}^3 s_{2,k} \sigma^3(b_k) & \gamma \sum_{k=0}^3 s_{3,k} \sigma^3(b_k) & \sum_{k=0}^3 s_{0,k} \sigma^3(b_k) \end{bmatrix} \quad (1.57)$$

Let \bar{x}_i denote the i^{th} column of \mathbf{X} , \bar{s}_j denote $[s_{j,0}, s_{j,1}, s_{j,2}, s_{j,3}]^T$ and \underline{b} denote $[b_0, b_1, b_2, b_3]$. By writing the codeword in the vectorial form, (1.57) is equivalent to:

$$\begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \bar{x}_4 \end{bmatrix} = \mathbf{M}_\mathcal{X} \begin{bmatrix} \bar{s}_0 \\ \bar{s}_1 \\ \bar{s}_2 \\ \bar{s}_3 \end{bmatrix} \quad (1.58)$$

where $\mathbf{M}_\mathcal{X}$ is the generation matrix of the 4×4 Perfect code:

$$\mathbf{M}_\mathcal{X} = \frac{1}{\sqrt{15}} \begin{bmatrix} \underline{b} & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma(\underline{b}) \\ \underline{0} & \underline{0} & \gamma\sigma^2(\underline{b}) & \underline{0} \\ \underline{0} & \gamma\sigma^3(\underline{b}) & \underline{0} & \underline{0} \\ \underline{0} & \underline{b} & \underline{0} & \underline{0} \\ \sigma(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma^2(\underline{b}) \\ \underline{0} & \underline{0} & \gamma\sigma^3(\underline{b}) & \underline{0} \\ \underline{0} & \underline{0} & \underline{b} & \underline{0} \\ \underline{0} & \sigma(\underline{b}) & \underline{0} & \underline{0} \\ \sigma^2(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma^3(\underline{b}) \\ \underline{0} & \underline{0} & \underline{0} & \underline{b} \\ \underline{0} & \underline{0} & \sigma(\underline{b}) & \underline{0} \\ \underline{0} & \sigma^2(\underline{b}) & \underline{0} & \underline{0} \\ \sigma^3(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \end{bmatrix} \quad (1.59)$$

CHAPTER 2

MIMO DECODER

As discussed in chapter 1, the NAF cooperation system can be given in a MIMO representation. To decode this cooperation system, we can apply the known decoding algorithms in the MIMO system. In context of MIMO system, we will some known decoder algorithms which concern the channel matrix processing and the lattice searching method.

The MIMO decoder is an important factor for the MIMO system or the virtual-MIMO system like NAF cooperation system. Our goal in this chapter is to find an “implementable” decoder algorithm that provides good decoding performance with acceptable complexity.

2.1 MIMO system description

We consider a MIMO system with M transmit antennas and N receive antennas. Let X denote the sent codeword for duration T and let Y denote the received signal. The system is represented by:

$$\mathbf{Y}_{N \times T} = \mathbf{H}_{N \times M} \mathbf{X}_{M \times T} + \mathbf{W}_{N \times T} \quad (2.1)$$

where \mathbf{H} is the channel matrix, $E\{\mathbf{X}^H \mathbf{X}\} = \mathbf{I}$ and \mathbf{W} denotes the independent and identically-distributed (i.i.d) AWGN terms, such that $w_{ij} \sim \mathcal{CN}(0, N_0)$.

We suppose the Channel State Information (CSI) is perfectly known at the receiver side and the Maximum Likelihood (ML) decoding aims to find the $\hat{\mathbf{X}}$ which satisfies:

$$\hat{\mathbf{X}} = \arg \max_{\mathbf{X} \in \mathcal{X}} \Pr\{\mathbf{X} | \mathbf{Y}, \mathbf{H}\} \quad (2.2)$$

where \mathcal{X} denotes the set of codewords.

Since the noise is AWGN, the ML decoding is equivalent to find the codeword $\hat{\mathbf{X}}$ that:

$$\hat{\mathbf{X}} = \arg \min_{\mathbf{X} \in \mathcal{X}} \|\mathbf{Y} - \mathbf{H}\mathbf{X}\|_F^2 \quad (2.3)$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm.

Let ω denote the transformation from a matrix to its vectorial form by writing the columns vertically such that:

$$\begin{aligned} \bar{\mathbf{y}} &= \omega(\mathbf{Y}) \\ \omega : \mathbf{Y} = [\bar{y}_1 \quad \bar{y}_2 \quad \cdots \quad \bar{y}_T] &\rightarrow \bar{\mathbf{y}} = \begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_T \end{bmatrix} \end{aligned} \quad (2.4)$$

Let Ω denote the following matrix transformation:

$$\begin{aligned} \tilde{\mathbf{H}} &= \Omega(\mathbf{H}) \\ \Omega : \mathbf{H} \rightarrow \tilde{\mathbf{H}} &= \left. \begin{bmatrix} \mathbf{H} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{H} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{H} \end{bmatrix} \right\} T \text{ times} \end{aligned} \quad (2.5)$$

The MIMO system can be equivalently expressed by:

$$\bar{\mathbf{y}} = \tilde{\mathbf{H}}\bar{\mathbf{x}} + \bar{\mathbf{w}} \quad (2.6)$$

where $\bar{\mathbf{x}} = \omega(\mathbf{X})$ and $\bar{\mathbf{w}} = \omega(\mathbf{W})$.

The transmitted signal \mathbf{X} can be a STBC codeword. When the STBC is applied, the codewords are linearly generated from the Quadrature Amplitude Modulation (QAM) symbols. Let $\bar{\mathbf{x}}$ denote the transmitted codeword and let $\bar{\mathbf{s}}$ denote the QAM symbols. The coding procedure can be expressed by:

$$\bar{\mathbf{x}} = K_{\text{mod}} \mathbf{M}_{\mathcal{X}}(\bar{\mathbf{s}} + \bar{\mathbf{c}}), \quad s_i \in \mathcal{S} \quad (2.7)$$

where $\mathbf{M}_{\mathcal{X}}$ is the STBC generation matrix as (1.45) and (1.59). \mathcal{S} is the set of possible constellation symbols. K_{mod} is the power normalization factor and $\bar{\mathbf{c}}$ is a constant offset such that $\mathbb{E}\{\bar{\mathbf{s}} + \bar{\mathbf{c}}\} = \bar{\mathbf{0}}$ and $\mathbb{E}\{K_{\text{mod}}^2(\bar{\mathbf{s}} + \bar{\mathbf{c}})(\bar{\mathbf{s}} + \bar{\mathbf{c}})^H\} = \mathbf{I}$.

Table 2.1 shows \mathcal{S} and K_{mod} for different constellations. The offset is configured as $\bar{\mathbf{c}} = 0.5$ for Binary Phase-Shift Keying (BPSK) and $\bar{\mathbf{c}} = 0.5(1 + i)$ for the other constellations, including Quadrature Phase-Shift Keying (QPSK), 16-QAM and 64-QAM.

Table 2.1: Lattice QAM symbols set \mathcal{S}

Constellation	$\Re(\mathcal{S})$	$\Im(\mathcal{S})$	K_{mod}
BPSK	$\{-1, 0\}$	$\{0\}$	2
QPSK	$\{-1, 0\}$	$\{-1, 0\}$	$2/\sqrt{2}$
16-QAM	$\{-2, -1, 0, 1\}$	$\{-2, -1, 0, 1\}$	$2/\sqrt{10}$
64-QAM	$\{-4, -3, -2, -1, 0, 1, 2, 3\}$	$\{-4, -3, -2, -1, 0, 1, 2, 3\}$	$2/\sqrt{42}$

Let the real signal be remarked by the letter r . We can denote the product of the channel matrix and the coding matrix by defining:

$$\mathbf{H}_{\text{eq}} = K_{\text{mod}} \tilde{\mathbf{H}} \mathbf{M}_{\mathcal{X}} \quad (2.8)$$

(2.6) turns to:

$$\bar{y} = \mathbf{H}_{\text{eq}}(\bar{s} + \bar{c}) + \bar{w} \quad (2.9)$$

The system can be expressed in using the real signals:

$$\bar{y}^r = \mathbf{H}_{\text{eq}}^r(\bar{s}^r + \bar{c}^r) + \bar{w}^r \quad (2.10)$$

where

$$\begin{cases} \bar{y}^r = \begin{bmatrix} \Re(\bar{y}) \\ \Im(\bar{y}) \end{bmatrix} \\ \bar{s}^r + \bar{c}^r = \begin{bmatrix} \Re(\bar{s} + \bar{c}) \\ \Im(\bar{s} + \bar{c}) \end{bmatrix} \\ \bar{w}^r = \begin{bmatrix} \Re(\bar{w}) \\ \Im(\bar{w}) \end{bmatrix} \end{cases} \quad (2.11)$$

and

$$\mathbf{H}_{\text{eq}}^r = \begin{bmatrix} \Re(\mathbf{H}_{\text{eq}}) & -\Im(\mathbf{H}_{\text{eq}}) \\ \Im(\mathbf{H}_{\text{eq}}) & \Re(\mathbf{H}_{\text{eq}}) \end{bmatrix} \quad (2.12)$$

The entries of \bar{s}^r are real integers such that $\bar{s}^r \in \mathcal{Z} \subset \mathbb{Z}^{2MT}$, where $\mathcal{Z} = \{\Re^{MT}(\mathcal{S}), \Im^{MT}(\mathcal{S})\}^T$.

Let the metric of \bar{s}^r be denoted by:

$$M(\bar{s}^r) = \left(\bar{y}^r - \mathbf{H}_{\text{eq}}^r(\bar{s}^r + \bar{c}^r) \right)^T \left(\bar{y}^r - \mathbf{H}_{\text{eq}}^r(\bar{s}^r + \bar{c}^r) \right) \quad (2.13)$$

The ML condition in (2.1) is equal to find $\hat{\bar{s}}^r$ that:

$$\hat{\bar{s}}^r = \arg \min_{\bar{s}^r \in \mathcal{Z}} M(\bar{s}^r) \quad (2.14)$$

To decode this system, an ‘‘exhaustive’’ way is to examine all the candidates in the code book. This decoding method can operate when the number of codewords is not too large. For example, in a MIMO system using 2×2 BLAST code with QPSK constellation, there are totally 16 candidates. It is possible to implement a parallel structure for the metric calculation of all the possible constellation points. However, when the dimension is high or the constellation is complex, this decoding method is too expensive to realize.

In the following sections, we will discuss the MIMO decoder problem with the real lattice generated by \mathbf{H}_{eq}^r :

$$\Lambda(\mathbf{H}_{\text{eq}}^r) = \{ \bar{y}^r | \bar{y}^r = \mathbf{H}_{\text{eq}}^r \bar{z}, \bar{z} \in \mathbb{Z}^n \} \quad (2.15)$$

The decoding problem turns to be the well-known Closest Lattice Point Search (CLPS) problem: to find in the $\Lambda(\mathbf{H}_{\text{eq}}^r)$ the nearest point to $\bar{y}^r - \mathbf{H}_{\text{eq}}^r \bar{c}^r$.

2.2 Left processing: MMSE-GDFE

Some matrix processings need to be taken before performing the lattice point searching. For the left-processing, the Minimum Mean Square Error Generalized Decision-Feedback

Equalizer (MMSE-GDFE) [14] is performed to minimize the noise impact and to give an upper triangular structure of the lattice generator matrix.

Let $\mathbf{H}_{\text{ext}} = \begin{bmatrix} \mathbf{H}_{\text{eq}}^r \\ \frac{1}{2}N_0\mathbf{I} \end{bmatrix}$ denote the extended matrix of \mathbf{H}_{eq}^r , where $\frac{1}{2}N_0$ is the real noise power.

We apply QR-decomposition to \mathbf{H}_{ext} that:

$$\mathbf{H}_{\text{ext}} = \mathbf{Q}_{\text{ext}}\mathbf{R} = \begin{bmatrix} \mathbf{Q} \\ \frac{1}{2}N_0\mathbf{R}^{-1} \end{bmatrix} \mathbf{R} \quad (2.16)$$

The matrix \mathbf{Q}_{ext} is orthogonal and \mathbf{R} is an upper triangular matrix with $\mathbf{R}^T\mathbf{R} = (\mathbf{H}_{\text{eq}}^r)^T \mathbf{H}_{\text{eq}}^r + \frac{1}{2}N_0\mathbf{I}$.

By left multiplying $\mathbf{P} = \mathbf{Q}^T$ at both sides of (2.10), we have:

$$\begin{aligned} \bar{y}' &= \mathbf{P}\bar{y}^r \\ &= \mathbf{R}(\bar{s}^r + \bar{c}^r) + \mathbf{P}\bar{w}^r + (\mathbf{PH} - \mathbf{R})(\bar{s}^r + \bar{c}^r) \\ &= \mathbf{R}(\bar{s}^r + \bar{c}^r) + \bar{w}' \end{aligned} \quad (2.17)$$

where $\bar{w}' = \mathbf{P}\bar{w}^r + (\mathbf{PH} - \mathbf{R})(\bar{s}^r + \bar{c}^r)$ is the equivalent noise term.

It is noticeable that the equivalent noise term contains the AWGN noise contribution $\mathbf{P}\bar{w}^r$ and the signal-dependent noise term $(\mathbf{PH} - \mathbf{R})(\bar{s}^r + \bar{c}^r)$.

The noise term $\mathbf{P}\bar{w}^r$ may not be i.i.d for the reason the matrix \mathbf{P} can be non-orthogonal and the signal-dependent term $(\mathbf{PH} - \mathbf{R})(\bar{s}^r + \bar{c}^r)$ is non-Gaussian. However, considering $E\{(\bar{s}^r + \bar{c}^r)(\bar{s}^r + \bar{c}^r)^T\} = \mathbf{K}_{\text{mod}}^{-2}\mathbf{I}$, it can be shown that the equivalent noise term \bar{w}' keeps white that $E\{\bar{w}'\bar{w}'^T\} = N_0'\mathbf{I}$ [15]. This operation will change the noise's AWGN property but the minimum distance criteria is expected to be only slightly sub-optimal [16].

The constant offset \bar{c}^r is removed from \bar{y}' by:

$$\begin{aligned} \bar{y}'' &= \bar{y}' - \mathbf{R}\bar{c}^r \\ &= \mathbf{R}\bar{s}^r + \bar{w}^r \end{aligned} \quad (2.18)$$

For the convenience of notation, in the following sections we use \bar{y} for \bar{y}'' , \bar{s} for \bar{s}^r and \bar{w} for \bar{w}^r .

2.3 Right processing : column permutation and lattice reduction

Let \mathbf{H} denote the real channel matrix. The QR-decomposition on \mathbf{H} may generate an “ill-conditioned” matrix \mathbf{R} whose diagonal entries can be very small. For the MIMO decoders, an ill-conditioned matrix introduces either large complexity or a bad performance. To solve this problem, it is suggested to perform the right-processing on the channel matrix before the lattice searching step.

This processing can be considered to decompose \mathbf{H} as:

$$\mathbf{H} = \mathbf{H}_{\text{red}}\mathbf{U} \quad (2.19)$$

where \mathbf{U} is a unimodular matrix such that $u_{i,j} \in \mathbb{Z}$ and $\mathbf{U}^T\mathbf{U} = \mathbf{I}$.

The matrix \mathbf{U} can be a permutation matrix which gives a decoding schedule. In the other hand, it can be a lattice reduction matrix which simplifies the lattice basis. In general, the lattice reduction matrix gives better performance than the permutation matrix. However, it is worthy to notice that it takes more time to calculate the lattice reduction matrix. Another inconvenience of the lattice reduction matrix is the shaping problem that the constellation set \mathcal{S} are changed for the lattice searching step.

For the permutation matrix, the algorithm Greedy Ordering (GO) proposed in [17, 18] can be applied to find a good decoding order. It is shown that the permutation matrix provides the optimal order for the Decision Feedback Equalizer (DFE) algorithm which will be presented in section 2.4.

Let \bar{h}_i denote the i^{th} column of channel matrix \mathbf{H} that $\mathbf{H} = [\bar{h}_1, \bar{h}_2, \dots, \bar{h}_M]$. For a given column permutation function π , the column-permuted matrix $\mathbf{H}(\pi)$ is defined by:

$$\mathbf{H}(\pi) = [\bar{h}_{\pi(1)}, \bar{h}_{\pi(2)}, \dots, \bar{h}_{\pi(M)}] \quad (2.20)$$

The GO algorithm aims to find a permutation function π such that for the matrix \mathbf{R} obtained by applying QR-decomposition on $\mathbf{H}(\pi)$, $\min_{1 \leq i \leq M} r_{i,i}$ is maximized. This algorithm yields the optimal permutation π in M steps. Let \mathcal{H}_k denote the index set of the not yet chosen columns at step k . For $k = 1, \dots, M$, the permutation function π is given by:

$$\pi(k) = \arg \max_{j \in \mathcal{H}_k} \left\{ \bar{h}_j^T \left[\mathbf{I} - \mathbf{H}_{k,j} \left(\mathbf{H}_{k,j}^T \mathbf{H}_{k,j} \right)^{-1} \mathbf{H}_{k,j}^T \right] \bar{h}_j \right\} \quad (2.21)$$

where $\mathbf{H}_{k,j}$ is a $N \times (k-1)$ matrix formed by the unchosen columns except \bar{h}_j : $\mathbf{H}_{k,j}$ is composed of the columns \bar{h}_i with $i \in \mathcal{H}_k - \{j\}$.

For the lattice reduction problem, we propose the famous Lenstra, Lenstra and Lovasz (LLL) reduction[19]. The LLL reduction algorithm aims to give the ‘‘shortest’’ basis from the initial lattice generate matrix \mathbf{H} . Therefore, the LLL reduction can greatly improve the sub-optimal algorithm or reduce the complexity of the ML decoding algorithm.

2.4 Lattice decoding

In this section, the ML decoding algorithm for lattice searching problem is presented as well as some sub-optimal algorithms. The ML decoding algorithm yields the best performance but it introduces also great complexity. In most cases, it is necessary to find a sub-optimal algorithm to provide a tradeoff between the decoding performance and the complexity.

We discuss two families of the algorithm: one with fixed latency and the other with variant latency. The fixed latency means the algorithm terminates with a predictable number of computations. This property is important for the hardware implementation of

such a decoder. At the other hand, the variant latency algorithms are generally applied in software level.

2.4.1 ML decoding algorithm

Sphere Decoder (SD) [20] and Schnorr-Euchner (SE) decoder [21] are the two ML algorithms most used in lattice decoding. In [22], the authors give a comparison on the complexity of these two algorithms. Briefly speaking, the SE algorithm has a little advantage to the SD algorithm for small amount of antennas and low SNR. We present the SE algorithm in Algorithm 1.

Algorithm 1 Schnorr Euchner (ML)

Let \mathbf{H} be the lattice generator matrix, \bar{y} be the received vector. This algorithm outputs coordinates $\bar{s} \in \mathbb{Z}^n$ of the closest point to \bar{y} in lattice $\Lambda(\mathbf{H}) = \mathbf{H}\mathbb{Z}^n$, such that $L(\bar{y}, \bar{s}, \mathbf{H}) = (\bar{y} - \mathbf{H}\bar{s})^T(\bar{y} - \mathbf{H}\bar{s}) \leq L(\bar{y}, \bar{s}', \mathbf{H}), \forall \bar{s}' \in \mathbb{Z}^n$ and also the value of $L(\bar{y}, \bar{s}, \mathbf{H})$.

```
1.[Prepare]    $\mathbf{H} = \mathbf{QR}$  (apply QR-decomposition to  $\mathbf{H}$ )
               $\bar{y} \leftarrow \mathbf{Q}^T \bar{y}$ 
2.[Start]      $\bar{\text{dist}} \leftarrow \{0, \dots, 0\}$ 
               $k \leftarrow n$ 
               $\text{bestdist} \leftarrow +\infty$ 
3.[Calculate]  $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
               $u_k \leftarrow \left[ \frac{e_k}{r_{k,k}} \right]_{\text{int}}$ 
               $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
               $\text{step}_k \leftarrow \text{sgn}(d)$ 
4.[Compare]    $\text{newdist} \leftarrow \text{dist}_k + d^2$ 
              if  $\text{newdist} > \text{bestdist}$ 
                  if  $k = n$ 
                      output  $\bar{s}$ ,  $\text{bestdist}$ 
                      terminate the algorithm
                  else
                      go to step 5
                  endif
              else
                  if  $k = 1$ 
                       $\bar{s} \leftarrow \bar{u}$ 
                       $\text{bestdist} \leftarrow \text{newdist}$ 
                      go to step 5
                  else
                      go to step 6
                  endif
              endif
5.[Backward]   $k \leftarrow k + 1$ 
               $u_k \leftarrow u_k + \text{step}_k$ 
               $\text{step}_k \leftarrow -\text{step}_k - \text{sgn}(\text{step}_k)$ 
               $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
              go to step 4
6.[Forward]    $k \leftarrow k - 1$ 
               $\text{dist}_k \leftarrow \text{newdist}$ 
              go to step 3
```

where $[x]_{\text{int}}$ returns the closest integer to x .

The [Prepare] step in Algorithm 1 is done in the MMSE-GDFE processing and \mathbf{R} is the

upper triangular defined by:

$$\mathbf{R} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,n} \\ 0 & r_{2,2} & \cdots & r_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{n,n} \end{bmatrix} \quad (2.22)$$

2.4.2 Algorithm Zero-Forcing

Let us review the MIMO system given in (2.10). The simplest way to decode this system is to project directly the received signal $\bar{\mathbf{y}}^r$ in \mathbb{Z}^n by multiplying $(\mathbf{H}_{\text{eq}}^r)^{-1}$:

$$\begin{aligned} \bar{\mathbf{y}}_{\text{proj}} &= (\mathbf{H}_{\text{eq}}^r)^{-1} \bar{\mathbf{y}}^r \\ &= \bar{\mathbf{s}}^r + \bar{\mathbf{c}}^r + (\mathbf{H}_{\text{eq}}^r)^{-1} \bar{\mathbf{w}}^r \end{aligned} \quad (2.23)$$

The detected vector is given by:

$$\hat{\mathbf{s}}^r = [(\mathbf{H}_{\text{eq}}^r)^{-1} \bar{\mathbf{y}}^r - \bar{\mathbf{c}}^r]_{\text{int}} \quad (2.24)$$

This method is called Zero-Forcing (ZF). When the noise power N_0 is available, the ZF algorithm can be extended to Minimum Mean Square Error (MMSE) detection by applying the projection matrix $\mathbf{R}^{-1} \mathbf{Q}^T$ instead of $(\mathbf{H}_{\text{eq}}^r)^{-1}$, where \mathbf{R} and \mathbf{Q} are defined in (2.16).

The ZF or MMSE detection is very simple and we can implement it for low dimension MIMO systems.

2.4.3 Heuristic algorithms: DFE and KSE

For high dimension MIMO systems, frequently we implement the algorithm DFE which operates on the system with an upper triangular channel matrix like (2.18). The DFE algorithm is given in Algorithm 2.

Algorithm 2 Decision-Feedback Equalizer lattice decoder

Let \mathbf{H} be the lattice generator matrix, \bar{y} be the received vector. This algorithm gives the coordinates $\bar{s} \in \mathbb{Z}^n$ of the closest point found in the lattice $\Lambda(\mathbf{H}) = \mathbf{H}\mathbb{Z}^n$ in an heuristic way. Both the lattice point and its Euclidian distance to \bar{y} are given.

```
1.[Prepare]    $\mathbf{H} = \mathbf{QR}$  (apply QR-decomposition to  $\mathbf{H}$ )
               $\bar{y} \leftarrow \mathbf{Q}^T \bar{y}$ 
2.[Start]     $k \leftarrow n$ 
              dist  $\leftarrow 0$ 
3.[Calculate]  $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
               $u_k \leftarrow \left[ \frac{e_k}{r_{k,k}} \right]_{\text{int}}$ 
               $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
4.[Recursion] dist  $\leftarrow \text{dist} + d^2$ 
              if  $k = 1$ 
                   $\bar{s} \leftarrow \bar{u}$ 
                  output  $\bar{s}$ , dist
                  terminate the algorithm
              else
                   $k \leftarrow k - 1$ 
                  go to step 3
              endif
```

This algorithm finds the so-called *Babai* point in n step. The QR-decomposition keeps the nature of i.i.d AWGN¹. The upper triangular channel matrix implies that the detection at step k has no impact on the precedent detection results. At step k , the precedents results u_j for $k < j \leq n$ are supposed to be correctly decoded and the detection result of u_k is independant to the precedent results.

However, the precedents results do have an impact on the current step. When an error occurs at step k , it will engender the error propagation by influencing the results u_j for $1 \leq j < k$. A solution to reduce the risk is to keep more candidates at each step and the generalized algorithm, named K-best Schnorr-Euchner (KSE) decoder, is proposed in Algorithm 3 where K candidates are taken into account at each step.

¹This property changes slightly when MMSE-GDFE is applied.

Algorithm 3 K-Best Schnorr-Euchner algorithm

Let \mathbf{H} denote the lattice generator matrix, \bar{y} be the received vector. Let $\{z_1, z_2, \dots, z_M\}$ denote the set of possible integer points at each real dimension. This algorithm gives the coordinates $\bar{s} \in \mathbb{Z}^n$ of the closest point found in the lattice $\Lambda(\mathbf{H}) = \mathbf{H}\mathbb{Z}^n$ in an heuristic way. Both the lattice point and its Euclidean distance to \bar{y} are given.

- 1.[Initialization] $\mathbf{H} = \mathbf{QR}$ (apply QR-decomposition to \mathbf{H})
 $\bar{y} \leftarrow \mathbf{Q}^T \bar{y}$
 $\{\text{dist}_1, \dots, \text{dist}_K\} \leftarrow \{+\infty, \dots, +\infty\}$
 $\{\bar{u}_1, \dots, \bar{u}_K\} \leftarrow \{\text{NULL}, \dots, \text{NULL}\}$
 $k \leftarrow n$
 $\text{dist}_1 \leftarrow 0$
- 2.[Forward] for $i = 1, \dots, K$
for $m = 1, \dots, M$
 $d \leftarrow y_k - \sum_{j=n+1}^k r_{j,k} u_{i,j} - r_{k,k} z_m$
 $D_{i,m} \leftarrow \text{dist}_k + d^2$
endfor
endfor
- 3.[Selection] Let $\{D_{i_1, m_1}, \dots, D_{i_K, m_K}\}$ denote the K smallest values among $\{D_{i,m}\}$ in increasing order
for $p = 1, \dots, K$
 $\bar{U}_p \leftarrow [\bar{u}_{i_p}, s_{m_p}]$
 $\text{dist}_p \leftarrow D_{i_p, m_p}$
endfor
- 4.[Judgement] if $k = 1$
 $\bar{s} \leftarrow \bar{U}_1$
output \bar{s}, dist_1
terminate algorithm
else
for $i = 1, \dots, K$
 $\bar{u}_i \leftarrow \bar{U}_i$
endfor
 $k \leftarrow k - 1$
go to step 2
endif

At each step, the K memorized nodes have the same degree and these candidates are considered the “best” at each step. Because the recursion is based on these K node, these candidates are the father nodes of the following detected nodes. By keeping more candidates, this processing avoids eventually eliminating the right candidate due to a false detection. Like DFE algorithm, this algorithm operates with a constant decoding complexity and the performance can be improved with larger K .

In addition, the diagonal coefficients of \mathbf{R} are very important to the performance of DFE or KSE algorithms. An efficient way to improve the system performance is to apply the right-processing like GO or LLL reduction.

2.4.4 Sequential decoder based algorithm : Fano decoder and stack decoder

The sequential decoders, like Fano decoder [23] and stack decoder [24], can be generalized to perform the lattice decoding in exploiting the upper triangular form of lattice generate matrix \mathbf{R} [16].

The principle of Fano decoder is to use a stage bias in order to give a “fair” metric for different decoding stages that the Fano metric is “comparable” for different level. The Fano metric is calculated from the accumulated squared Euclidean distance and we associate for each level a bias of the real noise power $N_0/2$:

$$f(\bar{x}_n^k) = \sum_{i=k}^n (y_i - \sum_{j=i}^n r_{i,j}x_j)^2 - (n-k)\frac{N_0}{2} \quad (2.25)$$

Following [16], Fano metric can take the general form:

$$f(\bar{x}_n^k) = \sum_{i=k}^n (y_i - \sum_{j=i}^n r_{i,j}x_j)^2 - b(n-k)\frac{N_0}{2} \quad (2.26)$$

It is notable that the stage bias factor b can be chosen arbitrarily which allows a flexible tradeoff between the performance and the complexity.

Fano lattice decoder algorithm is given in Algorithm 4.

We notice that the discrete metric threshold T is a multiple of the increment Δ so the configuration Δ is also important to the decoder’s performance: a too large value will degrade the decoding performance and a too small one will introduce unnecessary complexity.

The stack decoder described in Algorithm 5 is another type of sequential decoder where a stack is employed to memorize the K “best” candidates. The decoding starts from the root of the decoding tree. At each step, the child nodes of the best candidate in stack are examined. Then the metrics of the new generated child nodes and the rest candidates in stack are sorted and the K candidates with smallest metrics survive. This examination continues til the best candidate is a leaf node.

Unlike the KSE decoder, the candidates can have different length, so the Fano-like metric is applied in order to give a good comparison.

Algorithm 4 Fano lattice decoder

Let \mathbf{R} denote the upper-triangular lattice generator matrix, \bar{y} denote the received vector, b denote the stage bias and Δ denote the metric increment. This algorithm outputs coordinates $\bar{s} \in \mathbb{Z}^n$ of the closest point to \bar{y} in the lattice $\Lambda(\mathbf{R}) = \mathbf{R}\mathbb{Z}^n$ in an heuristic way. Both the lattice point and its relative metric are given.

```
1.[Initialization]  $k \leftarrow n, T \leftarrow 0, \bar{m} \leftarrow \mathbf{0}$ 
2.[Calculate]       $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
                    $u_k \leftarrow \left[ \frac{e_k}{r_{k,k}} \right]_{\text{int}}$ 
                    $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
                    $\text{step}_k \leftarrow \text{sgn}(d)$ 
3.[Compare]        $M \leftarrow m_k + d^2 - bN_0/2$ 
                   if  $M \leq T$ 
                       if  $k = 1$ 
                            $\bar{s} \leftarrow \bar{u}$ 
                           output  $\bar{s}, M$ 
                           terminate algorithm
                       else
                           go to step 4
                       endif
                   else
                       if  $k = n$  or  $m_{k+1} > T$ 
                            $T \leftarrow T + \Delta$ 
                           go to step 2
                       else
                            $k \leftarrow k + 1$ 
                            $u_k \leftarrow u_k + \text{step}_k$ 
                            $\text{step}_k \leftarrow -\text{step}_k - \text{sgn}(\text{step}_k)$ 
                           go to step 3
                       endif
                   endif
4.[Forward]        $k \leftarrow k - 1$ 
                    $m_k \leftarrow M$ 
                   if  $m_{k+1} > T - \Delta$ 
                       while  $m_k \leq T - \Delta$ 
                            $T \leftarrow T - \Delta$ 
                       endwhile
                   endif
                   go to step 2
```

Algorithm 5 Stack lattice decoder

Let \mathbf{R} be the upper-triangular lattice generator matrix, \bar{y} be the received vector, b be the stage bias. This algorithm outputs coordinates $\bar{s} \in \mathbb{Z}^n$ of the closest point to \bar{y} in the lattice $\Lambda(\mathbf{R}) = \mathbf{R}\mathbb{Z}^n$ in an heuristic way. Both the lattice point and its relative metric are given.

```
1.[Initialization] List = NULL,  $\overline{\text{deg}} \leftarrow \text{NULL}$ 
                    $\bar{m} \leftarrow \infty$ 
2.[Start]           $k \leftarrow n$ 
                   metric  $\leftarrow 0$ 
3.[Calculate]     if  $k = 0$ 
                    $\bar{s} \leftarrow \bar{u}$ 
                   output  $\bar{s}$ , metric
                   terminate algorithm
                   else
                    $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
                    $u_k \leftarrow \left\lfloor \frac{e_k}{r_{k,k}} \right\rfloor_{\text{int}}$ 
                    $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
                   step  $\leftarrow \text{sgn}(d)$ 
                   go to step 4
                   endif
4.[Compare]        $M \leftarrow \text{metric} + d^2 - bN_0/2$ 
                   if  $M < m_K$ 
                   go to step 5
                   else
                   go to step 7
                   endif
5.[Insertion]      $p \leftarrow \text{Pos}(M, \bar{m})$ 
                   List  $\leftarrow \{\text{List}_1^{p-1}, \bar{u}, \text{List}_p^{K-1}\}$ 
                    $\overline{\text{deg}} \leftarrow \{\overline{\text{deg}}_1^{p-1}, k-1, \overline{\text{deg}}_p^{K-1}\}$ 
                    $\bar{m} \leftarrow \{\bar{m}_1^{p-1}, M, \bar{m}_p^{K-1}\}$ 
                   if  $p = K$ 
                   go to step 7
                   else
                   go to step 6
                   endif
6.[Next]           $u_k \leftarrow u_k + \text{step}$ 
                    $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
                   step  $\leftarrow -\text{step} - \text{sgn}(\text{step})$ 
                   go to step 4
7.[Forward]        $\bar{u} \leftarrow \overline{\text{List}}_1, k \leftarrow \overline{\text{deg}}_1$ 
                   metric  $\leftarrow m_1$ 
                   List  $\leftarrow \{\text{List}_2^K, \text{NULL}\}$ 
                    $\overline{\text{deg}} \leftarrow \{\overline{\text{deg}}_2^K, \text{NULL}\}, \bar{m} \leftarrow \{\bar{m}_2^K, \infty\}$ 
                   go to step 3
```

where \bar{m} denotes the stack that records the candidates' metrics in non-decreasing order and the function $\text{Pos}(M, \bar{m})$ returns the index p such that $m_{p-1} \leq M < m_p$.

2.4.5 Generalized Schnorr-Euchner decoder: SEF decoder

The ML decoding in general demands huge computations for high dimension lattice. Inspired by the Fano metric, we associate a non-decreasing bias at each level which is integrated in the metric calculation by modifying the calculation of newdist in step 4 of Algorithm 1:

$$\text{newdist} \leftarrow \text{dist}_k + d^2 - b(n - k) \frac{N_0}{2} \quad (2.27)$$

This modified SE decoder is named SE decoder with Fano-like metric (SEF) [25]. The biased metric can effectively accelerate the searching speed that the complexity is reduced with trivial performance loss.

2.5 MIMO decoder performance

In this section, we will discuss the performance of MIMO decoders in terms of Word Error Rate (WER) and decoding complexity. We suppose the CSI is perfectly known at the receiver and the MMSE-GDFE is performed for the left-processing of channel matrix.

We group the MIMO decoders into 2 categories : the fixed-latency decoders and the variant-latency decoders.

The fixed-latency decoders include ZF decoder, DFE decoder and KSE decoder. As shown in section 2.4, the fixed-latency decoder uses the same structure in a recursive way. The decoding complexity is linear to the MIMO system's dimension which means the decoder's delay can be controlled. For the Wi-Fi system which demands a strict decoding latency², we are inclined to apply the fixed-latency decoders for hardware implementation [26].

The SE/SEF decoder, Fano decoder and stack decoder are considered as variant-latency decoders. The variant-latency decoder is not suitable for hardware level implementation since their processing delays are unpredictable. This type of decoders can be implemented in software level like the platforms with microprocessor support or for systems not sensitive to the latency. For the interest of complexity evaluation, we will evaluate the algorithm's efficiency by analysing the average number of visited points per level during the lattice searching.

For the right-processing mentioned in section 2.3, we will investigate 3 situations: without any right-processing (direct decoding), with permutation matrix using GO and with LLL reduction method.

We use the uncoded V-BLAST in all MIMO configurations. The ML performance obtained with SE decoder is provided as a reference. The MIMO channel is modeled as quasi-static Rayleigh fading channel that for channel matrix \mathbf{H} the channel coefficients $h_{i,j} \sim \mathcal{CN}(0, 1)$ are i.i.d. The MIMO systems of 2×2 , 4×4 and 16×16 ³ are studied with 2 constellations: QPSK, 16-QAM.

²In 802.11a/n, the maximum of PHY layer processing delay is $12\mu\text{s}$.

³The 4×4 MIMO system simulates the decoding of the Golden code while the 16×16 MIMO system simulates the case of 4×4 Perfect code.

2.5.1 Direct decoding

Let us first see the performances of the fixed-latency MIMO decoders which are shown in Figures 2.1-2.6. It is not astonishing that ZF algorithm gives the worst performance. Even in 2×2 MIMO case, the ZF's performance is far from ML performance. The DFE decoder outperforms the ZF algorithm in all the configurations, especially for high dimension MIMO cases. However, both ZF algorithm and DFE algorithm lose the system's diversity gain which is essential to the application of cooperative diversity. The KSE algorithm appears a good solution: with a buffer of size 4, its performance is very close to the ML curve in 2×2 and 4×4 MIMO system. For the 16×16 MIMO case, the gap between KSE and ML curves is greater that we lose about 6dB for $WER=10^{-3}$ in 16×16 MIMO system with 16-QAM.

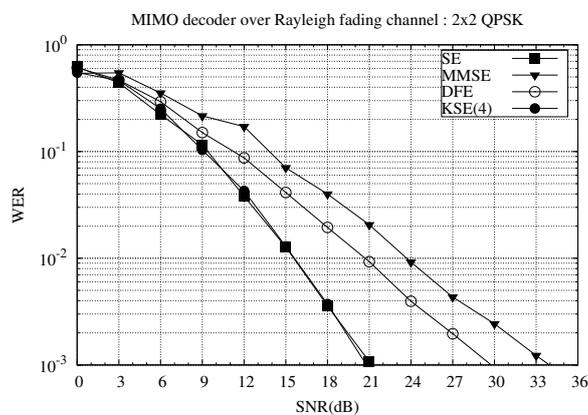


Figure 2.1: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 2x2 MIMO system with QPSK constellation

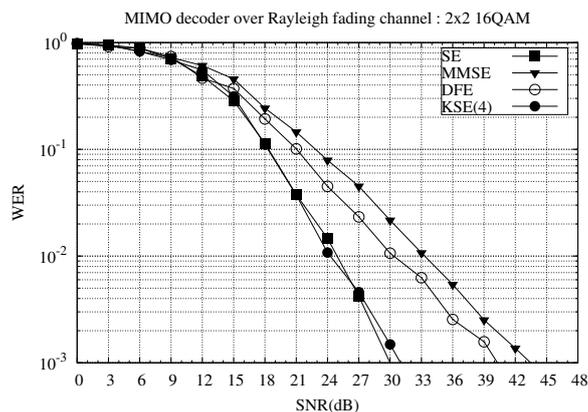


Figure 2.2: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 2x2 MIMO system with 16-QAM constellation

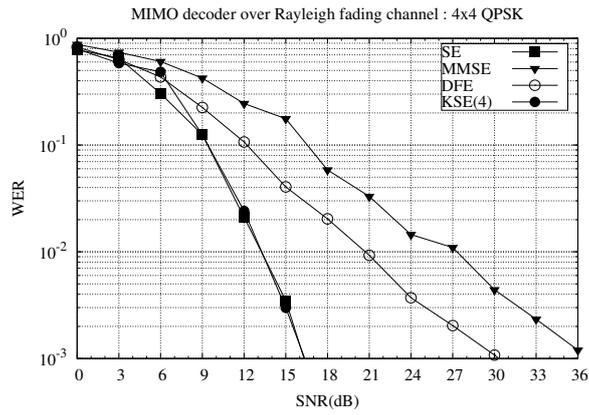


Figure 2.3: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 4x4 MIMO system with QPSK constellation

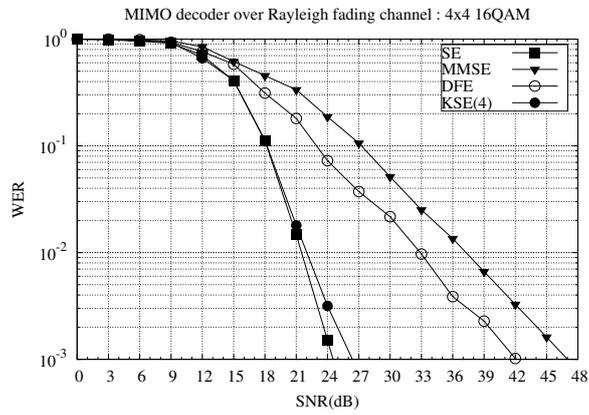


Figure 2.4: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 4x4 MIMO system with 16-QAM constellation

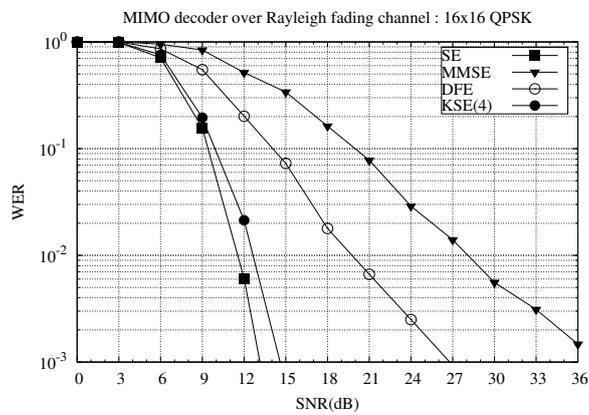


Figure 2.5: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 16x16 MIMO system with QPSK constellation

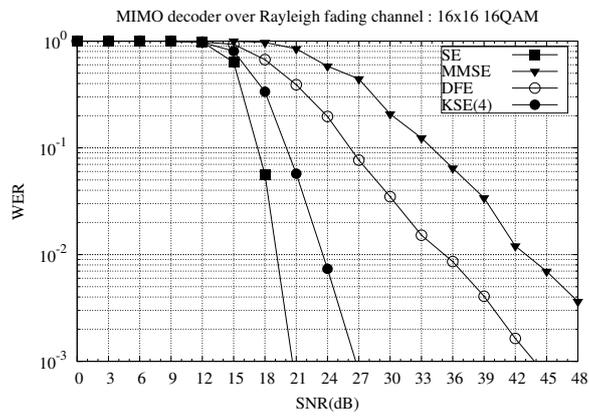


Figure 2.6: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder in 16x16 MIMO system with 16-QAM constellation

Now let us see the variant-latency decoders. For the parameter configuration, the bias b is set to 1 for all decoders, Δ is set to $N_0/8$ which gives good tradeoff between decoding performance and complexity. The buffer of stack decoder is set to 4.

As shown in Figures 2.7-2.12, Fano decoder and SEF decoder give both quasi-ML performance. Stack decoder achieves also very good performance in 2×2 and 4×4 MIMO systems but it loses about 8dB (WER= 10^{-3}) in 16×16 MIMO system with 16-QAM. We need to increase the stack size to improve the decoder performance for high dimension MIMO cases.

In terms of complexity, the sub-optimal decoders, Fano decoder, stack decoder and SEF decoder, show great advantage to the ML decoder when the MIMO system dimension or the number of constellation points increases. In 16×16 MIMO case, the complexity of ML decoder is too high which prevents the implementation of a ML decoder in high dimension MIMO, especially for the low and medium SNR ranges. In this case, the sub-optimal decoders are preferred for their excellent decoding performance and affordable calculation time.

