PHYSICAL LAYER NETWORK CODING USING LATTICE CODES

WILLIAM LIU

A Thesis Submitted in Fulfilment of Requirements for the Degree of
Doctor of Philosophy of Imperial College London and
Diploma of Imperial College

2016

Communications and Signal Processing Group
Department of Electrical and Electronic Engineering
Imperial College London

Abstract

In this thesis, using lattice codes and channel codes, we propose several solutions to improve the performance of physical-layer network coding (PNC) schemes over the two-way relay network as well as the multi-way relay network.

In the first part of our work, we propose an efficient equation-coefficient search algorithm for compute-and-forward over complex-valued channels, which guarantees to find the optimal coefficients over the ring of Gaussian integers as well as the ring of Eisenstein integers. We also propose an optimal coefficient search algorithm for the Integer-Forcing compute-and-forward receiver, when multiple receive antennas are available at the relay. The proposed algorithms are shown to have low polynomial complexity in transmitting user number and we use numerical results to verify the performance of our algorithms against other competitive search algorithms in literature.

In the second part of our work we investigate the performance of popular PNC schemes for the two-way relay network and derive their end-to-end achievable rates with channel inversion precoding over the Rayleigh fading channel. We also propose a power efficient precoding scheme based on compute-and forward-to improve the network throughput.

In the third part of our work, after introducing the novel concept of compatibility between channel code and lattice code, we propose several low complexity joint channel coded network coding (JCCNC) schemes for PAM and QAM two-way relay networks. Firstly, using LDPC codes, we propose the lattice-decoded JCCNC (LD-JCCNC) scheme. Secondly, we propose two JCCNC schemes using low density construction A (LDA) lattices and polar lattices, which we call LDA-JCCNC and PLA-JCCNC respectively. In both JCCNC schemes, channel coding and network coding operations are integrated through the use of lattice decoding at the relay, where the arithmetic addition of lattice codewords is directly decoded using their corresponding lattice decoding algorithm. In contrast with most existing JCCNC schemes which are restricted to set types of modulation schemes, we also give clear methodology on how our JCCNC schemes can be applied to general types of higher order PAM and QAM modulation schemes.

In the final part of our work, we extend our novel JCCNC schemes to the general multi-way relay network for the compute-and-forward scheme, and show how they can be used with general types of PAM and QAM modulation systems. Detailed simulation analysis are carried out to evaluate and compare the performance of all proposed JCCNC schemes in relation to existing schemes in literature.

Copyright Declaration

The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives licence. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the licence terms of this work.

Declaration of Originality

I hereby declare that this thesis is the product of my own work and has not been submitted in any form for another degree or diploma at any university. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given in the bibliography. The key original contributions of this thesis are the following:

- Efficient and Optimal equation coefficient search algorithms to solve the Shortest Vector Problem in both standard compute-and-forward and MIMO compute-and-forward over Rayleigh fading channel (Chapter 2).
- Derivation of end-to-end capacities of various physical-layer network coding schemes with channel inversion precoding for two-way relay over Rayleigh fading channel (Chapter 3).
- Proposal of a power-efficient precoding scheme for compute-and-forward over two-way relay channel (Chapter 3).
- Introduced the concept of lattice code and error correction code compatibility which allows network coding and channel coding to be integrated in their operations through use of lattice decoding at relay (Chapter 4)
- Proposal of lattice-decoded joint channel coded network coding (LD-JCCNC) scheme based on lattice code and channel code compatibility for both two way relay network and multi-way relay network. (Chapters 4 and Chapter 5)
- Proposal of LDA lattice aided channel coded network coding scheme, compatible with general QAM and PAM modulation, for two-way relay network and multi-way relay networks (Chapters 4 and Chapter 5)
- Proposal of polar lattice aided channel coded network coding scheme, compatible with general QAM and PAM modulation, for two-way relay network and multi-way relay networks (Chapters 4 and Chapter 5).

William Liu
28/09/2016

Acknowledgements

I would like to take this opportunity to express my deep gratitude to my PhD Supervisor Dr Cong Ling, who invested a lot of faith in me and worked hard to secure the funding for this work. His guidance and support has been invaluable to me over the last four years. Under his supervision, I was not only offered clear direction and insight but also a huge amount of flexibility and freedom to pursue research areas of personal interest. His great patience and encouragement was a major source of support through four years that was not without frustration.

I would also like thank Liu Ling, Yan Fei and Wang Zheng, all PhD students or former PhD students of Dr Cong Ling, who I have had the pleasure of working and collaborating with throughout the last four years. In particular, Yan Fei who generously helped me with my work on lattices, and Liu Ling who was ever present and collaborated with me for my work on the application polar lattices to channel coded network coding. With the help of these great friends, I was able to overcome the various challenges I faced, both in research and in life. I would also like to thank the staff and researchers and fellow students at Imperial College Communications and Signal Processing department, for making the last four years productive and enjoyable for me.

Finally I would like thank my parents, my brother and my beautiful Daria for their unconditional love, nonstop encouragement and relentless support. They taught me the necessity to be strong in the face of difficulty.

Contents

Abstract 2
Copyright Declaration 3
Declaration of Originality 4
Acknowledgements 5
Publications 9
List of Figures 10
List of Tables 12
Notations 13
Acronyms 14
1 Introduction 15
1.1 Physical Layer Network Coding 15
1.2 Compute-and-Forward Relaying Scheme 17
1.2.1 Compute and Forward System Structure 19
1.2.2 Decoding of Integer Linear Equations 20
1.2.3 MIMO Compute-and-Forward 24
1.3 Channel Coded Physical Layer Network Coding 25
1.3.1 Channel Coded Two-way Relay Systems 25
1.3.2 Channel coded Multi-way Relay Systems 29
1.4 Lattice codes 30
1.4.1 Lattice Properties 31
1.4.2 Some Important Lattices 34
1.4.3 Voronoi Constellation 35
1.5 Thesis Organization 37
2 Parameter Selection for Compute-and-Forward 38
2.1 Overview of Existing Coefficient Search Methods 40
2.2 Proposed Optimal algorithm for Complex valued channel 42
2.2.1 Proposed algorithm: C-F over Gaussian Integers 44
2.2.2 Proposed Algorithm: C-F over Eisenstein Integers 48
2.2.3 Proposed Algorithm for MIMO C-F over Complex channels 52
2.3 Complexity Analysis 54
2.4 Numerical Results 57
3 Capacity of PNC schemes on Two-way Relay 65

3.1 PNC Schemes for Two-Way Relay 65
3.2 Capacity of PNC Schemes over AWGN Channel 68
3.3 Capacity of PNC Schemes over Fading Channel 72
3.4 Capacity of Precoded PNC Schemes over Rayleigh Fading Channel 76
3.5 Capacity of Channel-aware Precoding for Compute-and-Forward over Fading Channel 81
4 New JCCNC Schemes for Two-way Relay Systems 85
4.1 Review of LDPC Code 87
4.1.1 LDPC Code 87
4.2 The Lattice-Decoded (LD)-JCCNC Schemes 93
4.2.1 LD-JCCNC Scheme for q-PAM Two-way relay Systems 95
4.2.2 LD-JCCNC Scheme for q^2-QAM Two-way relay Systems 99
4.2.3 Complexity Analysis 103
4.2.4 Simulation Verification 104
4.2.5 Summary 108
4.3 LDA-Aided JCCNC Scheme 108
4.3.1 LDA Lattice 108
4.3.2 LDA-JCCNC Scheme for PAM and QAM Two-way Relay Systems 111
4.3.3 Complexity Analysis 116
4.3.4 Simulation Verification 117
4.3.5 Summary 119
4.4 Polar Lattice Aided JCCNC Scheme 120
4.4.1 Polar Code 120
4.4.2 Polar Lattice 121
4.4.3 PLA-JCCNC Scheme for PAM and QAM Two-way Relay Systems 127
4.4.4 Simulation Verification 133
4.4.5 Summary 137
5 JCCNC Compute-and-Forward Scheme for Multi-way Relay 138
5.1 JCCNC Schemes for Compute-and-Forward 140
5.1.1 JCCNC Schemes for q-PAM Systems 141
5.1.1.1 Option-1 (The Standard JCCNC Scheme): 142
5.1.2 JCCNC schemes for q^2 QAM relay systems 147
5.1.2.1 Option-1 (Standard JCCNC Scheme): 148

5.1.3 JCCNC Schemes for q-QAM Systems 151
5.2 Simulation Results 154
5.3 Summary 158
6 Conclusions and Future Work 160
6.1 Conclusions 160
6.2 Future Work 162
Bibliography 163
Appendix 172

Publications

- Journal Papers
    1. W. Liu and C. Ling, "Efficient integer coefficient search for compute-and-forward," IEEE Transactions on Wireless Communications, vol. 15, no. 12, pp.8039-8050, Dec 2016.
- Conference Papers
    1. W. Liu, C. Ling, "LDA-Lattice Aided Network Coding for Two-Way Relay," IEEE Science and Information Conference (SAI), July, 2016.
    2. W. Liu, "Joint Channel Coding Network Coding for Multi-way Relay Systems," IEEE Science and Information Conference (SAI), July, 2016.
    3. W. Liu, "A new channel coding network coding scheme for two-way relay," IEEE Science and Information Conference (SAI), July, 2015.
    4. W. Liu, "Physical Layer Network Coding Schemes for Two-Way Relay," 9th IEEE/IET International Symposium on CSNDSP, July, 2014.
    5. W, Liu, "Compute-and-Forward for Two-Way Relay," 9th IEEE/IET International Symposium on CSNDSP, July, 2014.
    6. W, Liu, "Practical Aspects of Compute-and-Forward," International Conference On Advances In Computing, Electronics And Communication, 2013.

List of Figures

Figure 1-1: General model of compute-and-forward with L source nodes and M relay nodes. 19
Figure 1-2: Decoding structure of compute-and-forward. 21
Figure 1-3: Two-way relay with error correction coding and lattice coding 25
Figure 1-4: Schematic of channel coded decode-and-forward scheme 26
Figure 1-5: Schematic of channel coded Arithmetic-sum-and-forward scheme 27
Figure 1-6: Schematic of standard channel coded XOR-network coding scheme 28
Figure 1-7: Example of a nested lattice code with the A2 hexagonal lattice 33
Figure 1-8: Table of the best lattices for various dimensions, source: [64], Chapter 4. 35
Figure 1-9: Illustration of a Voronoi code decoder for a nested lattice created from the A2 hexagonal lattice. 37
Figure 2-1: Compute-and-forward rate loss incurred using integer coefficients obtained via complex-LLL approximation, for L=4 41
Figure 2-2: 3D plot of the optimization function in terms of α 46
Figure 2-3: Discontinuity representatives of Eisenstein integer points 50
Figure 2-4: Runtime performance ratio (Exhaustive Search CPU/Algorithm1 CPU) against SNR for L=2 61
Figure 2-5: Rate performance comparison for $L = 4$ 61
Figure 2-6: CPU time comparison for L=4 62
Figure 2-7: Rate performance comparison for L=8 62
Figure 2-8: CPU time comparison for L=8 63
Figure 2-9: Rate performance against transmitting user L, SNR=10dB 63
Figure 2-10: CPU time against transmitting user L, SNR=10dB 64
Figure 3-1 Model of two way relay channel 66
Figure 3-2: Achievable rates of relaying schemes over AWGN channels 72
Figure 3-3: Achievable rates of relaying schemes over fading channel without precoding. 76
Figure 3-4: Achievable rates of relaying schemes over fading channel with precoding 80
Figure 3-5: Achievable rates of relaying schemes over Rayleigh fading channel with channel aware precoding. 82
Figure 3-6: Symbol error rates of compute-and-forward over fading channel with and without the proposed channel precoding schemes. 84

Figure 4-1: Bipartite graph representation of an LDPC code. A short cycle of length 4 has been highlighted. 88
Figure 4-2: Message passing on the bipartite graph representing a parity check matrix 91
Figure 4-3: BPSK, BER performance with perfectly known channel information 105
Figure 4-4: BPSK, BER performance with noise power misestimated by 6 dB 106
Figure 4-5: 4-QAM, BER performance with perfectly known channel information 106
Figure 4-6: 4-QAM, BER performance with noise power misestimated by 6 dB 107
Figure 4-7: 4-QAM, BER performance with noise power misestimated by 8 dB\ 107
Figure 4-8: Factor graph of an integer LDA lattice 109
Figure 4-9: LDA lattice JCCNC scheme for the two-way relay 111
Figure 4-10: Constellation shape over Gaussian integers 112
Figure 4-11: Constellation shape over Eisenstein integers 112
Figure 4-12: SER performance on 5-PAM 118
Figure 4-13: SER performance on 5-QAM 118
Figure 4-14: Channel capacity for each level of partition chains 125
Figure 4-15: Encoding and decoding chain for a one-dimensional two-level polar lattice 126
Figure 4-16: Encoding and decoding chain for a two-dimensional four-level polar lattice 127
Figure 4-17: Example of polar lattice aided JCCNC scheme for two-way relay channel 129
Figure 4-18: BLER performance of 4-PAM PLA-JCCNC two-way relay system 135
Figure 4-19: BLER performance of 16-QAM PLA-JCCNC two-way relay system 136
Figure 5-1: A multi-way relay with ECC and lattice coding 140
Figure 5-2: Error performance of compute-and-forward for BPSK 155
Figure 5-3: Error performance of compute-and-forward for 3-PAM and 4-PAM 155
Figure 5-4: Error performance of compute-and-forward for 4-QAM 156
Figure 5-5: Error performance of compute-and-forward for 5-QAM 156