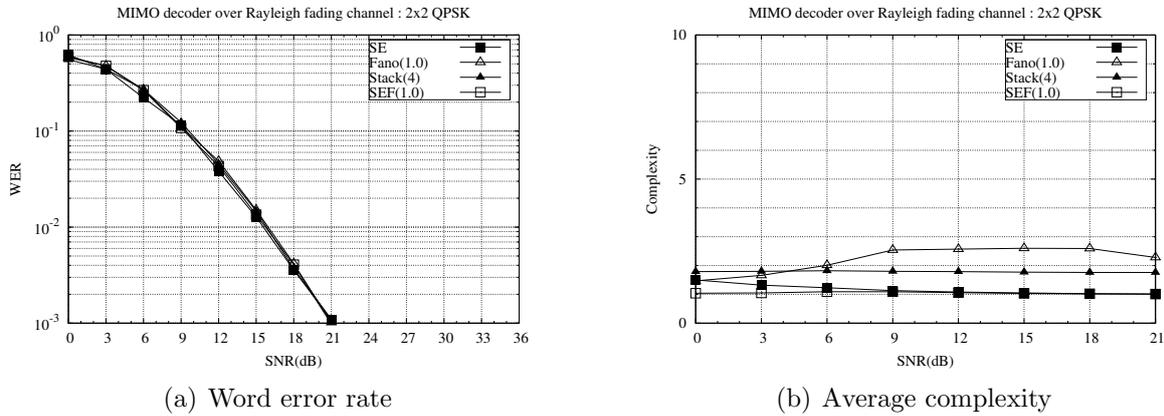


Figure 2.7: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 2x2 MIMO system with QPSK constellation

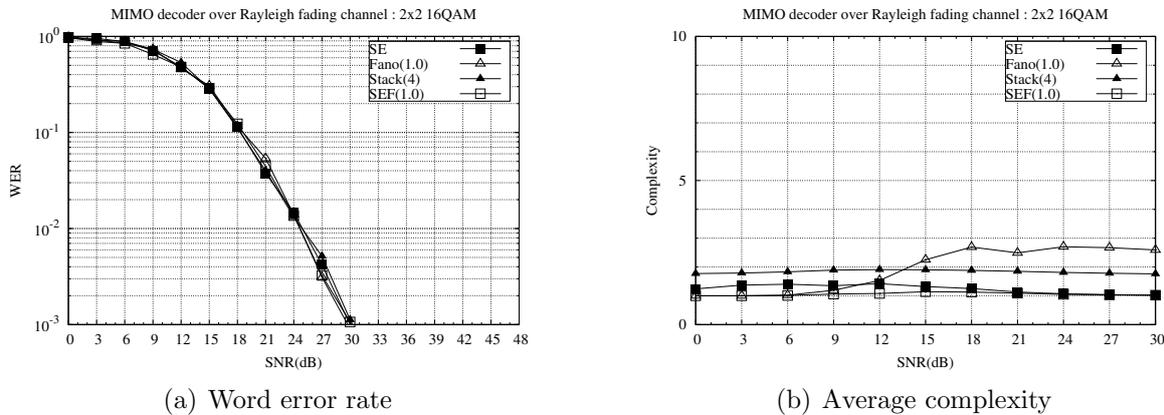
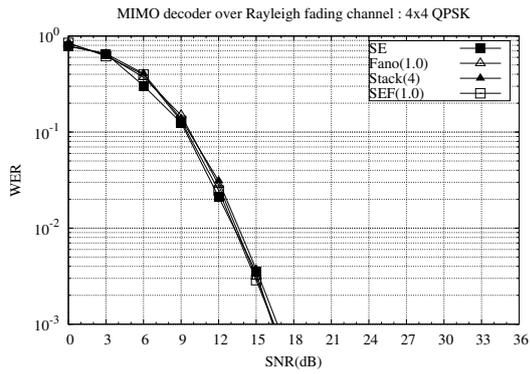
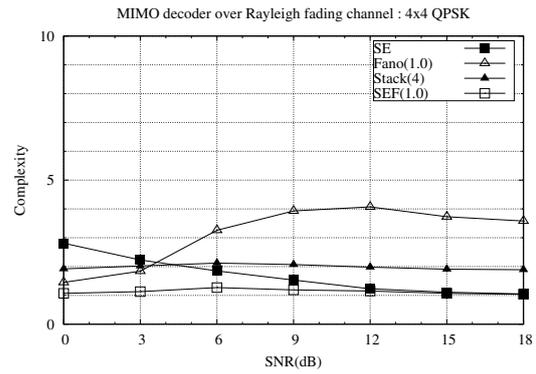


Figure 2.8: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 2x2 MIMO system with 16-QAM constellation

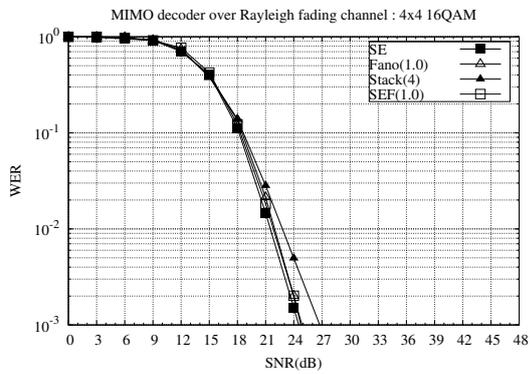


(a) Word error rate

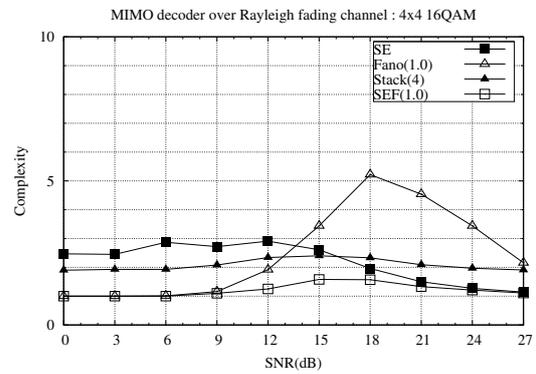


(b) Average complexity

Figure 2.9: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 4x4 MIMO system with QPSK constellation

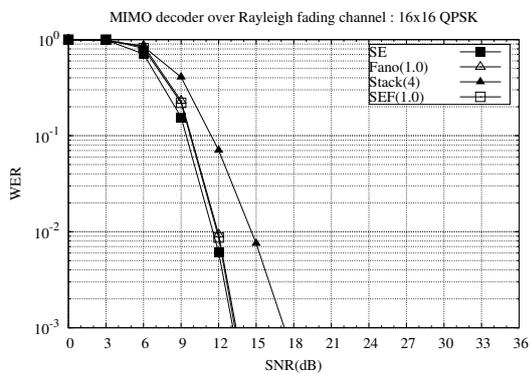


(a) Word error rate

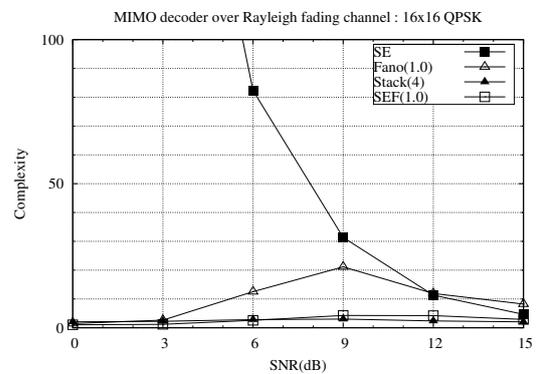


(b) Average complexity

Figure 2.10: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 4x4 MIMO system with 16-QAM constellation

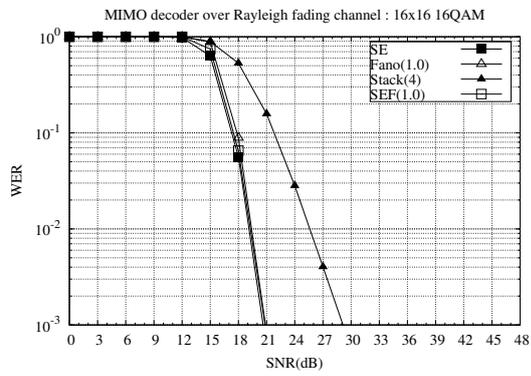


(a) Word error rate

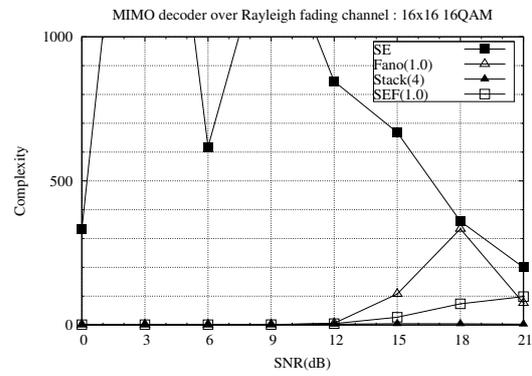


(b) Average complexity

Figure 2.11: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 16x16 MIMO system in QPSK constellation



(a) Word error rate



(b) Average complexity

Figure 2.12: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder in 16x16 MIMO system with 16-QAM constellation

2.5.2 Greedy ordering

A good decoding order can bring great improvement in the decoding performance. The simulation results illustrate the assistance of the GO method for the algorithms like DFE/KSE decoder, Fano decoder, stack decoder and SE/SEF decoder⁴.

The performances of DFE and KSE using GO are shown in Figure 2.13-2.18. The performance of DFE is significantly improved when applying GO and the decoding performance is close to ML performance for QPSK case. This improvement is more obvious with KSE decoder that achieves near ML performance: in 16×16 MIMO system with 16-QAM constellation, we lose only 1dB to the ML curve.

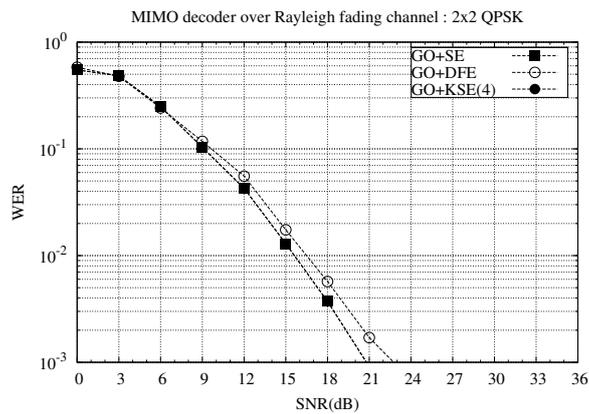


Figure 2.13: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 2x2 MIMO system with QPSK constellation

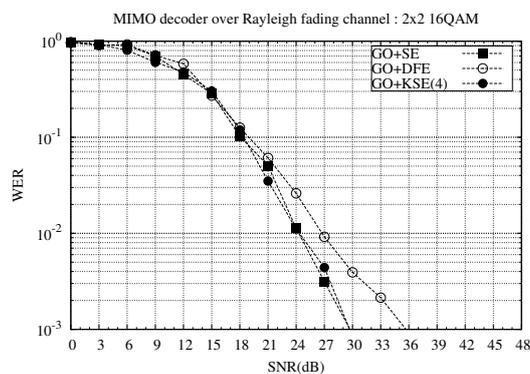


Figure 2.14: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 2x2 MIMO system with 16-QAM constellation

⁴The ordering method does not impact the performance of ZF decoding.

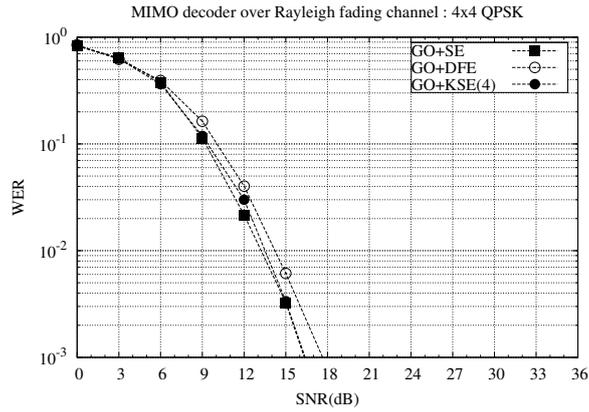


Figure 2.15: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 4x4 MIMO system with QPSK constellation

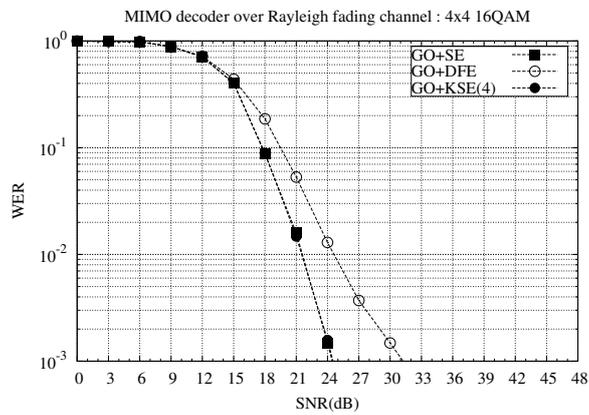


Figure 2.16: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 4x4 MIMO system with 16-QAM constellation

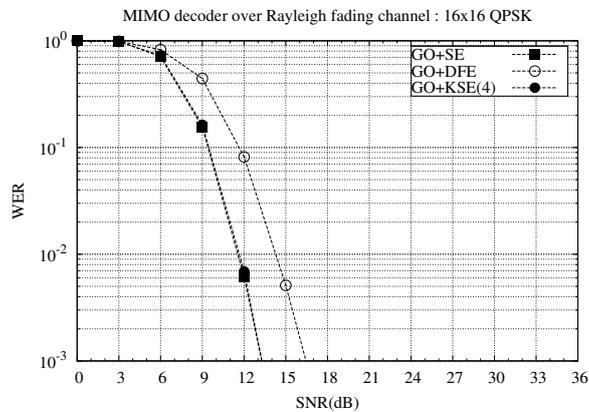


Figure 2.17: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 16x16 MIMO system with QPSK constellation

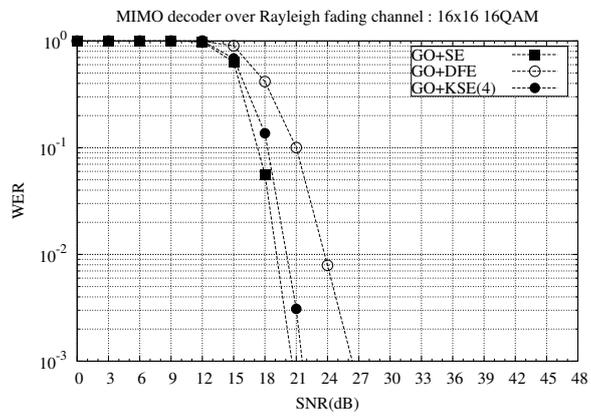


Figure 2.18: MIMO decoders' performance over Rayleigh fading channel: DFE decoder and KSE decoder with greedy ordering method in 16x16 MIMO system with 16-QAM constellation

The simulation results for Fano decoder, stack decoder and SEF decoder are shown in Figure 2.13-2.18. The GO processing improves the stack decoder's performance: the loss of stack decoder to ML performance is about 3dB ($WER=10^{-3}$) in 16×16 MIMO system with 16-QAM. Another advantage of this processing is that the complexity of SEF decoder and Fano decoder decreases a bit. This result is more notable for the ML decoder using SE decoder: in the 16×16 MIMO system, comparing Figure 2.17 and Figure 2.18, the complexity is much reduced.

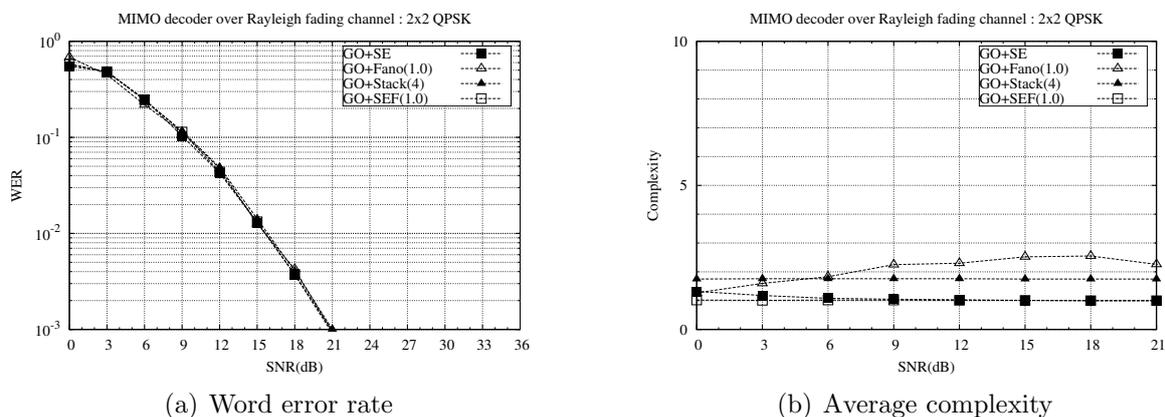


Figure 2.19: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 2x2 MIMO system with QPSK constellation

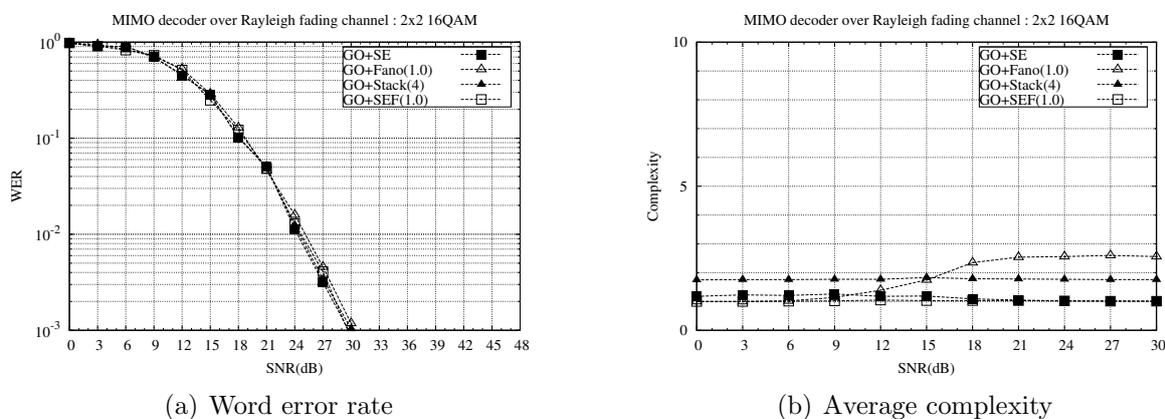
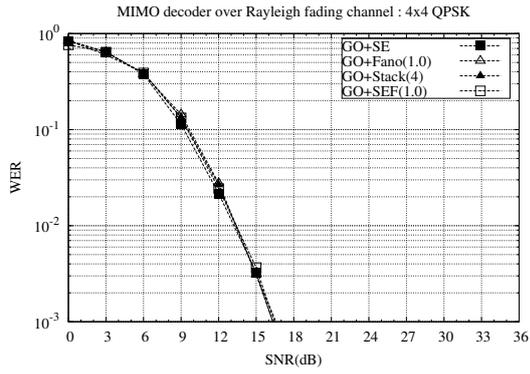
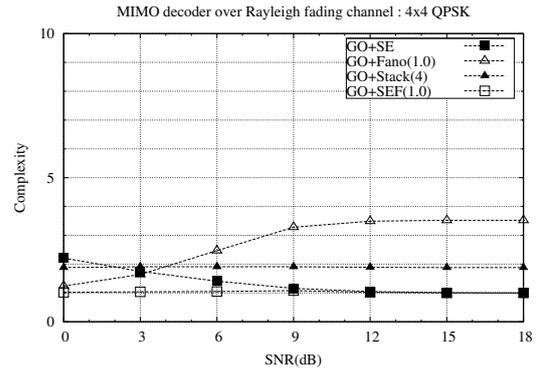


Figure 2.20: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 2x2 MIMO system with 16-QAM constellation

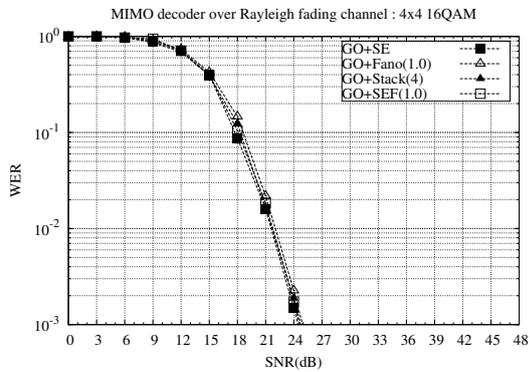


(a) Word error rate

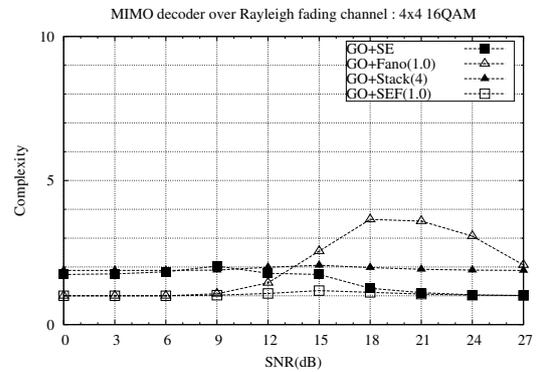


(b) Average complexity

Figure 2.21: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 4x4 MIMO system with QPSK constellation

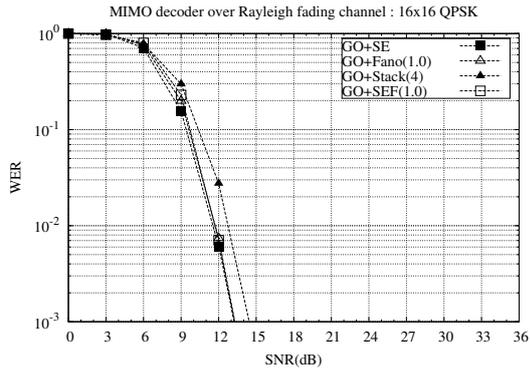


(a) Word error rate

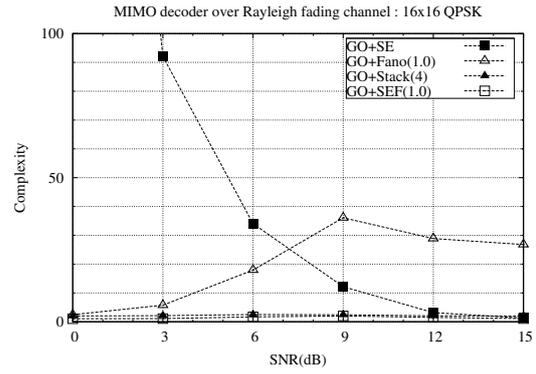


(b) Average complexity

Figure 2.22: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 4x4 MIMO system with 16-QAM constellation

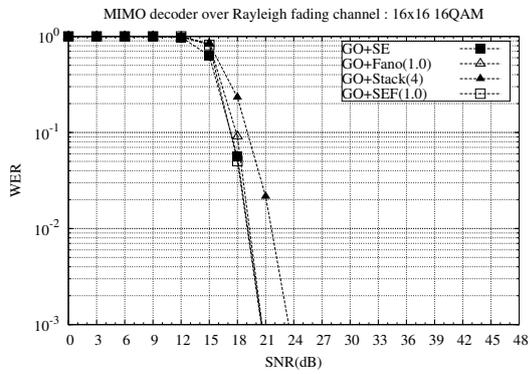


(a) Word error rate

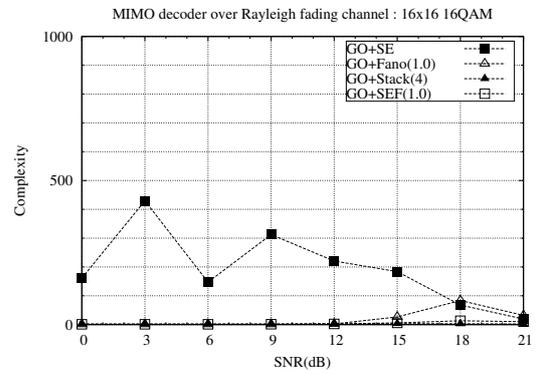


(b) Average complexity

Figure 2.23: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 16x16 MIMO system with QPSK constellation



(a) Word error rate



(b) Average complexity

Figure 2.24: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with greedy ordering method in 16x16 MIMO system with 16-QAM constellation

2.5.3 LLL reduction

The LLL reduction is a very powerful tool to simplify the channel matrix. We show the simulation results for ZF decoder, DFE decoder and KSE in Figures 2.25-2.30 and Fano decoder, Stack decoder and SEF decoder in Figures 2.31-2.36.

With LLL reduction, the channel matrix is close to an “orthogonal” matrix that even the ZF detection can achieve good performance. In 2×2 and 4×4 MIMO systems, the ZF decoder with LLL reduction loses very little to the ML decoder. For DFE decoder and KSE decoder, LLL reduction improves also the decoding performances which are slightly better than the performances with GO processing.

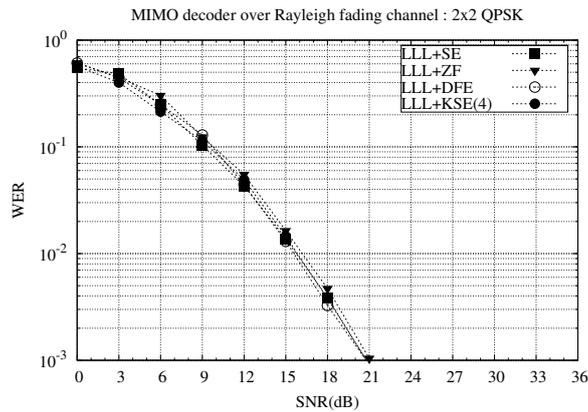


Figure 2.25: MIMO decoders’ performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 2x2 MIMO system with QPSK constellation

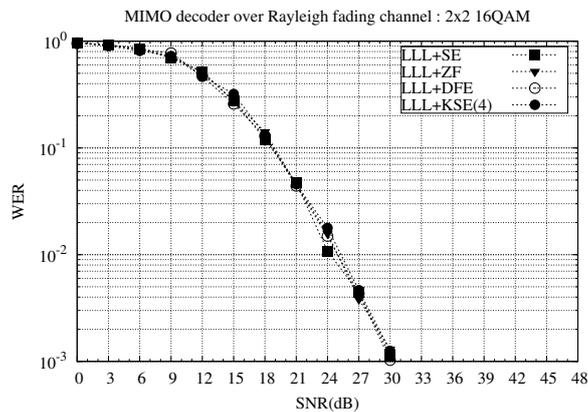


Figure 2.26: MIMO decoders’ performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 2x2 MIMO system with 16-QAM constellation

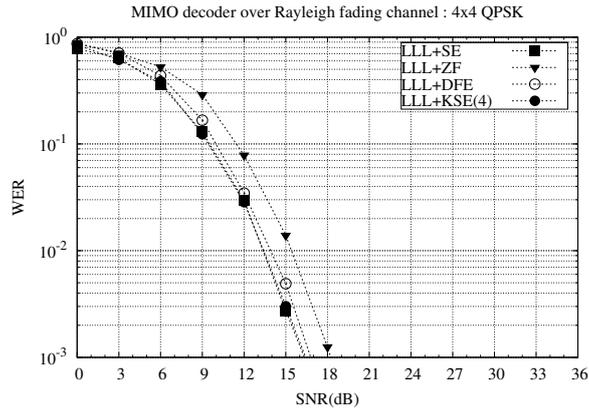


Figure 2.27: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DEF decoder and KSE decoder with LLL reduction in 4x4 MIMO system with QPSK constellation

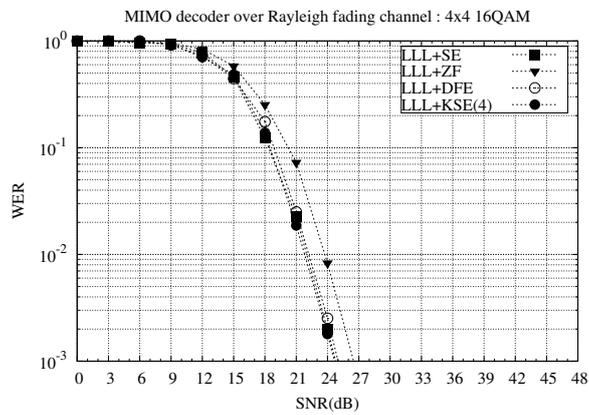


Figure 2.28: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 4x4 MIMO system with 16-QAM constellation

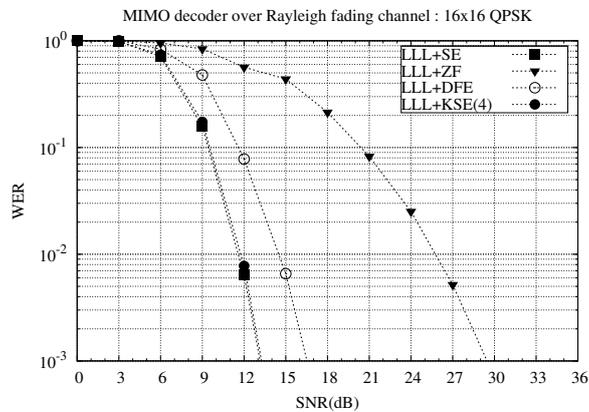


Figure 2.29: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 16x16 MIMO system with QPSK constellation

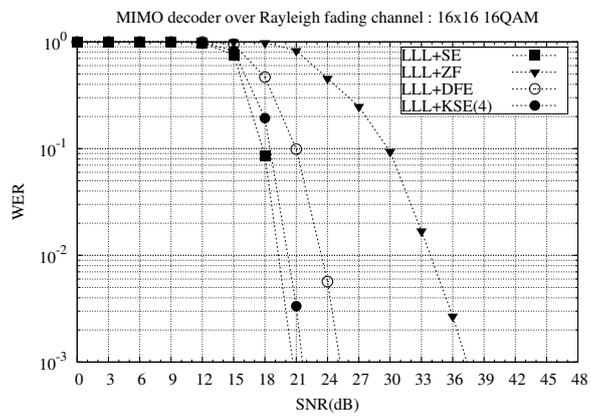


Figure 2.30: MIMO decoders' performance over Rayleigh fading channel: ZF decoder, DFE decoder and KSE decoder with LLL reduction in 16x16 MIMO system with 16-QAM constellation

Figures 2.31-2.36 show the simulation results of Fano decoder, stack decoder and SEF decoder. The LLL reduction improves the stack decoder's performance and decreases the complexity of Fano decoder and SE/SEF decoder.

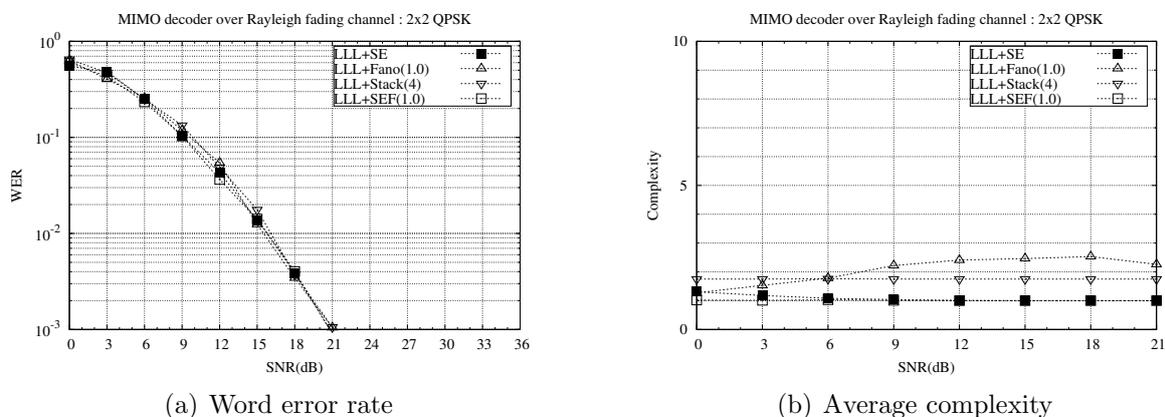


Figure 2.31: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 2x2 MIMO system with QPSK constellation

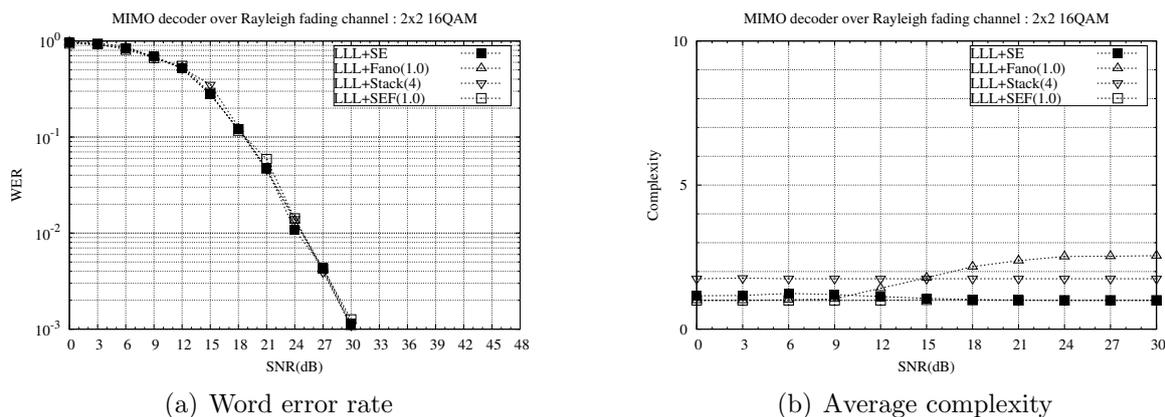
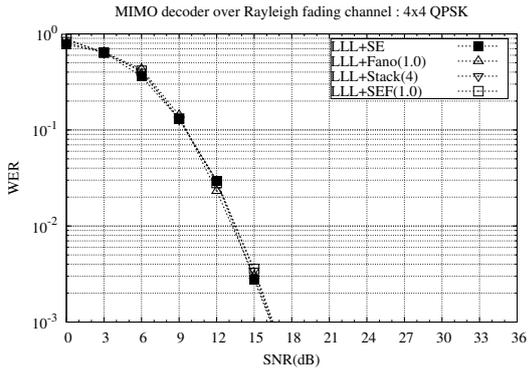
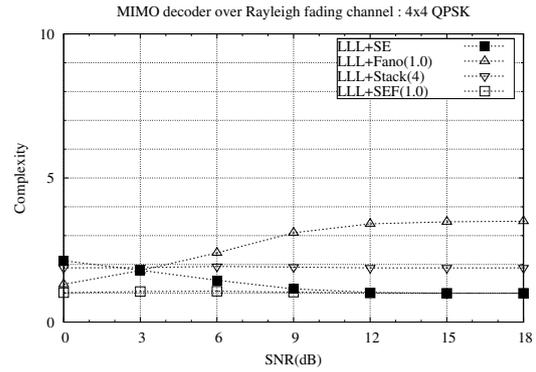


Figure 2.32: MIMO decoder performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 2x2 MIMO system with 16-QAM constellation

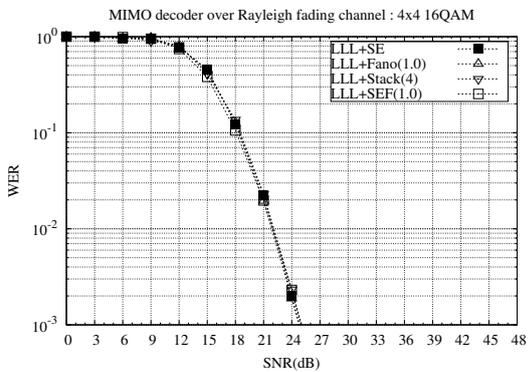


(a) Word error rate

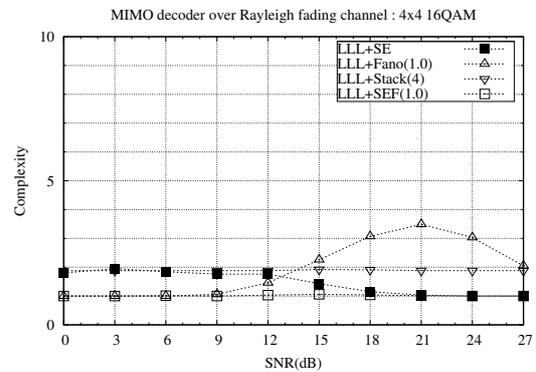


(b) Average complexity

Figure 2.33: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 4x4 MIMO system with QPSK constellation

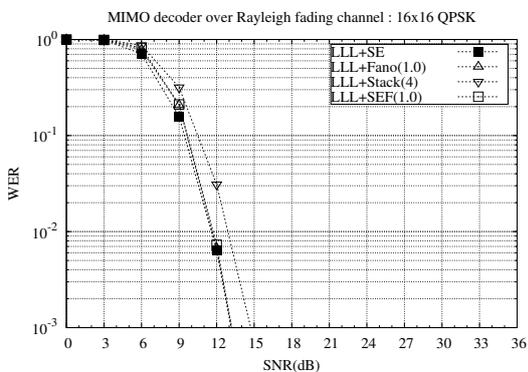


(a) Word error rate

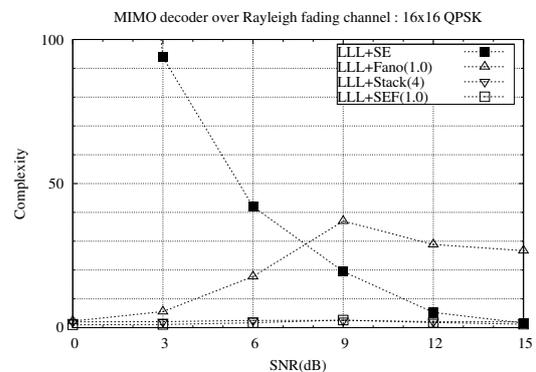


(b) Average complexity

Figure 2.34: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 4x4 MIMO system with 16-QAM constellation

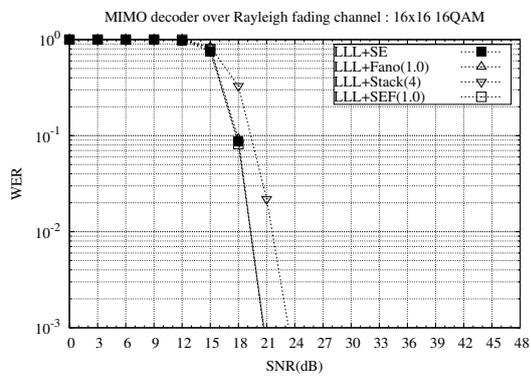


(a) Word error rate

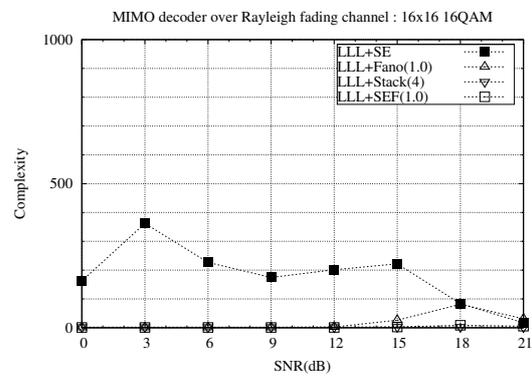


(b) Average complexity

Figure 2.35: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 16x16 MIMO system with QPSK modulation



(a) Word error rate



(b) Average complexity

Figure 2.36: MIMO decoders' performance over Rayleigh fading channel: Fano decoder, Stack decoder and SEF decoder with LLL reduction in 16x16 MIMO system with 16-QAM constellation

2.5.4 Conclusion

As shown in the simulation results, the right-processing on the channel matrix can bring effectively improvements on the sub-optimal decoders' performances. It is worthy to remark that either GO processing or LLL reduction demands extra computations. The advantage of the right-processing is that once it is done, the permutation or reduction matrix can be used for the decoding of all the received codewords. Therefore, when the channel is stable for long enough time, the GO processing or LLL reduction is acceptable for the channel processing.

In cases of low dimension MIMO systems, the performance of DFE decoder with GO processing or ZF detection with LLL reduction is very close to the ML performance. We can apply these decoders instead of a ML one for the interest of complexity reduction.

In cases of high dimension MIMO systems, Fano decoder and SEF decoder achieve both good performance. We suggest also to use the right-processing with the two decoders in order to reduce the complexity. With right-processing and sufficient stack size, stack decoder and KSE decoder give also quasi-ML performance. All these decoders can replace the ML decoder for a reasonable complexity.

For the interest of hardware implementation, we are glad to evaluate the performance of DFE/KSE decoder with greedy ordering. Hereafter, we use the ZF with LLL reduction in low dimension MIMO cases and SEF with LLL reduction in high dimension MIMO cases to generate a reference of "ML" performance. The DFE/KSE decoder with GO processing is then evaluated by adjusting the parameter K to find a tradeoff of the complexity and the sub-optimal performance regarding the ML performance.

CHAPTER 3

RELAY-SISO

The IEEE 802.11a-based Wi-Fi [27] system provides a rough data throughput up to 54Mbps. The OFDM modulation is used to simplify the equalization procedure for the channel's frequency-selectivity. This technique divides the bandwidth into orthogonal frequency sub-channels. The 20MHz bandwidth is divided into 64 sub-channels (sub-carriers) that the sub-carrier spacing is 312.5kHz. Considering the typical channel spreading is up to about 2000ns [28] which gives a coherent bandwidth about 500kHz, each sub-channel is considered as a flat channel that the channel equalization is easily performed.

Like other wireless communication systems, IEEE 802.11a system suffers from the fading phenomenon which can be detailed in 2 aspects: the slow fading and the fast fading. The slow fading is represented as a long-time degradation of SNR due to the shadowing effect between the source terminal and the destination terminal. The fast fading is related to the rapid environment's change or the terminals' mobility. The diversity technology like MIMO technique, can effectively enhance the system's robustness facing the fast fading.

However, the multiple antennas solution is not always easy to realize and the existing Wi-Fi devices are equipped generally with one antenna. The arising cooperative diversity technique seems to be a good approach to upgrade the current Wi-Fi network by building a virtual-MIMO system. In general, the terminals in a typical 802.11a network can be considered well synchronized in time domain since the propagation delay is negligible¹.

Another advantage of the cooperation network is the geographic gain of the relay terminal. Actually, the relay terminal can be found in a better transmission condition which means both the source to relay link and the relay to destination link are better than the direct source to destination link. A very simple example is when the relay is placed in the middle of the source and destination terminals that the source to relay link has less attenuation than the source to destination link, as well as the relay to destination link. This geographic advantage is defined by the SNR gain of indirect links to the direct link.

We are going to incorporate the single-relay single-antenna NAF cooperation system in the IEEE 802.11a standard by introducing minimum of modifications as possible. Both the compatibility with legacy IEEE 802.11a standard and the feasibility of cooperative

¹Consider an indoor environment where the distance between terminals is typically smaller than 30 meters. The propagation delay is about $0.1\mu\text{s}$ which is much smaller than the guard interval duration $0.8\mu\text{s}$.

diversity are considered in the proposal of this new cooperative physical layer, Relay-SISO.

3.1 IEEE 802.11a PHY layer

IEEE 802.11a [27] Physical (PHY) layer operates with 20MHz band occupation in the 5GHz band. This system adopts the Orthogonal Frequency-Division Multiplexing (OFDM) modulation which employs 64 sub-carriers, including 48 data sub-carriers, 4 pilots sub-carriers and 12 unused sub-carriers. This modulation is achieved in using Fast Fourier Transform (FFT) technique. The transmitter processing diagram of IEEE 802.11a PHY layer is illustrated in Figure 3.1.

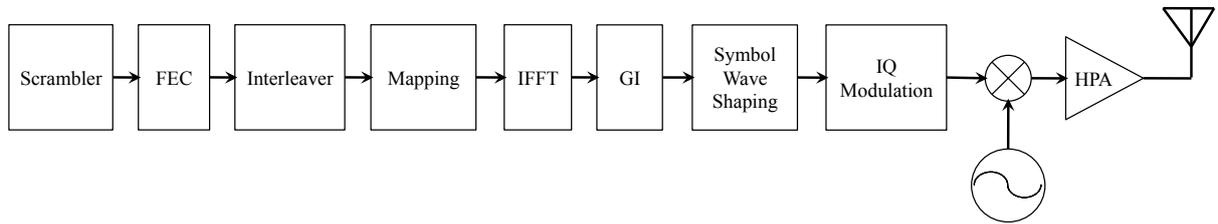


Figure 3.1: 802.11a general architecture of processing

IEEE 802.11a supports 8 possible data rates which are the combination of 3 convolutional coding rates, $R = 1/2, 2/3$ or $3/4$, and 4 constellations, BPSK, QPSK, 16-QAM and 64-QAM. The data rate dependent parameters are provided in Table 3.1.

Table 3.1: IEEE 802.11a rate-dependent parameters

Date rate (Mbits/s)	Modulation	Coding rate (R)	Coded bits per subcarrier (N_{BPSK})	Coded bits per OFDM symbol (N_{CBPS})	Data bits per OFDM symbol (N_{DBPS})
6	BPSK	1/2	1	48	24
9	BPSK	3/4	1	48	36
12	QPSK	1/2	2	96	48
18	QPSK	3/4	2	96	72
24	16-QAM	1/2	4	192	96
36	16-QAM	3/4	4	192	144
48	64-QAM	2/3	6	288	192
54	64-QAM	3/4	6	288	216

The Medium Access Control (MAC) layer sends the MAC Protocol Data Unit (MPDU) to the PHY layer where the MPDU is encapsulated into the Physical Service Data Unit (PSDU). The PSDU is scrambled and coded with an industry-standard convolutional code² in the Forward Error Correction (FEC) module, which is illustrated in Figure 3.2.

²The generator polynomials are $g_0 = (133)_8$ and $g_1 = (171)_8$.

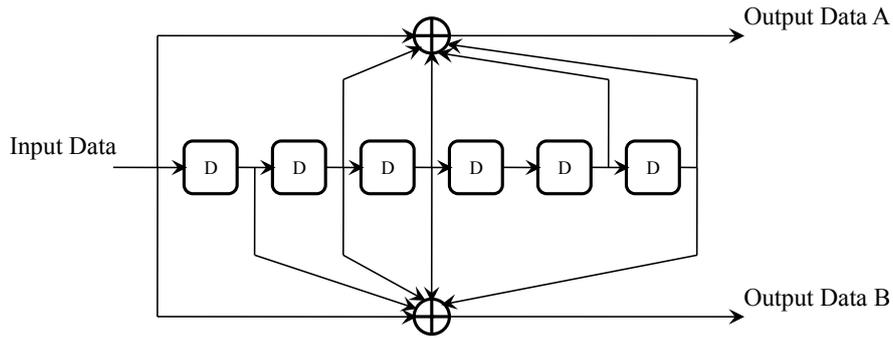


Figure 3.2: Convolutional encoder: $R = 1/2$ with $g_0 = (133)_8$ and $g_1 = (171)_8$

The encoded bits are assembled and then punctured if necessary for different coding rates, as shown in Figures 3.3-3.5.

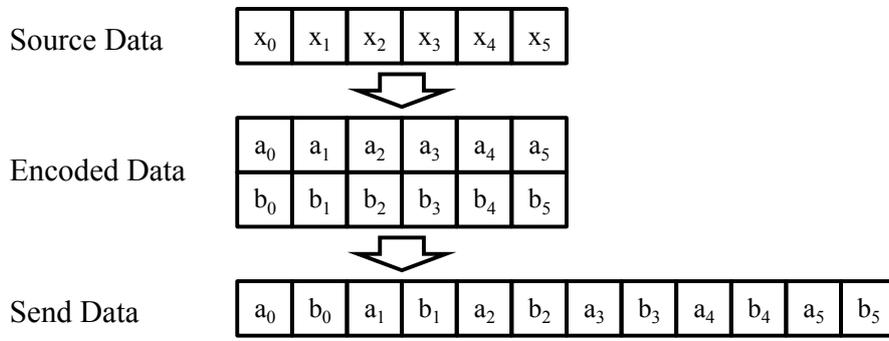


Figure 3.3: Output of convolutional code at $R = 1/2$

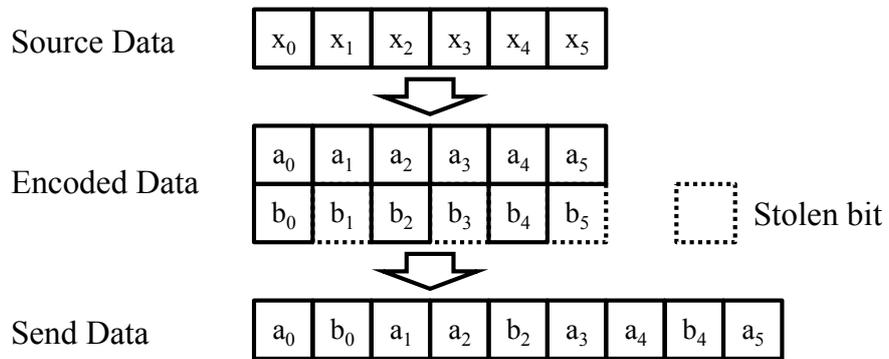


Figure 3.4: Output of convolutional code at $R = 2/3$

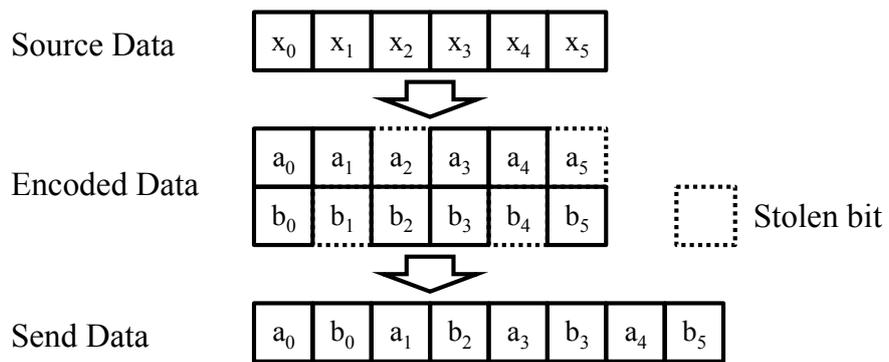


Figure 3.5: Output of convolutional code at $R = 3/4$

To avoid the frequency selective fading, the bit stream is processed in the classic rectangular interleaver followed by a bit-rotation operation. The binary stream is then grouped and mapped to constellation points using Gray code.

For the OFDM modulation, a 64-point IFFT/FFT block is used at the transmitter/receiver side. 52 subcarriers for frequency index -26 to -1 and +1 to +26 are used while the other 12 subcarriers including DC sub-carrier are set to 0. Each OFDM data symbol contains 48 data sub-carriers and 4 pilote sub-carriers. The frequency mapping is illustrated in Figure 3.6.

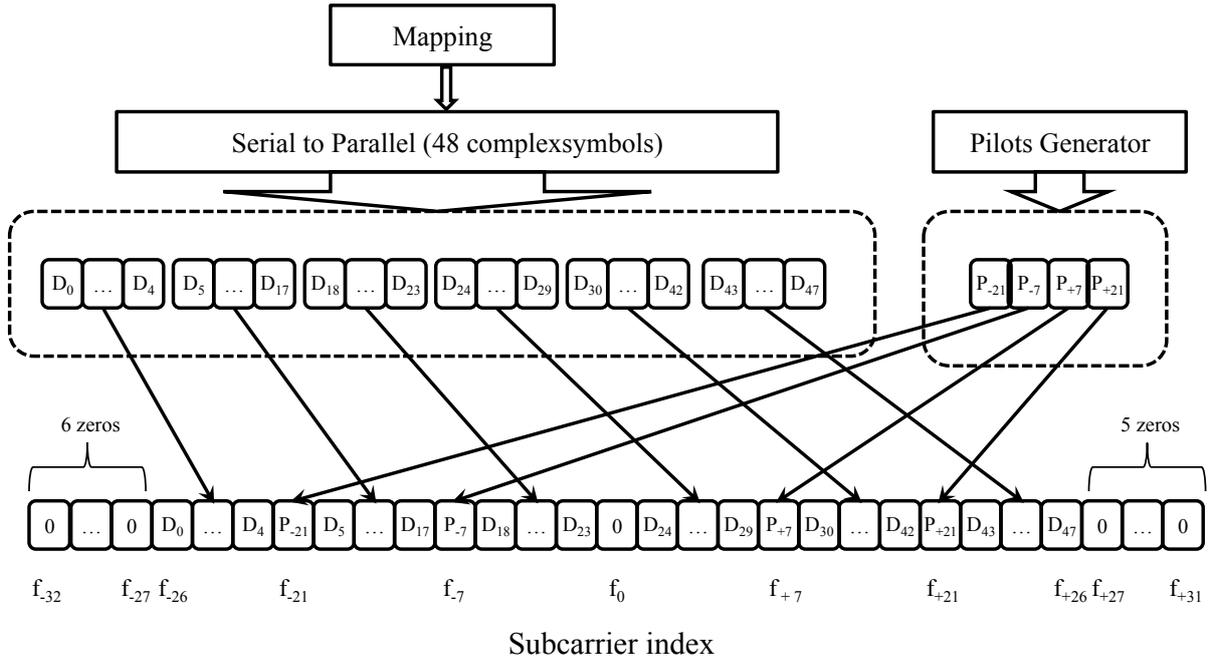


Figure 3.6: 802.11a frequency mapping for the OFDM symbol: 48 subcarriers are used for transmission of data symbol while 4 subcarriers are used for pilots at frequency index -21,-7,+7,+21; the other 12 subcarriers including DC are kept no used which generate protection band.

Let $N_{GI} = 16$ denote the length of the Guard Interval (GI) and let $N_{FFT} = 64$ denote number of points for FFT/IFFT. The GI is a Cyclic Prefix (CP) which is the copy of the last N_{GI} samples of an OFDM symbol. A complete OFDM symbol contains $N_{FFT} + N_{GI} = 80$ samples which are sent at a sample rate of 20MHz.

Figure 3.7 illustrates the IEEE 802.11a frame which is composed of the PLCP preamble field, the SIGNAL field of 1 OFDM symbol and the data field. The Physical Layer Convergence Procedure (PLCP) preamble field consists of 10 short symbols and 2 long symbols with a CP of $1.6\mu s$, which is known as Guard Interval for long training symbols (GI2) Each short symbol lasts for $0.8\mu s$ while the 2 long symbols last for $6.4\mu s$ without the GI2. The SIGNAL field is always coded with coding rate $R = 1/2$ using modulation BPSK. It contains the information on the length of MPDU, which represents the number of transmitting bytes, and the constellation and the coding rate used in the data OFDM symbols.

At the reveicer side, the short symbols will first be used for the Automatic Gain Control

Short symbols	Long symbols	SIGNAL	Data 1	Data 2	...	Data N_{SYM}
---------------	--------------	--------	--------	--------	-----	----------------

Figure 3.7: IEEE 802.11a frame

(AGC) training and IEEE 802.11a signal detection. The carrier leakage or Direct Current (DC) offset is also estimated during the reception of the short symbols and is removed for the rest signal. The two long symbols are used to estimate the channel coefficients of sub-carriers in use.

A crucial problem for this system is the Carrier Frequency Offset (CFO) problem. In fact, the OFDM system is very sensitive to the CFO since a frequency offset will bring inter-carrier interferences which break the orthogonality between sub-carriers. As demanded in the IEEE 802.11a recommendation, the transmitted center frequency tolerance shall be ± 20 ppm maximum, which is equal to ± 100 KHz frequency offset for the carrier frequency of 5GHz. This frequency offset should be estimated and removed for frequency domain processing.

Generally, we need to estimate the CFO in 2 steps. The first step is the short symbols and the long symbols based CFO estimation. Since the short symbols and long symbols are periodic, it is easy to evaluate the frequency offset with baseband signal. The second step of pilots based phase tracking. The estimation result in first step may be inaccurate so the second step is necessary for the residual CFO error tracking by pursuing the phase evolution of the pilots.

The Sampling Frequency Offset (SFO) problem is less serious than the CFO problem. We will focus only on the CFO problem in the system in supposing non SFO.

3.2 Relay-SISO PHY layer

As an extension of IEEE 802.11a PHY layer, the Relay-SISO PHY layer is proposed with the following modifications for the application of cooperative code. Based on the model single-relay single-antenna NAF cooperation schema as presented in section 1.2, this cooperation system contains three terminals, *source*, *destination* and *relay*.

At the source terminal, an STBC unit is added between the QAM mapping unit and IFFT unit, as shown in Figure 3.8.

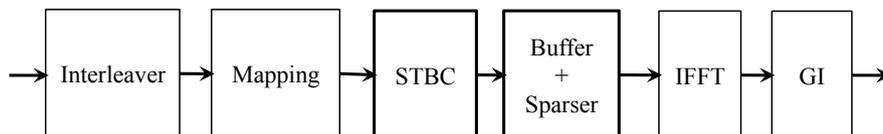


Figure 3.8: Relay-SISO PHY layer where an STBC unit is added between mapping unit and IFFT unit

Let $\{s_i\}$ denote the QAM symbol stream entering the STBC unit. The QAM symbols are grouped by 4 and each block is expressed by $\underline{s}_q = \{s_{q,1}, s_{q,2}, s_{q,3}, s_{q,4}\}$ where $s_{q,p} = s_{4q+p}$ for $p = 1, 2, 3, 4$.

We apply Golden coding on each s_q and the QAM symbol block is encoded into the 2×2 space-time codeword, noted by \mathbf{X}_q :

$$\mathbf{X}_q = \begin{bmatrix} X_{q,11} & X_{q,12} \\ X_{q,21} & X_{q,22} \end{bmatrix} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha(s_{q,1} + \theta s_{q,2}) & \alpha(s_{q,3} + \theta s_{q,4}) \\ \gamma \bar{\alpha}(s_{q,3} + \bar{\theta} s_{q,4}) & \bar{\alpha}(s_{q,1} + \bar{\theta} s_{q,2}) \end{bmatrix} \quad (3.1)$$

with

$$\begin{cases} \theta = \frac{1+\sqrt{5}}{2} \\ \bar{\theta} = \frac{1-\sqrt{5}}{2} \end{cases} \quad \text{and} \quad \begin{cases} \alpha = 1 - i - i\theta \\ \bar{\alpha} = 1 - i - i\bar{\theta} \\ \gamma = i \\ i^2 = -1 \end{cases}$$

The codeword stream is segmented by 48 that $\bar{\mathbf{X}}_n = \{\mathbf{X}_{48n}, \dots, \mathbf{X}_{48n+47}\}$, where $\bar{\mathbf{X}}_n$ represents a Cooperative Block (CB).

Let $\mathbf{X}_{n,k} = \begin{bmatrix} X_{n,k,11} & X_{n,k,12} \\ X_{n,k,21} & X_{n,k,22} \end{bmatrix}$ denote a codeword on sub-carrier k of the n^{th} CB and let N_{CB} denote the number of cooperative blocks. $\underline{\mathbf{X}}_k = \{\mathbf{X}_{0,k}, \dots, \mathbf{X}_{N_{\text{CB}}-1,k}\}$ is then the codeword stream on sub-carrier k .

The codeword streams are transmitted in 2 data fields: the first part is noted by data A and the second part is noted by data B.

Let $A_{m,k}$ be the symbol transmitted in data A on sub-carrier k in m^{th} OFDM symbol. The sequence $\underline{A}_k = \{A_{m,k}\}$ for $m = 0, \dots, 2N_{\text{CB}} - 1$ is transmitted with power π_{s1} :

$$\underline{A}_k = \sqrt{\pi_{s1}} \left\{ X_{0,k,11}, X_{0,k,12}, X_{1,k,11}, X_{1,k,12}, \dots, X_{N_{\text{CB}}-1,k,11}, X_{N_{\text{CB}}-1,k,12} \right\} \quad (3.2)$$

Using the same notation as data A, the sequence \underline{B}_k of data B is transmitted with power π_{s2} :

$$\underline{B}_k = \sqrt{\pi_{s2}} \left\{ X_{0,k,21}, X_{0,k,22}, X_{1,k,21}, X_{1,k,22}, \dots, X_{N_{\text{CB}}-1,k,21}, X_{N_{\text{CB}}-1,k,22} \right\} \quad (3.3)$$

where π_{s1} and π_{s2} denote the power factors as defined in section 1.2. This power manipulation can be realized by configuring the Power Amplifier (PA).

Regarding the format of IEEE 802.11a PHY frame, the Relay-SISO PHY frame sent by the source terminal, namely S frame, is proposed in Figure 3.9 where Short Training Field (STF) is the 10 short symbols, Long Training Field (LTF) is the 2 long symbols as defined in IEEE 802.11a. STF, LTF and the SIGNAL symbols are all transmitted with power factor π_{s1} .

STF	LTF	SIGNAL	Data A ₀	...	Data A _{2N_{CB}-1}	Idle field	Data B ₀	...	Data B _{2N_{CB}-1}
-----	-----	--------	------------------------	-----	--	------------	------------------------	-----	--

Figure 3.9: Relay-SISO S frame format

The SIGNAL uses the same format like IEEE 802.11a except the reserved bit³ is set to '1' in order to inform the relay and destination terminals the use of cooperation mode.

³By default the reserved bit is '0'.

The new data field consists of 3 parts: data A, idle field and data B. As defined in (3.2) and (3.3), data A and data B possess the same length. Since data B is transmitted after data A, the symbol streams $\underline{X}_{k,21}$ and $\underline{X}_{k,22}$ are memorized in the buffer when the data streams are encoded. A vacant period of N_I OFDM symbols is reserved for the idle field during which the relay terminal switches from reception mode to transmission mode and sends the additional preambles.

In this frame structure, $\{A_{2n,k}, A_{2n+1,k}, B_{2n,k}, B_{2n+1,k}\}$ are the symbols from the same codeword $\mathbf{X}_{n,k}$. This implies also the number of OFDM symbols should be a multiple of 4 and the number of QAM symbols should be a multiple of $4 \times 48 = 192$. Let N_{SYM} denote the number of OFDM data symbols, then the number of padding bits N_{PAD} is calculated as follows:

$$\begin{aligned} N_{\text{SYM}} &\leftarrow \lceil (16 + 8 \times \text{LENGTH} + 6) / N_{\text{DBPS}} \rceil \\ N_{\text{CB}} &\leftarrow \lceil N_{\text{SYM}} / 4 \rceil \\ N_{\text{SYM}} &\leftarrow 4 \times N_{\text{CB}} \\ N_{\text{PAD}} &\leftarrow N_{\text{SYM}} \times N_{\text{DBPS}} - (16 + 8 \times \text{LENGTH} + 6) \end{aligned} \quad (3.4)$$

where $\lceil x \rceil$ denotes the smallest interger greater than or equal to x . N_{DBPS} has been already defined in Table 3.1 and the 16-bit service sequence and the 6-bit tail for convolutional coding are both taken into account in the calculation. The LENGTH parameter denotes the number of bytes in the sending MPDU. Based on the LENGTH parameter, the PHY will calculate the frame's duration which is reported to MAC layer to avoid collision. However, for the same length of MPDU we need to add more padding bits to fill the CB in some cases. Besides the additional N_I idle OFDM symbols should also be taken into account. This means the cooperation frame is longer than the legacy IEEE 802.11a frame who is bearing the same quantity of information. Thus, here emerges a compatibility problem between the Relay-SISO terminal and the legacy IEEE 802.11a terminals.

Inspired by the mixed mode of IEEE 802.11n, a practicable way is to modify the LENGTH parameter to inform legacy terminals the real duration of the S frame. Considering in worst case 3 extra OFDM symbols will be added, the LENGTH parameter is recalculated by:

$$\text{LENGTH} \leftarrow \text{LENGTH} + N_{\text{DBPS}}(N_I + 3) / 8 \quad (3.5)$$

The LENGTH is increased by $N_{\text{DBPS}}(N_I + 3) / 8$, which is the number of bytes that can be transmitted during the idle OFDM symbols and 3 extra OFDM symbols using the same modulation and coding rate. The legacy terminals can update their Network Allocation Vector (NAV) regarding the LENGTH value although they do not interpret the data field. At the relay terminal and the destination terminal, this add-on value will be removed to get the real length.

For the relay terminal, we suppose in the network one terminal is selected to do the Amplify-and-Forward (AF) task. When the relay terminal receives the S frame, it stores the long symbols field of PLCP preamble and calculates the length of data A. The received signals are memorized and then will be amplified by applying the normalization factor b . This normalization factor is obtained by estimating the power of the received signal.

Due to the half-duplex constraint, the relay terminal needs some time, known as RX-to-TX turnaround time $\sim 2\mu\text{s}$ [27], to switch from reception mode to transmission mode.

Since the IEEE 802.11a PHY frame is slotted into OFDM symbols of duration $4\mu s$, we delay the relay's amplification by one OFDM symbol to ensure the time synchronization at the destination. This delay is the reason that we propose to group the OFDM symbols by data A and data B instead of transmitting the CBs in one-by-one manner: in that case at least one OFDM symbol is demanded for each CB.

The PHY frame format transmitted by the relay terminal is illustrated in Figure 3.10, noted by R frame.

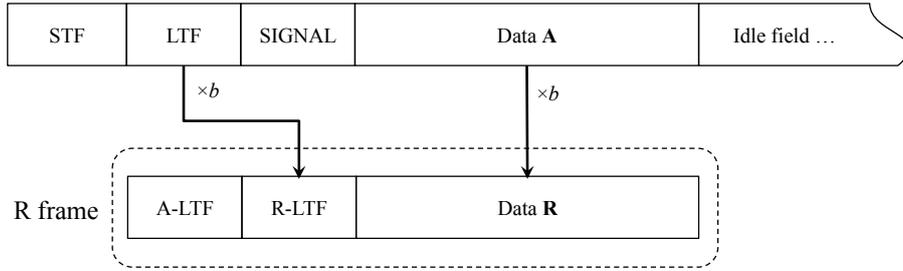


Figure 3.10: Relay-SISO R frame format

The R frame is sent with power π_{r2} . Extra preambles for the cooperation channel training are transmitted before the relayed data R, which is the amplified data A. The preamble Amplified Long Training Field (A-LTF) is the relay-generated LTF while Relayed Long Training Field (R-LTF) is the amplified LTF of S frame. The relay preambles are sent during the idle field of S frame such that the data R is temporally synchronized with the transmission of data B.

The destination terminal receives the combination of S frame and R frame, noted by D frame. We propose that the idle field occupies $N_1 = 5$ OFDM time symbols that 1 idle symbol for the turnaround time of relay terminal, 2 symbols for A-LTF and 2 symbols R-LTF. In order to enhance the CFO estimation accuracy, we suggest that the source terminal sends half STF which contains 5 short symbols between data A and A-LTF, as shown in Figure 3.11.

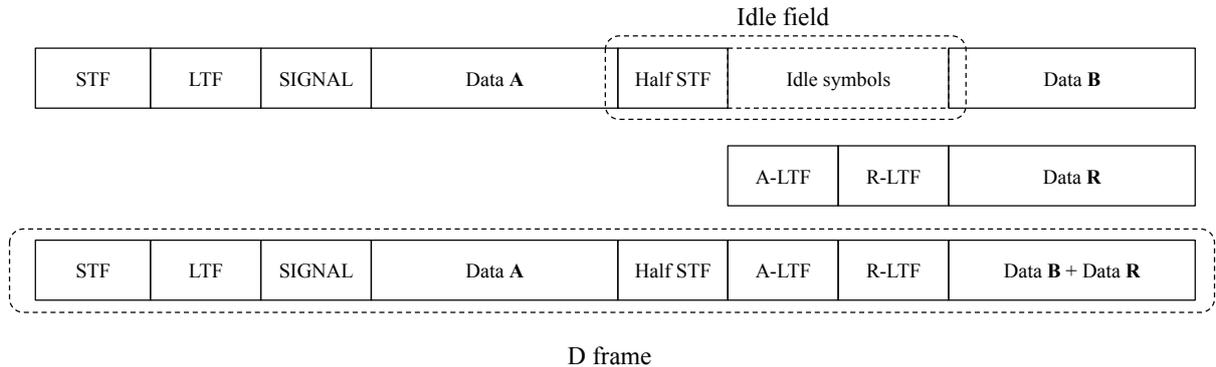


Figure 3.11: Relay-SISO D frame format

3.3 Relay-SISO preambles and cooperation procedure

In this section, the operation $[\underline{x} \star \underline{y}]_N$ denotes the N -length circular discrete convolution of sequence \underline{x} and \underline{y} . Without specifications, N is 64.

The operation $[\overline{X} \cdot \overline{Y}]$ or $[\underline{x} \cdot \underline{y}]$ denotes the element-by-element multiplication of vectors \overline{X} and \overline{Y} or row-vectors \underline{x} and \underline{y} .

For the notations, the time domain signal is denoted by a row-vector \underline{x} and \overline{X} denotes the N -point *unitary* Discrete Fourier Transform (DFT) of \underline{x} such that:

$$\overline{X} = \frac{1}{\sqrt{N}} \text{DFT}_N(\underline{x}) \quad (3.6)$$

By Parseval's theorem, we have $\|\overline{X}\|^2 = \|\underline{x}\|^2$.

The impulsional response of channel⁴ $p \rightarrow q$ is denoted by \underline{h}_{pq} which is sampled at 20MHz. The \overline{H}_{pq} is the corresponding frequency response which is the DFT of \underline{h}_{pq} :

$$\overline{H} = \text{DFT}_N(\underline{h}) \quad (3.7)$$

We notice that $\|\overline{H}\|^2 = N\|\underline{h}\|^2$.

We suppose the perfect time synchronization for the OFDM symbols that we do not need to consider the signal during the GI. For the signal within an OFDM symbol, we have:

$$[\underline{x} \star \underline{h}] = \overline{H} \cdot \overline{X} \quad (3.8)$$

where \underline{x} denotes the signal and \underline{h} denotes the channel response.

Considering there are totally 52 sub-carriers in use, the transmitted symbols are normalized in frequency domain by $K = 8/\sqrt{52}$ in order to have a power 1 in time domain.

Let \underline{h} and \overline{H} denote the channel response in time domain and frequency domain respectively. The channel coefficients on non-used sub-carriers have no effect on the data or training symbols. So the channel frequency responses are equivalent to $\overline{H}' = \{H'_k\}$ which is defined by:

$$\begin{cases} H'_k = KH_k, & k \text{ is a used sub-carrier} \\ H'_k = 0, & k \text{ is not a used sub-carrier} \end{cases} \quad (3.9)$$

Here K is considered as a part of the channel coefficient.

For the sake of simplicity, we use $\underline{h}' = \text{IDFT}_N(\overline{H}')$ to denote the channel impulsional response instead of \underline{h} .

3.3.1 Data field

Following the notation in section 1.2, we use letter x for the sending signal and y for the received signal. v, w stand for the AWGN terms at the relay terminal and destination terminal, respectively.

⁴The length of any channel's impulsional response is supposed less than GI.

For the OFDM symbol of data A, we have:

$$\begin{cases} \underline{y}_1^{(r)} = \sqrt{\pi_{s1}} \left[\underline{h}_{sr} \star \underline{x}_1^{(s)} \right] + \underline{v}_1 \\ \underline{y}_1^{(d)} = \sqrt{\pi_{s1}} \left[\underline{h}_{sd} \star \underline{x}_1^{(s)} \right] + \underline{w}_1 \end{cases} \quad (3.10)$$

which in frequency domain is expressed by:

$$\begin{cases} \bar{Y}_1^{(r)} = \sqrt{\pi_{s1}} \bar{H}_{sr} \cdot \bar{X}_1^{(s)} + \bar{V}_1 \\ \bar{Y}_1^{(d)} = \sqrt{\pi_{s1}} \bar{H}_{sd} \cdot \bar{X}_1^{(s)} + \bar{W}_1 \end{cases} \quad (3.11)$$

For the OFDM symbol of data B, the received signal can also be represented in frequency domain by:

$$\bar{Y}_2^{(d)} = \sqrt{\pi_{s2}} \bar{H}_{sd} \cdot \bar{X}_2^{(s)} + \sqrt{\pi_{r2}} \bar{H}_{rd} \cdot \bar{X}_2^{(r)} + \bar{W}_2 \quad (3.12)$$

where

$$\bar{X}_2^{(r)} = b \bar{Y}_1^{(r)} \quad (3.13)$$

Regrouping $\{A_{2n,k}, A_{2n+1,k}, B_{2n,k}, B_{2n+1,k}\}$ gives:

$$\mathbf{Y}_{n,k} = \mathbf{H}_k \mathbf{X}_{n,k} + \mathbf{V}_{n,k} + \mathbf{W}_{n,k} \quad (3.14)$$

where

$$\begin{cases} \mathbf{Y}_{n,k} = \begin{bmatrix} Y_{n,k,11} & Y_{n,k,12} \\ Y_{n,k,21} & Y_{n,k,22} \end{bmatrix} \\ \mathbf{H}_k = \begin{bmatrix} \sqrt{\pi_{s1}} H_{k,sd} & 0 \\ \sqrt{\pi_{s1} \pi_{r2}} b H_{k,rd} H_{k,sr} & \sqrt{\pi_{s2}} H_{k,sd} \end{bmatrix} \\ \mathbf{X}_{n,k} = \begin{bmatrix} X_{n,k,11} & X_{n,k,12} \\ X_{n,k,21} & X_{n,k,22} \end{bmatrix} \\ \mathbf{V}_{n,k} = \begin{bmatrix} 0 & 0 \\ \sqrt{\pi_{r2}} b H_{k,rd} V_{n,k,21} & \sqrt{\pi_{r2}} b H_{k,rd} V_{n,k,22} \end{bmatrix} \\ \mathbf{W}_{n,k} = \begin{bmatrix} W_{n,k,11} & W_{n,k,12} \\ W_{n,k,21} & W_{n,k,22} \end{bmatrix} \end{cases} \quad (3.15)$$

On each data sub-carrier, the system is a single-relay single-antenna NAF cooperation system. The problems are: first how to frequently synchronize the three terminals; secondly how estimate the CSI, including the channel coefficients and the noise power. To give the answer to these questions, new preambles for cooperation are introduced and we will discuss the use of these preambles as follows.

3.3.2 Carrier frequency offset estimation and correction

We suppose the channel response is a dirac function in the context of CFO estimation. The estimation and correction processing can be extended to multipath channel cases.

Let $x_{\text{BB}}(t)$ denote the transmitted baseband signal. The transmitted signal in Radio Frequency (RF) is given by:

$$x_{\text{RF}}(t) = \text{Re} \left(e^{2j\pi ft} x_{\text{BB}}(t) \right) \quad (3.16)$$

where f is the carrier frequency at transmitter's side.

We omit the noise term at the receiver. The received RF signal is demodulated with the receiver's carrier frequency f' :

$$y(t) = e^{-j(2\pi f' t + \phi_0)} x_{\text{RF}}(t) = \frac{1}{2} \left[e^{-j(2\pi \Delta f t + \phi_0)} + e^{-j(2\pi(f'+f)t + \phi_0)} \right] x_{\text{BB}}(t) \quad (3.17)$$

where $\Delta f = f' - f$ and ϕ_0 denotes the initial phase.

By applying Low-Pass Filter (LPF) with a gain of 2, the received based signal is given by:

$$y_{\text{BB}}(t) = e^{-j(2\pi \Delta f t + \phi_0)} x_{\text{BB}}(t) \quad (3.18)$$

The Δf is the CFO which needs to be estimated and removed in time domain.

In the single-relay cooperation system, let f_j denote the local carrier frequency of terminal j . We use the one of the source terminal f_s as the reference so the destination CFO is defined by $\Delta f_d = f_d - f_s$ and relay CFO is defined by $\Delta f_r = f_r - f_s$.

Both the relay terminal and the destination terminal will estimate Δf_d based on observation of STF and LTF. Without loss of generality, let us see the CFO estimation at the destination side. The transmitted signal is indicated by TX and the received signal is indicated by RX.

We propose to estimate the CFO in two steps with STF (short symbols) and LTF (long symbols) respectively. The STF is composed of 10 repetitions of the short symbol which represents 16 samples. The first CFO estimation is based on the 4 consecutive short symbols which are denoted by $\underline{s} = \{s_0, \dots, s_{63}\}$. Considering the presence of Δf_d , the received \underline{s} can be given by:

$$\underline{s}^{(\text{RX})} = \sqrt{\pi_{s1}} [\underline{s}^{(\text{TX})} \star \underline{h}^{(sd)}] \cdot \underline{\xi} + \underline{w} \quad (3.19)$$

with

$$\begin{cases} \underline{\xi} = \{e^{j\phi_k}\}, & k = 0, \dots, 63 \\ \phi_k = k\Delta\phi - \phi_0 \\ \Delta\phi = -2\pi\Delta f_d/f_s \end{cases} \quad (3.20)$$

where ϕ_0 is the initial phase, $\Delta\phi$ is the CFO phase error increment per sample and f_s the sampling frequency which is set to 20MHz.

The CFO estimation is equal to estimate the $\Delta\phi$. The STF based estimation is given by:

$$\Delta\hat{\phi}_{\text{STF}} = \frac{1}{32} \text{Arg} \left(\sum_{k=0}^{31} s_{k+32}^{(\text{RX})} s_k^{*(\text{RX})} \right) \quad (3.21)$$

The estimated $\Delta\hat{\phi}_{\text{STF}}$ is used to correct the phase error of LTF for channel coefficients' estimation.

The second CFO estimation is based on the 2 corrected long symbols which contain 128 samples. Let $\underline{l}_i = \{l_{i,0}, \dots, l_{i,63}\}$ denote i^{th} long symbol. In the same way, this estimation is:

$$\Delta\hat{\phi}_{\text{LTF}} = \frac{1}{64} \text{Arg} \left(\sum_{k=0}^{63} l_{2,k}^{(\text{RX})} l_{1,k}^{*(\text{RX})} \right) \quad (3.22)$$

The final CFO estimation result is the combination of $\Delta\hat{\phi}_{\text{STF}}$ and $\Delta\hat{\phi}_{\text{LTF}}$ such:

$$\Delta\hat{\phi} = \rho\Delta\hat{\phi}_{\text{STF}} + (1 - \rho)\Delta\hat{\phi}_{\text{LTF}} \quad (3.23)$$

where $\rho = 0.25$.

Let $\Delta\hat{\phi}_d$ be the estimation result of the destination and $\Delta\hat{\phi}_r$ be the one of the relay. At the destination's side, $\Delta\hat{\phi}_d$ is used to correct the signals behind LTF, including SIGNAL and the data field with cooperation preambles.

Let $\underline{y} = \{y_k\}$ denote the received signal from the beginning of SIGNAL til the end of data B(R), this correction can be described by:

$$y_k \leftarrow y_k e^{-jk\Delta\hat{\phi}_d} \quad (3.24)$$

At the relay's side, the SIGNAL symbol is corrected in the same way as (3.24) in using $\Delta\hat{\phi}_d$. However, the CFO correction is not necessary for the R-LTF and the data R. From (3.18), the received baseband signal (without noise) is given by:

$$y_{\text{BB}}^{(r)}(t) = e^{-j(2\pi\Delta f_r t + \phi_{0,r})} x_{\text{BB}}^{(s)}(t) \quad (3.25)$$

The amplification by b gives:

$$x_{\text{BB}}^{(r)}(t) = b y_{\text{BB}}^{(r)}(t) \quad (3.26)$$

The amplified signal is received by the destination terminal:

$$y_{\text{BB}}^{(d)} = e^{-j(2\pi(f_d - f_r)t + \phi_{0,d})} x_{\text{BB}}^{(r)}(t) = b e^{-j(\pi\Delta f_d t + \phi_{0,r} + \phi_{0,d})} x_{\text{BB}}^{(s)}(t) \quad (3.27)$$

We see that Δf_r is transparent to the destination terminal.

In fact, the field needs to be corrected is the relay-generated A-LTF which are sent with the relay's carrier frequency f_r .

Let $\underline{a} = \{a_k\}$ denote the transmit signal during A-LTF. This field are phase corrected by:

$$a_k \leftarrow a_k e^{jk\Delta\hat{\phi}_r} \quad (3.28)$$

Unlike the CFO correction in (3.24), this correction is performed to synchronize the carrier frequency of the relay transmitted signal to the one of source terminal. This processing helps the destination terminal to estimate its CFO without complicating the calculation.

For the residual CFO, both the relay terminal and the destination terminal can refine their estimation results in using the pilot sub-carriers⁵. For readers' convenience, we suppose the perfect CFO correction at both relay terminal and destination terminal in the following part.

⁵For the residual CFO tracking at the relay terminal, we distill only the pilot sub-carriers instead of the full frequency OFDM symbol.

3.3.3 Amplify-and-forward procedure

The relay terminal receives and retransmits the LTF and data A with amplification. As shown in the baseband model, the relay terminal calculates the amplify factor b in order to normalize the transmission power.

The received signal's power can be examined with the 2 long symbols of LTF. In frequency domain by, each transmitted long symbol can be expressed:

$$\bar{L} = [L_0, \dots, L_{63}]^T \quad (3.29)$$

As described in [27], L_k takes the value of +1 or -1 at the 52 used sub-carriers, otherwise $L_k = 0$.

The LTF is transmitted with power factor π_{s1} and the relay-received two long symbols are given by:

$$\underline{l}_i^{(\text{RX},r)} = \sqrt{\pi_{s1}} [\underline{l} \star \underline{h}_{sr}] + \underline{v}_i, \quad i = 1, 2 \quad (3.30)$$

where i is the index of long symbol and $\underline{v}_t = \{v_{k,t}\}$ represents the time domain noise that $v_{k,t} \sim \mathcal{CN}(0, N_0)$.

The received signal power is given by:

$$P = \pi_{s1} \|\underline{h}_{sr}\|^2 + N_0 \quad (3.31)$$

We see that:

$$\text{E} \left\{ \left\| \underline{l}_t^{(\text{RX},r)} \right\|^2 \right\} = \pi_{s1} \|\underline{l} \star \underline{h}_{sr}\|^2 + 64N_0 \quad (3.32)$$

where

$$\|\underline{l} \star \underline{h}_{sr}\|^2 = \|\bar{L} \cdot \bar{H}_{sr}\|^2 = \|\bar{H}_{sr}\|^2 = 64h_{sr} \quad (3.33)$$

The received signal power is estimated with the two long symbols by:

$$\hat{P} = \frac{1}{128} \left(\left\| \underline{l}_1^{(\text{RX},r)} \right\|^2 + \left\| \underline{l}_2^{(\text{RX},r)} \right\|^2 \right) \quad (3.34)$$

The estimation result is unbiased that:

$$\text{E} \left\{ \hat{P} \right\} = \pi_{s1} \|\underline{h}_{sr}\|^2 + N_0 \quad (3.35)$$

The normalization factor is calculated by:

$$b = \frac{1}{\sqrt{\hat{P}}} \quad (3.36)$$

The A-LTF consists of the two long symbols, denoted by \underline{a}_1 and \underline{a}_2 . This field is sent with power π_{r2} that:

$$\underline{a}_i^{(\text{TX},r)} = \sqrt{\pi_{r2}} \underline{l}, \quad i = 1, 2 \quad (3.37)$$

In fact, the normalization can be made on each sub-carrier for a better transmission spectrum. We use the same b on all the sub-carriers in order to simplify the amplify-and-forward procedure since the normalization can be done in time domain by scalar production.

Let \underline{r}_1 and \underline{r}_2 denote the two relayed long symbols of R-LTF. This field is also sent with power π_{r2} after normalization that:

$$\underline{r}_i^{(\text{TX},r)} = b\sqrt{\pi_{r2}}\underline{l}_i^{(\text{RX},r)} = b\sqrt{\pi_{s1}\pi_{r2}}[\underline{l} \star \underline{h}_{sr}] + b\sqrt{\pi_{r2}}\underline{v}_i, \quad i = 1, 2 \quad (3.38)$$

3.3.4 Channel coefficients estimation

As shown in the baseband model, to reform the MIMO-like system on each data subcarrier, we need the coefficients of the channel $s \rightarrow d$, noted by $H_{k,sd}$, the channel $s \rightarrow r$, noted by $H_{k,sr}$ and the channel $r \rightarrow d$, noted by $H_{k,rd}$ with k the frequency (sub-carrier) index. Besides, the noise power at both the relay terminal and the destination terminal are needed for the equivalent channel matrix.

These frequency parameters in Relay-SISO are estimated with the proposed preamble:

1. Use LTF to estimate the channel coefficients $\sqrt{\pi_{s1}}\overline{H}_{sd}$.
2. Use the preceding result to estimate the channel coefficients of data B, $\sqrt{\pi_{s2}}\overline{H}_{sd}$.
3. Use A-LTF to estimate the channel coefficients $\sqrt{\pi_{r2}}\overline{H}_{sr}$ and the power of destination noise $\sigma_w^2 = N_0$.
4. Use R-LTF to estimate the product channel coefficients of data R, $\sqrt{\pi_{s1}\pi_{r2}}b(\overline{H}_{rd} \cdot \overline{H}_{sr})$ and to estimate the total noise energy (introduced by the relay and the destination), noted by E_{v+w} .
5. Use the results above to estimate the relay amplified noise power in order to evaluate the noise normalisation factor $\underline{\rho} = \{\rho_k\}$ for each data sub-carrier.

This procedure is illustrated in Figure 3.12.

Details are provided in the following where \hat{x} denotes the estimation result of x .

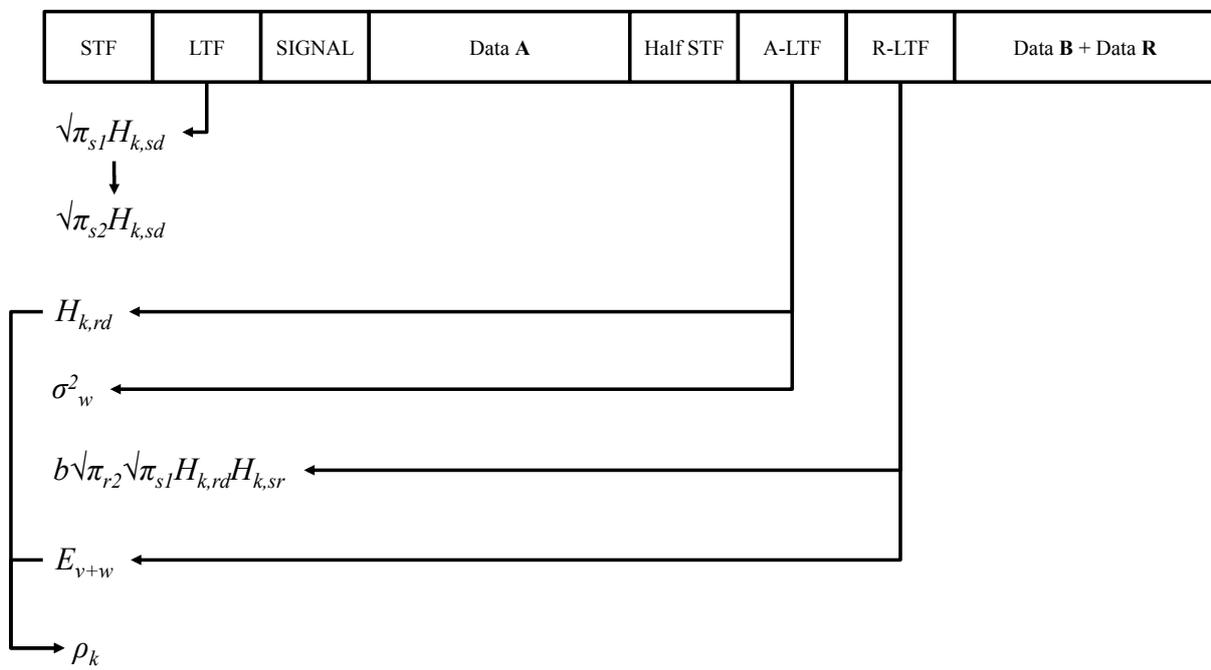


Figure 3.12: Relay-SISO channel parameters estimation

LTF

The coefficients of the channel $s \rightarrow d$ are estimated like in IEEE 802.11a case, where we suppose the channel's impulsive response \underline{h}_{sd} is shorter than GI. The received long symbols can be given by:

$$\underline{l}_i^{(\text{RX},d)} = \sqrt{\pi_{s1}}[\underline{l} \star \underline{h}_{sd}] + \underline{w}_i, \quad i = 1, 2 \quad (3.39)$$

where $\underline{w} = \{w_k\}$ represents the destination noise that $w_k \sim \mathcal{CN}(0, N_0)$.

The channel coefficients $\sqrt{\pi_{s1}}\overline{H}_{sd}$ are estimated using the average of the two received long symbols by:

$$\hat{\underline{l}}^{(\text{RX},d)} = \frac{1}{2} \left(\underline{l}_1^{(\text{RX},d)} + \underline{l}_2^{(\text{RX},d)} \right) \quad (3.40)$$

Considering \overline{L} is the unitary DFT of \underline{l} , we have:

$$\sqrt{\pi_{s1}}\hat{\overline{H}}_{sd} = \hat{\underline{L}}^{(\text{RX},d)} \cdot (\overline{L})^{-1} \quad (3.41)$$

where $(\overline{L})^{-1} = \{L_k^{-1}\}$ for non-zero sub-carriers.

The power factor π_{s1} and π_{s2} are known so the channel coefficients of data B are calculated by:

$$\sqrt{\pi_{s2}}\hat{\overline{H}}_{sd} = \sqrt{\frac{\pi_{s2}}{\pi_{s1}}} \sqrt{\pi_{s1}}\hat{\overline{H}}_{sd} \quad (3.42)$$

A-LTF

The channel coefficients $\sqrt{\pi_{r2}}\overline{H}_{rd}$ are estimated using A-LTF in the same way. The received 2 long training symbols of A-LTF are:

$$\underline{a}_i^{(\text{RX},d)} = \sqrt{\pi_{r2}}[\underline{l} \star \underline{h}_{rd}] + \underline{w}_i, \quad i = 1, 2 \quad (3.43)$$

The channel coefficients are given by:

$$\sqrt{\pi_{r2}}\hat{\overline{H}}_{rd} = \hat{\underline{A}}^{(\text{RX},d)} \cdot (\overline{L})^{-1} \quad (3.44)$$

where $\hat{\underline{A}}^{(\text{RX},d)}$ is the DFT of $\hat{\underline{a}}^{(\text{RX},d)}$ which is defined by:

$$\hat{\underline{a}}^{(\text{RX},d)} = \frac{1}{2} \left(\underline{a}_1^{(\text{RX},d)} + \underline{a}_2^{(\text{RX},d)} \right) \quad (3.45)$$

The $r \rightarrow d$ channel coefficients will be used to calculate the noise power normalization factor $\overline{\rho} = \{\rho_k\}$. In addition, the A-LTF is also used to estimate the noise power N_0 . Considering that:

$$\text{E} \left\{ \left\| \underline{a}_1^{(\text{RX},d)} - \underline{a}_2^{(\text{RX},d)} \right\|^2 \right\} = 128N_0 \quad (3.46)$$

N_0 is estimated by:

$$\hat{N}_0 = \frac{1}{128} \left\| \underline{a}_1^{(\text{RX},d)} - \underline{a}_2^{(\text{RX},d)} \right\|^2 \quad (3.47)$$

We define the the energy of channel $r \rightarrow d$ by:

$$E_H = 64\pi_{r2} \|\underline{h}_{rd}\|^2 \quad (3.48)$$

E_H is estimated by:

$$\hat{E}_H = \left\| \hat{\underline{a}}^{(\text{RX},d)} \right\|^2 - 32\hat{N}_0 \quad (3.49)$$

This estimated results come from the fact that:

$$\hat{\underline{a}}^{(\text{RX},d)} = \sqrt{\pi_{r2}} [\underline{l} \star \underline{h}_{rd}] + \frac{1}{2} (\underline{w}_1 + \underline{w}_2) \quad (3.50)$$

We see that:

$$\text{E} \left\{ \left\| \hat{\underline{a}}^{(\text{RX},d)} \right\|^2 \right\} = 64\pi_{r2} \|\underline{h}_{rd}\|^2 + 32N_0 \quad (3.51)$$

R-LTF

From (3.38), the received two long symbols of R-LTF are given by:

$$\underline{r}_i^{(\text{RX},d)} = b\sqrt{\pi_{s1}\pi_{r2}} [\underline{l} \star \underline{h}_{rd} \star \underline{h}_{sr}] + b\sqrt{\pi_{r2}} [\underline{v}_i \star \underline{h}_{rd}] + \underline{w}_i, \quad i = 1, 2 \quad (3.52)$$

The estimation of the product channel $b\sqrt{\pi_{s1}\pi_{r2}} (\overline{H}_{rd} \cdot \overline{H}_{sr})$ is based on the average of received symbols:

$$\hat{\underline{r}}^{(\text{RX},d)} = \frac{1}{2} \left(\underline{r}_1^{(\text{RX},d)} + \underline{r}_2^{(\text{RX},d)} \right) \quad (3.53)$$

and

$$b\sqrt{\pi_{s1}\pi_{r2}} \left(\hat{\overline{H}}_{rd} \cdot \hat{\overline{H}}_{sr} \right) = \hat{\overline{R}}^{(\text{RX},d)} \cdot (\overline{L})^{-1} \quad (3.54)$$

Let us denote the sum of noise energy of 64 samples by E_{v+w} which is defined by:

$$\begin{aligned} E_{v+w} &= \text{E} \left\{ \|b\sqrt{\pi_{r2}} [\underline{h}_{rd} \star \underline{v}] + \underline{w}\|^2 \right\} \\ &= (b^2 E_H + 64)N_0 \end{aligned} \quad (3.55)$$

Like (3.47), E_{v+w} is estimated by:

$$\hat{E}_{v+w} = \frac{1}{2} \left\| \underline{r}_1^{(rd)} - \underline{r}_2^{(rd)} \right\|^2 \quad (3.56)$$

For AWGN normalization, the destination terminal estimates the amplification factor b with \hat{N}_0 , \hat{E}_H and the noise term \hat{E}_{v+w} :

$$\hat{b}^2 = \frac{1}{\hat{E}_H \hat{N}_0} \left(\hat{E}_{v+w} - 64\hat{N}_0 \right) \quad (3.57)$$

The normalization factors $\hat{\rho}$ are then calculated on each sub-carrier k by:

$$\hat{\rho}_k = \frac{1}{\sqrt{1 + \left(\pi_{r2} \left| \hat{H}_{k,rd} \right|^2 \right) \hat{b}^2}} \quad (3.58)$$

where the channel coefficients $\sqrt{\pi_{r2}}\hat{H}_{rd}$ have been estimated precedently in (3.44). In practice, the noise power can be estimated with error that the relay amplification factor is actually given by:

$$\hat{b}^2 = \frac{1}{\hat{E}_H \hat{N}_0} \text{Max} \left(0, \hat{E}_{v+w} - 64\hat{N}_0 \right) \quad (3.59)$$

where $\text{Max}(x, y)$ denotes the maximum among x and y .

3.3.5 Cooperation procedure

For the Relay-SISO terminals, the cooperation procedure is guided by the MAC layer for the transmission scheduling. The source terminal will active cooperation mode when the cooperation mode is supported at the destination. Figure 3.13 gives a general processing diagram of Relay-SISO PHY layer at the source terminal.

At the other hand, the reception procedure is described in Figure 3.14⁶. This reception procedure is for both the relay terminal and the destination terminal. The MAC layer should indicate the PHY layer with “AF MODE” the mode of reception: “AF MODE” is enabled for the relay terminal and it is disabled for the destination terminal.

⁶At present, the half-STF is not used.