List of Tables

Table 4-1: Codeword elements and signal mappings of a noise free BPSK two-way relay system 96
Table 4-2: 4-QAM constellation mapping for GF(4) symbols 99
Table 4-3: GF(4) addition table 100
Table 4-4: Codeword elements and signal mappings of a noise free 4-QAM two-way relay system 100
Table 4-5: Codeword elements and signal mappings of a 5-PAM two-way relay system 114
Table 4-6: Codeword elements and signal mappings of a 5-QAM two-way relay system 115
Table 4-7: Complexity comparison 117

Notations

$\mathrm{Z}, \mathrm{Z}[j], \mathrm{Z}[\omega]$	Set of integers, Gaussian integers, Eisenstein integers respectively
$\mathrm{R}, \mathrm{R}^+, \mathrm{C}$	Set of real numbers, positive real numbers, and complex numbers respectively
$\lvert\cdot\rvert$	Absolute value
$\lVert\cdot\rVert$	Frobenius norm
$[\cdot]$	Quantize to the nearest integer with half integers rounded upwards
$[\cdot]_{\mathrm{Z}[j]}$	Quantize to the nearest Gaussian integer, with half integer components rounded upwards
$[\cdot]_{\mathrm{Z}[\omega]}$	Quantize to the nearest Eisenstein integer
$\mathrm{Re}(\cdot)$	Real component of a complex number
$\mathrm{Im}(\cdot)$	Real component of a complex number
$\otimes$	Kronecker product
$(\cdot)^*$	Complex conjugate
$(\cdot)^H$	Hermitian transpose
$(\cdot)^T$	Transpose
$(\cdot)^{-1}$	Inverse and pseudoinverse
mod	General modulus operator

Acronyms

AF	amplify-and-forward
AWGN	additive white Gaussian noise
BMS	binary memoryless symmetric
DF	decode-and-forward
BP	belief propagation
DNF	denoise-and-forward
CF	compute-and-forward
GF	Galois Field
JCCNC	joint channel coded network coding
LD-JCCNC	lattice decoding JCCNC
LDA	low density construction A lattice
LDA-JCCNC	LDA aided JCCNC
PLA-JCCNC	polar lattice aided JCCNC
MAC	multiple access channel
MLAN	modulo-lattice additive noise channel
SC	Successive Cancellation

1 Introduction

In this thesis, through the use of lattice codes, we tackle a selected number of contemporary problems in the research area of physical layer network coding (PNC), and propose ways to improve the PNC schemes both in performance and in efficiency. Our solutions to optimizing the performance of PNC schemes is threefold; an optimal equation coefficient search algorithm for compute-and-forward; thorough investigation into popular two-way relay PNC schemes under Rayleigh fading culminating in proposals to efficiently improve their performance through precoding; and finally the proposal of three novel joint channel coded network coding(JCCNC) schemes using high performance lattice codes. Our novel JCCNC schemes are first introduced for the two-way relay network, and then adapted for the general multi-way relay network.

We start by investigating lattice related problems within compute-and-forward and PNC in general for the two-way relay and the multi-way relay networks. We then extend into the area of channel coded PNC for relay networks, where we propose JCCNC schemes based on lattice codes to improve performance on the state of the art. We know that lattices are powerful tools in the wireless setting and form an important class of structured codes, yet there is a dearth of work on practical implementations of high capability lattices in network coding in general. Given recent results on the constructions of powerful capacity achieving lattices with great error correction performances, such as low density construction A (LDA) lattices and polar lattices, there is real potential for the application of these lattice codes to communication systems as constellation codes that effectively integrate channel coding and network coding in one efficient step.

1.1 Physical Layer Network Coding

The motivation for modern research in wireless networks is finding ways to match the demand for reliable high speed wireless services such as high speed internet and multimedia applications. In wireless networks, signal interference between transmitted messages, and additive noise experienced at receiving nodes are the two main problems affecting network throughput. The broadcast nature of the wireless medium means the transmitted messages arrive not only at the intended receiver, but also at neighbouring nodes in the wireless vicinity. Given the limitations in energy and bandwidth, the focus of research is on ways to manage interference and maximize network capacity. Relaying strategies for the wireless

communications network has become a high priority area of research, and identified as a promising area of development for future wireless networks.

Relaying strategies can be deployed in ad hoc and cellular networks. Conventionally nodes in ad hoc networks transmit data from one end to another using data routing techniques where each node in the network simply store and forward their received data without modifying its content. However proposals have recently been made to replace the conventional routing scheme with network coding strategy. With physical layer network coding (PNC), where conventional network coding is applied to the wireless physical layer, intermediate nodes can handle input from multiple source nodes to compute functions of input data. In doing so, transferring the same amount of information requires fewer intervals and network throughput is increased. In this way, network coding allows for more efficient use of wireless spectrum by exploiting the broadcast property of the wireless channel. A very simple but effective model to examine PNC is the two way relay channel where two users exchange information through an intermediate node. PNC has received considerable research attention in recent years and many cooperative relaying strategies have been proposed for PNC over the two way relay [1]-[12]. Much of the investigation work in this thesis will be focused on the two way relay network.

The idea of PNC for the wireless network originated from network coding for wired network, which can be traced back to as early as early 2000 [13]-[16]. The principle of network coding is that intermediate nodes in the network should forward functions of transmitted messages rather than individual messages to increase the throughput. In [2] three relaying schemes, namely decode-and-forward, amplify-and-forward, and denoise-and-forward are studied and the conditions which maximize their rate over two-way relay channel are expressed. In 2006, three research groups [17]-[19] independently proposed the idea of PNC for the wireless network. It was shown that when the relay decodes and relays the modulo-two sum (XOR) of the transmitted packets, the throughput of a two-way relay channel can be significantly improved [17]-[18]. In [18], the effects of channel fading and potential phase misalignment are considered for the same XOR decoding PNC scheme for two-way relay scheme. The authors were able to show that for a large range of SNR values, the scheme outperforms both amplify-and-forward and decode-and-forward. In [3], the authors explicitly considers the use of lattice encoding and decoding for a joint physical layer network coding scheme for the two way relay channel, and derived the maximum achievable rate for this scheme. The effects of imperfect phase synchronization between nodes on PNC

was studied in [4] where the authors showed that network rate achieved by PNC under imperfect synchronization is still superior compared to classical network coding schemes under perfect synchronization.