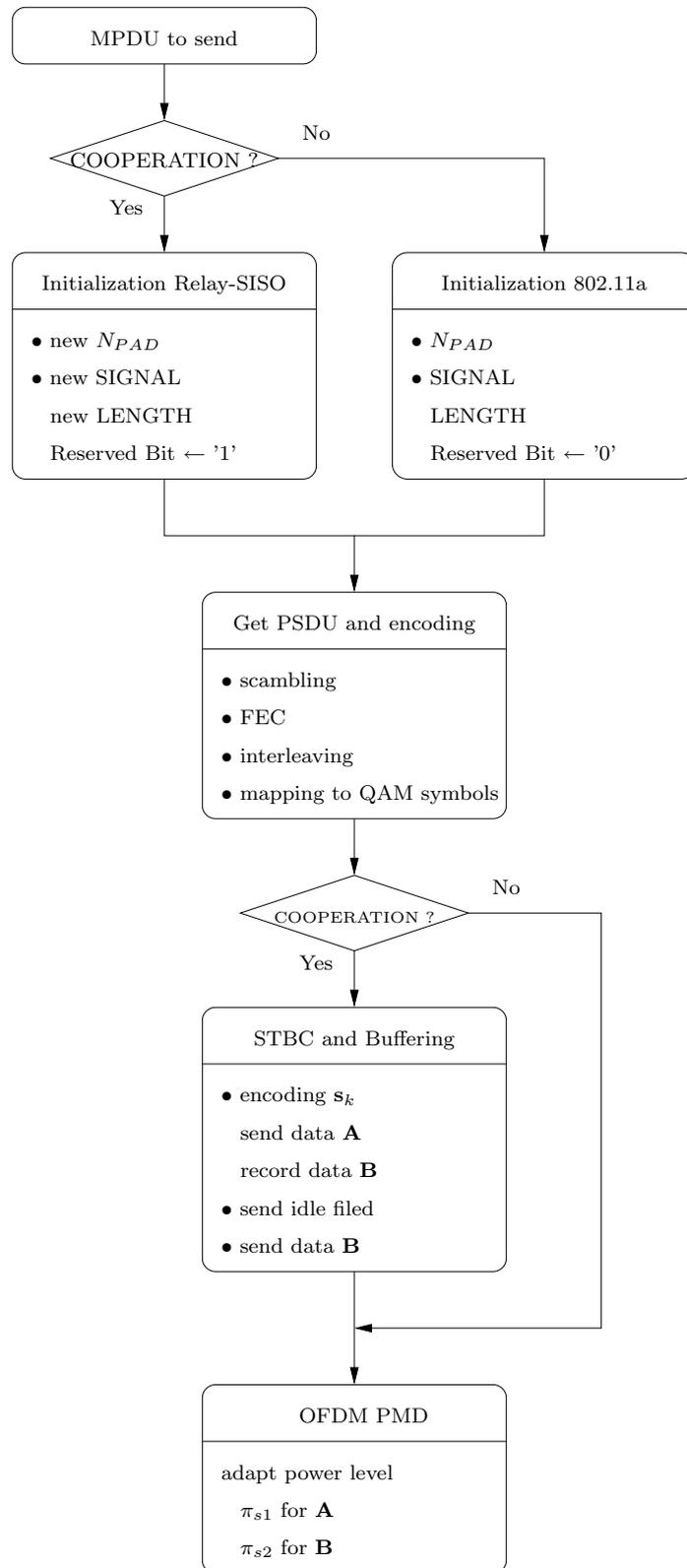


Figure 3.13: Relay-SISO PHY layer transmission procedure

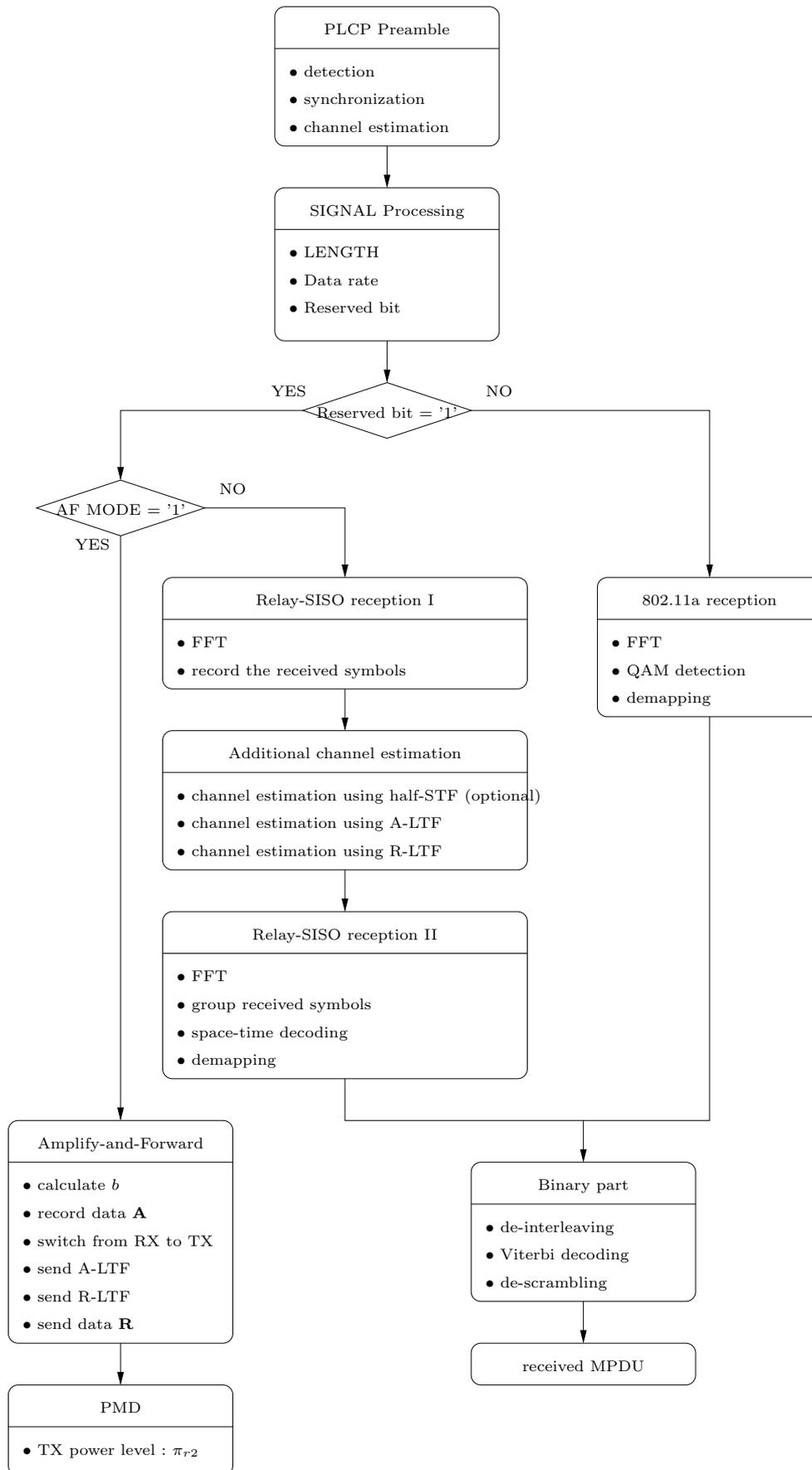


Figure 3.14: Relay-SISO PHY layer reception procedure

3.4 Relay-SISO Performance

The Frame Error Rate (FER) performances of Relay-SISO and IEEE 802.11a are simulated at data rate 12Mbps and 24Mbps with a MPDU=125 bytes. For the MIMO decoder issue, we chose the combination of GO and KSE algorithm with $K = 8$. This decoder achieves quasi-ML performance in cooperation mode. At the relay terminal, we take into account of a random delay due to the radio transmission which is uniformly distributed between $0\mu\text{s}$ and $0.1\mu\text{s}$. Both the flat fading channel model and the WLAN multipath channel are evaluated.

The multipath channel is modeled by several trajectories: each trajectory is characterized by its arrival time τ_i and its impulsional response h_i , where h_i is a complex random variable that $h_i \sim \mathcal{CN}(0, \sigma_i^2)$. The σ_i^2 is the power of each trajectory which are normalized in using the power of the first (direct) trajectory. In multipath channel case, SNR is defined by:

$$\text{SNR} = \frac{\sum_{i=1}^N \sigma_i^2}{N_0} \quad (3.60)$$

where N denotes the number of path and N_0 is the noise power.

We define the path gains of channel $s \rightarrow r$ and channel $r \rightarrow d$ by:

$$\begin{aligned} G_{sr}(\text{dB}) &= \text{SNR}_{sr}(\text{dB}) - \text{SNR}_{sd}(\text{dB}) \\ G_{rd}(\text{dB}) &= \text{SNR}_{rd}(\text{dB}) - \text{SNR}_{sd}(\text{dB}) \end{aligned} \quad (3.61)$$

In the simulations, two geographic configurations are considered: in the Low Path Gain (LPG) configuration, G_{sr} and G_{rd} are configured to 0dB to simulate the cooperative transmission in supposing that the direct transmission is not perturbed; in the High Path Gain (HPG) configuration, G_{sr} and G_{rd} are configured to 10dB in supposing that the direct transmission is severely perturbed. As explain in section 1.2, the power factor is chosen to meet the power condition $\pi_{s1} + \pi_{s2} + \pi_{r2} = 2$ and we choose $\pi_{s1} = 1$ and $\pi_{s2} = \pi_{r2} = 0.5$ in all the simulations in order to have a constant power during the transmission.

3.4.1 Flat fading channel

The flat fading channel is modeled as a simple Rayleigh fading channel with one trajectory. As shown in Figures 3.15 and 3.16, the performance is like the single carrier case, where “low” denotes LPG configuration and “high” denotes HPG configuration.

IEEE 802.11a system shows a FER performance of diversity 1 while Relay-SISO system gives a diversity gain of 2. At FER= 10^{-3} , Relay-SISO outperforms IEEE 802.11a: for data rate of 12Mbps, a system gain about 6dB in LPG configuration and about 11dB in HPG configuration; for data rate of 24Mbps, a little system gain in LPG configuration and about 6dB in HPG configuration.

The effecient throughput represents the successively transmitted data rate where the impact of padding bits and the preambles are neglected. For data rate of 12Mbps, the Relay-SISO in LPG configuration gives some improvement in throughput in medium and high SNR range.

However, when SNR is low there is no advantage of Relay-SISO. In all the configurations, there is a crossing point of Relay-SISO and IEEE 802.11a curves which can be considered as a SNR threshold. In the range where SNR is smaller than this threshold, the performance of IEEE 802.11a is better than Relay-SISO which means Relay-SISO system can outperform IEEE 802.11a only when the SNR is high enough. This phenomenon is more obvious in case of 24Mbps where the crossing point is found at $\text{SNR} \approx 27\text{dB}$ in HPG case and $\text{SNR} \approx 39\text{dB}$ in LPG case.

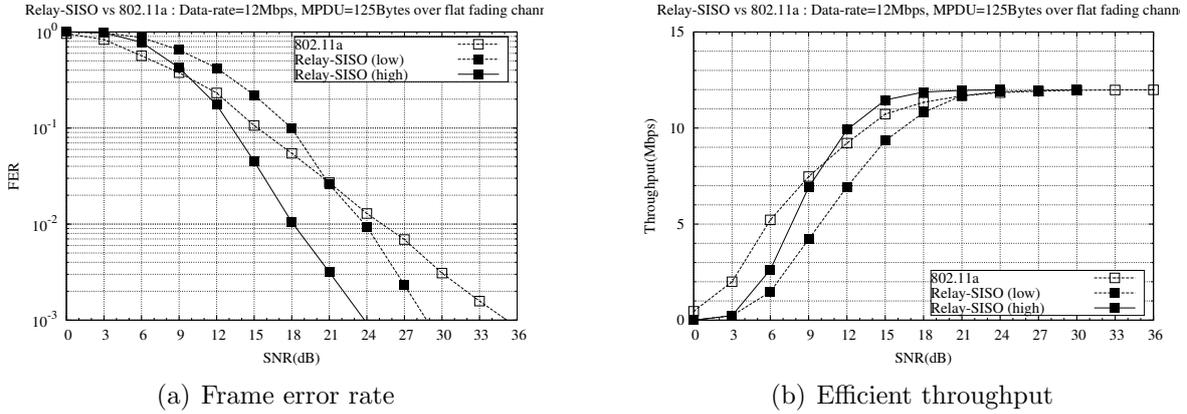


Figure 3.15: Relay-SISO and 802.11a over flat fading channel: each MPDU contains 125 bytes and it is sent at 12Mbps.

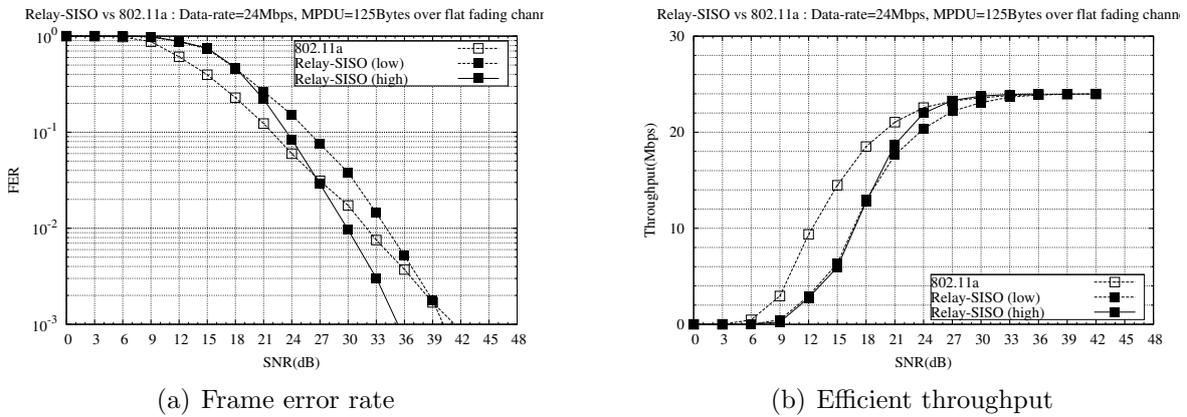


Figure 3.16: Relay-SISO and 802.11a over flat fading channel: each MPDU contains 125 bytes and it is sent at 24Mbps.

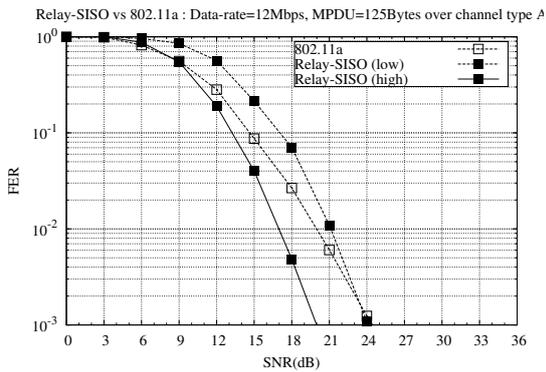
3.4.2 Multipath channel

We take the multipath channel type A in [28] as the multipath channel model. The parameters of this model are given in Table 3.2, where τ_i is the i^{th} path delay and σ_i^2 is the normalized path power.

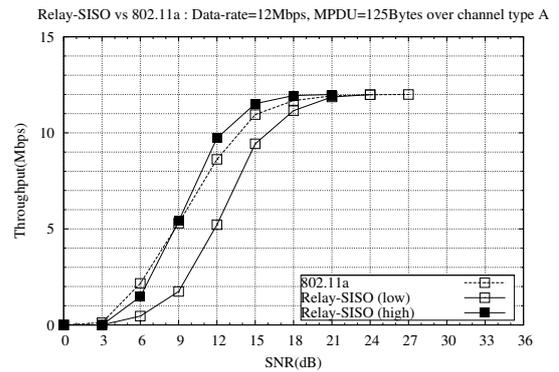
As shown in Figure 3.17 and 3.18, Relay-SISO shows a diversity gain greater than IEEE 802.11a's. At $\text{FER} = 10^{-3}$, Relay-SISO in HPG configuration brings a system gain about 4dB for 12Mbps which improves the efficient throughput. However in the other cases, the advantage of Relay-SISO is not observed.

Table 3.2: Multipath channel type A

τ (ns)	σ^2 (dB)
0	0.0
10	-0.9
20	-1.7
30	-2.6
40	-3.5
50	-4.3
60	-5.2
70	-6.1
80	-6.9
90	-7.8
110	-4.7
140	-7.3
170	-9.9
200	-12.5
240	-13.7
290	-18.0
340	-22.4
390	-26.7



(a) Frame error rate



(b) Efficient throughput

Figure 3.17: Relay-SISO and 802.11a over channel type A: each MPDU contains 125 bytes and it is sent at 12Mbps.

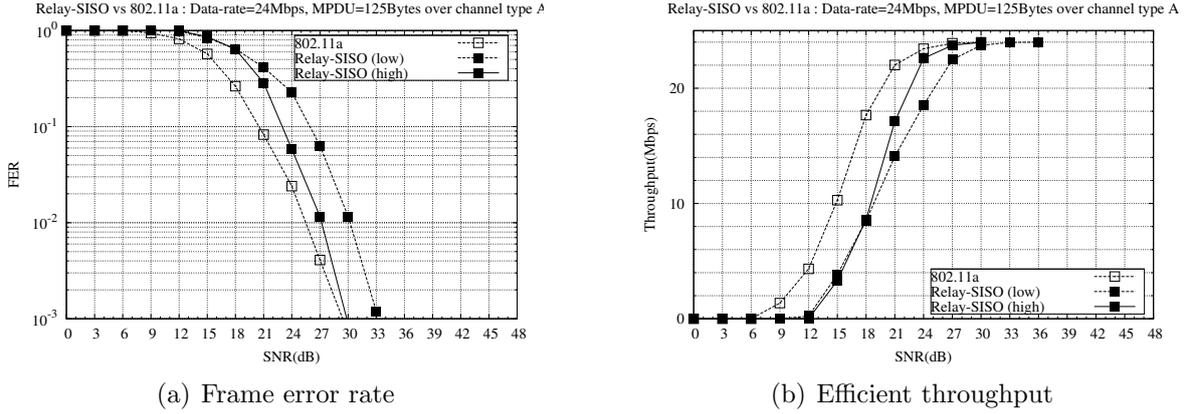


Figure 3.18: Relay-SISO and 802.11a over channel type A: each MPDU contains 125 bytes and it is sent at 24Mbps.

3.4.3 Discussion on performance

The proposed Relay-SISO PHY layer illustrates a significant cooperative diversity gain in flat fading channel as well as in multipath channel. In both channel environments, the performances of Relay-SISO are better than legacy 802.11a system with a considerable gain especially for low data rate and HPG configuration.

Application of cooperative diversity can enlarge the network coverage or alternatively allow a lower transmit power by providing good protection. This solution can be very interesting to enhance the system quality for the applications that are very demanding in terms of Quality of Service (QoS). For example the video broadcasting service needs a FER about 10^{-4} and Relay-SISO can be implemented to update the present WiFi networks with reasonable decoding complexity.

The inconvenience of Relay-SISO is that there is no significant gain in LPG configuration known as “crossing point” problem. This phenomenon is due to that in low SNR range the relay terminal amplifies the noise term that harms the reception quality. This problem can be the optimization works in the future.

CHAPTER 4

HYBRID COOPERATION

In chapter 3, a new PHY layer Relay-SISO employing an additional terminal as a relay node has been introduced. This single-relay single-antenna NAF system outperforms the legacy IEEE 802.11a in the moderate and high SNR region with HPG configuration. However, in the low-SNR region, Relay-SISO gives worse FER performance due to the fact that in low-SNR region, the relay terminal amplifies the noise. In that case, the NAF cooperation scenario is helpless or even destructive to the transmission quality.

We propose a new transmission strategy named Hybrid mode which is a combination of IEEE 802.11a system and Relay-SISO system. We suppose the channels are stable enough such that the performances of the cooperative and non-cooperative systems can be evaluated with a probing frame. It is clear that when Relay-SISO's performance is not better than IEEE 802.11a, we can disable the cooperation in order not to disturb the non-cooperative transmission. Our contribution in this chapter is that we realize a hybrid strategy in using a good metric to inform the source terminal the quality of cooperation channel.

4.1 Normalized minimum squared Euclidean distance

Let us first investigate the non-binary performance of a general MIMO system. We will give a metric in terms of the error probability.

Let \mathbf{Y} denote the received signal and let \mathbf{X} denote the transmitted codeword which is an element of the codewords ensemble \mathcal{X} . We use \mathbf{H} for the MIMO channel matrix and \mathbf{W} for the circular AWGN noise term.

The MIMO system is given by:

$$\mathbf{Y}_{N \times T} = \mathbf{H}_{N \times M} \mathbf{X}_{M \times T} + \mathbf{W}_{N \times T} \quad (4.1)$$

where $w_{ij} \sim \mathcal{CN}(0, N_0)$ and $\mathbb{E}\{\mathbf{W}\mathbf{W}^H\} = N_0 T \mathbf{I}_N$.

Let $\hat{\mathbf{X}}$ be the ML decoded codeword and the error probability of ML detection is expressed by:

$$P_e(\mathbf{H}, \mathcal{X}) = \Pr(\hat{\mathbf{X}} \neq \mathbf{X} \mid \mathbf{H}, \mathcal{X}) \quad (4.2)$$

In the communication systems, \mathcal{X} is a subset of an infinite linear code and we denote the infinite code by \mathcal{X}_{inf} . Instead of analyzing the error probability within \mathcal{X} , we will evaluate \mathcal{X}_{inf} that we investigate the Minimum Squared Euclidean Distance (MSED) of any two received codewords $d_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}})$.

Since \mathcal{X}_{inf} is linear, $d_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}})$ is given by:

$$d_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}}) = \min_{\mathbf{x} \in \mathcal{X}_{\text{inf}}, \mathbf{x} \neq \mathbf{0}} \|\mathbf{H}\mathbf{x}\|^2 \quad (4.3)$$

By normalizing this parameter in using the noise power, we define the Normalized Minimum Squared Euclidean Distance (NMSED) for a given channel \mathbf{H} and given code \mathcal{X}_{inf} by:

$$D_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}}) = \frac{d_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}})}{N_0} \quad (4.4)$$

The error probability is then approximated by:

$$P_e(\mathbf{H}, \mathcal{X}) \approx N_{\text{kiss}} Q\left(\sqrt{\frac{d_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}})}{2N_0}}\right) = N_{\text{kiss}} Q\left(\sqrt{\frac{D_{\text{min}}^2(\mathbf{H}, \mathcal{X}_{\text{inf}})}{2}}\right) \quad (4.5)$$

where Q -function is the normal cumulative distribution function and N_{kiss} is the kissing number that denotes the number of codewords having the same minimum distance.

Because the Q -function is mono-decreasing function, D_{min}^2 can be used as a metric in any MIMO systems when N_{kiss} is same. Bigger NMSED implies smaller P_e .

Now let us consider error probability in the non-cooperative and cooperative systems.

The noise power N_0 is supposed known as well as the channel matrix \mathbf{H} and codewords set \mathcal{X} .

The non-cooperative is a SISO system which is described by:

$$y = hx + w \quad (4.6)$$

where h is the channel coefficient and $w \sim \mathcal{CN}(0, N_0)$ is the noise term.

As detailed in chapter 2, the transmitted signal is:

$$x = K_{\text{mod}}(s + c) \quad (4.7)$$

where c is the offset such that $E\{x\} = 0$ and K_{mod} is the normalization factor of QAM symbol such that $E\{xx^*\} = 1$. s is a complex symbol in the constellation that $s \in \mathbb{Z}[i]$. An example of 16-QAM is illustrated in Figure 4.1.

It is easy to see the MSED in the constellation is 1, then MSED of the received symbols, denoted by $d_{\text{min},a}^2$ is given by:

$$d_{\text{min},a}^2 = K_{\text{mod}}^2 |h|^2 \quad (4.8)$$

In cooperative case, the system is equivalent to a MIMO system. We use the expression in (2.9) that the system is described by:

$$\bar{y}^r = \mathbf{H}_{\text{eq}}^r \bar{s}^r + \bar{w}^r \quad (4.9)$$

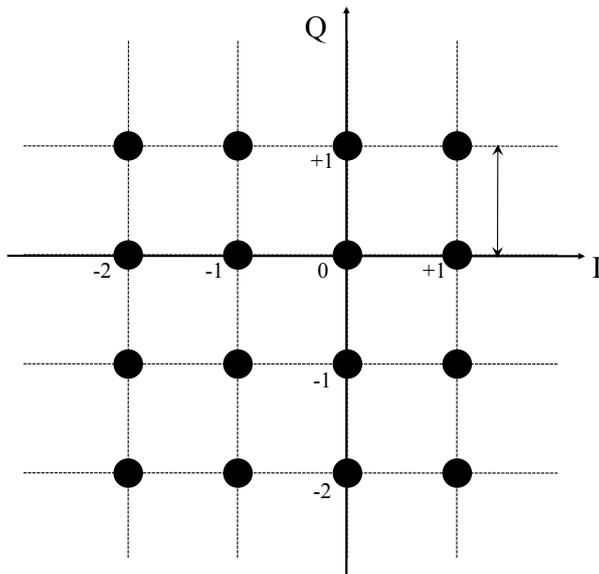


Figure 4.1: 16-QAM constellation

In the single-relay single-antenna NAF case, the \mathbf{H}_{eq}^r is a real matrix of dimension 8×8 . \bar{y}^r , \bar{s}^r and \bar{w}^r are all the real vectors with length 8. The noise term \bar{w}^r is AWGN that $\mathbb{E}\{\bar{w}^r \bar{w}^{rT}\} = 0.5N_0 2\mathbf{I}_8$.

Let $\mathcal{X} \subset \mathbb{Z}^8$ denote the set of constellation points that $\bar{s}^r \in \mathcal{X}$. We use $\Lambda_{\text{inf}}(\mathbf{H}_{\text{eq}}^r) = \mathbf{H}_{\text{eq}}^r \mathbb{Z}^8$ to denote the lattice generated by \mathbf{H}_{eq}^r and let $\Lambda(\mathbf{H}_{\text{eq}}^r) = \mathbf{H}_{\text{eq}}^r \mathcal{X}$ be its subset generated from \mathcal{X} .

Let $d_{\text{min},r}^2$ denote the MSED of the received symbols that we see:

$$d_{\text{min},r}^2 = \min_{\bar{x} \neq \bar{0}, \bar{x} \in \Lambda_{\text{inf}}(\mathbf{H}_{\text{eq}}^r)} \|\bar{x}\|^2 \quad (4.10)$$

Therefore, to find $d_{\text{min},r}^2$ is equivalent to find the shortest vector in lattice $\Lambda_{\text{inf}}(\mathbf{H}_{\text{eq}}^r)$. These problems, known as Finding Small Vectors in Lattices (FSVL) problems, are discussed in section 2.7.3 of [29] where Sphere Decoder based algorithms are used to search the small vectors. However, this method needs always to give an initial search radius and the algorithms may give several small vectors, not the smallest one. We propose in Algorithm 6 the method that gives the shortest vector in using the same architecture as decoder.

Algorithm 6 Shortest Vector

Let \mathbf{H} be the lattice generator matrix, this algorithm outputs the *minimal* non-zero vector $\bar{x} \in \mathbb{Z}^n$ such that $L(\bar{x}, \mathbf{H}) = \bar{x}^T \mathbf{H}^T \mathbf{H} \bar{x} \leq L(\bar{x}', \mathbf{H}), \forall \bar{x}' \neq \bar{0}, \bar{x}' \in \mathbb{Z}^n$ and also the value of $L(\bar{x}, \mathbf{H})$. Only one of the two vectors in pair $(\bar{x}, -\bar{x})$ is given.

```
1.[Initialize]   $\bar{y} \leftarrow \bar{0}$ 
                  $\bar{u} \leftarrow \{-1, 0, \dots, 0\}$ 
                  $\overline{\text{dist}} \leftarrow \{0, \dots, 0\}$ 
                  $\bar{e} \leftarrow \{0, \dots, 0\}$ 
                  $\overline{\text{step}} \leftarrow \{-1, \dots, -1\}$ 
                  $\mathbf{H} = \mathbf{QR}$  (apply QR-decomposition to  $\mathbf{H}$ )
2.[Start]       $k \leftarrow 1$ 
                  $\bar{x} \leftarrow \bar{u}$ 
                  $\text{bestdist} \leftarrow r_{1,1}^2$ 
                 go to step 5
3.[Calculate]   $d \leftarrow e_k - r_{k,k} \cdot u_k$ 
                  $\text{newdist} \leftarrow \text{dist}_k + d^2$ 
4.[Compare]   if  $\text{newdist} > \text{bestdist}$ 
                 if  $k = n$ 
                     output  $\bar{x}$ ,  $\text{bestdist}$ 
                     terminate the algorithm
                 else
                     go to step 5
                 endif
                 else
                     if  $k = 1$ 
                          $\bar{x} \leftarrow \bar{u}$ 
                          $\text{bestdist} \leftarrow \text{newdist}$ 
                         go to step 5
                     else
                         go to step 6
                     endif
                 endif
5.[Backward]   $k \leftarrow k + 1$ 
                  $u_k \leftarrow u_k + \text{step}_k$ 
                  $\text{step}_k \leftarrow -\text{step}_k - \text{sgn}(\text{step}_k)$ 
                 go to step 3
6.[Forward]    $k \leftarrow k - 1$ 
                  $\text{dist}_k \leftarrow \text{newdist}$ 
                  $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
                  $u_k \leftarrow \left[ \frac{e_k}{r_{k,k}} \right]_{\text{int}}$ 
                  $d \leftarrow e_k - r_{k,k} u_k$ 
                  $\text{step}_k \leftarrow \text{sgn}(d)$ 
                  $\text{newdist} \leftarrow \text{dist}_k + d^2$ 
                 go to step 4
```

where $[x]_{\text{int}}$ return the integer closest to x .

This algorithm could be seen as a modified SE algorithm with different initial condition: start from the $\bar{u} = \{-1, 0, \dots, 0\}$ instead of $\bar{0}$. As discussed in Algorithm 2.7.5 of [29], this method is not very efficient when the dimension is high and we can apply the LLL reduction to accelerate the searching as shown in Algorithm 7.

Algorithm 7 Shortest Vector for High Dimension Case

Let \mathbf{H} be the lattice generator matrix, this algorithm outputs the non-zero vectors $\bar{x} \in \mathbb{Z}^n$ such that $L(\bar{x}, \mathbf{H}) = \bar{x}^T \mathbf{H}^T \mathbf{H} \bar{x} \leq L(\bar{x}', \mathbf{H}), \forall \bar{x}' \neq \bar{0}, \bar{x}' \in \mathbb{Z}^n$ and also the value of $L(\bar{x}, \mathbf{H})$. Only one of the two vectors in pair $(\bar{x}, -\bar{x})$ is given.

- 1.[LLL-Reduction] $\mathbf{S} = \mathbf{H}\mathbf{U}$ (apply LLL reduction to \mathbf{H} , then \mathbf{S} is LLL reduced and \mathbf{U} is a unimodulo matrix.)
 - 2.[Permutation] Let \bar{s}_i denote the columns of \mathbf{S} and \bar{s}'_i denote the rows of \mathbf{S}^{-1} , find a permutation σ on $\{1, \dots, n\}$ such that $\|\bar{s}'_{\sigma(1)}\| \geq \|\bar{s}'_{\sigma(2)}\| \geq \dots \geq \|\bar{s}'_{\sigma(n)}\|$.
Set $\mathbf{S}' \leftarrow \mathbf{S}\mathbf{P}$ with \mathbf{P} the matrix of permutation that

$$p_{ij} = \begin{cases} 1 & i = \sigma(j) \\ 0 & i \neq \sigma(j) \end{cases}$$
 Set $\mathbf{U}' \leftarrow \mathbf{P}\mathbf{U}$ then $\mathbf{S}' = \mathbf{H}\mathbf{U}'$
 - 3.[Shortest Vector] Using Algorithm 6 on \mathbf{S}' , find the minimal nonzero vector \bar{y} and $L(\bar{y}, \mathbf{S}')$.
Set $\bar{x} \leftarrow \mathbf{U}'\bar{y}$
output $\bar{x}, L(\bar{x}, \mathbf{H}) = L(\bar{y}, \mathbf{S}')$
terminate the algorithm
-

By normalizing with N_0 , the NMSED is:

$$D_{\min,r}^2 = \frac{d_{\min,r}^2}{N_0} \quad (4.11)$$

The word error probability is approximated by:

$$P_e(\mathbf{H}_{\text{eq}}^r, \mathcal{X}) \approx N_{\text{kiss}} Q\left(\sqrt{\frac{D_{\min,r}^2}{2}}\right) \quad (4.12)$$

For comparison, the unity of transmission is a codeword of 4 QAM symbols in both cooperative and non-cooperative transmissions¹.

Since the the kissing number is less important than the NMSED, we will measure the WER only in function of D_{\min}^2 for both non-cooperation and cooperation systems with QPSK and 16-QAM constellations. In the cooperation system, we suppose all the Rayleigh fading channel condition and the simulation is in LPG configuration. The SNR is fixed to 15dB for QPSK and 18dB for 16-QAM. Simulation results are given in Figures 4.2 and 4.3:

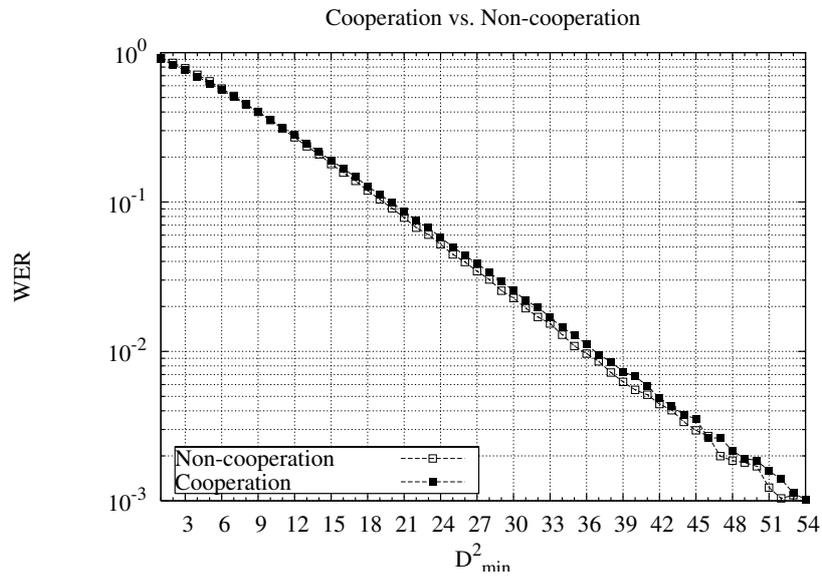


Figure 4.2: Word error rate in function of NMSED (D_{\min}^2) with QPSK modulation.

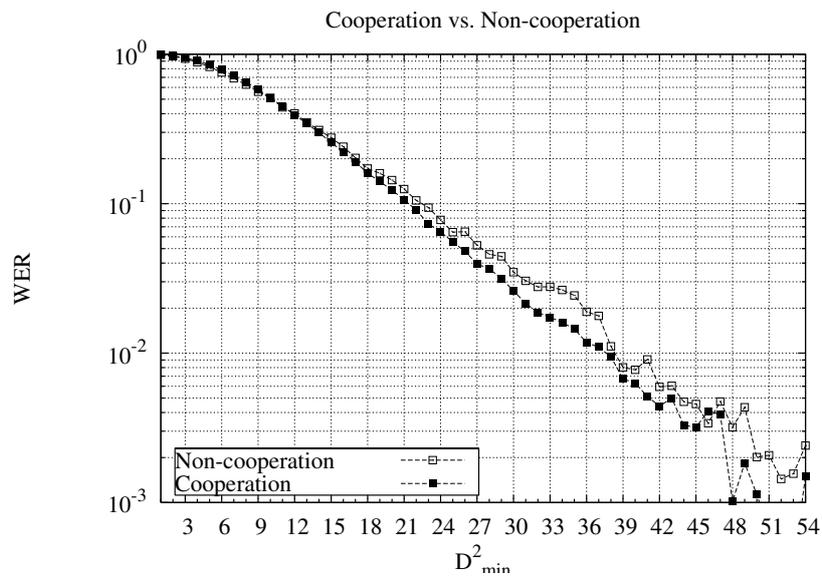


Figure 4.3: Word error rate in function of NMSED (D_{\min}^2) with 16-QAM modulation.

D_{\min}^2 gives a good indication of WER performance for both QPSK and 16-QAM constellations: we see especially for QPSK constellation the two curves are nearly the same. Using this metric, a hybrid transmission mode is proposed as follows:

We suppose full CSI at the source terminal. Let N_0 denote the power of noise, \mathcal{S} denote the constellation in use and let h_{sd} , h_{sr} and h_{rd} denote the coefficients. By comparing the NMSED metric which is noted by $D_{\min,r}^2$ for single-relay single-antenna NAF system and by $D_{\min,a}^2$ for non-cooperative system, select the mode whose NMSED is greater.

This strategy means to active cooperation mode only when $D_{\min,r}^2 > D_{\min,a}^2$. In Rayleigh-fading channel, the WER performances of this hybrid mode are illustrated in Figures 4.4 and 4.5.

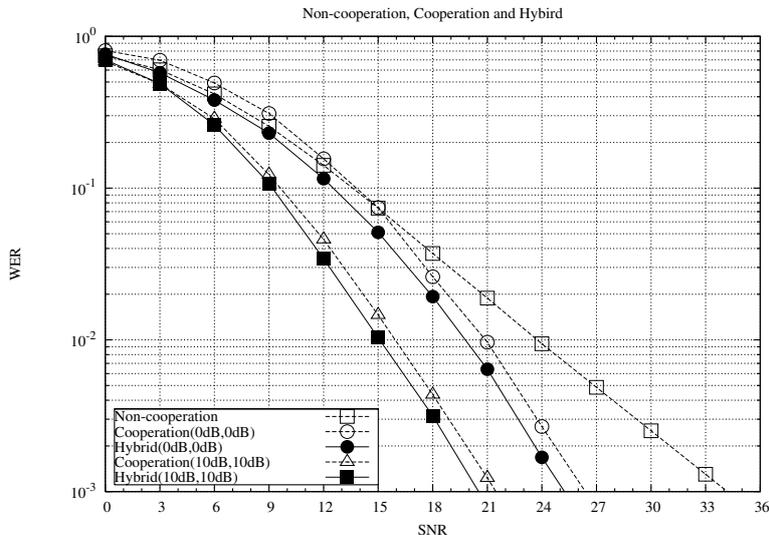


Figure 4.4: Non-cooperation, cooperation and Hybrid transmission in Rayleigh-fading channel with QPSK modulation: in cooperation(including Hybrid mode) case, the path gain are configured to $G_{sr} = 0\text{dB}$, $G_{rd} = 0\text{dB}$ and $G_{sr} = 10\text{dB}$, $G_{rd} = 10\text{dB}$.

In low SNR range, the performance of Hybrid mode is approaching to the non-cooperation performance; in high SNR range, it follows the cooperation performance in keeping the same diversity. This result is straightforward because for every realization of channel, we select the more robust transmission mode.

Besides since the Hybrid mode the cooperation is activated when necessary, the global complexity introduced by cooperation is reduced. In Figure 4.6 and Figure 4.7, the percentage of cooperation is illustrated. It is remarkable that in the case of QPSK with $G_{sr} = 0\text{dB}$, $G_{rd} = 0\text{dB}$, the cooperation occupies only about 10% transmissions and we get a coding gain about 9dB at $\text{WER} = 10^{-3}$ compared with non-cooperation case.

¹A transmitted codeword is $\bar{s} \in \mathcal{S}^4 \subset \mathbb{Z}^4(i)$.

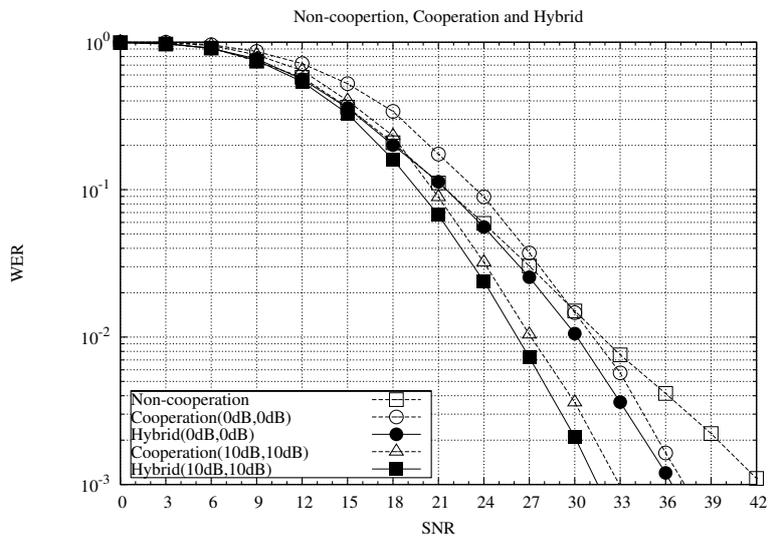


Figure 4.5: Non-cooperation, cooperation and Hybrid transmission in Rayleigh-fading channel with 16-QAM modulation: in cooperation(including Hybrid mode) case, the path gain are configured to $G_{sr} = 0\text{dB}$, $G_{rd} = 0\text{dB}$ and $G_{sr} = 10\text{dB}$, $G_{rd} = 10\text{dB}$.

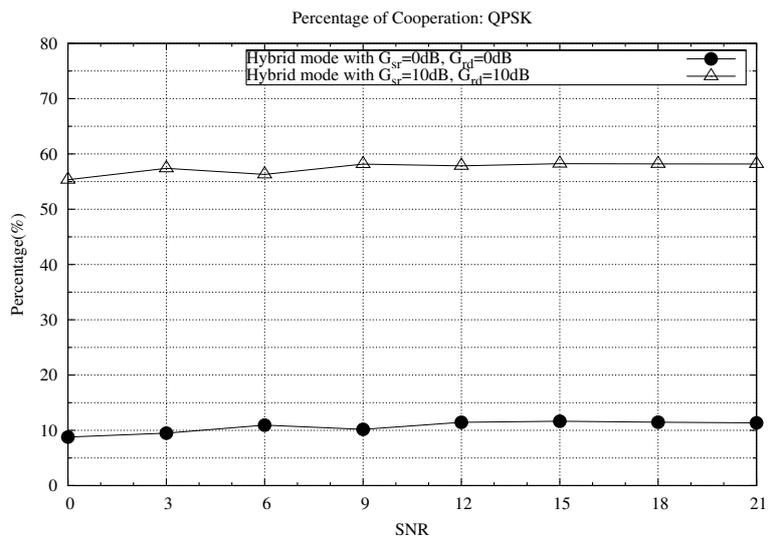


Figure 4.6: Percentage of cooperation in Hybrid mode with QPSK modulation

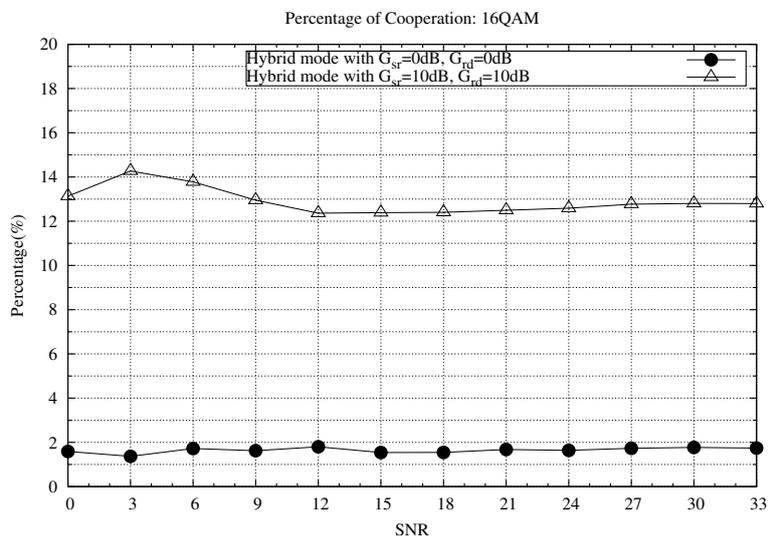


Figure 4.7: Percentage of cooperation in Hybrid mode with 16-QAM modulation

4.2 Equivalent metric for IEEE 802.11a and Relay-SISO

It is important to remark that the metric D_{\min}^2 in section 4.1 is related to the WER performance. However in the communication systems using FEC, the FER performance is not directly from the WER performance. More precisely, for the IEEE 802.11a and Relay-SISO systems who utilize the hard Viterbi decoder for the convolutional code, the Bit Error Rate (BER) performance at the output of QAM demapper or STBC decoder is the determinant to the FER performance.

Mapping is the passage from the information bits to the transmitted symbols. In order to minimize the BER, the Gray code (mapping) is often used such that the adjacent constellation points have only one bit difference or to say their Hamming distance is $d_H = 1$. Since the Golden code is generated from a rotated version of $\mathbb{Z}^4(i)$, we use the same Gray mapping in the cooperation case.

However, considering the virtual MIMO system, for the cooperation system one detection error is not translated into one bit error even for asymptotic performance when $\text{SNR} \rightarrow +\infty$. This fact holds true for all MIMO system where the adjacent points having d_{\min}^2 can give a Hamming distance $d_H > 1$: a simple illustration is given in Figure 4.8.

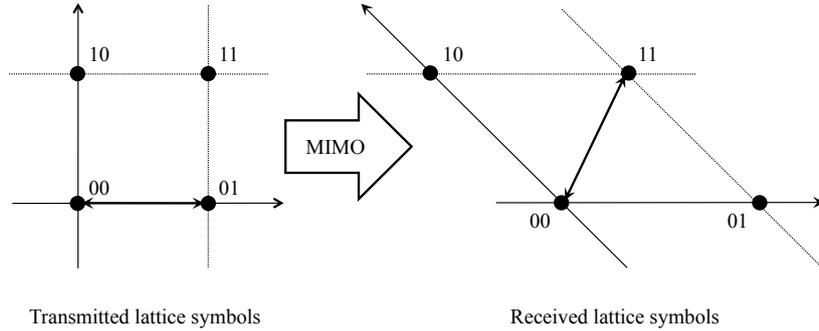


Figure 4.8: Received adjacent points having Hamming distance $d_h > 1$

Besides, we need to consider the randomization of the binary errors. The interleaving technique is often introduced to randomize the binary errors. In IEEE 802.11a, a written-in-row and read-in-column interleaver whose size is $3N_{\text{BPSK}} \times 16$. This interleaver separates the bits belonging to the same QAM symbol with 16 bits spacing as shown in Figure 4.9². In Relay-SISO case, since a codeword is generated from 4 QAM symbols, the correlation of binary errors is even more important than IEEE 802.11a. Therefore, in the proposition of Relay-SISO we use the interleaver of size $4N_{\text{BPSK}} \times 48$ that the bits belonging to the same codeword is separated with 48 bits spacing.

Although D_{\min}^2 gives a good indication on WER performance, it fails to give a metric in terms of FER or BER. As shown in Figure 4.10 and 4.11 where the simulations are taken over flat-fading channel with perfect knowledge of CSI, the performance of Relay-SISO in function of D_{\min}^2 is far from the one of IEEE 802.11a,

We consider an equivalent metric in terms of BER which is easily acquired from the D_{\min}^2 .

²A permutation of bits is also applied

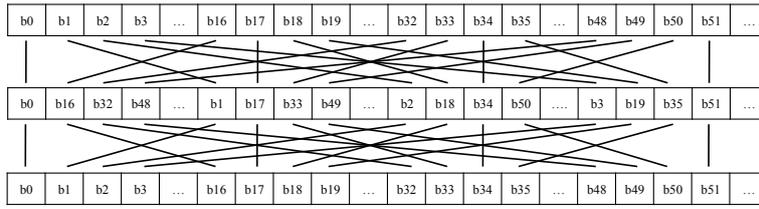


Figure 4.9: Example of IEEE 802.11a interleaving/deinterleaving for 16-QAM

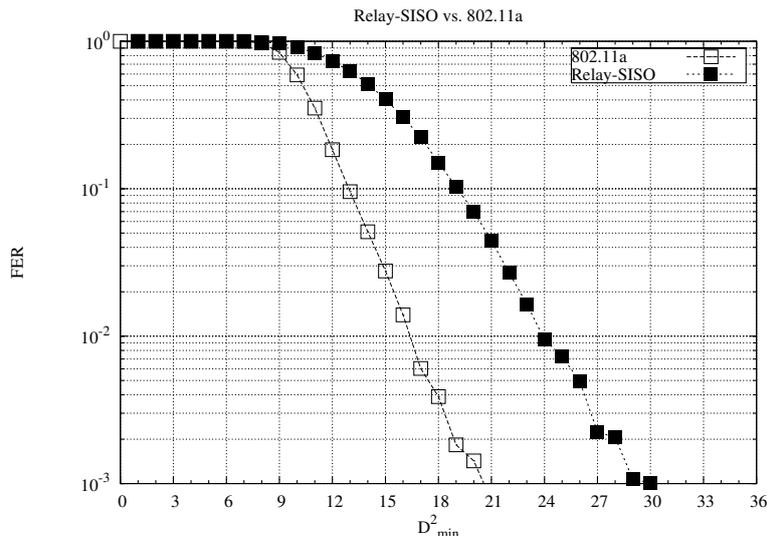


Figure 4.10: Frame error rate in function of NMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.

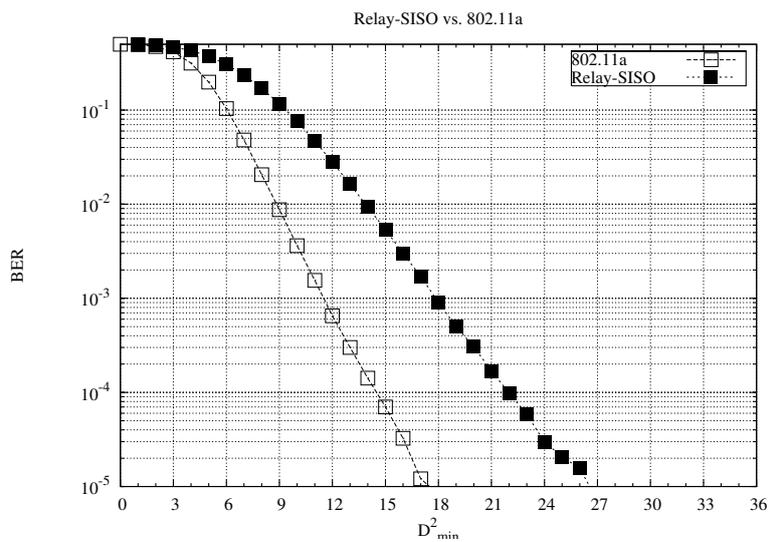


Figure 4.11: Bit error rate in function of NMSED (D_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.

Let \mathcal{X} denote the set of codewords whose size is $|\mathcal{X}| = 2^m$ and the binary mapping function is defined by:

$$\begin{aligned}\mu &: \{0, 1\}^m \rightarrow \mathcal{X} \\ \mu(\bar{b}) &= \mathbf{X}\end{aligned}\tag{4.13}$$

where, $\bar{b} = [b_1 b_2 \cdots b_m]^T \in \{0, 1\}^m$ and $\mathbf{X} \in \mathcal{X}$.

The binary mapping is bijective that we have the demapping function:

$$\begin{aligned}\mu^{-1} &: \mathcal{X} \rightarrow \{0, 1\}^m \\ \mu^{-1}(X) &= \bar{b}\end{aligned}\tag{4.14}$$

In the communication systems, the codeword set \mathcal{X} is taken from the infinite codeword set \mathcal{X}_{inf} . By applying the coset decomposition, the infinite codeword set \mathcal{X}_{inf} can be represented by:

$$\mathcal{X}_{\text{inf}} = \mathcal{X}_{\text{inf}}^{(c)} + \mathcal{X}\tag{4.15}$$

such that \mathcal{X} is the quotient set:

$$\mathcal{X} = \left[\mathcal{X}_{\text{inf}} / \mathcal{X}_{\text{inf}}^{(c)} \right]\tag{4.16}$$

For $\forall \mathbf{x} \in \mathcal{X}_{\text{inf}}$, we have then:

$$\mathbf{x} = \mathbf{x}^{(c)} + \mathbf{X}\tag{4.17}$$

with $\mathbf{x}^{(c)} \in \mathcal{X}_{\text{inf}}^{(c)}$ and $\mathbf{X} \in \mathcal{X}$.

We are going to analyze the BER performance in \mathcal{X}_{inf} instead of in \mathcal{X} . For this purpose, we define the μ -based coset labeling of the lattice \mathcal{X}_{inf} as follows:

Definition *μ -based Coset Labeling*: Let μ be a binary mapping of a codewords set \mathcal{X} . Let \mathcal{X}_{inf} denote the corresponding infinite lattice and then $\mathcal{X}'_{\text{inf}}$ is the sublattice of \mathcal{X}_{inf} that $\mathcal{X}_{\text{inf}} = \mathcal{X}'_{\text{inf}} + \mathcal{X}$. The μ -based coset labeling is then defined as:

$$\begin{aligned}\mathcal{M} &: \{0, 1\}^m \rightarrow \mathcal{X}_{\text{inf}} \\ \mathcal{M}(\bar{b}) &= \mathcal{X}'_{\text{inf}} + \mathbf{X}, \quad \mathbf{X} = \mu(\bar{b})\end{aligned}$$

In this way, the binary mapping of \mathcal{X} turns to a coset labeling of \mathcal{X}_{inf} . For $\mathbf{x} \in \mathcal{X}_{\text{inf}}$, the demapping is defined by:

$$\mu_c^{-1}(\mathbf{x}) = \mu^{-1}(\mathbf{x} \bmod \mathcal{X}'_{\text{inf}}) = \mu^{-1}(\mathbf{X}) = \bar{b}\tag{4.18}$$

Let $\mathbf{x} \in \mathcal{X}_{\text{inf}}$ be the transmitted codeword, the upper bound of BER is given by the union of pairwised binary error rate:

$$\begin{aligned}P_{eb, \mathbf{x}} &\leq \sum_{\mathbf{x}' \in \mathcal{X}_{\text{inf}}, \mathbf{x}' \neq \mathbf{x}} P_{ep, b}(\mathbf{x}' | \mathbf{x}, \mathbf{H}) \\ &= \sum_{\mathbf{x}' \in \mathcal{X}_{\text{inf}}, \mathbf{x}' \neq \mathbf{x}} d_H(\mathbf{x}, \mathbf{x}') P_{ep}(\mathbf{x}' | \mathbf{x}, \mathbf{H})\end{aligned}\tag{4.19}$$

where $d_H(\mathbf{x}, \mathbf{x}')$ denotes the Hamming distance between $\bar{b} = \mu_c^{-1}(\mathbf{x})$ and $\bar{b}' = \mu_c^{-1}(\mathbf{x}')$ and $P_{ep}(\mathbf{x}'|\mathbf{x}, \mathbf{H})$ denotes the pairwise error probability.

By defining $\Delta \mathbf{x} = \mathbf{x}' - \mathbf{x}$, the pairwise error probability depends only on $\Delta \mathbf{x}$ that:

$$P_{ep}(\mathbf{x}'|\mathbf{x}, \mathbf{H}) = P_{ep}(\Delta \mathbf{x}|\mathbf{H}) \quad (4.20)$$

However, this property does not hold always for $d_H(\mathbf{x}, \mathbf{x}')$ because $d_H(\mathbf{x}, \mathbf{x}')$ depends on \mathbf{x} that:

$$\begin{aligned} d_H(\mathbf{x}, \mathbf{x}') &= d_H(\Delta \mathbf{x}, \mathbf{x}) \\ &= \sum_i d_H(\Delta x_i, x_i) \end{aligned} \quad (4.21)$$

where x_i is the entry of \mathbf{x} and $\Delta x_i = x'_i - x_i$.

The second equality holds because the binary mapping is the same in each dimension. With Gray mapping, it is easy to see $d_H(\Delta x_i, x_i)$ is independent to x_i for BPSK, QPSK and 16-QAM that:

$$d_{H,\text{BPSK/QPSK}}(\Delta x_i) = \begin{cases} 0, & \Delta x_i \text{ is even} \\ 1, & \Delta x_i \text{ is odd} \end{cases} \quad (4.22)$$

and

$$d_{H,16\text{-QAM}}(\Delta x_i) = \begin{cases} 0, & \Delta x_i \equiv 0 \pmod{4} \\ 1, & \Delta x_i \equiv 1, 3 \pmod{4} \\ 2, & \Delta x_i \equiv 2 \pmod{4} \end{cases} \quad (4.23)$$

However, for 64-QAM it turns:

$$d_{H,64\text{-QAM}}(\Delta x_i) = \begin{cases} 0, & \Delta x_i \equiv 0 \pmod{8} \\ 1, & \Delta x_i \equiv 1, 7 \pmod{8} \\ 2, & \Delta x_i \equiv 2, 4, 6 \pmod{8} \\ 1 \text{ or } 3, & \Delta x_i \equiv 3, 5 \pmod{8} \end{cases} \quad (4.24)$$

We see that there are 2 possible values of d_H for $x_i \pmod{8} = 3$ or 5 . We define the function $d_H^{(\max)}(\Delta x_i)$ for 64-QAM by:

$$d_H^{(\max)}(\Delta x_i) = \max_{x_i} d_H(\Delta x_i, x_i) \quad (4.25)$$

Then we have:

$$d_{H,64\text{-QAM}}^{(\max)}(\Delta x_i) = \begin{cases} 0, & \Delta x_i \equiv 0 \pmod{8} \\ 1, & \Delta x_i \equiv 1, 7 \pmod{8} \\ 2, & \Delta x_i \equiv 2, 4, 6 \pmod{8} \\ \underline{3}, & \Delta x_i \equiv 3, 5 \pmod{8} \end{cases} \quad (4.26)$$

With this notation, we see:

$$d_H^{(\max)}(\Delta \mathbf{x}) = \sum_i d_H^{(\max)}(\Delta x_i) \quad (4.27)$$

From (4.19),(4.20) and (4.21), the upper bound of P_{eb} is given by:

$$P_{eb}(\mathbf{H}) \leq \sum_{\mathbf{x} \in \mathcal{X}_{\text{inf}}, \mathbf{x} \neq \mathbf{0}} d_H^{(\text{max})}(\mathbf{x}) P_{ep}(\mathbf{x}|\mathbf{H}) \quad (4.28)$$

The goal is to give a D_{min}^2 -like metric in both Relay-SISO and IEEE 802.11a using D_{min}^2 . Therefore, the equivalent D_{min}^2 metric is defined as below:

Definition ENMSED: Let \mathbf{H} denote the channel matrix. The metric Equivalent Normalized Minimum Squared Euclidean Distance (ENMSED) is the value of D_{min}^2 which verifies the condition $P_e(D_{\text{min}}^2) = P_{eb}(\mathbf{H})$.

The new metric ENMSED by definition is a mathematical indicator in terms of BER and we can evaluate it as decribed (4.28).

To obtain this metric, we notice that:

$$P_{ep}(\mathbf{x}|\mathbf{H}) = Q \left(\sqrt{\frac{\|\mathbf{H}\mathbf{x}\|^2}{2N_0}} \right), \quad \mathbf{x} \neq \mathbf{0} \quad (4.29)$$

The Euclidean distance decides the ordre of error probability whereas the $d_H^{(\text{max})}$ is only a scalar factor. Thus we need only examine several nearest points to $\mathbf{0}$ to evaluate the BER. For example, we find K nearests points and then we calculate D_{min}^2 with these points.

For the first step, Algorithm 2.7.5 (Short Vectors) and 2.7.7 (Fincke-Pohst) in [29] have already given the methods in both low dimensions case and high dimensions case, respectively. However, these SD-based algorithms can not give the exact number of smallest vectors. For this reason, some SE-based algorithms are introduced here to search K shortest non-zero vectors: see Algorithm 8³.

As we are also interested in the hardware level implementation, we can use a sub-optimal Algorithm 9 which is based on K-Best algorithm.

By excluding the $\bar{\mathbf{0}}$, we find the $K - 1$ nonzero shortest vectors. The K-Best algorithm could be seen as a variant of the Decision-Feedback Equilizer, DFE. Instead of keeping only 1 best candidate at each step, this algorithm keeps K best candidates, thus the last K best candidates give the possible K shortest vectors (including $\bar{\mathbf{0}}$).

For the second question on generation of D_{min}^2 , we suppose $\{\bar{x}_i\}$ are coordinates of the K smallest vectors in lattice generated by \mathbf{H} . Let $\{d_i^2\} = \{\|\mathbf{H}\bar{x}_i\|^2\}$ denote the correponding squared Euclidean distances which are normalized with the noise power such that $D_i^2 = d_i^2/N_0$. The BER is upper bounded by:

$$P_{eb}(\mathbf{H}) \leq \sum_i d_H^{(\text{max})}(\bar{x}_i) Q \left(\sqrt{\frac{D_i^2}{2}} \right) \quad (4.30)$$

³As same as algorithm Fincke-Pohst, we could apply the LLL-reduction and permutation for high dimensions.

Algorithm 8 SE-Based K Shortest Vectors

Let \mathbf{H} denote the lattice generator matrix, this algorithm outputs the K shortest vectors $\mathcal{S} = \{\bar{s}_i\}, \bar{s}_i \in \mathbb{Z}^n$ for $i = 1, 2, \dots, K$ that $L(\bar{s}) = \bar{s}^T \mathbf{H}^T \mathbf{H} \bar{s} \leq L(\bar{s}'), \forall \bar{s} \in \mathcal{S}, \forall \bar{s}' \neq \bar{0}, \bar{s}' \in \mathbb{Z}^n$ but $\bar{s}' \notin \mathcal{S}$ in lattice.

```
1.[Initialize]  $\bar{y} \leftarrow \bar{0}, \bar{u} \leftarrow [-1, 0, \dots, 0]$ 
 $\overline{dist} \leftarrow [0, \dots, 0], \bar{e} \leftarrow [0, \dots, 0]$ 
 $\overline{step} \leftarrow [-1, \dots, -1]$ 
 $\mathbf{H} = \mathbf{QR}$  (apply QR-decomposition to  $\mathbf{H}$ )
2.[Start]  $k \leftarrow 1, c \leftarrow 1$ 
while  $c \leq K$ 
     $\bar{x}_c \leftarrow \bar{u}, \text{bestdist}_c \leftarrow r_{1,1}^2 u_k^2$ 
     $\text{step}_k \leftarrow -\text{step}_k - \text{sgn}(\text{step}_k)$ 
     $u_k \leftarrow u_k + \text{step}_k, c \leftarrow c + 1$ 
endwhile
find the maximum among  $\{\text{bestdist}_c\}, \text{bestdist}_{\max}$ 
let  $c_{\max}$  denote its index that
     $\text{bestdist}_{\max} = \text{bestdist}_{c_{\max}}$ 
go to step 5
3.[Calculate]  $d \leftarrow e_k - r_{k,k} u_k, \text{newdist} \leftarrow \text{dist}_k + d^2$ 
4.[Compare] if  $\text{newdist} > \text{bestdist}_{\max}$ 
    if  $k = n$ 
        output  $\{\bar{x}_c\}, \{\text{bestdist}_c\}$ 
        terminate the algorithm
    else
        go to step 5
    endif
else
    if  $k = 1$ 
         $\bar{x}_{c_{\max}} \leftarrow \bar{u}, \text{bestdist}_{c_{\max}} \leftarrow \text{newdist}$ 
        find the maximum among  $\{\text{bestdist}_c\}, \text{bestdist}_{\max}$ 
        let  $c_{\max}$  denote its index that
             $\text{bestdist}_{\max} = \text{bestdist}_{c_{\max}}$ 
        go to step 5
    else
        go to step 6
    endif
endif
5.[Backward]  $k \leftarrow k + 1, u_k \leftarrow u_k + \text{step}_k$ 
 $\text{step}_k \leftarrow -\text{step}_k - \text{sgn}(\text{step}_k)$ 
go to step 3
6.[Forward]  $k \leftarrow k - 1, \text{dist}_k \leftarrow \text{newdist}$ 
 $e_k \leftarrow y_k - \sum_{j=k+1}^n r_{k,j} u_j$ 
 $u_k \leftarrow \lfloor \frac{e_k}{r_{k,k}} \rfloor_{\text{int}}, d \leftarrow e_k - r_{k,k} u_k$ 
 $\text{step}_k \leftarrow \text{sgn}(d), \text{newdist} \leftarrow \text{dist}_k + d^2$ 
go to step 4
```

Algorithm 9 K-Best Based K Shortest Vectors

Let \mathbf{H} denote the lattice generator matrix, this algorithm outputs the K shortest vectors (including $\bar{\mathbf{0}}$) $\mathcal{S} = \{\bar{s}_i\}$, $\bar{s}_i \in \mathbb{Z}^n$ for $i = 1, 2, \dots, K$ that $L(\bar{s}) = \bar{s}^T \mathbf{H}^T \mathbf{H} \bar{s} \leq L(\bar{s}')$, $\forall \bar{s} \in \mathcal{S}, \forall \bar{s}' \neq \bar{\mathbf{0}}, \bar{s}' \in \mathbb{Z}^n$ and $\bar{s}' \notin \mathcal{S}$ in lattice in an heuristic way. Let $\{Z_{min}, \dots, Z_{max}\}$ denote the set of possible integer points at each real dimension then the points to be examined at each dimension is $\{z_1, \dots, z_M\} = \{-(Z_{max} - Z_{min}), \dots, +(Z_{max} - Z_{min})\}$.

- 1.[Initialize] $\bar{y} \leftarrow \mathbf{0}$
 $\{\text{dist}_1, \dots, \text{dist}_K\} \leftarrow \{0, +\infty, \dots, +\infty\}$
 $k \leftarrow n$
 $\{\bar{u}_1, \bar{u}_2, \dots, \bar{u}_K\} \leftarrow \{\mathbf{0}, \dots, \mathbf{0}\}$
 $\mathbf{H} = \mathbf{QR}$ (QR-decomposition)
 - 2.[Forward] for $i = 1, \dots, K$
for $m = 1, \dots, M$
 $d \leftarrow \sum_{j=n+1}^N r_{n,j} u_{i,j} + r_{n,n} z_m$
 $D_{i,m} \leftarrow \text{dist}_i + d^2$
endfor
endfor
 - 3.[Selection] Let $\{D_{i_1, m_1}, \dots, D_{i_K, m_K}\}$ denote the K smallest values among $\{D_{i,m}\}$
for $p = 1, \dots, K$
 $\bar{U}_p \leftarrow [\bar{u}_{i_p}^T, s_{m_p}]^T$
 $\text{dist}_p \leftarrow D_{i_p, m_p}$
endfor
 - 4.[Judgement] if $k = 1$
output $\{\bar{U}\}$
terminate algorithm
else
for $i = 1, \dots, K$
 $\bar{u}_i \leftarrow \bar{U}_i$
endfor
 $k \leftarrow k - 1$
go to step 2
endif
-

By definition, we are going to find \mathcal{D}_{\min}^2 that:

$$P_e(\mathcal{D}_{\min}^2) = P_{eb}(\mathbf{H}) \quad (4.31)$$

which gives:

$$Q\left(\sqrt{\frac{\mathcal{D}_{\min}^2}{2}}\right) = \sum_i d_H^{(\max)}(\mathbf{x}_i) Q\left(\sqrt{\frac{\mathcal{D}_i^2}{2}}\right) \quad (4.32)$$

This procedure is achieved in 2 steps:

- Step 1, we calculate \mathcal{D}_i^2 by:

$$Q\left(\sqrt{\frac{\mathcal{D}_i^2}{2}}\right) = d_H^{(\max)}(\mathbf{x}_i) Q\left(\sqrt{\frac{\mathcal{D}_i^2}{2}}\right)$$

- Step 2, we calculate \mathcal{D}_{\min}^2 from the previous $\{\mathcal{D}_i^2\}$ by:

$$Q\left(\sqrt{\frac{\mathcal{D}_{\min}^2}{2}}\right) = \sum_i Q\left(\sqrt{\frac{\mathcal{D}_i^2}{2}}\right)$$

Step 1 can be seen as a multiplication function on D^2 that:

$$\mathcal{D}^2 = \mathcal{D}^2\left(D^2, d_H^{(\max)}\right) \quad (4.33)$$

In the same way, step 2 is considered as a series of additions that:

$$Q\left(\sqrt{\frac{\mathcal{D}^2}{2}}\right) = Q\left(\sqrt{\frac{\mathcal{D}_1^2}{2}}\right) + Q\left(\sqrt{\frac{\mathcal{D}_2^2}{2}}\right) \quad (4.34)$$

which is noted by:

$$\mathcal{D}^2 = \mathcal{D}^2(\mathcal{D}_1^2, \mathcal{D}_2^2) \quad (4.35)$$

The lookup tables can be implemented for quick calculation. The multiplication table is shown in Table 4.1 and the addition table is shown in Table 4.1.

In our implementation, the entries of both the mutiplication table and addition table are rounded to intergers and we consider the cases d_H less than 6 and \mathcal{D}^2 less than 50. Then the multiplication table is of dimension 50×5 and the addition table is of dimension 50×50 . In the simulations, the number of searching points in lattice is configured as $K = 16$ for both QPSK and 16-QAM. This setting gives a good indication of Relay-SISO's performance as shown in Figure 4.12 and 4.13.

Table 4.1: Simplified multiplication table of $\mathcal{D}^2(\mathcal{D}^2, d_H)$

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	1	0	0	0	0	0
2	2	0	0	0	0	0
3	3	1	0	0	0	0
4	4	2	1	0	0	0
5	5	3	2	1	1	0
6	6	4	3	2	1	1
7	7	5	4	3	2	2
8	8	6	4	4	3	2
9	9	7	5	4	4	3
10	10	8	6	5	5	4
11	11	9	7	6	6	5
12	12	10	8	7	6	6
13	13	11	9	8	7	7
14	14	12	10	9	8	8
15	15	13	11	10	9	9
16	16	14	12	11	10	10
17	17	14	13	12	11	11
18	18	15	14	13	12	12
19	19	16	15	14	13	13

Table 4.2: Simplified addition table of $\mathcal{D}^2(\mathcal{D}_1^2, \mathcal{D}_2^2)$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	1	1	1	1	2	2	2	2	2	2	2	2	2
3	0	0	1	1	2	2	2	2	2	3	3	3	3	3	3	3
4	0	0	1	2	2	2	3	3	3	3	4	4	4	4	4	4
5	0	1	1	2	2	3	3	4	4	4	4	5	5	5	5	5
6	0	1	1	2	3	3	4	4	5	5	5	5	5	6	6	6
7	0	1	2	2	3	4	4	5	5	6	6	6	6	6	7	7
8	0	1	2	2	3	4	5	5	6	6	7	7	7	7	7	8
9	0	1	2	3	3	4	5	6	6	7	7	7	8	8	8	8
10	0	1	2	3	4	4	5	6	7	7	8	8	8	9	9	9
11	0	1	2	3	4	5	5	6	7	7	8	9	9	9	10	10
12	0	1	2	3	4	5	5	6	7	8	8	9	10	10	10	11
13	0	1	2	3	4	5	6	6	7	8	9	9	10	11	11	11
14	0	1	2	3	4	5	6	7	7	8	9	10	10	11	12	12
15	0	1	2	3	4	5	6	7	8	8	9	10	11	11	12	13

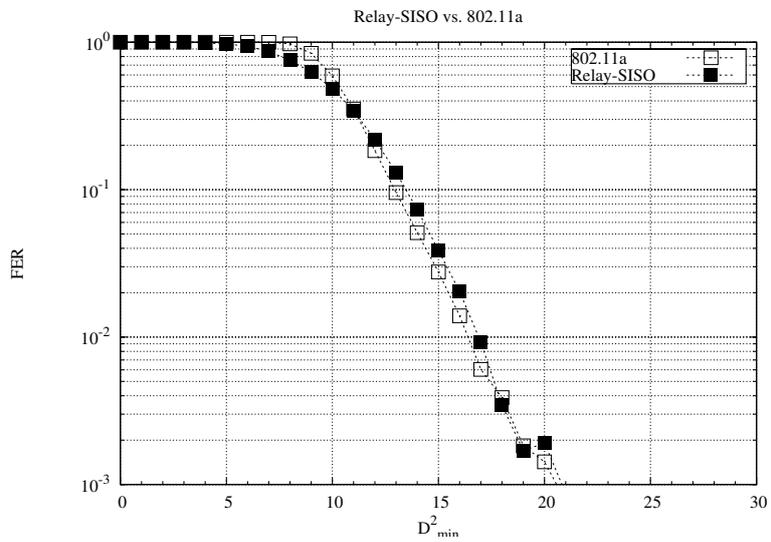


Figure 4.12: Frame error rate in function of ENMSED (\mathcal{D}_{min}^2) over flat fading channel: one frame contains a MPDU of 200 bytes at datarate 12Mbps.

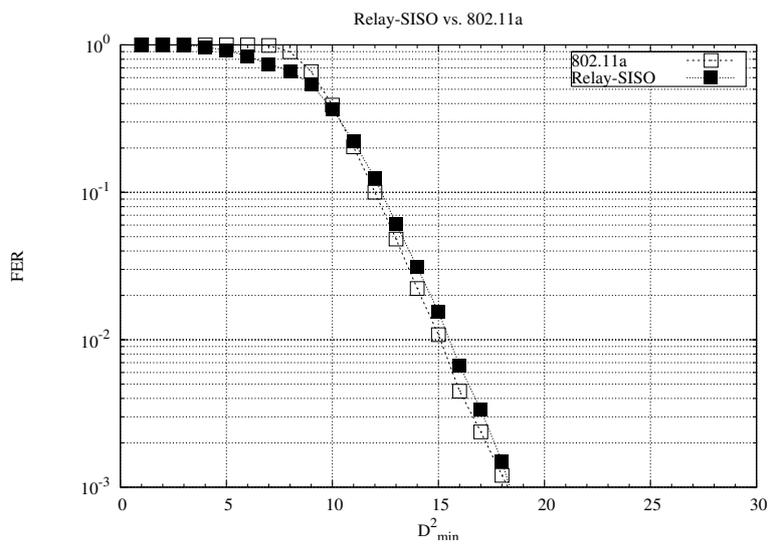


Figure 4.13: Frame error rate in function of ENMSED (\mathcal{D}_{min}^2) over flat fading channel: one frame contains a MPDU of 400 bytes at datarate 24Mbps.

4.3 Hybrid mode and performance

In the multi-carrier transmissions, the ENMSED is calculated on each sub-carrier then we take the average of 6 smallest metrics as the system metric.

As the Hybrid mode described in section 4.2, we use the metric \mathcal{D}_{\min}^2 as an indicator of FER performance. We apply the same hybrid principle in IEEE 802.11a/Relay-SISO system and this Relay-SISO-Hybrid method is described as follows:

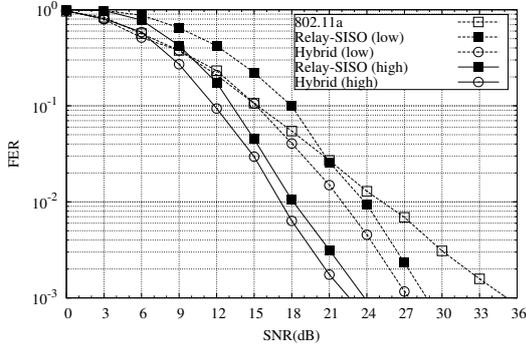
Before the transmission of MPDU in Relay-SISO mode, the terminals source, relay and destination accomplish a cooperative transmission by transmitting a sounding frame to probe the cooperation channel with data rate indication. The destination terminal calculates the ENMSED metric of Relay-SISO system noted by $\mathcal{D}_{\min,r}^2$ and the one of IEEE 802.11a system, noted by $\mathcal{D}_{\min,a}^2$. Based on the two metrics, the destination terminal informs the source terminal the mode decision by choosing Relay-SISO mode when $\mathcal{D}_{\min,r}^2 > \mathcal{D}_{\min,a}^2$.

The performance of Relay-SISO-Hybrid is simulated with data rate 12Mbps and 24Mbps over flat fading channel and multipath channel type A where “low” denotes the LPG configuration and “high” denotes the HPG configurations. Simulation results are provided in Figures 4.14, 4.15, 4.16 and 4.17.

The Hybrid mode can effectively improve the performance of cooperation system. In low or medium SNR range, the Hybrid performance follows the IEEE 802.11a while in high SNR range, Hybrid keeps the cooperative diversity gain with even a little coding gain. As a result, Hybrid mode outperforms both 802.11a and Relay-SISO.

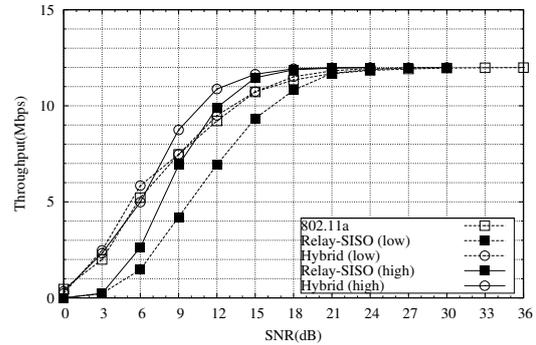
The crossing point problem disappears for the transmissions over the flat fading channel. For the multipath channel, we notice that in case of 24Mbps with LPG configuration the hybrid mode improves the cooperative performance but with a little loss in SNR compared with the IEEE 802.11a system. This is due to the fact that the average of ENMSED is less precise for frequency selective channel. Thus, the future research works can be taken on this aspect.

Relay-SISO, 802.11a and Hybrid: Data-rate=12Mbps, MPDU=125Bytes over flat fading ch



(a) Frame error rate

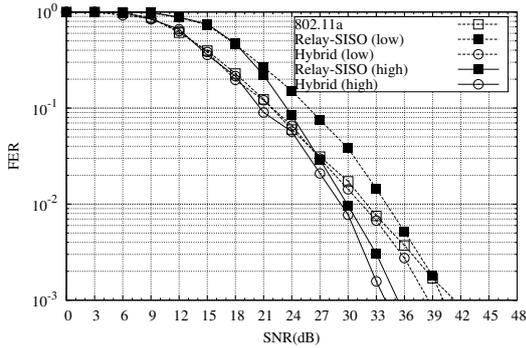
Relay-SISO, 802.11a and Hybrid: Data-rate=12Mbps, MPDU=125Bytes over flat fading ch



(b) Efficient throughput

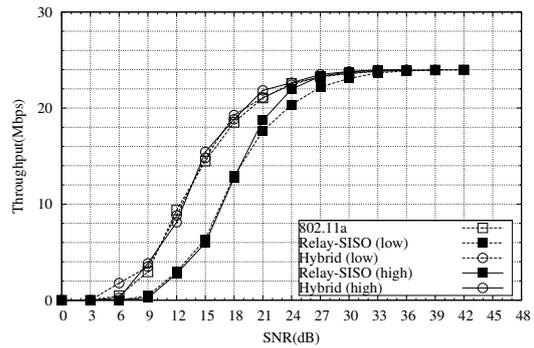
Figure 4.14: Relay-SISO, IEEE 802.11a and Hybrid over flat fading channel: each MPDU contains 125 bytes and it is sent at 12Mbps.

Relay-SISO, 802.11a and Hybrid: Data-rate=24Mbps, MPDU=125Bytes over flat fading c



(a) Frame error rate

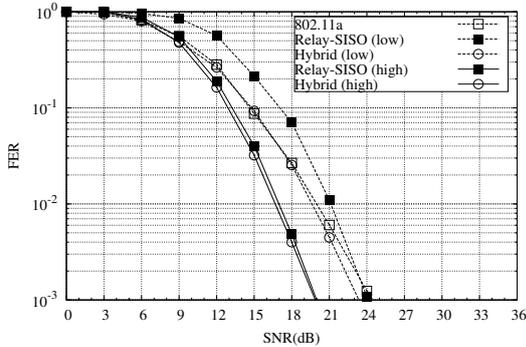
Relay-SISO, 802.11a and Hybrid: Data-rate=24Mbps, MPDU=125Bytes over flat fading cl



(b) Efficient throughput

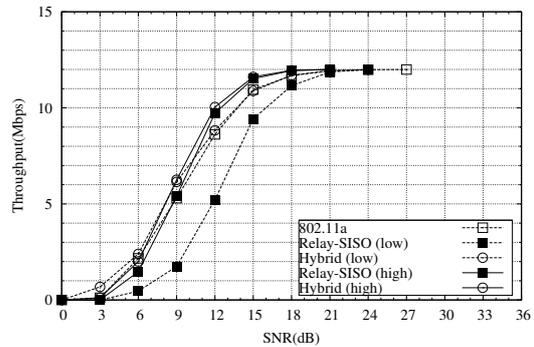
Figure 4.15: Relay-SISO, IEEE 802.11a and Hybrid over flat fading channel: each MPDU contains 125 bytes and it is sent at 24Mbps.

Relay-SISO, 802.11a and Hybrid: Data-rate=12Mbps, MPDU=125Bytes over channel ty



(a) Frame error rate

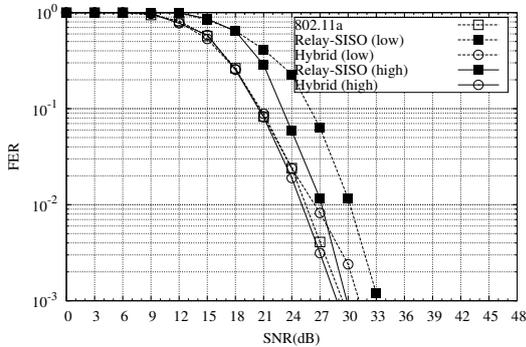
Relay-SISO, 802.11a and Hybrid: Data-rate=12Mbps, MPDU=125Bytes over channel ty



(b) Efficient throughput

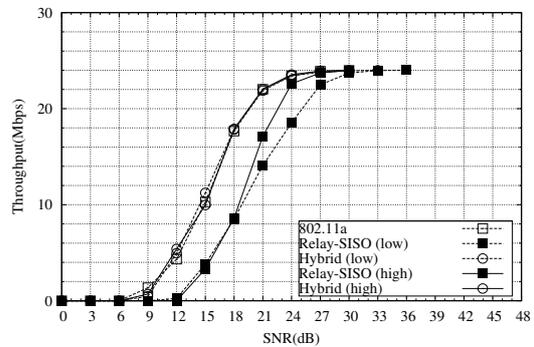
Figure 4.16: Relay-SISO, IEEE 802.11a and Hybrid over channel type A: each MPDU contains 125 bytes and it is sent at 12Mbps.

Relay-SISO, 802.11a and Hybrid: Data-rate=24Mbps, MPDU=125Bytes over channel ty



(a) Frame error rate

Relay-SISO, 802.11a and Hybrid: Data-rate=24Mbps, MPDU=125Bytes over channel ty



(b) Efficient throughput

Figure 4.17: Relay-SISO, IEEE 802.11a and Hybrid over channel type A: each MPDU contains 125 bytes and it is sent at 24Mbps.

CHAPTER 5

GOLDEN CODE COSET AND ITS APPLICATION IN COOPERATION SYSTEM

The Golden code is previously presented in section 1.3. This is a full-diversity and full-throughput 2×2 STBC with non-vanishing determinant. Research works on this family of STBC show that this code can be extended by using the partition method to enlarge the determinant gain.

However, as explained in [30], the direct use of partitioned Golden code will not bring any gain of system since the gain in determinant is recompensated by the loss in spectral efficiency. In order to take advantage of this determinant gain, the authors propose to use a trellis coded Golden code by combining the Trellis Coded Modulation (TCM) with the partitioned Golden code. This method sheds lights on the application of the partitioned Golden code and it inspires the following research works.

In this chapter, we propose the use of the partitioned Golden code with IEEE 802.11a standard convolutional code to enhance the transmission quality. Simulation results show the improvements of cooperative transmissions with a very little complexity cost.

5.1 Golden code and partition

Let us review the construction of Golden code in section 1.3.

Let \mathbb{K} denote the quadratic extension of $\mathbb{F} = \mathbb{Q}(i)$:

$$\mathbb{K} = \{s_1 + s_2\theta, \text{ for } s_1, s_2 \in \mathbb{F}\} \quad (5.1)$$

where

$$\begin{aligned} \theta &= \frac{1 + \sqrt{5}}{2} \\ \bar{\theta} &= \frac{1 - \sqrt{5}}{2} \end{aligned} \quad (5.2)$$

with minimal polynomial $\mu(X) = X^2 - X - 1$.

With this construction, the basis $B_{\mathbb{K}} = [1, \theta]$ are also the integral basis of the intergers ring of \mathbb{K} , denoted by $\mathcal{O}_{\mathbb{K}} = \mathbb{Z}[i][\theta]$.

The CDA \mathcal{A} is defined over \mathbb{K}/\mathbb{F} by:

$$\mathcal{A}(\mathbb{K}/\mathbb{F}, \gamma, \sigma) = \{x + e \cdot y, \text{ for } x, y \in \mathbb{K}/\mathbb{F}\} \quad (5.3)$$

with

$$\begin{cases} e^2 = \gamma \in \mathbb{F} \\ x \cdot e = e \cdot \sigma(x), \forall x \in \mathbb{K} \end{cases} \quad (5.4)$$

In the matrix representation we have:

$$e = \begin{bmatrix} 0 & 1 \\ \gamma & 0 \end{bmatrix} \quad (5.5)$$

and

$$x = \text{diag}(x, \sigma(x)) \quad (5.6)$$

where σ is the generator of the Galois group $\text{Gal}(\mathbb{K}/\mathbb{F})$:

$$\text{Gal}(\mathbb{K}/\mathbb{F}) = \{\sigma^0, \sigma^1\} \quad (5.7)$$

with $\sigma(\theta) = \bar{\theta}$.

The canonical embedding is defined by:

$$\begin{aligned} \Sigma : \mathbb{K} &\rightarrow \mathbb{C}^2 \\ \Sigma(x) &= [x, \sigma(x)]^T \end{aligned} \quad (5.8)$$

and the complex generator matrix of $\Lambda_{\mathcal{O}}$ is obtained by embedding the integer basis:

$$\mathbf{M}_{\mathcal{O}} = \Sigma([1, \theta]) = \begin{bmatrix} 1 & \theta \\ 1 & \bar{\theta} \end{bmatrix} \quad (5.9)$$

Let \mathcal{I} denote the principal ideal of \mathbb{K} generated by $\alpha = 1 + i - i\theta$ such that $\mathcal{I} = \mathcal{I}(\alpha)$.

The relative basis of \mathcal{I} is then given by $B_{\mathcal{I}} = [\alpha, \alpha\theta]$ which gives the complex lattice generator matrix:

$$\mathbf{M}_{\mathcal{I}} = \begin{bmatrix} \alpha & \alpha\theta \\ \bar{\alpha} & \bar{\alpha}\bar{\theta} \end{bmatrix} \quad (5.10)$$

It is easy to see that:

$$\mathbf{M}^H \mathbf{M} = 5\mathbf{I}_2 \quad (5.11)$$

Therefore, the obtained lattice $\Lambda(\mathcal{I})$ is a rotated version of $\sqrt{5}\mathbb{Z}^2(i)$.

By restricting $[x, y] \in \mathcal{I}$, the infinite Golden codewords can be expressed by:

$$\mathcal{G}_{\infty} = \left\{ \mathbf{X} = \frac{1}{\sqrt{5}} \begin{bmatrix} x & y \\ \gamma\sigma(y) & \sigma(x) \end{bmatrix}, \text{ for } x, y \in \mathcal{I} \right\} \quad (5.12)$$

where $\sqrt{5}$ is for power normalization and γ is chosen equal to i which is not a norm of \mathbb{K} .

By (5.10) and (5.12), the general form of Golden code is given by:

$$\mathcal{G}_{\infty} = \left\{ \mathbf{X} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha(s_1 + s_2\theta) & \alpha(s_3 + s_4\theta) \\ i\bar{\alpha}(s_3 + s_4\theta) & \bar{\alpha}(s_1 + s_2\theta) \end{bmatrix}, \quad s_i \in \mathbb{Z}(i) \right\} \quad (5.13)$$

The infinite Golden code is viewed as an order of \mathcal{A} that $\mathcal{G}_\infty = (\mathcal{A}, \mathcal{I}(\alpha))$. We define the reduced norm of $g \in \mathcal{G}_\infty$ by the determinant of its matrix representation:

$$N_{\text{red}}(g) = \det(\mathbf{M}(g)) \quad (5.14)$$

The information symbols $\{s_1, s_2, s_3, s_4\}$ are chosen from the constellation assigned QAM symbols set, denoted by \mathcal{C} . The Golden code is given by:

$$\mathcal{G}_{\mathcal{C}} = \left\{ \mathbf{X} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha(s_1 + s_2\theta) & \alpha(s_3 + s_4\theta) \\ i\bar{\alpha}(s_3 + s_4\theta) & \bar{\alpha}(s_1 + s_2\theta) \end{bmatrix}, \quad s_i \in \mathcal{C} \subset K_{\text{mod}}\mathbb{Z}(i) \right\} \quad (5.15)$$

where K_{mod} is the constellation normalization factor.

Now consider the left principal ideal of $\mathcal{A}(\mathbb{K}/\mathbb{F}, \gamma, \sigma)$ which is generated by

$$\mathcal{A}(\beta_L) = \{\beta_L \cdot z, \quad \text{for } z \in \mathcal{A}\} \quad (5.16)$$

In \mathcal{A} , the determinants of the elements that are generated by the integer ring \mathcal{O} are still in $\mathbb{Z}(i)$ that $N_{\text{red}}(a) \in \mathbb{Z}(i)$ for $a \in \mathcal{A}$ and $N_{\text{red}} = 0$ only when $a = 0$. Let $\delta_{\min}(\mathcal{A}, \mathcal{O}_{\mathbb{K}})$ denote the minimum squared determinant of $(\mathcal{A}, \mathcal{O}_{\mathbb{K}})$ and we have:

$$\begin{aligned} \delta_{\min}(\mathcal{A}, \mathcal{O}_{\mathbb{K}}) &= \min \{|N_{\text{red}}(x + e \cdot y)|^2, \quad x, y \in \mathcal{O}_{\mathbb{K}} \text{ and } (x, y) \neq (0, 0)\} \\ &= 1 \end{aligned} \quad (5.17)$$

The minimum squared determinant of \mathcal{G}_∞ is then given by:

$$\begin{aligned} \delta_{\min}(\mathcal{G}_\infty) &= \frac{1}{25} \cdot \min \{N_{\text{red}}(g), \quad g \in \mathcal{I} \text{ and } g \neq 0\} \\ &= \frac{1}{25} \cdot |N_{\mathbb{K}/\mathbb{Z}(i)}(\alpha)|^2 |\delta_{\min}(\mathcal{A}, \mathcal{O}_{\mathbb{K}})|^2 = \frac{1}{5} \end{aligned} \quad (5.18)$$

In matrix representation, for the determinant of the non-zero elements in $\mathcal{A}(\beta_L)$ we have:

$$N_{\text{red}}(\beta_L \cdot z) = N_{\text{red}}(\beta_L) \cdot N_{\text{red}}(z) \quad (5.19)$$

Then the minimum determinant of $(\mathcal{A}(\beta_L), \mathcal{O}_{\mathbb{K}})$ is:

$$\delta_{\min}(\mathcal{A}(\beta_L), \mathcal{O}_{\mathbb{K}}) = |N_{\text{red}}(\beta_L)|^2 |\delta_{\min}(\mathcal{A}, \mathcal{O}_{\mathbb{K}})|^2 = |N_{\text{red}}(\beta_L)|^2 \quad (5.20)$$

The same operation can be taken for the right side in generating a right principal ideal of \mathcal{A} by multiplying β_R :

$$\mathcal{A}(\beta_R) = \{z \cdot \beta_R, \text{ for } z \in \mathcal{A}\} \quad (5.21)$$

As the left principal ideal, the minimal determinant of $(\mathcal{A}(\beta_R), \mathcal{O}_{\mathbb{K}})$ is

$$\delta_{\min}(\mathcal{A}(\beta_R), \mathcal{O}_{\mathbb{K}}) = |N_{\text{red}}(\beta_R)|^2 \quad (5.22)$$

The same results can be applied to \mathcal{G}_∞ by generating:

$$\begin{aligned} \mathcal{G}_\infty(\beta_R) &= \{g \cdot \beta_R, \text{ for } g \in \mathcal{G}_\infty\} \\ \mathcal{G}_\infty(\beta_L) &= \{\beta_L \cdot g, \text{ for } g \in \mathcal{G}_\infty\} \end{aligned} \quad (5.23)$$

then

$$\begin{aligned}\delta_{\min}(\mathcal{G}_{\infty}(\beta_R)) &= \frac{1}{5} |N_{\text{red}}(\beta_R)|^2 \\ \delta_{\min}(\mathcal{G}_{\infty}(\beta_L)) &= \frac{1}{5} |N_{\text{red}}(\beta_L)|^2\end{aligned}\tag{5.24}$$

In [31] and [32], the conception of set partition chain allows to construct the complex lattice from \mathbb{Z}^n . We consider a chain of partitions from \mathcal{G}_{∞} to $2\mathcal{G}_{\infty}$ that:

$$[\mathcal{G}_{\infty}]/[\mathcal{G}_{\infty} \cdot \beta_R]/[\beta_L \cdot \mathcal{G}_{\infty} \cdot \beta_R]/[(1+i)\beta_L \cdot \mathcal{G}_{\infty}]/[2\mathcal{G}_{\infty}]/\cdots\tag{5.25}$$

where we define:

$$\begin{cases} \beta_L = \beta_R = \bar{\theta} + e \cdot \theta \\ \tilde{\beta}_R = -\theta + e \cdot \theta \\ \tilde{\beta}_L = -i\theta + ie \cdot \theta \end{cases}\tag{5.26}$$

The element β_R or β_L is obtained as follows: consider an algebraic extension of \mathbb{K} of degree 2 with the minimum polynomial $\mu(X) = X^2 - i$. In this way, the CDA \mathcal{A} is associated to \mathbb{L} since $e^2 = i$. Lattice $(2)\mathbb{L}$ can be factorized by $(2)\mathcal{O}_{\mathbb{L}} = \mathcal{I}^2(\eta)$ where the principal ideal is generated by $\eta = \bar{\theta} + \bar{\theta}\sqrt{i}$. By replacing \sqrt{i} with $(\cdot e)$, we can find a one-to-one correspondance that $\Phi : \mathbb{L} \rightarrow \mathcal{A}$ and $\beta = \Phi(\eta) = \bar{\theta} + \bar{\theta} \cdot e = \bar{\theta} + e \cdot \theta$.