1.2 Compute-and-Forward Relaying Scheme

In 2009, Nazer et al [20] extended PNC from the two way relay network to the general multi-way relay framework by introducing a new PNC and network relaying strategy, called compute-and-forward (C-F). The compute-and-forward PNC scheme exploits, rather than combat, the multiple access interference in a wireless relay network, resulting in an improved network throughput. In this PNC scheme, the relays, instead of recovering individual messages, attempt to reliably recover and pass an integer linear combination of transmitted messages, referred to as an "equation" of transmitted messages, to the destination. By receiving sufficient number of equations of transmitted messages, the destination is able to recover the individual messages by simply solving these linear equations. The scheme relies on the use of structured linear codes, specifically nested lattice codes which have good algebraic and statistical properties making it suitable for the linear computations of the Gaussian channel. With lattices, assuming all encoders use the same lattice codebook, then in additive Gaussian channel, an integer linear combination of the transmitted lattice codewords is still a lattice codeword. Standard random codes lack this property so that the decoder can only recover individual codewords at a much lower rate.

The compute-and-forward decoding at relay nodes in [20] is facilitated by the use of nested lattice codes constructed over the ring of integers, which ensures that an integer linear combinations of codewords is also a codeword. C-F using lattices created using Construction A over the ring of Gaussian integers and the ring of Eisenstein integers are covered in [23][24][25] respectively. It was shown in [24] that C-F with lattices over Eisenstein integers achieves superior performance in outage rate and error probability than C-F over Gaussian integers. A C-F scheme utilizing lattices over rings of imaginary quadratic integers is given in [26].

From a practical point of view, there are several open problems in compute-and-forward. One such problem is how to efficiently obtain the optimal equation coefficients to maximize the computational rate of the scheme. Aside from the optimal but computationally-taxing exhaustive search method, a number of efficient but suboptimal solutions have been proposed in literature. The vast majority of those proposed solutions are limited in their practicality

either due to poor performance for large number of transmitting users [27][28][31] or because they assume a simplified channel model [32]. In Chapter 2, we address this problem in great detail and propose an efficient equation coefficient search algorithm that guarantees to find the optimal equation coefficients for compute and forward over Rayleigh fading channel. We also adapt our search algorithm to obtain the optimal equation coefficients for C-F under the MIMO setting, when multiple antennas are available at the relay.

Another open area of research in compute-and-forward is the performance limitations of the scheme that occur as a result of scaling and approximation of arbitrary valued channel coefficients using integer coefficients. Due to this phenomenon, referred to as Diophantine approximation problem, it was shown in [33] that compute-and-forward achieves achieve at most 2 degrees of freedom (DoF). The trade-off between the quality of the channel coefficient approximation and the degree of noise amplification is known as the Diophantine trade-off and is the limiting factor in the performance of C-F.

On the other hand, while lattice decoding for compute-and-forward has been shown to be limited from DoF point of view, the lattice decoding algorithms are very fast and efficient by fully utilizing congruent and symmetric properties of lattice codes. Therefore we want to find ways to fully realise the structural benefits of lattice codes in compute and forward, but at the same time, not be subjected to performance limitations that occur due to channel quantization in compute-and-forward. We tackle this problem in Chapter 3 where we investigate the performance of compute-and-forward and other popular PNC schemes for the two way relay channel. In addition we propose a power efficient transmission scheme based on compute and forward to significantly improve its performance over the Rayleigh fading channel.

In the final part of our work, we focus on the application of powerful lattice codes to PNC over a number of network settings. We note that so far, almost all the research works published on the application of lattice codes in compute-and-forward are based on studies of its achievable asymptotic rate from information theory point of view. In particular [42][44][45] are the only work covering the practical performance of compute-and-forward scheme under lattice encoding and decoding. Even so, only one and two dimensional integer lattices with basic shaping were assumed in those schemes. Therefore in those implementations, the benefits of using lattice codes in compute-and-forward and PNC in general are not fully realized. In recent years new capacity-achieving lattices have been proposed, which are constructed from powerful error correction codes, in particular the polar lattice and the low density construction A (LDA) lattice. Therefore, there exists great

potential to apply these lattices to compute-and-forward as constellation codes in order to optimise the overall network through integrated channel coding and network coding.

1.2.1 Compute and Forward System Structure

In compute-and-forward relaying scheme, nested lattice codes are employed by the requirement that an integer linear combination of codewords must still be a codeword. For a multi-way relay network with many source nodes and relay nodes, the general computation model of compute-and-forward can be summarized below in Figure 1-1.

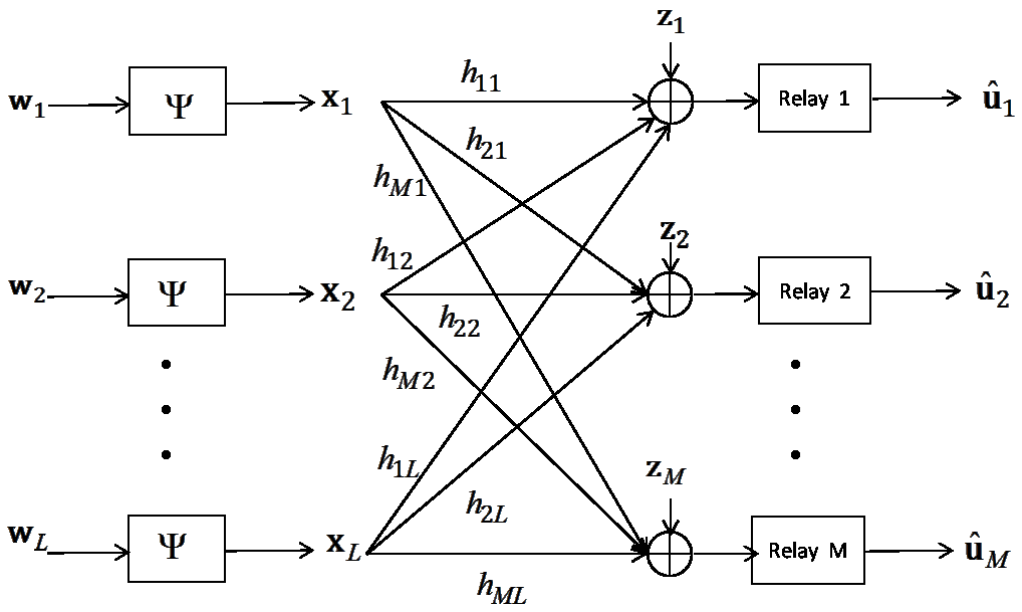


Figure 1-1: General model of compute-and-forward with L source nodes and M relay nodes.