In fact, there are many chains of partition from \mathcal{G}_{∞} to $2\mathcal{G}_{\infty}$ by choosing the left or right principal ideals and the element generating the principal ideal. We select the element β as the element of both the left ideal $\beta_L \cdot \mathcal{G}_{\infty}$ and right principal ideal $\mathcal{G}_{\infty} \cdot \beta_R$.

The $\tilde{\beta}_L$ and $\tilde{\beta}_R$ are calculated by writing:

$$\begin{cases} \tilde{\beta}_L \cdot \beta_L = 1 - i \\ \beta_R \cdot \tilde{\beta}_R = 1 + i \end{cases}\tag{5.27}$$

It is remarkable that with this construction, we have:

$$\begin{cases} |N_{\text{red}}(\beta_L)|^2 = |N_{\text{red}}(\beta_R)|^2 = |-1 + i|^2 = 2 \\ |N_{\text{red}}(\tilde{\beta}_R)|^2 = |-1 + i|^2 = 2 \\ |N_{\text{red}}(\tilde{\beta}_L)|^2 = |1 - i|^2 = 2 \end{cases}\tag{5.28}$$

The minimum squared determinant increases by 2 for each partition:

$$\begin{aligned}
\mathcal{G}_\infty &\rightarrow \delta_{\min}(\mathcal{G}_\infty) = \frac{1}{5} \\
&\Downarrow \\
\mathcal{G}_\infty \cdot \beta_R &\rightarrow \delta_{\min}(\mathcal{G}_\infty \cdot \beta_R) = \frac{2}{5} \\
&\Downarrow \\
\beta_L \cdot \mathcal{G}_\infty \cdot \beta_R &\rightarrow \delta_{\min}(\beta_L \cdot \mathcal{G}_\infty \cdot \beta_R) = \frac{4}{5} \\
&\Downarrow \\
\beta_L \cdot \mathcal{G}_\infty \cdot \beta_R \cdot \tilde{\beta}_R &= (1+i)\beta_L \cdot \mathcal{G}_\infty \rightarrow \delta_{\min}((1+i)\beta_L \cdot \mathcal{G}_\infty) = \frac{8}{5} \\
&\Downarrow \\
(1+i)\tilde{\beta}_L \cdot \beta_L \cdot \mathcal{G}_\infty &= 2\mathcal{G}_\infty \rightarrow \delta_{\min}(2\mathcal{G}_\infty) = \frac{16}{5}
\end{aligned}$$

5.2 Golden code partition and coset bit mapping

The Golden code partition chain gives a method to increase the minimum squared determinant and now we consider the bit mapping problem with this partition.

For Golden code, the bit mapping is achieved within 2 steps: first we generate the 4 complex QAM symbols $\{s_1, s_2, s_3, s_4\}$ from $4N_{\text{BPSC}}$ by Gray code; then we encode $\{s_1, s_2, s_3, s_4\}$ as shown in (5.15). This bit mapping of Golden code is simple since Golden code is a rotated $\mathbb{Z}^4(i)$.

However, this bit mapping method does not fit the partitioned Golden code. We propose in this section a bit mapping method for the coset partition.

Let us take the partition $[\mathcal{G}_\infty \cdot \beta_R]$ for example. Let \mathcal{Q} denote the quotient ring of $[\mathcal{G}_\infty / (\mathcal{G}_\infty \cdot \beta_R)]$ and Golden code is therefore decomposed by:

$$[\mathcal{G}_\infty] = [\mathcal{G}_\infty \cdot \beta_R] + [\mathcal{Q}] \quad (5.29)$$

The order of \mathcal{Q} is denoted by $|\mathcal{Q}|$ which gives $|\mathcal{Q}| = 4$ for $N_{\mathbb{L}}(\eta) = 4$. Let $\{q_0, q_1, q_2, q_3\}$ denote the 4 elements of \mathcal{Q} .

According to the ring property, \mathcal{Q} contains the element 0 and $(q_i + q_j) \bmod \beta_R \in \mathcal{Q}$. Without loss of generality, let $q_0 = 0$, q_1 , q_2 and $q_3 = q_1 + q_2 \bmod \beta_R$ denote the elements of the quotient ring.

Let Ψ denote the labelling such that:

$$\begin{aligned}
\Psi : \mathcal{G}_\infty &\rightarrow \mathbb{Z}^4(i) \\
\Psi(x) &= [s_1, s_2, s_3, s_4]
\end{aligned} \quad (5.30)$$

where $[s_1, s_2, s_3, s_4]$ generate g as shown in (5.13).

We see that:

$$\begin{cases} q_1 = 1 \\ q_2 = \theta \\ q_3 = 1 + \theta \end{cases} \quad (5.31)$$

which gives

$$\begin{cases} \Psi(q_1) = [1, 0, 0, 0] \\ \Psi(q_2) = [0, 1, 0, 0] \\ \Psi(q_3) = [1, 1, 0, 0] \end{cases} \quad (5.32)$$

with

$$\begin{cases} N_{\text{red}}(q_1) = 1 \\ N_{\text{red}}(q_2) = 1 \\ N_{\text{red}}(q_3) = 1 \end{cases} \quad (5.33)$$

Because $N_{\text{red}}(\beta) = -1 + i$, $\{q_1, q_2, q_3\}$ are neither in $\mathcal{G}_\infty(\beta_R)$ nor in $\mathcal{G}_\infty(\beta_L)$, so $\mathcal{Q} = \{q_0, q_1, q_2, q_3\}$ is the quotient ring of $\mathcal{G}_\infty(\beta_R)/\mathcal{G}_\infty$ and $\mathcal{G}_\infty(\beta_L)/\mathcal{G}_\infty$ ¹.

We can continue the decomposition in (5.29) according to (5.25):

$$\begin{aligned} [\mathcal{G}_\infty] &= [\mathcal{G}_\infty \cdot \beta_R] + [\mathcal{Q}] \\ &= [\beta_L \cdot \mathcal{G}_\infty \cdot \beta_R] + [\mathcal{Q} \cdot \beta_R] + [\mathcal{Q}] \\ &= [(1+i)\beta_L \cdot \mathcal{G}_\infty] + [\beta_L \cdot \mathcal{Q} \cdot \beta_R] + [\mathcal{Q} \cdot \beta_R] + [\mathcal{Q}] \\ &= [2\mathcal{G}_\infty] + [(1+i)\beta_L \cdot \mathcal{Q}] + [\beta_L \cdot \mathcal{Q} \cdot \beta_R] + [\mathcal{Q} \cdot \beta_R] + [\mathcal{Q}] \end{aligned} \quad (5.34)$$

Let $[\mathcal{Q}^{(0)}]$ denote $[\mathcal{Q}]$, $[\mathcal{Q}^{(1)}]$ denote $[\mathcal{Q} \cdot \beta_R]$, $[\mathcal{Q}^{(2)}]$ denote $[\beta_L \cdot \mathcal{Q} \cdot \beta_R]$, $[\mathcal{Q}^{(3)}]$ denote

¹ \mathcal{Q} is also the quotient ring of $\mathcal{G}_\infty(\tilde{\beta}_R)/\mathcal{G}_\infty$ and $\mathcal{G}_\infty(\tilde{\beta}_L)/\mathcal{G}_\infty$.

$[(1+i)\beta_L \cdot \mathcal{Q}]$ and their coordinates are:

$$\begin{aligned}
& \begin{cases} \Psi(q_0^{(0)}) = [0, 0, 0, 0] \\ \Psi(q_1^{(0)}) = [1, 0, 0, 0] \\ \Psi(q_2^{(0)}) = [0, 1, 0, 0] \\ \Psi(q_3^{(0)}) = [1, 1, 0, 0] \end{cases} \\
& \begin{cases} \Psi(q_0^{(1)}) = [0, 0, 0, 0] \\ \Psi(q_1^{(1)}) = [1, 1, 1, 1] \\ \Psi(q_2^{(1)}) = [1, 0, 1, 0] \\ \Psi(q_3^{(1)}) = [0, 1, 0, 1] \end{cases} \\
& \begin{cases} \Psi(q_0^{(2)}) = [0, 0, 0, 0] \\ \Psi(q_1^{(2)}) = [i, 1, i, 1] \\ \Psi(q_2^{(2)}) = [1+i, 1+i, 0, 0] \\ \Psi(q_3^{(2)}) = [1+2i, 2+i, i, 1] \end{cases} \\
& \begin{cases} \Psi(q_0^{(3)}) = [0, 0, 0, 0] \\ \Psi(q_1^{(3)}) = [1+i, 1+i, 1+i, 1+i] \\ \Psi(q_2^{(3)}) = [1+i, 0, 1+i, 0] \\ \Psi(q_3^{(3)}) = [2+2i, 1+i, 2+2i, 1+i] \end{cases}
\end{aligned} \tag{5.35}$$

For each partition, the quotient elements can be labelled by 2 bits that $(b_1, b_0) \leftrightarrow q_i$. For example in \mathcal{Q} , we have:

$$\begin{cases} (0, 0) \leftrightarrow q_0 = [0, 0, 0, 0] \\ (0, 1) \leftrightarrow q_1 = [1, 0, 0, 0] \\ (1, 0) \leftrightarrow q_2 = [0, 1, 0, 0] \\ (1, 1) \leftrightarrow q_3 = [1, 1, 0, 0] \end{cases} \tag{5.36}$$

Therefore, the elements of the quotient ring $[\mathcal{G}_\infty/2\mathcal{G}_\infty]$ can be labelled by 8 bits noted by $(b_7 \cdots b_0)$. The coordinates can be calculated from $(b_7 \cdots b_0)$ by:

$$\bar{s} = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} = \mathbf{G}_{2\mathcal{G}}^{(c)} \cdot [b_7 \cdots b_0]^T \tag{5.37}$$

where $\mathbf{G}_{2\mathcal{G}}^{(c)}$ is given by:

$$\mathbf{G}_{2\mathcal{G}}^{(c)} = \begin{bmatrix} 1+i & 1+i & 1+i & i & 1 & 1 & 0 & 1 \\ 0 & 1+i & 1+i & 1 & 0 & 1 & 1 & 0 \\ 1+i & 1+i & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1+i & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.38}$$

The generation of the $[\mathcal{G}_\infty/2\mathcal{G}_\infty]$ holds for modulo $2\mathcal{G}$ and the following operation generates the quotient element:

$$\bar{s} = \left(\mathbf{G}_{2\mathcal{G}}^{(c)} \cdot [b_7 \cdots b_0]^T \right) \bmod 2 \tag{5.39}$$

By separating the real part and imaginaire part, we have:

$$\begin{bmatrix} \Re(\bar{s}) \\ \Im(\bar{s}) \end{bmatrix} = \mathbf{G}_{2\mathcal{G}} \cdot [b_7 \cdots b_0]^T \text{ mod } 2 \quad (5.40)$$

where

$$\mathbf{G}_{2\mathcal{G}} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.41)$$

In fact, the bit mapping matrix \mathbf{G} is unimodular and its inversion is given by:

$$\mathbf{G}_{2\mathcal{G}}^{-1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \quad (5.42)$$

such that

$$\mathbf{G}_{2\mathcal{G}} \cdot \mathbf{G}_{2\mathcal{G}}^{-1} = \mathbf{I}_8 \text{ mod } 2 \quad (5.43)$$

The demapping procedure can be easily realized by:

$$[b_7 \cdots b_0]^T = \mathbf{G}_{2\mathcal{G}}^{-1} \cdot \begin{bmatrix} \Re(\bar{s}) \\ \Im(\bar{s}) \end{bmatrix} \text{ mod } 2 \quad (5.44)$$

This coset labelling can be developed to $4\mathcal{G}_\infty$ where the chain of partitions following $[4\mathcal{G}_\infty/2\mathcal{G}_\infty/\mathcal{G}_\infty]$. In the same way, the matrix generating the quotient ring $[4\mathcal{G}_\infty/\mathcal{G}_\infty]$ is a bit mapping matrix which maps 16 bits into 8 integers (or 4 complex integers) as follows:

$$\begin{aligned} \mathbf{G}_{4\mathcal{G}} &= \begin{bmatrix} 2 & 2 & 2 & 0 & 2 & 2 & 0 & 2 & 3 & 1 & 3 & 2 & 3 & 1 & 0 & 1 \\ 0 & 2 & 2 & 2 & 0 & 2 & 2 & 0 & 0 & 3 & 1 & 3 & 0 & 1 & 1 & 0 \\ 2 & 2 & 0 & 2 & 2 & 2 & 0 & 0 & 3 & 1 & 2 & 1 & 3 & 1 & 0 & 0 \\ 0 & 2 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 3 & 2 & 3 & 0 & 3 & 0 & 0 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 0 & 3 & 1 & 3 & 3 & 0 & 0 & 0 & 0 \\ 0 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{G}_{4\mathcal{G}/2\mathcal{G}} & \mathbf{G}_{4\mathcal{G}}^* \end{bmatrix} \end{aligned} \quad (5.45)$$

where $\mathbf{G}_{4\mathcal{G}}^*$ and $\mathbf{G}_{4\mathcal{G}/2\mathcal{G}}$ are both 8×8 matrix and we notice that $\mathbf{G}_{4\mathcal{G}/2\mathcal{G}}$ is $2\mathbf{G}_{2\mathcal{G}}$.

The coordinates of the quotient ring $[4\mathcal{G}_\infty/\mathcal{G}_\infty]$ are generated by:

$$\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} = \mathbf{G}_{4\mathcal{G}} \cdot [b_{15} \cdots b_0]^T \text{ mod } 4 \quad (5.46)$$

Now the question is how to demap from \bar{s} to $[b_{15} \cdots b_0]^T$. Considering the enties of $\mathbf{G}_{4\mathcal{G}/2\mathcal{G}} = 2\mathbf{G}_{2\mathcal{G}}$, we can project $[\mathfrak{R}(\bar{s}), \mathfrak{S}(\bar{s})]^T$ by modulo 2 that:

$$\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix}_{\text{proj}} = \begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} \text{ mod } 2 \quad (5.47)$$

From (5.48), we have:

$$\begin{aligned} \begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix}_{\text{proj}} &= \mathbf{G}_{4\mathcal{G}} \cdot [b_{15} \cdots b_0]^T \text{ mod } 2 \\ &= (\mathbf{G}_{4\mathcal{G}} \text{ mod } 2) \cdot [b_{15} \cdots b_0]^T \end{aligned} \quad (5.48)$$

where

$$\mathbf{G}_{4\mathcal{G}} \text{ mod } 2 = \begin{bmatrix} (\mathbf{G}_{4\mathcal{G}/2\mathcal{G}} \text{ mod } 2) & (\mathbf{G}_{4\mathcal{G}}^* \text{ mod } 2) \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{G}_{2\mathcal{G}} \end{bmatrix} \quad (5.49)$$

Here $(\mathbf{G}_{4\mathcal{G}}^* \text{ mod } 2)$ coincides with $\mathbf{G}_{2\mathcal{G}}$ because the partition chain follows $[4\mathcal{G}_\infty/2\mathcal{G}_\infty/\mathcal{G}_\infty]$.

We have then:

$$\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix}_{\text{proj}} \text{ mod } 2 = \mathbf{G}_{2\mathcal{G}} \cdot [b_7 \cdots b_0]^T \text{ mod } 2 \quad (5.50)$$

Therefore, we can give a demapping procedure of $[b_7 \cdots b_0]$ from the projected symbol as we do in $[2\mathcal{G}_\infty/2\mathcal{G}_\infty]$ case that

$$[b_7 \cdots b_0]^T = \mathbf{G}_{2\mathcal{G}}^{-1} \cdot \begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix}_{\text{proj}} \text{ mod } 2 \quad (5.51)$$

The demapping of $[b_{15} \cdots b_8]$ is based on the result of $[b_7 \cdots b_0]$. Considering that:

$$\mathbf{G}_{4\mathcal{G}/2\mathcal{G}} \cdot [b_{15} \cdots b_8]^T \text{ mod } 4 = \left(\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} - \mathbf{G}_{4\mathcal{G}}^* \cdot [b_7 \cdots b_0]^T \right) \text{ mod } 4 \quad (5.52)$$

It is remarkable $\left(\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} - \mathbf{G}_{4\mathcal{G}}^* \cdot [b_7 \cdots b_0]^T \right)$ is a multiple of 2, then we have:

$$\mathbf{G}_{2\mathcal{G}} \cdot [b_{15} \cdots b_8]^T \text{ mod } 2 = \frac{1}{2} \left(\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} - \mathbf{G}_{4\mathcal{G}}^* \cdot [b_7 \cdots b_0]^T \right) \text{ mod } 2 \quad (5.53)$$

The demapping of $[b_{15} \cdots b_8]$ is given by

$$[b_{15} \cdots b_8]^T = \mathbf{G}_{2\mathcal{G}}^{-1} \cdot \frac{1}{2} \left(\begin{bmatrix} \mathfrak{R}(\bar{s}) \\ \mathfrak{S}(\bar{s}) \end{bmatrix} - \mathbf{G}_{4\mathcal{G}}^* \cdot [b_7 \cdots b_0]^T \right) \text{ mod } 2 \quad (5.54)$$

The mapping and demapping procedures are very easy to implemented in hardware level since all the processings are done in binary format.

5.3 Application and performance

Now let us consider the Relay-SISO case where we use 1/2 based convolutional code as the FEC. In order to adapt different throughputs, the convolutional code is punctured to rate 2/3 and 3/4. For the partition chain of Golden code, each partition will bring a determinant gain of 2 but also a loss of 2 bits in STBC coding. We propose to use the punctured convolutional code to compensate the 2 bits loss.

It is true that the error-correction capacity will decrease with punctured convolutional code. In Figure 5.1(a), the convolutional code performances of the 3 rates are evaluated as well as the non-code case: we see the 2/3-rate code is with 0.5dB loss compared to the 1/2-rate one and the 3/4-rate code is with about 1.0dB loss. Asymptotically, the gap between the punctured convolutional codes and the original one can be shortened. For Rayleigh fading channel, the loss due to the puncture can be even smaller, see Figure 5.1(b): there is nearly no loss for the 2/3-rate case while about 0.7dB loss in 3/4-rate case. This performance is not astonishing since the channel fading plays a role more important than the coding gain in this configuration. Thus a simple combination of the partitioned Golden code with the punctured convolutional code can work in this case.

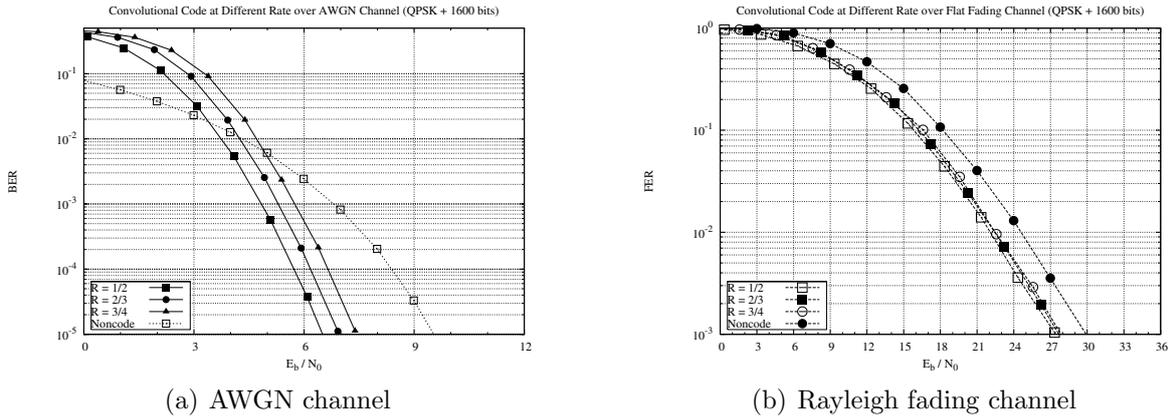


Figure 5.1: 802.11a convolutional code (133,171) with different puncture mode

5.3.1 New modulation and coding scheme for data rate 12Mbps in Relay-SISO

Let us investigate the data rate 12Mbps case as shown in Table 3.1 where the constellation is QPSK and the convolutional code rate is 1/2. A codeword of Golden code contains 8 bits in this case. This codeword is generated from the 8 bits in using the bit mapping procedure as described in 5.2.

Now let us restrict the coding in only in $[\mathcal{G} \cdot \beta_R]$ such that a codeword is generated from 6 bits by forcing the b_1 and b_0 to 0 in 5.37. This processing is equivalent to associate a coding rate of 3/4 in the STBC coding procedure. For the FEC part, instead of using 1/2-rate convolutional code, we use the punctured version for a code rate of 2/3. Totally, we have 1/2 code rate while the STBC part provides a determinant gain of 2.

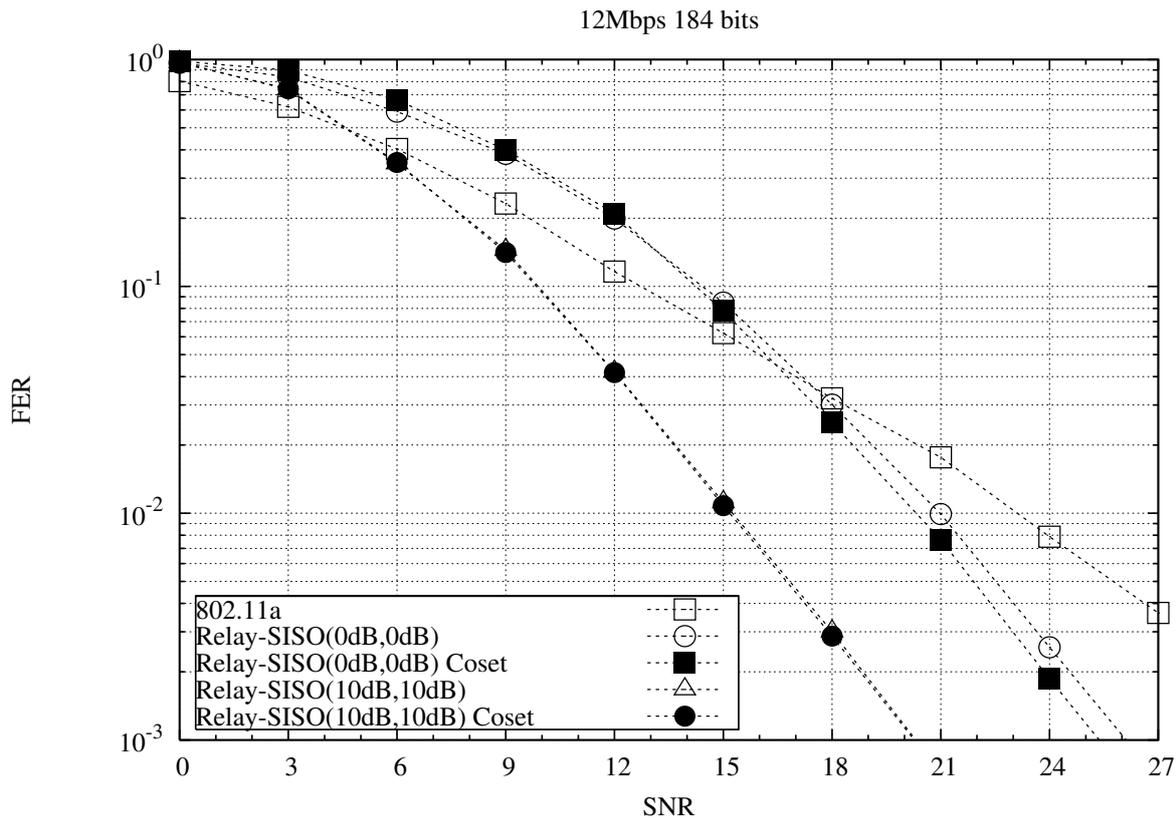


Figure 5.2: Partitioned Golden code for Relay-SISO 12Mbps

As shown in Figure 5.2, we find a coding gain about 0.8dB for the Relay-SISO in LPG configuration at $FER=10^{-3}$. However, there is nearly no gain for the HPG configuration.

5.3.2 New modulation and coding scheme for data rate 24Mbps in Relay-SISO

The 24Mbps transmission uses the 16-QAM modulation and the 1/2-rate convolutional code. By applying the partition chain, a Golden codeword will be generated from 12 bits where we take the candidates in $[\beta_L \cdot \mathcal{G} \cdot \beta_R]$ instead of $[\mathcal{G}]$. This operation will bring a code rate of 3/4 for the STBC part and we keep use the 2/3-rate punctured convolutional code as FEC.

In Figure 5.3, the coding gain is much important than the 12Mbps case since the extended Golden code provides nearly 6dB gain in determinant while the FEC is still 2/3-rate convolutional code. For $FER=10^{-3}$, a coding gain about 2.2dB is found for the LPG configuration and about 3dB is found for the HPG configuration.

5.3.3 Application in 2×2 MIMO system

We can apply these modulation and coding schemes in the context of classic MIMO system where the Golden code is used conjointly with convolutional code. As shown in

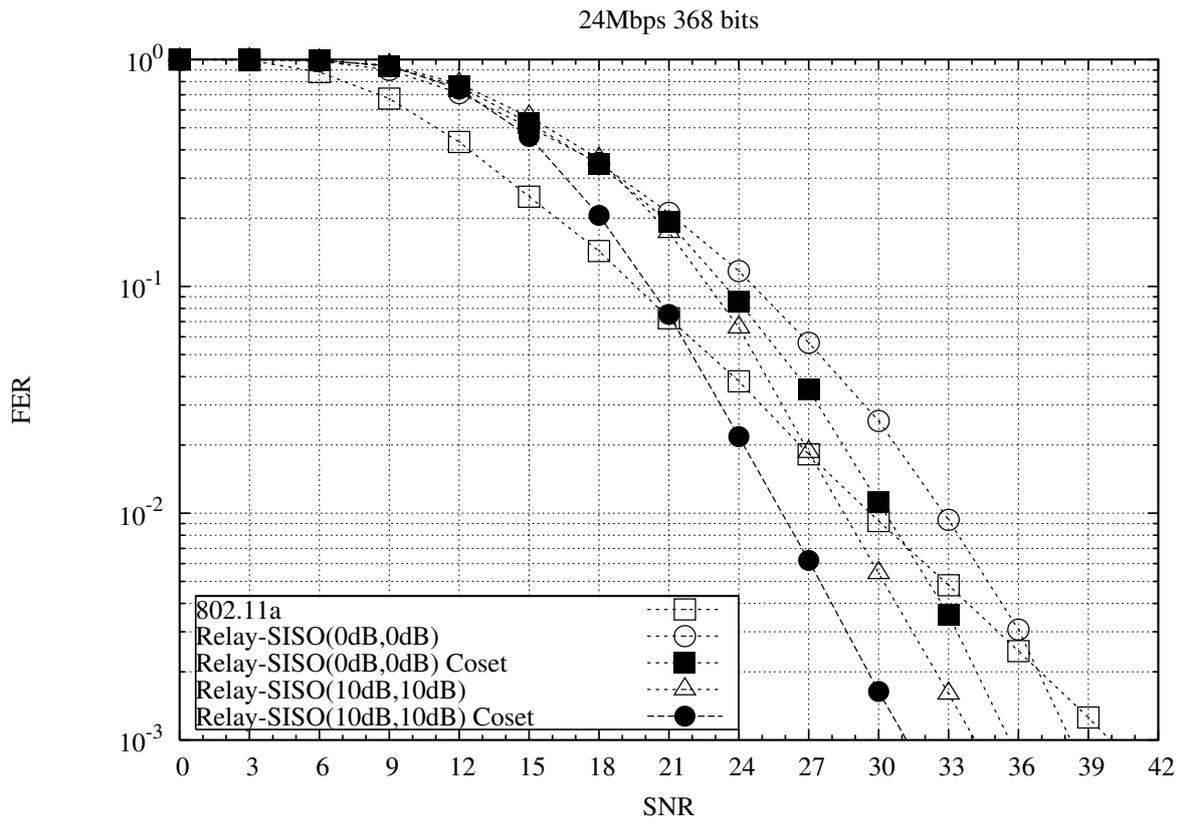


Figure 5.3: Partitioned Golden code for Relay-SISO 24Mbps

Figure 5.4, this combination shows better performance than the unpartitioned Golden coding with 1/2-rate convolutional code. In the configuration of 16-QAM with 1/2-rate convolutional code, this proposition can achieve a coding gain of 3dB.

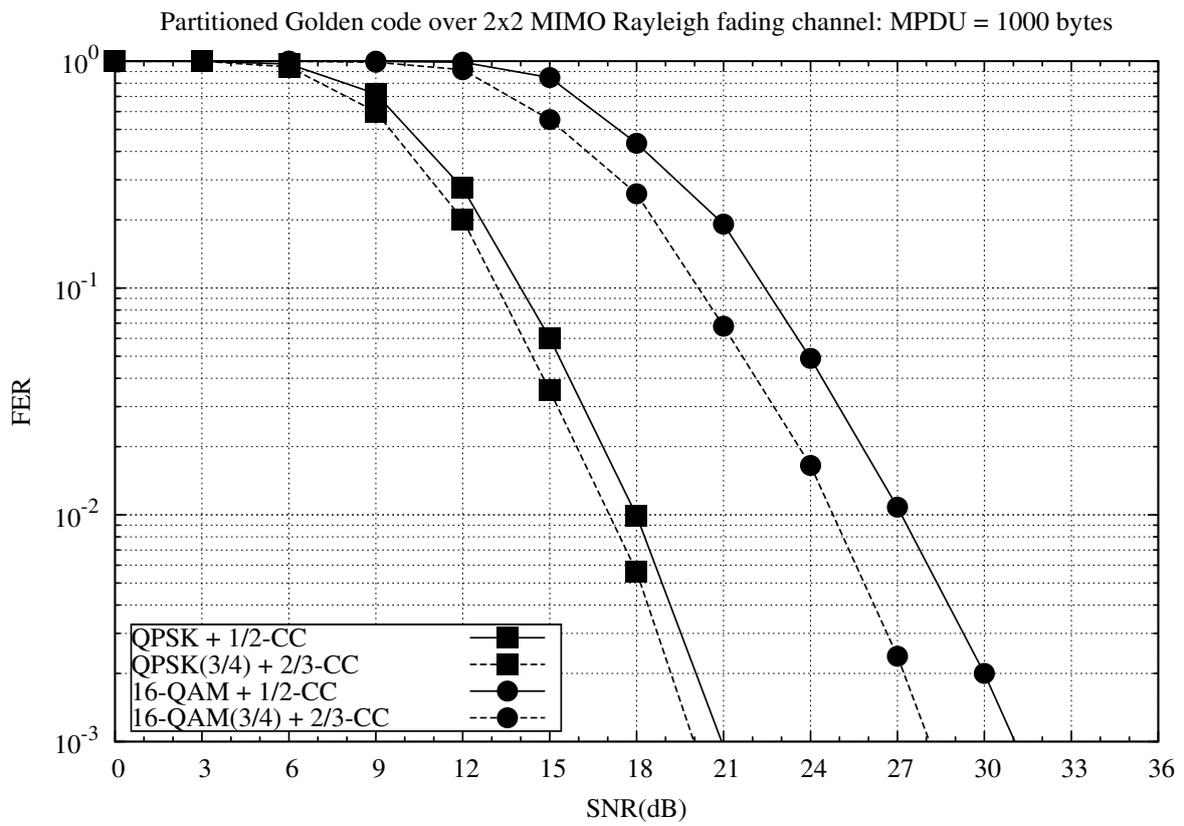


Figure 5.4: Partitioned Golden code in MIMO Rayleigh fading channel

CHAPTER 6

RELAY-MIMO

In the previous chapters, the application of cooperative transmission in the context of IEEE 802.11a PHY layer has been presented. We will introduce in this chapter a cooperative system which is based on the single-relay two-antenna NAF scheme as described in section 1.2.

In the new arriving IEEE 802.11n [33] standard, the MIMO technology is applied to increase both the robustness, *diversity gain*, and the throughput, *multiplexing gain*. The cooperation system, named Relay-MIMO PHY layer, is proposed as an extension of IEEE 802.11n. This system combines the cooperative diversity with the MIMO diversity to enhance the quality of transmission. Like Relay-SISO system, this new PHY protocol is realized by introducing new STBC encoder/decoder in IEEE 802.11n system.

The cooperative scenario consists still of 3 terminals, respectively the source terminal, the destination terminal and the relay terminal. We suppose that each terminal is equipped with two antennas and the 4×4 Perfect code is applied. As shown in section 1.3, this system is equivalent to a 16×16 MIMO system thus the decoding complexity can be very high even for simple constellation. We use the sub-optimal decoder which is studied in chapitre 2 to give a quick and near-ML performance. Simulation results are reported in both flat fading channel context and multipath channel context.

6.1 IEEE 802.11n PHY Layer

We present the IEEE 802.11n PHY layer protocol in this section.

The IEEE 802.11n PHY layer operates with 20MHz/40MHz bandwidth, which corresponds a 64-FFT or 128-FFT based OFDM modulation. As an extension of IEEE 802.11a PHY layer, this standard has a lot of improvements related to use of MIMO technique. We will focus on the modifications for the MIMO system.

Considering the compatibility with legacy IEEE 802.11a system, the Mixed Mode (MM) is introduced as a very important transmission mode in the PLCP protocol. The other supported modes are the legacy IEEE 802.11a mode and the Green Field (GF) mode. The GF mode is incompatible with legacy system that we are interested only at the MM transmission which is also the most implemented mode on the products in the market.

The frame format on MM is shown in Figure 6.1.

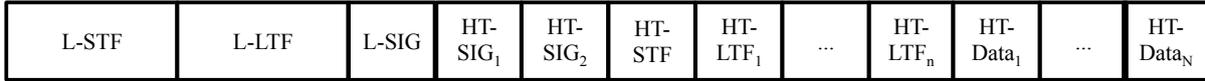


Figure 6.1: 802.11n mixed mode frame

This new frame format contains the legacy compatible preambles and a new portion for the High-Throughput (HT) transmission. The legacy portion is composed of Legacy Short Training Field (L-STF), Legacy Long Training Field (L-LTF) and Legacy SIGNAL Field (L-SIG) and High-Throughput SIGNAL Field (HT-SIG). The HT portion is composed of High-Throughput Short Training Field (HT-STF), High-Throughput Long Training Field (HT-LTF) and High-Throughput Data Field (HT-DATA). The legacy IEEE 802.11a terminals will translate the HT-SIG and HT portion as a valid IEEE 802.11a data field which will be rejected as a wrong frame by the MAC layer.

Like in IEEE 802.11a, L-STF is used for AGC training, the signal detection and timing recovery together with GI2 of L-LTF. L-LTF is used for the legacy channel coefficient training which is demanded for the L-SIG and HT-SIG decoding. The HT-SIG field is modulated in using Quadrature BPSK (Q-BPSK) modulation for the discrimination of HT mode from legacy mode. The HT-STF is used for additional CFO estimation. The HT-LTFs provide the MIMO channel coefficients estimation.

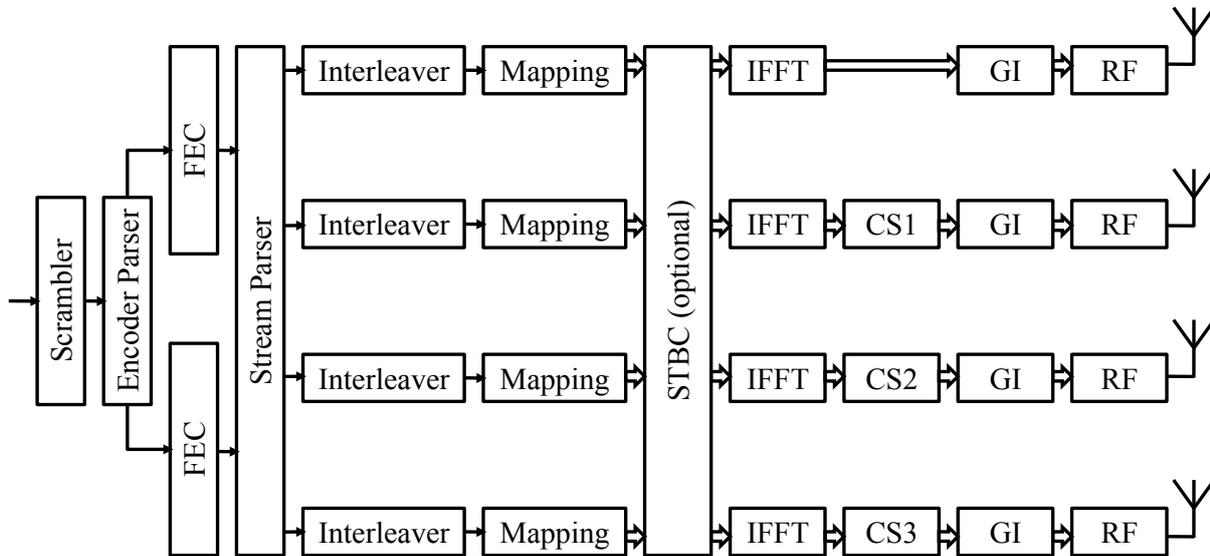


Figure 6.2: 802.11n general architecture of processing

Figure 6.2 gives a general 802.11n processing scheme. An MPDU is scrambled then fed into the FEC module. In the FEC module, IEEE 802.11n standard introduces the Low-Density Parity-Check (LDPC) code and adds the 5/6 convolutional code as shown in Figure 6.3. With the encoder parser module, the binary stream is fed alternatively into two convolutional encoders. This processing enables the use of two Viterbi decoders at the receiver side in order to accelerate the decoding when the throughput is high.

The dimension of interleaver is configured to $4N_{\text{BPSCS}}(i_{\text{SS}}) \times 13$ for 20MHz case and $6N_{\text{BPSCS}}(i_{\text{SS}}) \times 18$ for 40MHz case where i_{SS} is the index of spatial stream and $N_{\text{BPSCS}}(i_{\text{SS}})$

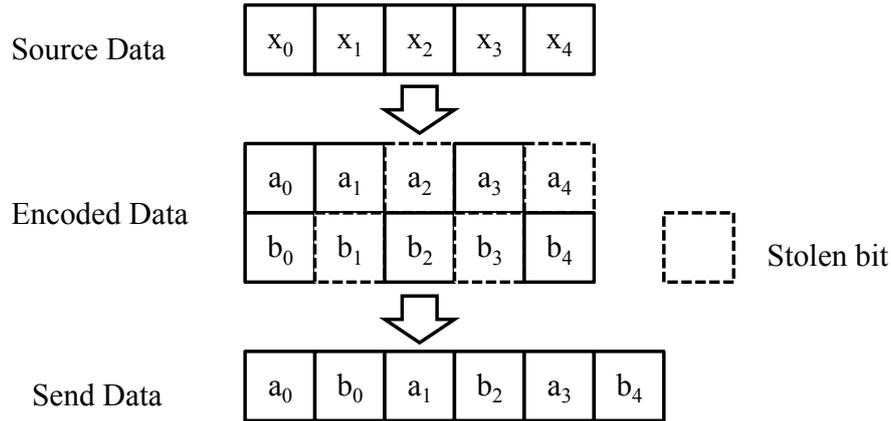


Figure 6.3: Output of convolutional code at $R = 5/6$

is the number of coded bits per single carrier for each spatial stream. In addition to the bits permutation, a frequency rotation is applied when there are more than one spatial stream.

IEEE 802.11n supports up to $N_{SS} = 4$ spatial streams. In the module of STBC, the optional Alamouti STBC can be applied after the QAM symbol mapping and it will generate N_{STS} space-time streams¹. When beamforming or spatial expansion processing is needed, spatial mapping will project N_{STS} space-time streams to N_{TX} transmit antennas. This operation is realized by multiplying at each sub-carrier the spatial mapping matrix \mathbf{Q}_k whose dimension is $N_{TX} \times N_{STS}$.

A Cyclic Shift (CS) is also applied when there are more than one antenna: either by multiplying the Cyclic Shift Delay (CSD) matrix² in frequency domain or by direct shifting in time domain when $N_{TX} = N_{STS}$. The CS parameters are detailed in Tables 6.1 and 6.2.

Table 6.1: 802.11n cyclic shift value definition for non-HT portion

Num. of Tx	CS for chain 1	CS for chain 2	CS for chain 3	CS for chain 4
1	0ns	-	-	-
2	0ns	-200ns	-	-
3	0ns	-100ns	-200ns	-
4	0ns	-50ns	-100ns	-150ns

Table 6.2: 802.11n cyclic shift value definition for HT portion

Num. of Tx	CS for chain 1	CS for chain 2	CS for chain 3	CS for chain 4
1	0ns	-	-	-
2	0ns	-400ns	-	-
3	0ns	-400ns	-200ns	-
4	0ns	-400ns	-200ns	-600ns

¹In case of non STBC, $N_{SS} = N_{STS}$.

²The spatial mapping matrix \mathbf{Q}_k takes into account the CSD matrix.

At last, the complex symbols streams are modulated in the IFFT module where a new frequency mapping using 56 sub-carriers is applied for HT portion. An illustration of this frequency mapping for the bandwidth of 20MHz is shown in Figure 6.4.

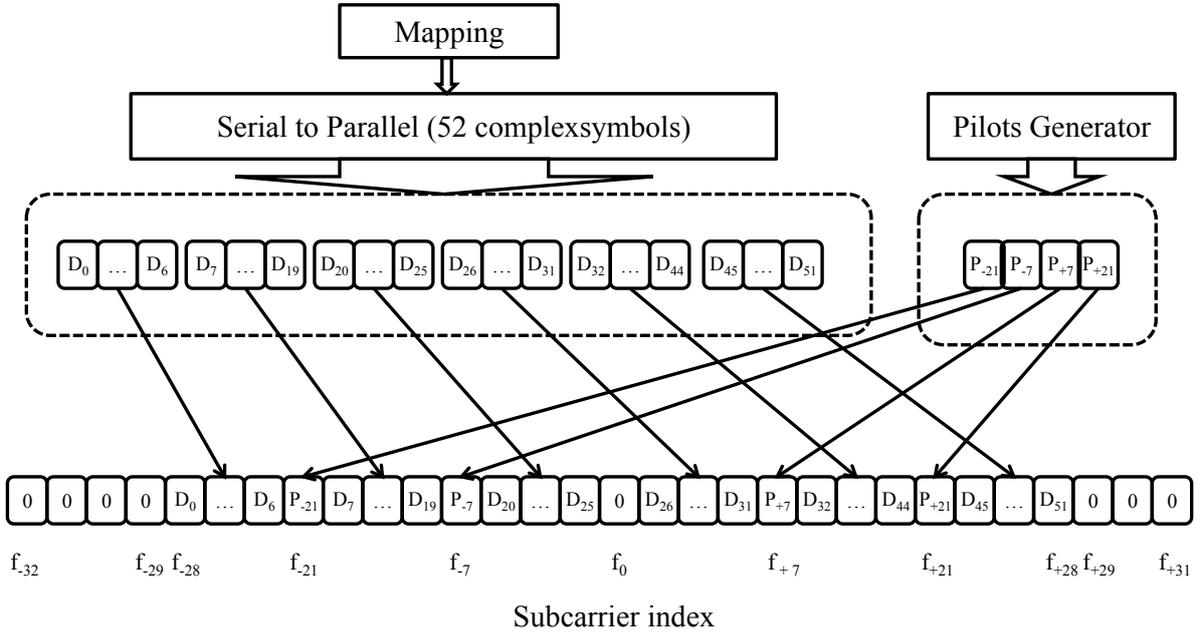


Figure 6.4: IEEE 802.11n 20MHz frequency mapping for the HT portion OFDM symbol: 52 subcarriers are used for transmission of data symbol while 4 subcarriers are used for pilots at frequency index $-21, -7, +7, +21$; the other 8 subcarriers including DC are kept no used which generate protection band.

We will introduce the single-relay two-antenna NAF system in the mandatory 20MHz part of IEEE 802.11n standard with $N_{SS} = N_{STS} = N_{TX} = 2$ with Equal Modulation (EQM) configuration. The supported Modulation and Coding Scheme (MCS) for $N_{SS} = 2$ is listed in Table 6.3, where N_{SS} denotes the number of spatial streams, N_{SD} denotes the number of data sub-carriers, N_{SP} denotes the number of pilots sub-carriers, N_{CBPS} denotes the number of coded bits per OFDM symbol and N_{DBPS} denotes the number of data bits per OFDM symbol. The Alamouti coding and spatial mapping options are not applied in this context.

Table 6.3: IEEE 802.11n MCS parameters for 20MHz, $N_{SS} = 2$, GI=800ns, EQM

MCS index	Date rate (Mbps)	Mod.	Coding rate (R)	N_{BPSCS} (i_{SS})	N_{SD}	N_{SP}	N_{CBPS}	N_{DBPS}
8	13.0	BPSK	1/2	1	52	4	104	52
9	26.0	QPSK	1/2	2	52	4	208	104
10	39.0	QPSK	3/4	2	52	4	208	156
11	52.0	16-QAM	1/2	4	52	4	416	208
12	78.0	16-QAM	3/4	4	52	4	416	312
13	104.0	64-QAM	2/3	4	52	4	624	416
14	117.0	64-QAM	3/4	4	52	4	624	468
15	130.0	64-QAM	5/6	4	52	4	624	520

6.2 Relay-MIMO PHY Layer

The proposed Relay-MIMO PHY layer is based on the single-relay two-antenna NAF scenario described in section 1.2. This system is an adaption of the IEEE 802.11n standard with cooperative transmission support.

Relay-MIMO operates with 2 spatial streams. Let $\underline{s}^{(1)} = \{s_p^{(1)}\}$ and $\underline{s}^{(2)} = \{s_p^{(2)}\}$ denote the 2 QAM symbol streams. We group them by 16:

$$\underline{s}_q = \{s_{q,1}, \dots, s_{q,16}\} = \{s_{8q}^{(1)}, \dots, s_{8q+7}^{(1)}, s_{8q}^{(2)}, \dots, s_{8q+7}^{(2)}\} \quad (6.1)$$

We apply the 4×4 Perfect code to each QAM block as in equation 1.57.