In the encoding process, the messages $\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_L \in \mathrm{F}_q^k$ from the L source nodes are k-dimensional row vectors with individual elements drawn from a GF(q) finite field. The L message vectors are encoded to L number of n-dimensional complex-valued codewords $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_L \in \mathrm{C}^n$, which are then transmitted over a channel with complex-valued coefficients h_{ml} for $l = 0,....,L$ and $m = 0,....,M$. Denote Λ_1 as the fine lattice and Λ as the coarse lattice, the codewords to be sent are drawn from a nested lattice code defined by the quotient Λ_1 / Λ where $\Lambda \subseteq \Lambda_1$. The received signal at each relay node is a linear combination of the transmitted signals from all the source nodes plus additive noise which can be expressed as:

$$\mathbf{y}_m = \sum_{l=1}^{L} h_{ml}\mathbf{x}_l + \mathbf{z}_m . \tag{1-1}$$

The channel coefficients between the source nodes and the relay nodes are assumed perfectly known at the relay nodes. The n-dimensional codewords are subject to average power constraint of $\frac{1}{n}E\|\mathbf{x}_l\|^2 \leq P$, and $\mathbf{z}_m \sim \mathrm{CN}(0, \sigma^2\mathbf{I}_n)$ is a circularly-symmetric jointly-Gaussian complex random vector where $\mathbf{I}_n$ is the $n \times n$ identity matrix.

The target of the decoder at each relay is to decode from the noisy signal in (1-1) an estimate $\hat{\mathbf{u}}_m$ of an integer linear combination of transmitted codewords from all the source nodes, given by:

$$\mathbf{u}_m = \left[\sum_{l=1}^{L} a_{ml}\mathbf{w}_l\right] \bmod \Lambda , \tag{1-2}$$

where Λ is the shaping lattice and $a_{ml} \in Z + jZ$ is the Gaussian integer coefficient to be determined in the decoding process in order to maximize the achievable rate of compute-and-forward. The integer equation coefficients are therefore a key design parameter of compute-and-forward. The nature of the integer coefficients depends on the underlying channel and takes on the form of rational integers for real valued channels and typically Gaussian integers for complex valued channels. A compute-and-forward scheme over the ring of Eisenstein integers is also proposed in [24] and [25].

Note that because of the one to one mapping between messages and lattice codewords, decoding an integer linear combination of codeword vectors is equivalent to decoding an integer linear combination of message vectors. The decoded integer linear combinations of codewords, also known as equations, are forwarded to the destination node, where after receiving enough number of linear independent equations, the transmitted messages can be recovered. The decoding error rate is determined by using the decision rule $P_e = \Pr(\hat{\mathbf{u}}_m \neq \mathbf{u})$.

1.2.2 Decoding of Integer Linear Equations

As the design principle of compute-and-forward is based on the algebraic property that an integer linear combination of a lattice codeword is still a lattice codeword, the goal of the decoder at the relay is to reliably recover an integer linear combination of the transmitted codewords. However, in reality, the channel coefficients will take on arbitrary values, so that the linear combination of transmitted codewords received at the relay is no longer an integer

linear combination. Therefore the task of the decoder is to find an integer linear combination which is as co-linear as possible to the received signal without considering the noise. The decoding method of compute-and-forward can be summarized below in Figure 1-2.

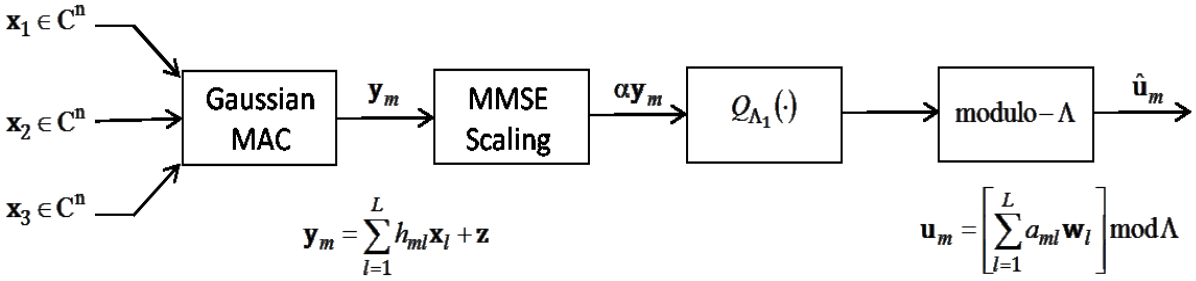


Figure 1-2: Decoding structure of compute-and-forward.

The decoding process includes scaling the received signal with an optimal scaling factor, and then quantizing the scaled signal to the nearest lattice point. The purpose of scaling by a factor α is to make the scaled signal $\tilde{\mathbf{y}}_m$ as close to an integer linear combination of the transmitted codewords as possible under the condition of simultaneously minimizing the effective noise of C-F in equation (1-4). The scaled signal can be expressed as:

$$\begin{aligned}\tilde{\mathbf{y}}_m &= \alpha\mathbf{y}_m = \alpha\left(\sum_{l=1}^{L} h_{ml}\mathbf{x}_l + \mathbf{z}_m\right)\\ &= \sum_{l=1}^{L} a_{ml}\mathbf{x}_l + \sum_{l=1}^{L}(\alpha h_{ml} - a_{ml})\mathbf{x}_l + \alpha\mathbf{z}_m \\ &= \sum_{l=1}^{L} a_{ml}\mathbf{x}_l + \mathbf{n}_{eff}\end{aligned} \tag{1-3}$$

where,

$$\mathbf{n}_{eff} = \sum_{l=1}^{L}(\alpha h_{ml} - a_{ml})\mathbf{x}_l + \alpha\mathbf{z}_m \tag{1-4}$$

From (1-3), it can be seen that the scaled signal consists of three components. The first component is the desired integer linear combination of the codewords. The second component is the noise caused by approximation error when approximating arbitrary valued channels by integer coefficients. The last component is related to the additive white Gaussian noise.