The coding procedure will generate a stream of 4×4 codewords, denoted by $\{\mathbf{X}_q\}$:

$$\mathbf{X}_q = \begin{bmatrix} X_{q,11} & X_{q,12} & X_{q,13} & X_{q,14} \\ X_{q,21} & X_{q,22} & X_{q,23} & X_{q,24} \\ X_{q,31} & X_{q,32} & X_{q,33} & X_{q,34} \\ X_{q,41} & X_{q,42} & X_{q,43} & X_{q,44} \end{bmatrix} \quad (6.2)$$

The number of data sub-carriers is 52 for the HT portion. Thus, the codeword stream is grouped by 52 and a CB is defined by $\bar{\mathbf{X}}_n$:

$$\bar{\mathbf{X}}_n = \{\mathbf{X}_{52n}, \dots, \mathbf{X}_{52n+51}\} \quad (6.3)$$

Let $\mathbf{X}_{n,k}$ denote the codeword on sub-carrier k of the n^{th} CB and let $X_{n,k,ij}$ denote the i^{th} -row j^{th} -column entry. $\underline{\mathbf{X}}_k = \{\mathbf{X}_{0,k}, \dots, \mathbf{X}_{N_{\text{CB}}-1,k}\}$ is the codeword stream on sub-carrier k , where N_{CB} is the number of CB.

We follow the transmission pattern of Relay-SISO by transmitting in 2 parts of data, denoted by data A and data B respectively. Then the coding processing is shown in Figures 6.5 and 6.6 where the 4×4 Perfect encoder and a buffer are added.

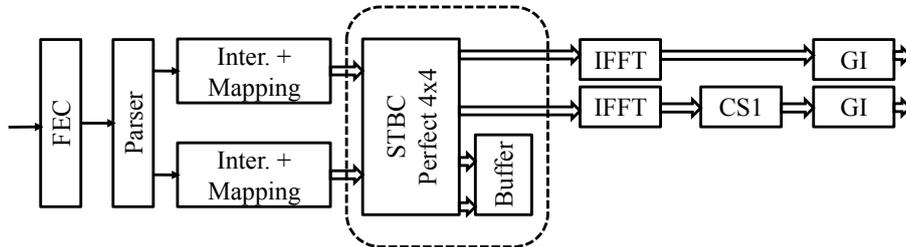


Figure 6.5: Relay-MIMO PHY layer for data A: the 4×4 Perfect code is used where the inputs are 2 parallel QAM symbol streams. The coded signal is partially sent on 2 Tx antennas as in IEEE 802.11n configuration while the rests are kept in a buffer.

Let $A_{i,m,k}$ denote the symbol of data A transmitted on i^{th} antenna on sub-carrier k in m^{th} OFDM symbol and $B_{i,m,k}$ for the symbol of data B. The transmitted sequences is

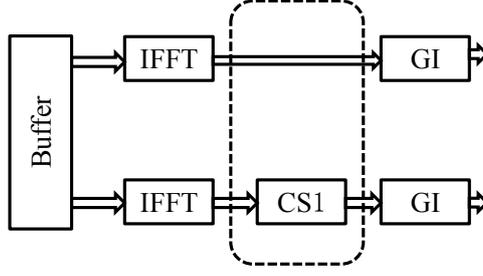


Figure 6.6: Relay-MIMO PHY layer for data B: the memoried codewords are transmitted

given by:

$$\left\{ \begin{array}{l} \underline{A}_{1,k} = \sqrt{\pi_{s1}/2} \{ X_{0,k,11}, \dots, X_{0,k,14}, \dots, X_{N_{CB}-1,k,11}, \dots, X_{N_{CB}-1,k,14} \} \\ \underline{A}_{2,k} = \sqrt{\pi_{s1}/2} \{ X_{0,k,21}, \dots, X_{0,k,24}, \dots, X_{N_{CB}-1,k,21}, \dots, X_{N_{CB}-1,k,24} \} \\ \underline{B}_{1,k} = \sqrt{\pi_{s2}/2} \{ X_{0,k,31}, \dots, X_{0,k,34}, \dots, X_{N_{CB}-1,k,31}, \dots, X_{N_{CB}-1,k,34} \} \\ \underline{B}_{2,k} = \sqrt{\pi_{s2}/2} \{ X_{0,k,41}, \dots, X_{0,k,44}, \dots, X_{N_{CB}-1,k,41}, \dots, X_{N_{CB}-1,k,44} \} \end{array} \right. \quad (6.4)$$

where π_{s1} and π_{s2} are the power factors and we normalize the transmit power by $1/\sqrt{2}$.

In such way:

$$\left\{ \begin{array}{l} \left[\begin{array}{l} A_{1,4n,k}, \dots, A_{1,4n+3,k} \\ A_{2,4n,k}, \dots, A_{2,4n+3,k} \\ B_{1,4n,k}, \dots, B_{1,4n+3,k} \\ B_{2,4n,k}, \dots, B_{2,4n+3,k} \end{array} \right] \end{array} \right. \quad (6.5)$$

are from the same 4×4 Perfect codeword $\mathbf{X}_{n,k}$.

The system parameters are calculated as follows:

$$N_{SYM} \leftarrow \lceil (16 + 8 \times LENGTH + 6) / N_{DBPS} \rceil$$

$$N_{CB} \leftarrow \lceil N_{SYM} / 16 \rceil$$

$$N_{SYM} \leftarrow 16 \times N_{CB} \quad (6.6)$$

$$N_{PAD} \leftarrow N_{SYM} \times N_{DBPS} - (16 + 8 \times LENGTH + 6) \quad (6.7)$$

where N_{DBPS} takes the value defined in Table 6.3.

The LENGTH in HT-SIG is modified by:

$$LENGTH \leftarrow LENGTH + N_{DBPS} \times (N_I + 7) / 8 \quad (6.8)$$

where the N_I is the time occupation for the idle field and we consider Relay-MIMO adds 7 padding OFDM time symbols in the worst case. The legacy LENGTH in the L-SIG should be also modified as shown in [33].

In addition, considering the CS processing applied in 802.11n, we remark that during the transmission for data A, the signal sent on antenna 1 is non-shifted while the signal sent on antenna 2 is cyclic-shifted by the CS1 block, see Table 6.2. The same processing is taken on data B where the signal transmitted on antenna 1 is non-shifted and the signal transmitted on antenna 2 is processed with the CS1 block.

At the destination's side, the system can be considered as a 4×4 virtual-MIMO system as shown in Figure 6.7. This configuration helps to implement the STBC encoder/decoder in the full transceiver chain.

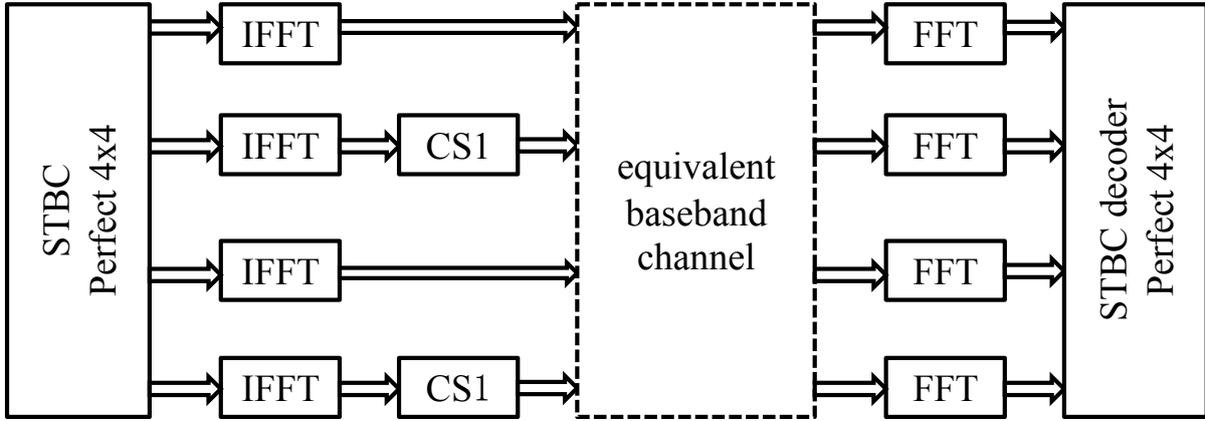


Figure 6.7: Relay-MIMO baseband equivalent transmission viewed at the receiver's side

As Relay-SISO, we define the 3 cooperative frame formats: the S frame in Figure 6.8, the R frame in Figure 6.9 and the D frame in Figure 6.10.



Figure 6.8: Relay-MIMO S frame: the S frame is sent by the source terminal



Figure 6.9: Relay-MIMO R frame: the R frame is sent by the relay terminal



Figure 6.10: Relay-MIMO D frame: the D frame is received by the destination terminal

In the S frame, the reserved bit of L-SIG is modified to inform the cooperation mode and the IEEE 802.11n preambles are kept. The HT data field contains 3 subfields: the data A field, the data B field and the idle field which is dedicated to the half-STF symbol and the relay added preambles.

In the R frame, the relay added preambles are detailed in Figure 6.11. The legacy training field consists of A-LTF and R-LTF. The HT training field consists of Amplified High-Throughput Long Training Field (AHT-LTF) and Relayed High-Throughput Long Training Field (RHT-LTF),

The AHT-LTFs contains the same HT training symbols like HT-LTFs while the RHT-LTFs are the relayed HT-LTFs. For the CFO consideration, the relay generated AHT-LTF is phase-corrected like A-LTF as explained in section 3.3.

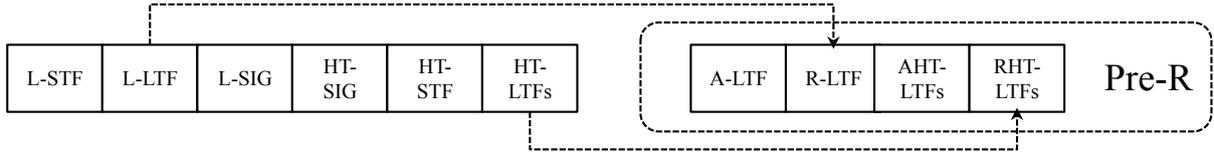


Figure 6.11: Relay-MIMO R frame's preambles

There is no extension HT-LTF³. We remark that the training symbols HT-LTFs are not repeated that the noise power estimation should be performed with legacy long training symbols, A-LTF and R-LTF. AHT-LTF are employed to estimate the the coefficients of channel $r \rightarrow s$ and RHT-LTF is for the product channel. Both AHT-LTF and RHT-LTF are configured with same number of HT long training symbols as HT-LTF.

In case of $N_{SS} = 2$, HT-LTF contains 2 OFDM training symbols which is the same for AHT-LTF and RHT-LTF. By taking into account the half-STF, A-LTF and R-LTF, we attribute $N_I = 9$ OFDM symbols in the idle field.

This structure of frame will be compatible for IEEE 802.11a, IEEE 802.11n and Relay-SISO systems because these system will update their NAV by verifying the LENGTH parameter in L-SIG or HT-SIG field, as shown in Figure 6.12.

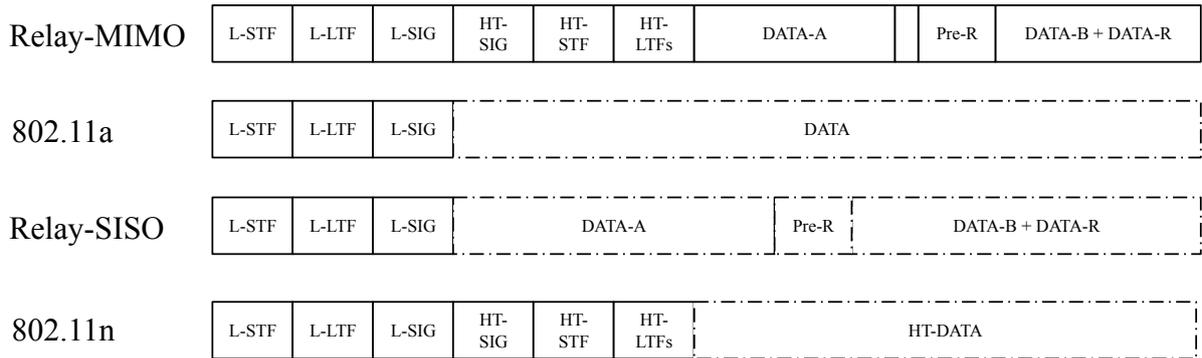


Figure 6.12: Relay-MIMO compatibility with IEEE 802.11a, IEEE 802.11n and Relay-SISO PHY layer

6.3 Relay-MIMO preambles and cooperation procedure

We use the same notations as in section 3.3.

In the non-HT portion, 52 sub-carriers are used and the transmitted symbols are normalized in frequency domain by $K = 8/\sqrt{52}$; in the HT portion, 56 sub-carriers are used then transmitted symbols are normalized by $K_{HT} = 8\sqrt{56}$. The normalization factors K and K_{HT} are both considered as a part of the channel coefficients.

³In IEEE 802.11n system, more HT-LTFs can be added as an extension of data HT-LTF for the purpose of channel probing.

6.3.1 HT data field

The channels between the terminals are all 2×2 MIMO channel. Let $\underline{h}_{pq,i,j}$ denote the channel impulsional response from TX antenna j of terminal p to RX antenna i of terminal q .

In frequency domain, we have:

$$\bar{H}_{pq,i,j} = \text{DF}T_{64}(\bar{h}_{pq,i,j}) = \left\{ H_{k,pq,i,j} \right\}^T, \quad i, j = 1, 2 \quad (6.9)$$

On sub-carrier k , let $\mathbf{H}_{pq,k}$ denote the coefficients matrix of channel $p \rightarrow q$:

$$\mathbf{H}_{pq,k} = \begin{bmatrix} H_{pq,k,1,1} & H_{pq,k,1,2} \\ H_{pq,k,2,1} & H_{pq,k,2,2} \end{bmatrix} \quad (6.10)$$

By regrouping the symbols of data A and data B which belong to the same codeword, the system can be described on sub-carrier k by:

$$\mathbf{Y}_{n,k} = \mathbf{H}_k \mathbf{X}_{n,k} + \mathbf{V}_{n,k} + \mathbf{W}_{n,k} \quad (6.11)$$

where we have:

$$\begin{cases} \mathbf{H}_k = \begin{bmatrix} \sqrt{\pi_{s1}} \mathbf{H}_{sd,k} & 0 \\ \sqrt{\pi_{s1} \pi_{r2}} \mathbf{H}_{rd,k} \mathbf{B}_k \mathbf{H}_{sr,k} & \sqrt{\pi_{s2}} \mathbf{H}_{sd,k} \end{bmatrix} \\ \mathbf{V}_{n,k} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \sqrt{\pi_{r2}} \mathbf{H}_{rd,k} \mathbf{B}_k \mathbf{V}_{n,k,1} & \sqrt{\pi_{r2}} \mathbf{H}_{rd,k} \mathbf{B}_k \mathbf{V}_{n,k,2} \end{bmatrix} \end{cases} \quad (6.12)$$

The CS is also taken into account in MIMO channel coefficients matrix. Since the HT data transmission are the same as the HT portion training field, we will explain more precisely in the following parts.

The cooperative system's parameters are estimated with the legacy and HT channel preambles. We suppose the CFO is perfectly estimated and processed.

6.3.2 Cyclic shift processing

For notation, let j denote the source antenna index, m denote the relay antenna index and i denote the destination antenna index. The transmitted signal is marked by TX and the received signal is marked by RX. t is the index of OFDM symbol in each preamble field.

The long training symbol of the legacy preamble is denoted as follows: \underline{l} for L-LTF, \underline{a} for A-LTF and \underline{r} for R-LTF. The long training symbol of the HT preambles is marked by HT: $\underline{l}_{\text{HT}}$ for HT-LTF, $\underline{a}_{\text{HT}}$ for AHT-LTF and $\underline{r}_{\text{HT}}$ for RHT-LTF.

In the configuration of 2 antennas, the time symbol stream on second antenna is a cyclic-shifted version of the first antenna. According to Table 6.1, the second antenna is cyclic-shifted by 4 samples in the non-HT portion and 8 samples in the HT portion.

Let $[\underline{x}]_n$ be the cyclic-shift operation of \underline{x} by n samples such that:

$$[\underline{x}]_n = [x_{(n) \bmod N}, \dots, x_{(N-1+n) \bmod N}] \quad (6.13)$$

where $N = 64$ is the length of \underline{x} .

For A-LTF and L-LTF, the transmitted signals can be given by:

$$\begin{cases} \underline{l}_{j=1,t}^{(\text{TX})} = \sqrt{\pi_{s1}/2} \underline{l} \\ \underline{l}_{j=2,t}^{(\text{TX})} = \sqrt{\pi_{s1}/2} [\underline{l}]_4 \\ \underline{a}_{m=1,t}^{(\text{TX})} = \sqrt{\pi_{r2}/2} \underline{l} \\ \underline{a}_{m=2,t}^{(\text{TX})} = \sqrt{\pi_{r2}/2} [\underline{l}]_4 \end{cases}, \quad t = 1, 2 \quad (6.14)$$

where \underline{l} is the legacy long symbol time sequence as defined in [27].

The sequence of HT-LTF in frequency domain is given by $\bar{L}_{\text{HT}} = K_{\text{HT}} [L_{\text{HT},0}, \dots, L_{\text{HT},63}]$. In IEEE 802.11n standard, the HT-LTFs are transmitted in an orthogonal manner that for q^{th} HT-LTF symbol, we send on the p^{th} transmit antenna $P_{pq} \bar{L}_{\text{HT}}$ where P_{pq} is the row- p column- q entry of a 4×4 Hadamard matrix which is given by:

$$\mathbf{P}_{\text{HTLTF}} = \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ -1 & 1 & 1 & 1 \end{bmatrix} \quad (6.15)$$

For HT-LTF and AHT-LTF, the transmitted signal can be given by:

$$\begin{cases} \underline{l}_{\text{HT},j=1,t=1}^{(\text{TX})} = \sqrt{\pi_{s1}/2} \underline{l}_{\text{HT}} \\ \underline{l}_{\text{HT},j=2,t=1}^{(\text{TX})} = \sqrt{\pi_{s1}/2} [\underline{l}_{\text{HT}}]_8 \\ \underline{l}_{\text{HT},j=1,t=2}^{(\text{TX})} = -\sqrt{\pi_{s1}/2} \underline{l}_{\text{HT}} \\ \underline{l}_{\text{HT},j=2,t=2}^{(\text{TX})} = \sqrt{\pi_{s1}/2} [\underline{l}_{\text{HT}}]_8 \end{cases} \quad (6.16)$$

and

$$\begin{cases} \underline{a}_{\text{HT},m=1,t=1}^{(\text{TX})} = \sqrt{\pi_{r2}/2} \underline{l}_{\text{HT}} \\ \underline{a}_{\text{HT},m=2,t=1}^{(\text{TX})} = \sqrt{\pi_{r2}/2} [\underline{l}_{\text{HT}}]_8 \\ \underline{a}_{\text{HT},m=1,t=2}^{(\text{TX})} = -\sqrt{\pi_{r2}/2} \underline{l}_{\text{HT}} \\ \underline{a}_{\text{HT},m=2,t=2}^{(\text{TX})} = \sqrt{\pi_{r2}/2} [\underline{l}_{\text{HT}}]_8 \end{cases} \quad (6.17)$$

where $\underline{l}_{\text{HT}}$ is the time sequence of HT long symbol.

6.3.3 AF procedure

The AF procedure concerns the power estimation and the relay preambles' generation.

For the relay terminal, the received training symbols of L-LTF are given by:

$$\underline{l}_{m,t}^{(\text{RX})} = \sum_{j=1}^2 [\underline{l}_{j,t}^{(\text{TX})} \star \underline{h}_{sr,m,j}] + \underline{v}_{m,t}, \quad t = 1, 2 \quad (6.18)$$

where $v_{m,t} \sim \mathcal{CN}(0, \sigma_v^2)$ with $\sigma_v^2 = N_0$ is the AWGN noise term⁴.

By (6.14), we have:

$$\begin{aligned} \underline{l}_{j=2,t}^{(\text{TX})} \star \underline{h}_{sr,m,j=2} &= \sqrt{\frac{\pi_{s1}}{2}} \left[[l]_4 \star \underline{h}_{sr,m,j=2} \right] \\ &= \sqrt{\frac{\pi_{s1}}{2}} \left[l \star [\underline{h}_{sr,m,j=2}]_4 \right] \end{aligned} \quad (6.19)$$

where the channel response $\underline{h}_{sr,m,j=2}$ is considered cyclic-shifted by 4 samples⁵. Hence, the cyclic-shift is transparent to the receiver and this processing is considered as a part of the channel response. For the non-HT portion part, the channel impulsive responses from 2 TX antennas are overlapping in time domain.

Let $\underline{h}_{sr,m}$ denote the sum of $\underline{h}_{sr,m,j=1}$ and $[\underline{h}_{sr,m,j=2}]_4$, equation (6.18) can be given by:

$$\begin{aligned} \underline{l}_{m,t}^{(\text{RX})} &= \sqrt{\frac{\pi_{s1}}{2}} \left[l \star \left(\underline{h}_{sr,m,j=1} + [\underline{h}_{sr,m,j=2}]_4 \right) \right] + \underline{v}_{m,t} \\ &= \sqrt{\frac{\pi_{s1}}{2}} \left[l \star \underline{h}_{sr,m} \right] + \underline{v}_{m,t} \end{aligned} \quad (6.20)$$

For the HT-LTF, the received HT long training symbols are given by:

$$\underline{l}_{\text{HT},m,t}^{(\text{RX})} = \sqrt{\frac{\pi_{s1}}{2}} \sum_{j=1}^2 \left[\underline{l}_{\text{HT},j,t}^{(\text{TX})} \star \underline{h}_{sr,m,j} \right] + \underline{v}_{m,t}, \quad t = 1, 2 \quad (6.21)$$

Considering that:

$$\underline{l}_{\text{HT},j=2,t}^{(\text{TX})} \star \underline{h}_{sr,m,j=2} = \sqrt{\frac{\pi_{s1}}{2}} \underline{l}_{\text{HT}} \star [\underline{h}_{sr,m,j=2}]_8 \quad (6.22)$$

By (6.16) and (6.19), we have:

$$\begin{cases} \underline{l}_{\text{HT},m,t=1}^{(\text{RX})} = \sqrt{\pi_{s1}/2} \left[\underline{l}_{\text{HT}} \star \left(\underline{h}_{sr,m,j=1} + [\underline{h}_{sr,m,j=2}]_8 \right) \right] + \underline{w}_{m,t=1} \\ \underline{l}_{\text{HT},m,t=2}^{(\text{RX})} = \sqrt{\pi_{s1}/2} \left[\underline{l}_{\text{HT}} \star \left(-\underline{h}_{sr,m,j=1} + [\underline{h}_{sr,m,j=2}]_8 \right) \right] + \underline{w}_{m,t=2} \end{cases} \quad (6.23)$$

For the normalization factor b , let \underline{x} denote one of the transmitted HT portion OFDM symbols. The received OFDM symbol \underline{y} is given by:

$$\underline{y}_{m,t}^{(\text{RX})} = \left[\underline{x}_{j=1,t}^{(\text{TX})} \star \underline{h}_{sr,m,j=1} \right] + \left[\underline{x}_{j=2,t}^{(\text{TX})} \star [\underline{h}_{sr,m,j=2}]_8 \right] + \underline{w}_{m,t} \quad (6.24)$$

The transmitted signals are independent and normalized that:

$$\begin{aligned} \text{E} \left\{ \text{Tr} \left[\left(\underline{x}_{j,t}^{(\text{TX})} \right)^T \underline{x}_{j,t}^{*(\text{TX})} \right] \right\} &= 32\pi_{s1} \\ \text{E} \left\{ \left(\underline{x}_{j=1,t}^{(\text{TX})} \right)^T \underline{x}_{j=2,t}^{*(\text{TX})} \right\} &= \mathbf{0} \end{aligned} \quad (6.25)$$

⁴We use $w_{i,t} \sim \mathcal{CN}(0, \sigma_w^2)$ with $\sigma_w^2 = N_0$ for the destination terminal.

⁵We suppose the all the channel responses are in the GI.

Then the received signal power on the HT portion is given by:

$$\mathbb{E}\left\{\|y_{m=1,t}^{(\text{RX})}\|^2 + \|y_{m=2,t}^{(\text{RX})}\|^2\right\} = 32\pi_{s1} \sum_{m=1}^2 \sum_{j=1}^2 \|h_{sr,m,j}\|^2 + 128N_0 \quad (6.26)$$

Therefore, the repeated L-LTF can not be used to estimate the power of the received signal for HT data portion due to the overlapping of the channel responses. We use the long training symbols of HT-LTF which are temporally orthogonal since:

$$\mathbb{E}\left\{\sum_{t=1}^2 \|l_{\text{HT},m,t}^{(\text{RX})}\|^2\right\} = 32\pi_{s1} \sum_{j=1}^2 \|h_{sr,m,j}\|^2 + 128N_0 \quad (6.27)$$

The received power is estimated by:

$$\hat{P} = \frac{1}{256} \sum_{m=1}^2 \sum_{t=1}^2 \|l_{\text{HT},m,t}^{(\text{RX})}\|^2 \quad (6.28)$$

The received signals are normalized by multiplying the matrix \mathbf{B} which is a diagonal matrix:

$$\mathbf{B} = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix} \quad (6.29)$$

where

$$b = \frac{1}{\sqrt{\hat{P}}} \quad (6.30)$$

The amplified legacy training symbols R-LTF, denoted by $r_{m,t}^{(\text{TX})}$, are given by:

$$\begin{cases} r_{m=1,t}^{(\text{TX})} = b\sqrt{\pi_{r2}/2}l_{m=1,t}^{(\text{RX})} \\ r_{m=2,t}^{(\text{TX})} = b\sqrt{\pi_{r2}/2}[l_{m=2,t}^{(\text{RX})}]_4 \end{cases} \quad (6.31)$$

The amplified HT training symbols of RHT-LTF, denoted by $r_{m,t}^{(\text{TX})}$, are given by:

$$\begin{cases} r_{\text{HT},m=1,t}^{(\text{TX})} = b\sqrt{\pi_{r2}/2}l_{\text{HT},m=1,t}^{(\text{RX})} \\ r_{\text{HT},m=2,t}^{(\text{TX})} = b\sqrt{\pi_{r2}/2} [l_{\text{HT},m=2,t}^{(\text{RX})}]_8 \end{cases} \quad (6.32)$$

We remark that the non-HT portion CS is applied on R-LTF and the HT portion CS is applied on RHT-LTF. The HT portion CS is also applied on the relayed data field R. The A-LTF and AHT-LTF are defined as (6.14) and (6.17) except the transmit power for the amplified frame is transmitted with power π_{r2} .

By (6.20) and (6.23), we have:

$$\begin{cases} r_{m=1,t}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}[l \star h_{sr,m=1}] + b v_{m=1,t} \\ r_{m=2,t}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}[l \star [h_{sr,m=2}]_4] + b [v_{m=2,t}]_4 \end{cases}, \quad t = 1, 2 \quad (6.33)$$

and

$$\begin{cases} \underline{r}_{\text{HT},m=1,t=1}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}} \left[\underline{l}_{\text{HT}} \star \left(\underline{h}_{sr,m=1,j=1} + [\underline{h}_{sr,m=1,j=2}]_8 \right) \right] + b\underline{v}_{m=1,t=1} \\ \underline{r}_{\text{HT},m=1,t=2}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}} \left[\underline{l}_{\text{HT}} \star \left(-\underline{h}_{sr,m=1,j=1} + [\underline{h}_{sr,m=1,j=2}]_8 \right) \right] + b\underline{v}_{m=1,t=2} \\ \underline{r}_{\text{HT},m=2,t=1}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}} \left[\underline{l}_{\text{HT}} \star \left([\underline{h}_{sr,m=2,j=1}]_8 + [\underline{h}_{sr,m=2,j=2}]_{16} \right) \right] + b [\underline{v}_{m=2,t=1}]_8 \\ \underline{r}_{\text{HT},m=2,t=2}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}} \left[\underline{l}_{\text{HT}} \star \left(-[\underline{h}_{sr,m=2,j=1}]_8 + [\underline{h}_{sr,m=2,j=2}]_{16} \right) \right] + b [\underline{v}_{m=2,t=2}]_8 \end{cases} \quad (6.34)$$

6.3.4 Channel coefficient estimation

Let $\overline{\mathbf{H}}_{pq} = \{\mathbf{H}_{pq,k}\}$ denote the frequency responses of channel $p \rightarrow q$ for the HT portion data and let \hat{x} denote the estimation result of x .

The Relay-MIMO channel parameters estimation is proposed as follows:

1. Use HT-LTF to estimate the direct channel matrix, $\sqrt{\pi_{s1}}\overline{\mathbf{H}}_{sd}$.
2. Use the preceding result to estimate the channel matrix for data B, $\sqrt{\pi_{s2}}\overline{\mathbf{H}}_{sd}$.
3. Use AHT-LTF to estimate the channel matrix $\sqrt{\pi_{r2}}\overline{\mathbf{H}}_{sr}$.
4. Use A-LTF to estimate the destination noise $\sigma_w^2 = N_0$.
5. Use RHT-LTF to estimate the product channel matrix for data R, $b\sqrt{\pi_{s1}\pi_{r2}}\overline{\mathbf{H}}_{rd} \cdot \overline{\mathbf{H}}_{sr}$ where $\mathbf{B} = b\mathbf{I}$.
6. Use R-LTF to estimate the accumulated noise energy, noted by E_{v+w} .
7. Use the results above to estimate the power normalization factor b in order to evaluate the noise normalization factor $\underline{\rho} = \{\rho_k\}$ at each data sub-carrier.

This procedure is illustrated in Figure 6.13.

L-LTF

Let $\overline{H}_{sd,i,\text{Non-HT}}$ denote the legacy channel coefficients on RX antenna i which are the DFT of $\underline{h}_{sd,i}$. The estimation is performed by using the average of two long training symbols as shown in section 3.3.

HT-LTF

By (6.23), the destination received HT-LTF is given by:

$$\begin{cases} \underline{l}_{\text{HT},i,t=1}^{(\text{RX})} = \sqrt{\pi_{s1}/2} \left[\underline{l}_{\text{HT}} \star \left(\underline{h}_{sd,i,j=1} + [\underline{h}_{sd,i,j=2}]_8 \right) \right] + \underline{w}_{i,t=1} \\ \underline{l}_{\text{HT},i,t=2}^{(\text{RX})} = \sqrt{\pi_{s1}/2} \left[\underline{l}_{\text{HT}} \star \left(-\underline{h}_{sd,i,j=1} + [\underline{h}_{sd,i,j=2}]_8 \right) \right] + \underline{w}_{i,t=2} \end{cases} \quad (6.35)$$

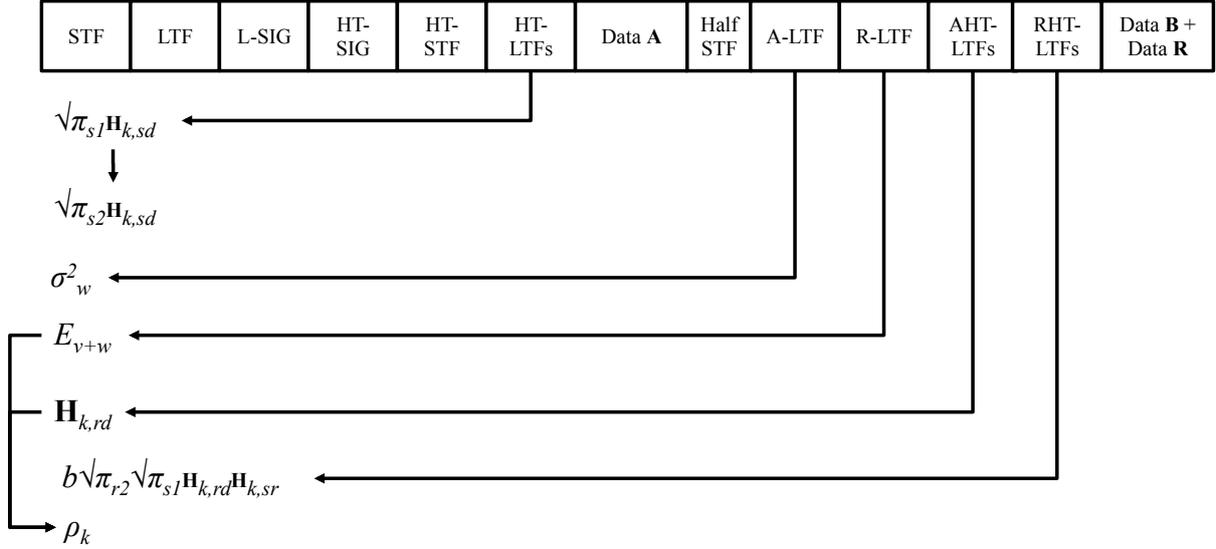


Figure 6.13: Relay-MIMO channel parameters estimation

In frequency domain, the system is described by:

$$\begin{cases} \bar{L}_{\text{HT},i,t=1}^{(\text{RX})} = \sqrt{\pi_{s1}/2\bar{L}_{\text{HT}}} \cdot (\bar{H}_{sd,i,j=1} + \bar{H}_{sd,i,j=2}) + \bar{W}_{i,t=1} \\ \bar{L}_{\text{HT},i,t=2}^{(\text{RX})} = \sqrt{\pi_{s1}/2\bar{L}_{\text{HT}}} \cdot (-\bar{H}_{sd,i,j=1} + \bar{H}_{sd,i,j=2}) + \bar{W}_{i,t=2} \end{cases} \quad (6.36)$$

where $\bar{H}_{sd,i,j=2}$ is the DFT of $[h_{sd,i,j=2}]_8$.

On sub-carrier k , let the received HT-LTF be noted by:

$$\mathbf{L}_{\text{HT},k}^{(\text{RX})} = \begin{bmatrix} L_{\text{HT},i=1,t=1}^{(\text{RX})} & L_{\text{HT},i=1,t=2}^{(\text{RX})} \\ L_{\text{HT},i=2,t=1}^{(\text{RX})} & L_{\text{HT},i=2,t=2}^{(\text{RX})} \end{bmatrix} \quad (6.37)$$

The channel matrix is denoted by:

$$\mathbf{H}_{sd,k} = \begin{bmatrix} H_{sd,k,i=1,j=1} & H_{sd,k,i=1,j=2} \\ H_{sd,k,i=2,j=1} & H_{sd,k,i=2,j=2} \end{bmatrix} \quad (6.38)$$

By writing $\bar{\mathbf{L}}_{\text{HT}}^{(\text{RX})} = \{\mathbf{L}_{\text{HT},k}^{(\text{RX})}\}$ and $\bar{\mathbf{H}}_{sd} = \{\mathbf{H}_{sd,k}\}$, equation (6.36) turns to:

$$\bar{\mathbf{L}}_{\text{HT}}^{(\text{RX})} = \sqrt{\frac{\pi_{s1}}{2}} \bar{\mathbf{H}}_{sd} \cdot (\mathbf{P}_2 \otimes \bar{\mathbf{L}}_{\text{HT}}) + \bar{\mathbf{W}} \quad (6.39)$$

where $x \otimes y$ is the Kronecker production and:

$$\mathbf{P}_2 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \quad (6.40)$$

The matrix \mathbf{P}_2 is the upper-left part of $\mathbf{P}_{\text{HTLTF}}$ in equation (6.15).

The estimation of MIMO channel coefficients is realized by multiplying the $\mathbf{P}_2^{-1} \otimes (\bar{\mathbf{L}}_{\text{HT}})^{-1}$:

$$\sqrt{\pi_{s1}} \hat{\mathbf{H}}_{sd} = \sqrt{2} \bar{\mathbf{L}}_{\text{HT}}^{(\text{RX})} \cdot (\mathbf{P}_2^{-1} \otimes (\bar{\mathbf{L}}_{\text{HT}})^{-1}) \quad (6.41)$$

where $(\bar{L}_{\text{HT}})^{-1} = \{L_k^{-1}\}$ for non-zero sub-carriers and

$$\mathbf{P}_2^{-1} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \quad (6.42)$$

The channel matrix for data B is given by:

$$\sqrt{\pi_{s2}} \hat{\mathbf{H}}_{sd} = \sqrt{\frac{\pi_{s2}}{\pi_{s1}}} \sqrt{\pi_{s1}} \hat{\mathbf{H}}_{sd} \quad (6.43)$$

A-LTF

The A-LTF is like L-LTF that the received A-LTF is given by:

$$\underline{a}_{i,t}^{(\text{RX})} = \sqrt{\frac{\pi_{rd}}{2}} [l \star \underline{h}_{rd,i}] + \underline{w}_{i,t}, \quad t = 1, 2 \quad (6.44)$$

where

$$\underline{h}_{rd,i} = \underline{h}_{rd,i,m=1} + [\underline{h}_{rd,i,m=2}]_4 \quad (6.45)$$

Like in Relay-SISO system, the noise power at detination terminal is estimated by:

$$\hat{N}_0 = \frac{1}{256} \sum_{i=1}^2 \|\underline{a}_{i,t=1}^{(\text{RX})} - \underline{a}_{i,t=2}^{(\text{RX})}\|^2 \quad (6.46)$$

R-LTF

The R-LTF reception is described as follows:

$$\underline{r}_{i,t}^{(\text{RX})} = \sum_{m=1}^2 [\underline{r}_{m,t}^{(\text{TX})} \star \underline{h}_{rd,i,m}] + \underline{w}_{i,t} \quad (6.47)$$

By (6.31), we can develop it by:

$$\begin{aligned} \underline{r}_{i,t}^{(\text{RX})} &= \frac{b\sqrt{\pi_{s1}\pi_{r2}}}{2} \left[l \star \left(\underline{h}_{sr,m=1} \star \underline{h}_{rd,i,m=1} + [\underline{h}_{sr,m=2}]_4 \star \underline{h}_{rd,i,m=2} \right) \right] \\ &\quad + b\sqrt{\frac{\pi_{r2}}{2}} \left([\underline{v}_{m=1,t} \star \underline{h}_{rd,i,m=1}] + [[\underline{v}_{m=2,t}]_4 \star \underline{h}_{rd,i,m=2}] \right) + \underline{w}_{i,t} \end{aligned} \quad (6.48)$$

The sum of noise energy is defined by:

$$\begin{aligned} E_{v+w} &= \mathbb{E} \left\{ \sum_{i=1}^2 \left\| b\sqrt{\frac{\pi_{r2}}{2}} \left([\underline{v}_{m=1,t} \star \underline{h}_{rd,i,m=1}] + [[\underline{v}_{m=2,t}]_4 \star \underline{h}_{rd,i,m=2}] \right) + \underline{w}_{i,t} \right\|^2 \right\} \\ &= (b^2 E_H + 128) N_0 \end{aligned} \quad (6.49)$$

where the sum of the $r \rightarrow d$ channel energy:

$$E_H = 32\pi_{r2} \sum_{i=1}^2 \sum_{m=1}^2 \|\underline{h}_{rd,i,m}\|^2 \quad (6.50)$$

E_H will be estimated with AHT-LTF.

The estimation of E_{v+m} is given by

$$\hat{E}_{v+w} = \frac{1}{2} \sum_{i=1}^2 \|r_{i,t=1}^{(\text{RX})} - r_{i,t=2}^{(\text{RX})}\|^2 \quad (6.51)$$

AHT-LTF

The AHT-LTFs are used for the coefficients estimation of channel $r \rightarrow d$. This procedure is the same like HT-LTF. We use $\hat{\mathbf{H}}_{rd}$ to denote the estimation results.

Besides, the AHT-LTFs are used to estimate E_H .

Considering:

$$\begin{cases} a_{\text{HT},i,t=1}^{(\text{RX})} = \sqrt{\pi_{r2}/2} \left[l_{\text{HT}} \star \left(h_{rd,i,j=1} + [h_{rd,i,j=2}]_8 \right) \right] + w_{i,t=1} \\ a_{\text{HT},i,t=2}^{(\text{RX})} = \sqrt{\pi_{r2}/2} \left[l_{\text{HT}} \star \left(-h_{rd,i,j=1} + [h_{rd,i,j=2}]_8 \right) \right] + w_{i,t=2} \end{cases} \quad (6.52)$$

We have:

$$\mathbb{E} \left\{ \sum_{i=1}^2 \sum_{t=1}^2 \|a_{\text{HT},i,t}^{(\text{RX})}\|^2 \right\} = 64\pi_{r2} \sum_{i=1}^2 \sum_{m=1}^2 \|h_{rd,i,m}\|^2 + 256N_0 \quad (6.53)$$

Then E_H is estimated as follows:

$$\hat{E}_H = \frac{1}{2} \left(\sum_{i=1}^2 \sum_{t=1}^2 \|a_{\text{HT},i,t}^{(\text{RX})}\|^2 \right) - 128\hat{N}_0 \quad (6.54)$$

From (6.49) and (6.54), we estimate the normalization factor by:

$$\hat{b}^2 = \frac{1}{\hat{E}_H} \left(\hat{E}_{v+w} - 128\hat{N}_0 \right) \quad (6.55)$$

On each data subcarrier, the $\mathbf{\Sigma}$ matrix is calculated by:

$$\mathbf{\Sigma}_k = \left(\hat{b}^2 \pi_{r2} \hat{\mathbf{H}}_{rd,k} \hat{\mathbf{H}}_{rd,k}^H + \mathbf{I} \right)^{-1} \quad (6.56)$$

The noise normalization follows the procedure described in section 1.2.

RHT-LTF

The received signals RHT-LTF are expressed by:

$$r_{\text{HT},j,t}^{(\text{RX})} = \sum_{m=1}^2 r_{\text{HT},m,t}^{(\text{TX})} \star h_{rd,j,m} + w_{j,t} \quad (6.57)$$

which can be described in frequency domain by:

$$\overline{R}_{\text{HT},j,t}^{(\text{RX})} = \sum_{m=1}^2 \overline{R}_{\text{HT},m,t}^{(\text{TX})} \cdot \overline{H}_{rd,j,m} + \overline{W}_{j,t} \quad (6.58)$$

On sub-carrier k , let:

$$\left\{ \begin{array}{l} \mathbf{R}_{\text{HT},k}^{(\text{RX})} = \begin{bmatrix} R_{\text{HT},k,j=1,t=1}^{(\text{RX})} & R_{\text{HT},k,j=1,t=2}^{(\text{RX})} \\ R_{\text{HT},k,j=2,t=1}^{(\text{RX})} & R_{\text{HT},k,j=2,t=2}^{(\text{RX})} \end{bmatrix} \\ \mathbf{H}_{rd,k} = \begin{bmatrix} H_{rd,k,j=1,m=1} & H_{rd,k,j=1,m=2} \\ H_{rd,k,j=2,m=1} & H_{rd,k,j=2,m=2} \end{bmatrix} \\ \mathbf{R}_{\text{HT},k}^{(\text{TX})} = \begin{bmatrix} R_{\text{HT},k,m=1,t=1}^{(\text{TX})} & R_{\text{HT},k,m=1,t=2}^{(\text{TX})} \\ R_{\text{HT},k,m=2,t=1}^{(\text{TX})} & R_{\text{HT},k,m=2,t=2}^{(\text{TX})} \end{bmatrix} \\ \mathbf{W}_k = \begin{bmatrix} W_{k,i=1,t=1} & W_{k,i=1,t=2} \\ W_{k,i=2,t=1} & W_{k,i=2,t=2} \end{bmatrix} \end{array} \right. \quad (6.59)$$

The received signal can be given by:

$$\mathbf{R}_{\text{HT},k}^{(\text{RX})} = \mathbf{H}_{rd,k} \mathbf{R}_{\text{HT},k}^{(\text{TX})} + \mathbf{W}_k \quad (6.60)$$

From (6.34), we have:

$$\left\{ \begin{array}{l} \overline{R}_{\text{HT},m=1,t=1}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}\overline{L}_{\text{HT}} \cdot (\overline{H}_{sr,m=1,j=1} + \overline{H}_{sr,m=1,j=2}) + b\overline{V}_{m=1,t=1} \\ \overline{R}_{\text{HT},m=1,t=2}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}\overline{L}_{\text{HT}} \cdot (-\overline{H}_{sr,m=1,j=1} + \overline{H}_{sr,m=1,j=2}) + b\overline{V}_{m=1,t=2} \\ \overline{R}_{\text{HT},m=2,t=1}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}\overline{L}_{\text{HT}} \cdot (\overline{H}_{sr,m=2,j=1} + \overline{H}_{sr,m=2,j=2}) + b\overline{V}_{m=2,t=1} \\ \overline{R}_{\text{HT},m=2,t=2}^{(\text{TX})} = b/2\sqrt{\pi_{s1}\pi_{r2}}\overline{L}_{\text{HT}} \cdot (-\overline{H}_{sr,m=2,j=1} + \overline{H}_{sr,m=2,j=2}) + b\overline{V}_{m=2,t=2} \end{array} \right. \quad (6.61)$$

where

$$\left\{ \begin{array}{l} \overline{H}_{sr,m=1,j=2} = \text{DFT} \left(\left[\underline{h}_{sr,m=1,j=2} \right]_8 \right) \\ \overline{H}_{sr,m=2,j=1} = \text{DFT} \left(\left[\underline{h}_{sr,m=2,j=1} \right]_8 \right) \\ \overline{H}_{sr,m=2,j=2} = \text{DFT} \left(\left[\underline{h}_{sr,m=2,j=1} \right]_{16} \right) \\ \overline{V}_{m=2,t=1} = \text{DFT} \left(\left[\underline{v}_{m=2,t=1} \right]_8 \right) \\ \overline{V}_{m=2,t=2} = \text{DFT} \left(\left[\underline{v}_{m=2,t=2} \right]_8 \right) \end{array} \right. \quad (6.62)$$

Then the transmitted RHT-LTF can be noted on sub-carrier k by:

$$\mathbf{R}_{\text{HT},k}^{(\text{TX})} = \frac{b\sqrt{\pi_{s1}\pi_{r2}}}{2} \mathbf{H}_{sr,k} \mathbf{P}_2 L_{\text{HT},k} + b\mathbf{V}_k \quad (6.63)$$

with

$$\left\{ \begin{array}{l} \mathbf{H}_{sr,k} = \begin{bmatrix} H_{sr,k,m=1,j=1} & H_{sr,k,m=1,j=2} \\ H_{sr,k,m=2,j=1} & H_{sr,k,m=2,j=2} \end{bmatrix} \\ \mathbf{V}_k = \begin{bmatrix} V_{k,m=1,t=1} & V_{k,m=1,t=2} \\ V_{k,m=2,t=1} & V_{k,m=2,t=2} \end{bmatrix} \end{array} \right. \quad (6.64)$$

By (6.60) and (6.63):

$$\mathbf{R}_{\text{HT},k}^{(\text{RX})} = \frac{b\sqrt{\pi_{s1}\pi_{r2}}}{2} \mathbf{H}_{rd,k} \mathbf{H}_{sr,k} \mathbf{P}_2 L_{\text{HT},k} + b\mathbf{H}_{rd,k} \mathbf{V}_k + \mathbf{W}_k \quad (6.65)$$

The product channel matrix $b\sqrt{\pi_{r2}\pi_{s1}}\mathbf{H}_{rd,k}\mathbf{H}_{sr,k}$ is estimated by:

$$b\sqrt{\pi_{r2}\pi_{s1}}\hat{\mathbf{H}}_{rd,k}\hat{\mathbf{H}}_{sr,k} = 2\mathbf{R}_{\text{HT},k}^{(\text{RX})}\mathbf{P}_2^{-1}L_{\text{HT},k}^{-1} \quad (6.66)$$

6.4 Relay-MIMO Performance

Here we give the simulation results of Relay-MIMO PHY layer and the original IEEE 802.11n PHY layer in the configuration 2×2 using BLAST code. The MIMO decoding algorithm is very important: we use the KSE decoder with $K=8$ to decode the BLAST code in IEEE 802.11n system and KSE decoder with $K=16$ to decode the Perfect 4×4 code in Relay-MIMO system. In both decoders, GO processing is applied before performing the lattice searching.