Standard lattice decoder denoted by $Q_{\Lambda_1}(\cdot)$ is then used to quantize the scaled signal to the nearest point in the fine lattice Λ_1. The quantized signal is then modulo operated over the

coarse lattice Λ to decode an estimate $\hat{\mathbf{u}}_m$ of the integer linear combination, effectively decoding the compute-and-forward processed signal to a point in the original lattice codebook.

$$\hat{\mathbf{u}}_m = Q_{\Lambda_1}(\alpha \mathbf{y}_m) \bmod \Lambda .$$

Therefore a main task of the compute-and-forward decoder is to select the coefficient vector $\mathbf{a}_m = [a_{m1}, a_{m2}, \cdots, a_{mL}]$ to minimize the effective noise $\mathbf{n}_{eff}$ expressed by equation (1-4).

1.2.2.1 Achievable Rate

It has been proved in [20] that for a complex valued wireless network with channel coefficient vector $\mathbf{h}_m = [h_{m1}, h_{m2}, \cdots h_{mL}] \in \mathrm{C}^L$ and equation coefficient vector $\mathbf{a}_m = [a_{m1}, a_{m2}, \cdots, a_{mL}] \in \{Z + jZ\}^L$, the following computation rate is achievable at the relay receiver:

$$\Re(\mathbf{h}_m, \mathbf{a}_m) = \max_{\alpha \in C} \log\left(\frac{P}{|\alpha|^2 + P\|\alpha \mathbf{h}_m - \mathbf{a}_m\|^2} \right). \tag{1-5}$$

where the computation rate is defined as the rate at which the relay can recover its desired integer linear combination of messages.

It has also been proven that this computation rate can be uniquely maximized by further choosing α to be the minimum mean square error (MMSE) coefficient which is given by:

$$\alpha_{\mathrm{MMSE}} = \frac{\mathbf{a}_m \mathbf{h}_m^H P}{1 + P\|\mathbf{h}_m\|^2}, \tag{1-6}$$

and that the computational rate is greater than zero for the range:

$$\|\mathbf{a}_m\| \le \Phi = \sqrt{1 + P\|\mathbf{h}_m\|^2} . \tag{1-7}$$

By substituting the MMSE scaling factor α_{MMSE} into (1-5) the resulting achievable computation rate can be written as:

$$\Re(\mathbf{h}_m, \mathbf{a}_m) = \log\left(\left(\|\mathbf{a}_m\|^2 - \frac{P\left|\mathbf{a}_m \mathbf{h}_m^H\right|^2}{1 + P\|\mathbf{h}_m\|^2}\right)^{-1}\right). \tag{1-8}$$

1.2.2.2 Parameter Selection

Selecting a larger value for the scaling factor α will allow for a better approximation of the noisy signal to an integer linear combination of codewords, which reduces any noise incurred from poor approximation of channel values to integer coefficients. However, it can be observed from (1-5) that a larger scaling factor also results in greater amplification of the original noise in (1-4). This trade-off between integer-approximation and noise amplification is known as the Diophantine trade-off [33]. In fact, the design criterion of the compute-and-forward receiver is to select the scaling factor α and the equation coefficient vector $\mathbf{a}_m$ so as to maximize the computation rate. This is also equivalent to minimizing the power of the effective noise of compute-and-forward [22]:

$$\mathbf{n}_{eff}(\alpha, \mathbf{a}_m) = \sigma^2 |\alpha|^2 + P\|\alpha \mathbf{h}_m - \mathbf{a}_m\|^2 \tag{1-9}$$

$$(\alpha^{\text{opt}}, \mathbf{a}^{\text{opt}}) = \underset{\alpha \in C, \mathbf{a}_m \neq 0}{\arg\min}\ \mathbf{n}_{eff}(\alpha, \mathbf{a}_m) \tag{1-10}$$

Using (1-6), the effective noise term in (1-9) can be rewritten as:

$$\mathbf{n}_{eff}(\mathbf{a}_m) = \mathbf{a}_m \mathbf{M} \mathbf{a}_m^H \tag{1-11}$$

where,

$$\mathbf{M} = P\left(\mathbf{I} - \frac{P}{1 + P\|\mathbf{h}_m\|^2} \mathbf{h}_m^H \mathbf{h}_m\right) \tag{1-12}$$

Because the matrix $\mathbf{M}$ is Hermitian and positive definite, it has a Cholesky decomposition $\mathbf{M} = \mathbf{L}\mathbf{L}^H$, where $\mathbf{L}$ is a lower triangular matrix. Thus (1-9) becomes:

$$\mathbf{n}_{eff}(\mathbf{a}_m) = \mathbf{a}_m \mathbf{M} \mathbf{a}_m^H = \|\mathbf{a}_m \mathbf{L}\|^2. \tag{1-13}$$

Therefore the optimal equation coefficient vector is given by,

$$\mathbf{a}_m^{\text{opt}} = \underset{\mathbf{a}_m \in Z[j]^L \setminus \mathbf{a}_m = 0}{\arg\min} f(\mathbf{a}) = \underset{\mathbf{a}_m \in Z[j]^L \setminus \mathbf{a}_m = 0}{\arg\min} \mathbf{a}_m \mathbf{M} \mathbf{a}_m^H = \underset{\mathbf{a}_m \in Z[j]^L \setminus \mathbf{a}_m = 0}{\arg\min} \|\mathbf{a}_m \mathbf{L}\|^2 \tag{1-14}$$

Thus finding the optimal parameters $\mathbf{a}_m$ in compute-and-forward is a shortest vector problem (SVP) of finding the non-zero minimum Euclidean norm point in a lattice with Gram matrix

$\mathbf{M}$. Since the performance of compute-and-forward depends heavily on the ability to select the optimal equation coefficients for any set of channel instances, parameter selection is the most important step in the scheme. In Chapter 2, we provide an overview of existing parameter search methods for compute-and-forward and propose an efficient and optimal method to solve the SVP in compute-and-forward over complex valued channel.

1.2.3 MIMO Compute-and-Forward

In [49] [50] MIMO C-F was introduced where the relay aims to decode the best integer linear combination of codewords using multiple antennas. The MIMO C-F system model consists of L transmitting users sending to relays with k receive antennas. Take $\mathbf{H}$ to be a $k \times L$ channel matrix and let $\mathbf{b} \in [b_1, b_2, \ldots b_k] \in \mathrm{C}^L$ be a judiciously chosen pre-processing vector which is applied to the channel output $\mathbf{y}$. Each relay aims to select the optimal coefficient vector $\mathbf{a} \in [a_1, a_2, \ldots a_k] \in \mathrm{Z}[j]^L$ to maximise the achievable computational rate of MIMO C-F, which is proven in [50] to be,

$$\Re(\mathbf{H}, \mathbf{a}, \mathbf{b}) = \max_{\alpha \in C} \log\left(\frac{P}{|\mathbf{b}|^2 + P\|\mathbf{bH} - \mathbf{a}\|^2} \right). \tag{1-15}$$

By taking derivative of the denominator of (1-15) with respect to $\mathbf{b}$, setting the result to 0 and solving for $\mathbf{b}$, it can be shown that the optimal integer forcing vector that maximises the achievable rate in (1-15) at each relay is given by:

$$\mathbf{b}^{\mathrm{opt}} = \mathbf{a}\mathbf{H}^H \left(P^{-1}\mathbf{I} + \mathbf{H}\mathbf{H}^H \right)^{-1}. \tag{1-16}$$

To maximize the computational rate, it is shown in [50] that it is sufficient to check the space of all Gaussian integer vectors $\mathbf{a}$ satisfying:

$$\mathbf{a} \le \Phi = \sqrt{1 + P\lambda_{\max}} \tag{1-17}$$

where $\lambda_{\max}$ is the maximum singular value of $\mathbf{H}$. Substituting optimal integer forcing vector $\mathbf{b}^{\mathrm{opt}}$ into (1-15), the rate optimization problem can be expressed as,

$$\begin{aligned} \mathbf{a}^{\mathrm{opt}} = \underset{\mathbf{a} \in \mathrm{Z}[j]^L \setminus \mathbf{a} \neq 0}{\arg\min}\ f(\mathbf{a}) &= \underset{\mathbf{a} \in \mathrm{Z}[j]^L \setminus \mathbf{a} = 0}{\arg\min} \left(\mathbf{a}\mathbf{M}\mathbf{a}^H \right) \\ &= \underset{\mathbf{a} \in \mathrm{Z}[j]^L \setminus \mathbf{a} = 0}{\arg\min} \left(\mathbf{a}\mathbf{V}\mathbf{D}\mathbf{V}^H\mathbf{a}^H \right) \end{aligned} \tag{1-18}$$

where $\mathbf{V} \in \mathrm{C}^{L \times L}$ is composed of the right singular vectors of $\mathbf{H}$ and $\mathbf{D} \in \mathrm{C}^{L \times L}$ is a diagonal matrix whose first k diagonal elements satisfy $r_i = (1 + P\lambda_i^2)^{-1}$, $i = 1, \ldots, k$ and the

remaining $L-k$ elements equal to 1. λ_i is the i-th singular value of $\mathbf{H}$. Therefore the search for the optimal integer coefficients vector for MIMO C-F is equivalent to finding the shortest vector in a lattice with gram matrix $\mathbf{VDV}^H$. In Chapter 2, we also propose an optimal deterministic algorithm to solve the SVP of compute-and-forward under the MIMO setting.

1.3 Channel Coded Physical Layer Network Coding

Error correction codes can be applied to relay networks the same way as it can be applied to point to point channels to increase overall network throughout. The general idea is that redundancy bits are added to the original information bits at the user nodes so that channel coded messages are transmitted from the users. Error correction codes (ECC) can be applied to relay systems via end-to-end approach or through link-to-link approach. In link to link approach, the relay performs channel decoding and re-encoding as well as PNC mapping. In the end to end approach, channel decoding is only performed by the end user. The link-to-link approach is superior in error performance as it 'cleans up' the information bits throughout the network and prevents accumulation of noise, especially when there are multiple intermediate relays between sender and receiver. In this work, we consider specifically the link-to-link approach and we assume perfect synchronisation in transmission so that the symbols of user nodes arrive at relay node perfectly aligned.

1.3.1 Channel Coded Two-way Relay Systems

Techniques of applying channel coding to PNC schemes for the two way relay network received significant interest in recent years. In this thesis, we will refer channel coded PNC schemes as joint channel-coded network coding (JCCNC) schemes in general. Figure 1-3 below shows a channel coded two-way relay system over additive white Gaussian noise (AWGN) channel. The messages *e* are first channel coded to a codeword *c* and then mapped to a lattice codeword *x* before transmission.

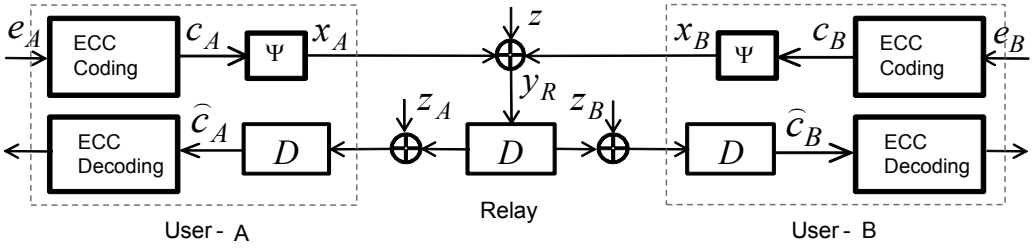


Figure 1-3: Two-way relay with error correction coding and lattice coding

The received superimposed signal at the relay node from the two source nodes is $y_R = x_A + x_B + z$, where z is Gaussian noise. The ultimate aim of channel coded network coding schemes is to decode at the relay the XOR of the transmitted messages $e_R = e_A \oplus e_B$ when messages are drawn from GF(2) field. When messages are drawn from GF(q), where q is a prime, then the relay aims to decode the modulo addition of the transmitted messages in GF(q). In this section, we review the state of the art JCCNC schemes in literature and classify them into three distinct categories:

JCCNC-1: Decode-and-forward strategy with channel coding. In this scheme, the relay node directly decodes messages e_A and e_B sent from the two source nodes separately before applying network coding and channel coding for the broadcast phase. In general, the performance of this scheme is suboptimal since the signals from the two source nodes interfere with each other. However, if joint soft decoding is employed, the performance can still be improved significantly such as in [53] and [54]. In [57], the authors proposed to use binary ECC codes with higher order PAM modulation for decode-and-forward PNC scheme. In [58], the authors proposed binary LDPC codes for the two-way relay under decode-and-forward strategy with an interference cancellation joint decoder. In [61], the authors presented a quaternary decoding method for BPSK-modulated binary LDPC coded two-way relay systems to decode the codewords sent from the two source nodes simultaneously at the relay. Simulation results show that the quaternary decoding method has better BER performance than direct binary decoding of the modulo addition of the two codewords at the relay.