We will compare the performance of IEEE 802.11n and Relay-MIMO in the context of flat fading channel and multipath channel in supposing that there is no correlation between the TX/RX antennas. The transmission of a MPDU=125 Bytes is simulated with MCS=9 for data-rate of 26Mbps and MCS=11 for data-rate of 52Mbps. We define still two cooperation scenarios: the LPG configuration with $G_{sr} = G_{rd} = 0\text{dB}$ and the HPG configuration with $G_{sr} = G_{rd} = 10\text{dB}$. Both the FER and the efficient throughput are reported.

6.4.1 Flat fading channel

In the flat fading channel environment, the simulation results in Figures (6.14) and (6.15) show the important diversity gain of Relay-MIMO. In MCS=9, Relay-MIMO gives a system gain about 7dB in HPG configuration and 3dB in LPG configuration for $\text{FER}=10^{-3}$. This system gain is shown in throughput where we find an improvement about 2Mbps in HPG configuration around $\text{SNR}=15\text{dB}$.

In MCS=11, the advantage of Relay-MIMO is less evident that only a small gain in SNR is reported in HPG and there is nearly no gain in throughput.

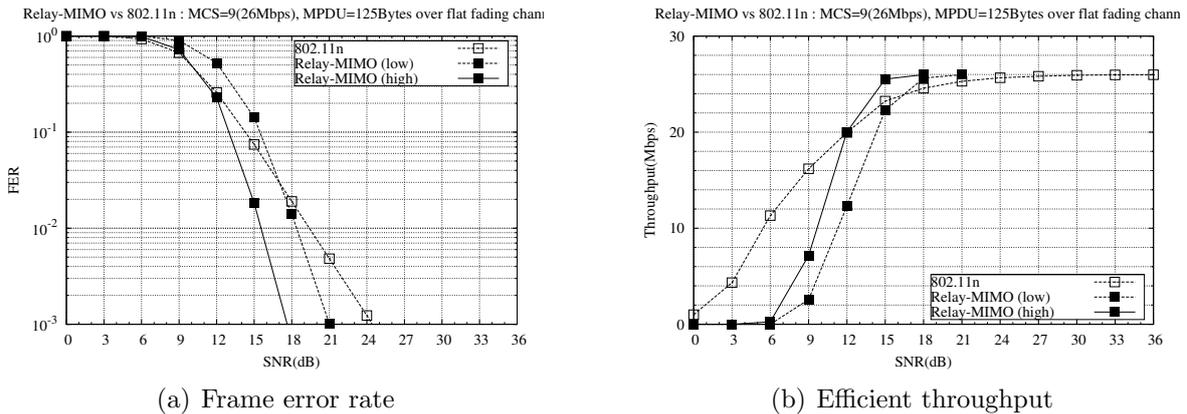


Figure 6.14: Relay-MIMO and IEEE 802.11n over 2×2 MIMO flat fading channel: each MPDU contains 125 bytes and it is sent at MCS=9(26Mbps).

6.4.2 Multipath channel

In the multipath channel environment, we still use the channel type A which is shown in section 3.4. The diversity gain of Relay-MIMO is not obvious. As shown in Figures

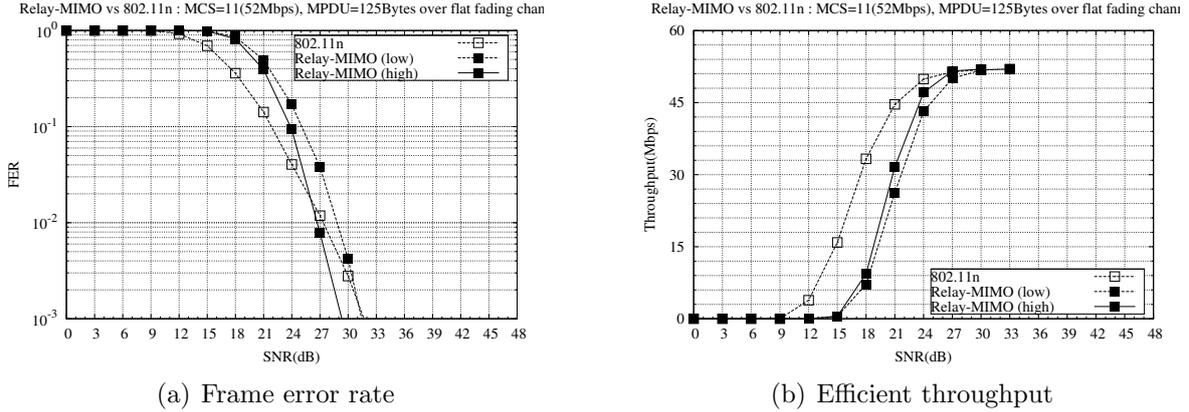


Figure 6.15: Relay-MIMO and IEEE 802.11n over 2×2 MIMO flat fading channel: each MPDU contains 125 bytes and it is sent at MCS=11(52Mbps).

(6.16) and (6.17), the IEEE 802.11n system makes full use of the combination of MIMO diversity and multipath diversity that it achieves good performance in both MCS=9 and MCS=11. Relay-MIMO in HPG configuration gives nearly the same performance as IEEE 802.11n for MCS=9 and loss about 3dB for MCS=11. In this context, Relay-MIMO has hardly advantage.

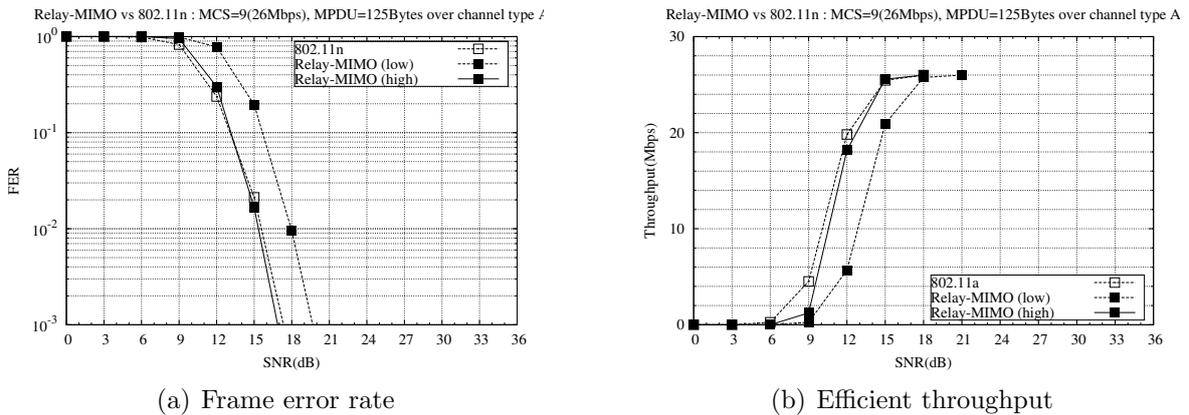


Figure 6.16: Relay-MIMO and 802.11n over 2×2 MIMO channel type A: each MPDU contains 125 bytes and it is sent at MCS=9(26Mbps).

6.4.3 Discussion on performance

Compared with 802.11n, Relay-MIMO provides extra cooperative diversity which is significant in flat-fading channel for HPG configuration and low data rate transmission. However, the cooperative diversity gain becomes less obvious when we increase the data rate or change HPG condition to LPG condition. In the case of multipath channel, the cooperative diversity is shadowed by the multipath diversity such that the Relay-MIMO diversity gain is less significant. We find that Relay-MIMO scheme can not outperform 802.11n system since current IEEE 802.11n system makes full use of multi-antenna diversity as well as multi-path diversity.

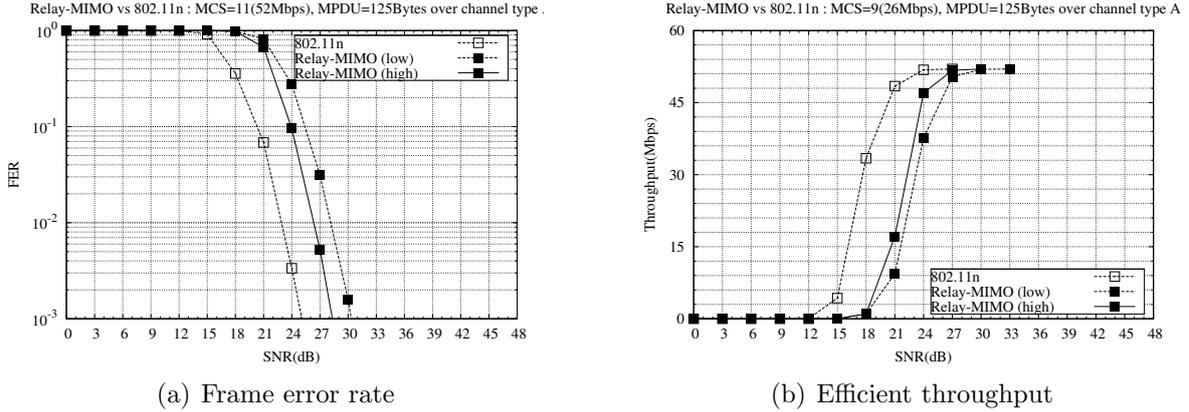


Figure 6.17: Relay-MIMO and 802.11n over 2×2 MIMO channel type A: each MPDU contains 125 bytes and it is sent at MCS=11(52Mbps).

Thus, the interest to apply cooperative diversity can be in flat-fading channel environment. The hybrid strategy in chapter 4 can also be introduced to solve the “crossing” point problem which can be the future work.

In the context of multipath channel which is a typical Wireless Local Area Network (WLAN) environment, it will not be worthy to implement such cooperative system since current IEEE 802.11n system provides already sufficient robustness.

6.5 Algebraic reduction for 4×4 Perfect code

For the MIMO system in very high dimension, the LLL algorithm in section 2.3 can be very efficient to simplify the channel matrix. The LLL reduction with simple ZF projection can achieve a good performance. However, the complexity of this reduction method can be very high. In this section, we propose a new strategy of STBC decoder in using the Algebraic Reduction (AR) where the structure of STBC is taken into account. The algebraic reduction has been studied in [34] for the STBC generated from the algebraic extension \mathbb{K}/\mathbb{F} . Based on this method, we propose a recursive decoder for the 4×4 Perfect code which is applied in Relay-MIMO system.

6.5.1 Algebraic reduction

The algebraic reduction method can be applied on the fast fading channel using algebraic extension as the space-time code. In this case, the system is described by:

$$\bar{y} = \mathbf{H}\bar{x} + \bar{w} \quad (6.67)$$

where $\mathbf{H} = \text{diag}(h_1, h_2, \dots, h_n)$ is an $n \times n$ diagonal matrix and the received signal \bar{y} , the transmitted signal \bar{x} and the AWGN term \bar{w} are all n -dimensional vectors.

Let \mathbb{K} denote an n -degree algebraic extension on $\mathbb{F} = \mathbb{Q}(i)$ and let $\{\theta_1, \theta_2, \dots, \theta_n\}$ denote

its integral basis. Its integer ring $\mathcal{O}_{\mathbb{K}}$ is generated by:

$$\mathcal{O}_{\mathbb{K}} = \left\{ \sum_{k=1}^n s_k \theta_k, s_k \in \mathbb{Z}(i) \right\} \quad (6.68)$$

Its galois group $\text{Gal}_{\mathbb{K}/\mathbb{F}}$ is given by $\{\sigma_1, \sigma_2, \dots, \sigma_n\}$. The canonical embedding of the integers basis ring of \mathbb{K} , which is denoted by $\mathcal{O}_{\mathbb{K}}$, in fact generates a lattice $\Lambda(\mathcal{O}_{\mathbb{K}})$.

The generation matrix of lattice $\Lambda(\mathcal{O}_{\mathbb{K}})$ is given by:

$$\Phi_{\Lambda(\mathcal{O})} = \begin{bmatrix} \sigma_1(\theta_1) & \sigma_1(\theta_2) & \cdots & \sigma_1(\theta_n) \\ \sigma_2(\theta_1) & \sigma_2(\theta_2) & \cdots & \sigma_2(\theta_n) \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_n(\theta_1) & \sigma_n(\theta_2) & \cdots & \sigma_n(\theta_n) \end{bmatrix} \quad (6.69)$$

For Golden code or 4×4 Perfect code, the integer ring $\mathcal{O}_{\mathbb{K}}$ is replaced by its ideal $\mathcal{I}_{\mathbb{K}}$. In case of Golden code, the ideal is principal $\mathcal{I} = \alpha \mathcal{O}_{\mathbb{K}}$ and the related lattice is generated by:

$$\Phi_{\Lambda(\mathcal{I})} = \frac{1}{\sqrt{5}} \begin{bmatrix} \alpha & \alpha\theta \\ \alpha & \alpha\bar{\theta} \end{bmatrix} \quad (6.70)$$

where $\sqrt{5}$ is the normalization factor that $\Phi_{\Lambda(\mathcal{I})}^H \Phi_{\Lambda(\mathcal{I})} = \mathbf{I}$.

In case of 4×4 Perfect code, the ideal is generated by the basis $[b_1, b_2, b_3, b_4]$. In using canonical embedding, the lattice generator is:

$$\Phi_{\Lambda(\mathcal{I})} = \frac{1}{\sqrt{15}} \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \\ \sigma(b_0) & \sigma(b_1) & \sigma(b_2) & \sigma(b_3) \\ \sigma^2(b_0) & \sigma^2(b_1) & \sigma^2(b_2) & \sigma^2(b_3) \\ \sigma^3(b_0) & \sigma^3(b_1) & \sigma^3(b_2) & \sigma^3(b_3) \end{bmatrix} \quad (6.71)$$

By writing:

$$c = \left| \prod_{k=1}^n h_k \right|^{\frac{1}{n}} \quad (6.72)$$

the system in (6.67) can be given by:

$$\bar{y} = c \Psi \mathbf{H}' \Phi \bar{x} + \bar{w} \quad (6.73)$$

where $\Psi = \text{diag}(e^{i\phi_1}, e^{i\phi_2}, \dots, e^{i\phi_n})$ and $\mathbf{H}' = \text{diag}(h'_1, h'_2, \dots, h'_n)$ are both diagonal matrix with ϕ_i the phase of h_i and $h'_i = h_i/c$.

The principal of algebraic reduction is to find a “unit” $u \in \mathcal{O}_{\mathbb{K}}$ that its canonical embedding $\mathbf{U} = \text{diag}(\sigma_1(u), \sigma_2(u), \dots, \sigma_n(u))$ is approaching to \mathbf{H}' .

Let us define the norm of the elements in \mathbb{K}/\mathbb{F} by $N_{\mathbb{K}/\mathbb{F}}(x) = \prod_{k=1}^n \sigma_k(x)$ then the units are the elements in $\mathcal{O}_{\mathbb{K}}$ with $N_{\mathbb{K}/\mathbb{F}} = 1$.

Let $\mathbf{U} = \text{diag}(\sigma_1(u), \sigma_2(u), \dots, \sigma_n(u))$ such that $\det(\mathbf{U}) = N_{\mathbb{K}/\mathbb{F}}(u) = 1$.

The unit has a very interesting property that:

$$\mathbf{U}\Phi = \Phi\mathbf{T} \quad (6.74)$$

where \mathbf{T} is unimodular matrix with $t_{i,j} \in \mathbb{F}$.

In ideal case, if the matrix of channel coefficients gives $\mathbf{H}' = \mathbf{U}$, we have:

$$\begin{aligned} \bar{y} &= c\Psi\mathbf{H}'\Phi\bar{x} + \bar{w} \\ &\Leftrightarrow \\ \bar{y} &= c\Psi\Phi\mathbf{T}\bar{x} + \bar{w} \end{aligned} \quad (6.75)$$

Since $\mathbf{T}\mathbb{Z}^n(i) = \mathbb{Z}^n(i)$, we can denote $\mathbf{T}\bar{x}$ by $\bar{z} \in \mathbb{Z}^n(i)$.

The system turns to:

$$\bar{y} = c\Psi\Phi\bar{z} + \bar{w} \quad (6.76)$$

Since the matrix Φ is unitary, a simple ZF detector can give ML performance by left-multiplying $(c\Psi\Phi)^{-1}$. The \bar{x} is then obtained by $\mathbf{T}^{-1}\bar{z}$.

In general cases, the principle of algebraic reduction is to find a unit u such that \mathbf{U} gives a good approximation to the \mathbf{H}' in order to perform a ZF detection in a quasi-orthogonal space.

The Dirichlet's theorem [35] shows that for an algebraic extension \mathbb{K}/\mathbb{F} with signature $(r, s)^6$, there exist $r + s - 1$ fundamental units $\{u_1, u_2, \dots, u_{r+s-1}\}$ such that a unit u can be given by:

$$u = \epsilon \prod_{i=1}^{r+s-1} u_{k_i} \quad (6.77)$$

where ϵ is a root of the unit 1 and $k_i \in \mathbb{Z}$ is the power of each fundamental element.

Let us define the logarithmic canonical embedding as following:

$$\Gamma : \mathbb{K} \mapsto \mathbb{R}^n \quad (6.78)$$

$$x \rightarrow [\log |\sigma_1(x)|, \log |\sigma_2(x)|, \dots, \log |\sigma_{r+s}(x)|]^T \quad (6.79)$$

According to (6.77), this logarithmic canonical embedding generates a lattice with generation matrix:

$$\mathbf{G}_{\mathbf{U}} = \begin{bmatrix} \log |\sigma_1(u_1)| & \log |\sigma_1(u_2)| & \cdots & \log |\sigma_1(u_{r+s-1})| \\ \log |\sigma_2(u_1)| & \log |\sigma_2(u_2)| & \cdots & \log |\sigma_2(u_{r+s-1})| \\ \vdots & \vdots & \ddots & \vdots \\ \log |\sigma_{r+s}(u_1)| & \log |\sigma_{r+s}(u_2)| & \cdots & \log |\sigma_{r+s}(u_{r+s-1})| \end{bmatrix} \quad (6.80)$$

By Γ embedding, a unit u in \mathbb{K} is mapped to:

$$\Gamma(u) = \bar{u}_{\log} = [\log |\sigma_1(x)|, \log |\sigma_2(x)|, \dots, \log |\sigma_{r+s}(x)|]^T \quad (6.81)$$

⁶ r is the number of real roots and s is the number of imaginaire root pairs.

Let $\bar{k} = \{k_1, k_2, \dots, k_{r+s-1}\}$ denote its coordinates in the logarithmic lattice that:

$$\bar{u}_{\log} = \mathbf{G}_{\mathbf{U}}\bar{k} \quad (6.82)$$

By transforming the matrix \mathbf{H}' in its logarithm form, we aim to decode such system:

$$\bar{h}_{\log} = \mathbf{G}_{\mathbf{U}}\bar{k} + \bar{v} \quad (6.83)$$

In this system, \bar{h}_{\log} is considered as the received signal, \bar{k} is the signal to decode and \bar{v} is viewed as a “noise” term.

The decoding procedure of this matrix will be taken in real domain. Since $\mathbf{G}_{\mathbf{U}}$ is fixed, the lattice reduction method can be pre-performed easily.

For 4×4 Perfect code, the algebraic extension is $\mathbb{K}/\mathbb{F} = \mathbb{Q}(i, \theta)$ with $\theta = \xi_{15} + \xi_{15}^{-1}$. This is a 4 degree extension of \mathbb{F} with signature $(4, 0)$.

The 3 fundamental units are:

$$\begin{cases} u_1 = 1 + 3\theta - \theta^2 - \theta^3 \\ u_2 = 1 - \theta \\ u_3 = 1 - 3\theta + \theta^3 \end{cases} \quad (6.84)$$

The fundamental units generate the logarithmic lattice with generation matrix:

$$\mathbf{G}_{\mathbf{U}} = \begin{bmatrix} -0.1898 & -1.0839 & -0.4812 \\ 1.0839 & -0.1987 & 0.4812 \\ 0.1898 & 1.0839 & -0.4812 \\ -1.0839 & 0.1898 & 0.4812 \end{bmatrix} \quad (6.85)$$

By QR-decomposition, we have $\mathbf{G}_{\mathbf{U}} = \mathbf{Q}_{\mathbf{G}}\mathbf{R}_{\mathbf{G}}$ with

$$\begin{cases} \mathbf{Q}_{\mathbf{G}} = \begin{bmatrix} -0.1220 & -0.6965 & -0.5000 \\ 0.6965 & -0.1220 & 0.5000 \\ 0.1220 & 0.6965 & -0.5000 \\ -0.6965 & 0.1220 & 0.5000 \end{bmatrix} \\ \mathbf{R}_{\mathbf{G}} = \begin{bmatrix} 1.5563 & 0 & 0 \\ 0 & 1.5563 & 0 \\ 0 & 0 & 0.9624 \end{bmatrix} \end{cases} \quad (6.86)$$

We find the matrix $\mathbf{R}_{\mathbf{G}}$ is a diagonal matrix that the ML lattice decoding can be achieved by the ZF algorithm.

6.5.2 Algebraic reduction for 4×4 Perfect code

As explained in section 1.2, a general MIMO system using 4×4 Perfect code can be described as in (2.6):

$$\bar{y} = \tilde{\mathbf{H}}\bar{x} + \bar{w} \quad (6.87)$$

We recall the coding matrix $\mathbf{M}_{\mathcal{X}}$ in (1.57) is:

$$\mathbf{M}_{\mathcal{X}} = \begin{bmatrix} \underline{b} & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma(\underline{b}) \\ \underline{0} & \underline{0} & \gamma\sigma^2(\underline{b}) & \underline{0} \\ \underline{0} & \gamma\sigma^3(\underline{b}) & \underline{0} & \underline{0} \\ \underline{0} & \underline{b} & \underline{0} & \underline{0} \\ \sigma(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma^2(\underline{b}) \\ \underline{0} & \underline{0} & \gamma\sigma^3(\underline{b}) & \underline{0} \\ \underline{0} & \underline{0} & \underline{b} & \underline{0} \\ \underline{0} & \sigma(\underline{b}) & \underline{0} & \underline{0} \\ \sigma^2(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} & \gamma\sigma^3(\underline{b}) \\ \underline{0} & \underline{0} & \underline{0} & \underline{b} \\ \underline{0} & \underline{0} & \sigma(\underline{b}) & \underline{0} \\ \underline{0} & \sigma^2(\underline{b}) & \underline{0} & \underline{0} \\ \sigma^3(\underline{b}) & \underline{0} & \underline{0} & \underline{0} \end{bmatrix} \quad (6.88)$$

We will first do some processing with the matrix $\mathbf{M}_{\mathcal{X}}$ by writing :

$$\mathbf{M}_{\mathcal{X}} = \mathbf{M}_p \mathbf{D}_{\mathcal{X}} \quad (6.89)$$

where we have:

$$\left\{ \begin{array}{l} \mathbf{D}_{\mathcal{X}} = \text{diag}(\Phi, \Phi, \dots, \Phi) = \begin{bmatrix} \Phi & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Phi & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Phi & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \Phi \end{bmatrix} \\ \Phi = \frac{1}{\sqrt{15}} \begin{bmatrix} b_0 & b_1 & b_2 & b_3 \\ \sigma(b_0) & \sigma(b_1) & \sigma(b_2) & \sigma(b_3) \\ \sigma^2(b_0) & \sigma^2(b_1) & \sigma^2(b_2) & \sigma^2(b_3) \\ \sigma^3(b_0) & \sigma^3(b_1) & \sigma^3(b_2) & \sigma^3(b_3) \end{bmatrix} \end{array} \right. \quad (6.90)$$

The matrix \mathbf{M}_p is a permutation matrix with rotation at certain rows:

$$\mathbf{M}_p = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & i & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & i & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & i & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & i \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6.91)$$

Then the system can be given by:

$$\bar{y} = \tilde{\mathbf{H}}\mathbf{M}_p\mathbf{D}\chi\bar{s} + \bar{w} \quad (6.92)$$

Let $\hat{\mathbf{H}}$ denote the product $\tilde{\mathbf{H}}\mathbf{M}_p$ that:

$$\hat{\mathbf{H}} = \begin{bmatrix} h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} & 0 & 0 & ih_{13} & 0 & 0 & ih_{12} & 0 & 0 \\ h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} & 0 & 0 & ih_{23} & 0 & 0 & ih_{22} & 0 & 0 \\ h_{31} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{34} & 0 & 0 & ih_{33} & 0 & 0 & ih_{32} & 0 & 0 \\ h_{41} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} & 0 & 0 & ih_{43} & 0 & 0 & ih_{42} & 0 & 0 \\ 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} & 0 & 0 & ih_{13} & 0 \\ 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} & 0 & 0 & ih_{23} & 0 \\ 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{34} & 0 & 0 & ih_{33} & 0 \\ 0 & h_{42} & 0 & 0 & h_{41} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} & 0 & 0 & ih_{43} & 0 \\ 0 & 0 & h_{13} & 0 & 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} \\ 0 & 0 & h_{23} & 0 & 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} \\ 0 & 0 & h_{33} & 0 & 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{34} \\ 0 & 0 & h_{43} & 0 & 0 & h_{42} & 0 & 0 & h_{41} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} \\ 0 & 0 & 0 & h_{14} & 0 & 0 & h_{13} & 0 & 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{24} & 0 & 0 & h_{23} & 0 & 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{34} & 0 & 0 & h_{33} & 0 & 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{44} & 0 & 0 & h_{43} & 0 & 0 & h_{42} & 0 & 0 & h_{41} & 0 & 0 & 0 \end{bmatrix} \quad (6.93)$$

By permuting the rows, we can give a new representation in matrix form:

$$\mathbf{M}_q\hat{\mathbf{H}} = \begin{bmatrix} h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} & 0 & 0 & ih_{13} & 0 & 0 & ih_{12} & 0 & 0 \\ 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} & 0 & 0 & ih_{13} & 0 \\ 0 & 0 & h_{13} & 0 & 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{14} \\ 0 & 0 & 0 & h_{14} & 0 & 0 & h_{13} & 0 & 0 & h_{12} & 0 & 0 & h_{11} & 0 & 0 & 0 \\ 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} & 0 & 0 & ih_{23} & 0 \\ 0 & 0 & h_{23} & 0 & 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} \\ 0 & 0 & 0 & h_{24} & 0 & 0 & h_{23} & 0 & 0 & h_{22} & 0 & 0 & h_{21} & 0 & 0 & 0 \\ h_{21} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{24} & 0 & 0 & ih_{23} & 0 & 0 & ih_{22} & 0 & 0 \\ 0 & 0 & h_{33} & 0 & 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{34} \\ 0 & 0 & 0 & h_{34} & 0 & 0 & h_{33} & 0 & 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 \\ h_{31} & 0 & 0 & 0 & 0 & 0 & 0 & h_{34} & 0 & 0 & h_{33} & 0 & 0 & ih_{32} & 0 & 0 \\ 0 & h_{32} & 0 & 0 & h_{31} & 0 & 0 & 0 & 0 & 0 & h_{34} & 0 & 0 & ih_{33} & 0 & 0 \\ 0 & 0 & 0 & h_{44} & 0 & 0 & h_{43} & 0 & 0 & h_{42} & 0 & 0 & h_{41} & 0 & 0 & 0 \\ h_{14} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} & 0 & 0 & ih_{43} & 0 & 0 & ih_{42} & 0 & 0 \\ 0 & h_{24} & 0 & 0 & h_{41} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} & 0 & 0 & ih_{43} & 0 \\ 0 & 0 & h_{34} & 0 & 0 & h_{42} & 0 & 0 & h_{41} & 0 & 0 & 0 & 0 & 0 & 0 & ih_{44} \end{bmatrix} \quad (6.94)$$

where the row-permutation matrix \mathbf{M}_q is :

$$\mathbf{M}_q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6.95)$$

Notice that the matrix $\mathbf{M}_q \hat{\mathbf{H}}$ is a sparse matrix and it has a very special form: by decomposition into 4×4 blocks, the matrix can be represented by:

$$\mathbf{M}_q \hat{\mathbf{H}} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 & \mathbf{C}_1 & \mathbf{D}_1 \\ \mathbf{D}_2 & \mathbf{A}_2 & \mathbf{B}_2 & \mathbf{C}_2 \\ \mathbf{C}_3 & \mathbf{D}_3 & \mathbf{A}_3 & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{C}_4 & \mathbf{D}_4 & \mathbf{A}_4 \end{bmatrix} \quad (6.96)$$

The matrix \mathbf{A}_i are all diagonal:

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & 0 & 0 \\ 0 & a_2 & 0 & 0 \\ 0 & 0 & a_3 & 0 \\ 0 & 0 & 0 & a_4 \end{bmatrix}$$

The matrix \mathbf{B}_i are under form:

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 & b_4 \\ b_1 & 0 & 0 & 0 \\ 0 & b_2 & 0 & 0 \\ 0 & 0 & b_3 & 0 \end{bmatrix}$$

The matrix \mathbf{C}_i are under form:

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & c_4 \\ c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \end{bmatrix}$$

The matrix \mathbf{D}_i are under form:

$$\mathbf{D} = \begin{bmatrix} 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_4 \\ d_1 & 0 & 0 & 0 \end{bmatrix}$$

Let \bar{y}' denote the product of $\mathbf{M}_q \bar{y}$ and we have:

$$\bar{y}' = \mathbf{M}_q \hat{\mathbf{H}} \mathbf{D}_\chi \bar{s} + \bar{w} \quad (6.97)$$

Let us write $\bar{y}' = \{\bar{y}'_1, \bar{y}'_2, \bar{y}'_3, \bar{y}'_4\}^T$, $\bar{s}' = \{\bar{s}'_1, \bar{s}'_2, \bar{s}'_3, \bar{s}'_4\}^T$ and $\bar{w}' = \{\bar{w}'_1, \bar{w}'_2, \bar{w}'_3, \bar{w}'_4\}^T$. The MIMO system turns to be:

$$\begin{bmatrix} \bar{y}'_1 \\ \bar{y}'_2 \\ \bar{y}'_3 \\ \bar{y}'_4 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 & \mathbf{C}_1 & \mathbf{D}_1 \\ \mathbf{D}_2 & \mathbf{A}_2 & \mathbf{B}_2 & \mathbf{C}_2 \\ \mathbf{C}_3 & \mathbf{D}_3 & \mathbf{A}_3 & \mathbf{B}_3 \\ \mathbf{B}_4 & \mathbf{C}_4 & \mathbf{D}_4 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \Phi & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Phi & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Phi & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \Phi \end{bmatrix} \begin{bmatrix} \bar{s}_1 \\ \bar{s}_2 \\ \bar{s}_3 \\ \bar{s}_4 \end{bmatrix} + \begin{bmatrix} \bar{w}_1 \\ \bar{w}_2 \\ \bar{w}_3 \\ \bar{w}_4 \end{bmatrix} \quad (6.98)$$

The particular form of the matrix \mathbf{A}_i , \mathbf{B}_i , \mathbf{C}_i and \mathbf{D}_i allows us to perform a block-wise QR decomposition, which gives :

$$\begin{bmatrix} \bar{y}'_1 \\ \bar{y}'_2 \\ \bar{y}'_3 \\ \bar{y}'_4 \end{bmatrix} = \begin{bmatrix} \mathbf{A}'_1 & \mathbf{B}'_1 & \mathbf{C}'_1 & \mathbf{D}'_1 \\ \mathbf{0} & \mathbf{A}'_2 & \mathbf{B}'_2 & \mathbf{C}'_2 \\ \mathbf{0} & \mathbf{0} & \mathbf{A}'_3 & \mathbf{B}'_3 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}'_4 \end{bmatrix} \begin{bmatrix} \Phi & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Phi & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Phi & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \Phi \end{bmatrix} \begin{bmatrix} \bar{s}_1 \\ \bar{s}_2 \\ \bar{s}_3 \\ \bar{s}_4 \end{bmatrix} + \begin{bmatrix} \bar{w}'_1 \\ \bar{w}'_2 \\ \bar{w}'_3 \\ \bar{w}'_4 \end{bmatrix} \quad (6.99)$$

In fact, this structure is divided system into 4 layers. For each layer, the system has a diagonal channel matrix form. This system contains a generation matrix of an algebraic extension Φ that we can perform the algebraic reduction algorithm onto it.

We propose the following DFE-like strategy to decode this MIMO system:

- Start from the layer 4. We use algebraic reduction to decode the system:

$$\bar{y}'_4 = \mathbf{A}'_4 \Phi \bar{s}_4 + \bar{w}'_4$$

We obtain $\hat{\bar{s}}_4$;

- For layer 3, use the precedent result $\hat{\bar{s}}_4$ to remove the inter-layer interference:

$$\bar{y}'_3 \leftarrow \bar{y}'_3 - \mathbf{B}'_3 \hat{\bar{s}}_4$$

Decode the following system using algebraic reduction:

$$\bar{y}'_3 = \mathbf{A}'_3 \Phi \bar{s}_3 + \bar{w}'_3$$

We obtain $\hat{\bar{s}}_3$;

- For layer 2, we have:

$$\bar{y}'_2 \leftarrow \bar{y}'_2 - \mathbf{B}'_2 \hat{\bar{s}}_3 - \mathbf{C}'_2 \hat{\bar{s}}_4$$

Perform algebraic reduction algorithm to decode system:

$$\bar{y}'_2 = \mathbf{A}'_2 \Phi \bar{s}_2 + \bar{w}'_2$$

We obtain $\hat{\bar{s}}_2$;

- For layer 1:

$$\bar{y}'_1 \leftarrow \bar{y}'_1 - \mathbf{B}_1 \hat{s}_2 - \mathbf{C}_1 \hat{s}_3 - \mathbf{D}_1 \hat{s}_4$$

Using algebraic reduction, decode system:

$$\bar{y}'_1 = \mathbf{A}'_1 \Phi \bar{s}_1 + \bar{w}'_1$$

We get \hat{s}_1 .

It is easy to find that the QR-decomposition pre-processing is also simplified since this processing is taken on block-wised diagonal matrix. In addition, the pre-processing is realized with only permutation operations, which is very easy for hardware level implementation. For each layer, after the AR, we perform the ZF decoder in the lattice searching phase that the global complexity is reduced to the DFE decoder level.

6.5.3 Performance

This AR-DFE method can be implemented to decode the 4×4 Perfect code with very low complexity. The performance using AR-DFE method is given in Figure 6.18 where the performance of AR-DFE in single-relay two-antenna NAF scenario is compared with the BLAST code decoded with classic DFE decoder. Each codeword contains 16 symbols with QPSK constellation. The performance of cooperation system is reported in LPG configuration with $G_{sr} = G_{rd} = 0\text{dB}$ and in HPG configuration with $G_{sr} = G_{rd} = 10\text{dB}$. Simulation results show that the cooperative 4×4 Perfect code using AR-DFE decoder outperforms the non-cooperative BLAST system using DFE decoder.

Although the AF-DFE's performance is far from the ML decoder, this algorithm is low complexity solution for the application of 4×4 Perfect code for low data rate. More advanced algebraic reduction technique is our goal in the future.

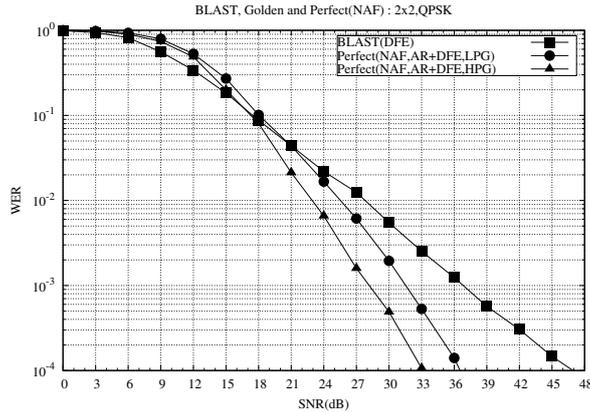


Figure 6.18: ZF-DFE decoder with algebraic reduction for 4x4 Perfect code in single-relay two-antenna NAF system

CONCLUSION AND PERSPECTIVES

In this thesis, we present the baseband of NAF cooperation system and the STBC coding scheme which can be utilized in the single-relay single-antenna NAF system and single-relay two-antenna NAF system. There are many other cooperation systems using DF protocol or OAF protocol. Compared with the DF solution, the AF solution is more simple to implemented and more adapted to the Wi-Fi network. The OAF solution can be a simple cooperation system, however, the orthogonality brings a loss of spectral efficiency. Therefore, we propose the NAF cooperation in the current Wi-Fi system with cooperative STBC.

In order to implement the proposed STBC, many MIMO decoders are analyzed in terms of WER and complexity in chapter 2. Regarding the processing latency, the sub-optimal decoders are grouped into the fixed latency decoders which are for hardware level implementation and the variant latency decoders which are more adapted for the software solution. Based on the simulation results, SEF and KSE decoders are selected in our simulations for their quick and quasi-ML performances.

Based on IEEE 802.11a PHY layer, we propose a cooperative system Relay-SISO in realizing the single-relay single-antenna NAF scenario in chapter 3. The new cooperative PHY layer contains the full definitions of frame structures, the AF procedure and the cooperation channel estimation methods. The CFO problem is also discussed addressed in this system. Simulation results confirm the advantage of cooperative diversity in flat fading channel and in multipath channel.

However, in the low SNR range Relay-SISO system gives a worse performance due to the amplification of noise term. In order to solve this problem, a new strategy is proposed by combining the IEEE 802.11a and Relay-SISO system in chapter 4. Named Hybrid mode, this method is realized by probing the cooperation channel to decide the transmission mode. In order to examine the channel's quality, we propose to generate a channel metric for both Relay-SISO and IEEE 802.11a system by reusing the MIMO decoder's architecture. According to our simulations, this metric provides a good indication and the "crossing-point" problem is generally resolved.

Another optimization on Relay-SISO system is to use the partitioned Golden code with punctured convolutional code. The partition chain over Golden code is studied where each partition gives a determinant gain of 2. The direct use of partitioned Golden code can not improve the system's performance for the loss of spectral efficiency. We propose to recompensate the loss of spectral efficiency by puncturing the convolutional code for higher coding rate. For the implementation, we propose an easy bit mapping and demapping method. Simulation results show the coding gain of partitioned Golden code in Relay-SISO system as well as in MIMO system.

For the IEEE 802.11n standard, we achieve the cooperative diversity in the MIMO context. As an extension of IEEE 802.11n in mixed mode, the proposed PHY layer Relay-MIMO is compatible with IEEE 802.11a system and Relay-SISO system. This cooperative system operates with the proposed frame formats and channel estimation procedures. As shown in the simulations results, this system reveals the cooperative diversity gain. However, in terms of coding gain or efficient throughput, we constate that IEEE 802.11n system is good enough specially in multipath environment that the advantage of Relay-MIMO is very limited. In addition, we propose a specific decoder using algebraic reduction method for the 4×4 Perfect code. This decoder solution sheds some light on hardware implementation of the Perfect code for its very low complexity.

For the future works, the MIMO decoder is always a challenge for the applications of the cooperative STBC. This task includes the channel processing, the lattice searching method and possibly the specific STBC decoder like algebraic reduction decoder. The research works on MIMO decoder can aim to provide not only a decoder fo simulations but also a decoder prototype for hardware implementation. The tradeoff between performance and complexity is very important for the sub-optimal decoders regarding the system demands.

The other types of cooperation scenarios can be eventually studied for the application of cooperative diversity in the Wi-Fi system. For the NAF based Relay-SISO system and Relay-MIMO system, , there are still a lots of optimization works to do. First is the power allocation optimization. The first can be the power allocation problem since the power allocation may have a great influence on the coding gain. Another optimization can be made by constructing new STBC for multi-carrier cases. As we see from the simulation results, the performance with multipath channel model is very different from the one with flat fading channel. Thus, it is always a challenge to apply the research achievements in the real environment.

PUBLICATIONS

1. Zhipeng Zhao and Belfiore J.C., "Application of Cooperative Diversity in 802.11a Ad-hoc Networks", *Computer Communications and Networks*, ICCCN 2007, 13-17 Aug, 2007, Honolulu, HI, page 1016.
2. Zhipeng Zhao, "Wi-Fi in High-Speed Transport Communications", *Intelligent Transport Systems Telecommunications*, ITST 2009 9th International Conference, 20-22 Oct, 2009, Lille, Page 430-434.
3. Zhipeng Zhao, "MIMO: From Theory to Implementation" chapter IEEE 802.11n, *COST 2100 Training School*, edited by Alain Sibille, Claude Oestges, Alberto Zanella, Academic Press, 1st edition, November 15, 2010.

BIBLIOGRAPHY

- [1] J. N. Laneman and G. W. Wornell, “Energy-efficient antenna sharing and relaying for wireless networks,” in *IEEE Wireless Comm. and Networking Conf. (WCNC)*, Chicago, IL, USA, Sep. 2000.
- [2] —, “Exploiting distributed spatial diversity in wireless networks,” in *Alerton Conf. Communications, Control, and Computing*, Monticello, IL, USA, Sep. 2000.
- [3] K. Azarian, H. E. Gamal, and P. Schniter, “On the achievable diversity-multiplexing tradeoff in half-duplex cooperative channels,” *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4152–4172, Dec. 2005.
- [4] S. Yang and J.-C. Belfiore, “Optimal space-time codes for the mimo amplify-and-forward cooperative channel,” *IEEE Trans. Inf. Theory*, vol. 53, no. 2, pp. 647–663, Feb. 2007.
- [5] L. Zheng and D. N. C. Tse, “Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels,” *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.
- [6] A. Sendonaris, E. Erkip, and B. Aazhang, “User cooperation diversity - part i: System description,” *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.
- [7] —, “User cooperation diversity - part ii: Implementation aspects and performance analysis,” *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1939–1948, Nov. 2003.
- [8] T. M. Cover and A. A. E. Gamal, “Capacity theorems for the relay channel,” pp. 572–584, Sep. 1979.
- [9] J. N. Laneman, G. W. Wornell, and D. N. C. Tse, “An efficient protocol for realizing cooperative diversity in wireless networks,” in *ISIT*, Washington DC, USA, Jun. 2001.
- [10] J. N. Laneman and G. W. Wornell, “Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks,” pp. 2415–2425, Oct. 2003.
- [11] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, “Cooperative diversity in wireless networks: Efficient protocols and outage behavior,” *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.
- [12] J.-C. Belfiore, G. Rekaya, and E. Viterbo, “The golden code : A 2×2 full-rate space-time code with nonvanishing determinants,” *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1432–1436, Apr. 2005.

- [13] F. Oggier, G. Rekaya, J.-C. Belfiore, and E. Viterbo, "Perfect space time block codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 3885–3902, Sep. 2006.
- [14] M. O. Damen, H. E. Gamal, and G. Caire, "Mmse-gdfe lattice decoding for under-determined linear channels," *IEEE International Symposium on Information Theory*, p. 565, Jul. 2004.
- [15] H. E. Gamal, G. Caire, and M. O. Damen, "Lattice coding and decoding achieve the optimal diversity-multiplexing tradeoff of mimo channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 968–985, Jun. 2004.
- [16] A. D. Murugan, H. E. Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: Rediscovering sequential decoding," *IEEE Trans. Inf. Theory*, vol. 52, pp. 933–953, Mar. 2006.
- [17] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, "Simplified processing for high spectral efficiency wireless communication employing multi-element arrays," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 11, pp. 1841–1852, Nov. 1999.
- [18] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, pp. 2389–2402, 2003.
- [19] A. K. Lenstra, A. W. L. Jr., and L. Lovasz, "On factoring polynomials with rational coefficients," *Math. Annalen*, no. 261, pp. 515–534, 1982.
- [20] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," vol. 45, no. 5, pp. 1639–1642, Jul. 1999.
- [21] 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.
- [22] G. Rekaya and J.-C. Belfiore, "On the complexity of ml lattice decoders for decoding linear full rate space-time codes," in *ISIT*, Yokohama, Japan, Jul. 2003.
- [23] R. M. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inf. Theory*, vol. 9, no. 2, pp. 64–74, Apr. 1963.
- [24] F. Jelinek, "Fast sequential decoding algorithm using a stack," *IBM J. Res. Develop.*, vol. 13, pp. 675–685, Nov. 1969.
- [25] Z. Guo and P. Nilsson, "Reduced complexity schnorr-euchner decoding algorithms for mimo systems," vol. 8, no. 5, pp. 286–288, May 2004.
- [26] —, "Algorithm and implementation of the k-best sphere decoding for mimo detection," pp. 491–503, Mar. 2006.
- [27] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-Speed Physical Layer in the 5 GHz Band*, IEEE 802.11a Std., 1999.

- [28] J. Medbo and P. Schramm, *Channel Models for HIPERLAN/2 in Different Indoor Scenarios*, 1998.
- [29] H. Cohen, *A Course in Computational Algebraic Number Theory*. New York: Springer-Verlag, 1996.
- [30] D. Champion, J.-C. Belfiore, G. Rekaya, and E. Viterbo, “Partitionning the golden code : A framework to the design of space-time coded modulation,” in *Canadian Workshop on Information Theory (CWIT)*, Montreal, Canada, Sep. 2005.
- [31] G. Forney, “Coset codes - part i: Introduction and geometrical classification,” *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1123–1151, Sep. 1988.
- [32] ———, “Coset codes - part ii: Binary lattices and related codes,” *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1152–1187, Sep. 1988.
- [33] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification: Amendment 5: Enhancements for Higher Throughput*, IEEE P802.11n/D5.0 Std., 2008.
- [34] G. Rekaya, J.-C. Belfiore, and E. Viterbo, “A very efficient lattice reduction tool on fast fading channels,” in *International Symposium on Information Theory and its Applications*, Parma, Italy, October 10-13 2004.
- [35] H. Cohen, *Advanced Number Theory*. New York: Dover Publication Inc., 1980.