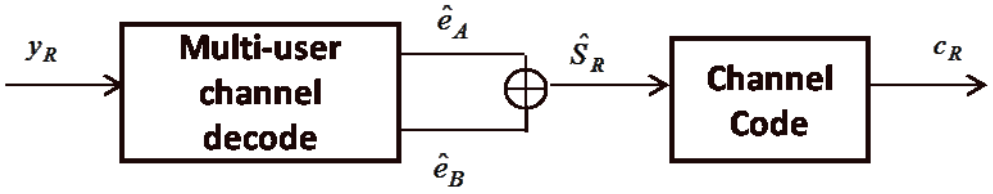


Figure 1-4: Schematic of the decoding process for channel coded decode-and-forward scheme at the relay

JCCNC-2: This is classified as the channel coded arithmetic-sum-and-forward (AS) scheme where probabilistic information on the arithmetic addition of the channel coded codewords is obtained first, and then used to obtain the modulo sum of the transmitted messages via integrated channel decoding and PNC mapping. This scheme was investigated

in [7][53][54][59][60][63]. So far, JCCNC-2 schemes have only been proposed with BPSK, 3-PSK, QPSK, and 5-PSK modulation schemes. In early work presented in [60], the authors proposed a channel coded AS scheme of using parity-check code for the two-way relay over binary symmetric channel. Another AS strategy was proposed in [7] where the authors adopted a repeat accumulate (RA) channel code with BPSK modulation for two-way relay, and redesigned the belief propagation decoding algorithm of the RA code for the purpose of decoding at the relay.

In [59], the authors presented a partial decoding method at the relay, called pairwise check decoding (PCD), for two-way relay fading channels by using non-binary LDPC code. The proposed PCD method formed a so called check-relationship table for the superimposed LDPC-coded packet pair during the multiple access phase. Meanwhile, it incorporated adaptive network coding by using closest-neighbour clustering mapping to compensate the phase deviation of the fading channels. This method is essentially a realization of the AS scheme with channel coding and non-linear network coding.

In [63], the authors presented non-binary constellations for channel coded two-way relay systems over fading channel. In this paper, the two source nodes choose their symbols from a finite field GF(q). After channel coding, the symbols are then q-PSK modulated (q = 2, 3, 4, 5) and sent to the relay. A linear combination of the codewords can be decoded directly at the relay node. Simulation results show that finite fields GF(3), GF(4), and GF(5) outperform the binary case over quasi-static Rayleigh fading channel when concatenated RS and convolutional code are used at the source nodes. When LDPC codes are used, finite fields GF(3) and GF(4) outperform the binary case over Rayleigh fading channel.

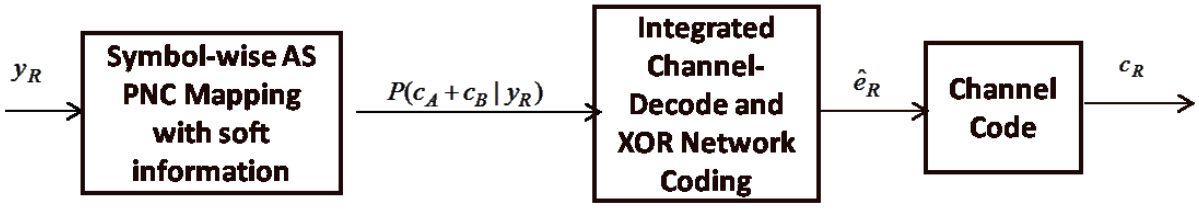


Figure 1-5: Schematic of decoding process of channel coded Arithmetic-sum-and-forward scheme at the relay

JCCNC-3: This is the channel coded XOR network coding scheme, which we will refer to as the 'standard' JCCNC scheme in this thesis. When the messages are drawn from a non-binary prime finite field, JCCNC-3 is also known as the modulo sum network coding scheme.

In this scheme, the modulo addition of the two transmitted messages is decoded using soft information on the linear combination of channel coded codewords. Channel coding is reapplied for the broadcast phase. This scheme is presented in [7][53][61]. It is known that decoding the modulo addition of codewords loses useful information compared with decoding of the arithmetic addition of codewords [7][53], therefore in most scenarios the performance of JCCNC-3 scheme will generally be worse than JCCNC-2 scheme. In [53]-[56], the authors presented sum-product algorithms for binary LDPC coded two-way relay systems with BPSK modulation. These iterative decoding methods were used at the relay node to decode individual codeword (decode-and-forward), to decode the modulo addition of the codewords (compute-and-forward), and to decode the arithmetic addition of the codewords (denoise-and-forward) respectively. The authors also extended the proposed method from BPSK modulation to QPSK modulation two-way relay systems in [54].

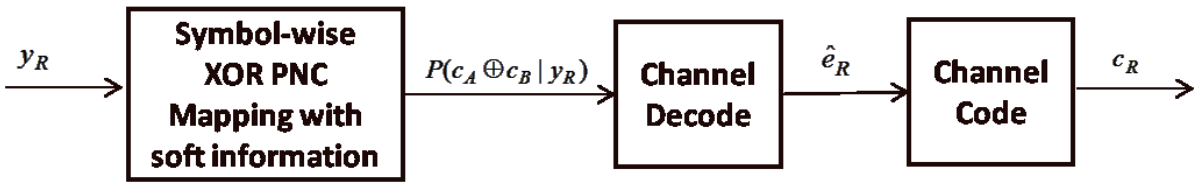


Figure 1-6: Schematic of the decoding process of standard channel coded XOR-network coding scheme at the relay

So far existing JCCNC-3 (XOR-JCCNC) schemes have only been applied to BPSK and QPSK modulation systems. For higher modulation schemes, excessive complexity is incurred in calculating the soft information for the network coded symbol, making the scheme infeasible for higher order modulations. Furthermore the existing JCCNC-2 schemes have only been applied to BPSK, 3-PSK, QPSK, and 5-PSK modulation systems. The reason is that decoding algorithms proposed for JCCNC-2 require the tanner graph used for sum-product algorithm decoding to be reconstructed for error correction code over different finite fields, corresponding to different modulation orders. Different sum-product algorithms must be designed based on the superimposed encoder for different sized finite fields. This introduces unnecessary additional complexity into the design and implementation of this scheme since a real communication system may adaptively select the modulation order based on the channel conditions.

We address the two problems outlined above in Chapter 4 where we present four novel JCCNC schemes that employ lattice decoding strategy at the relay. The proposed JCCNC schemes can be efficiently applied to general types of PAM and QAM modulation systems, with little added complexity for higher order modulations. The key condition for the proposed JCCNC schemes is that the channel code and the lattice code are compatible, a condition which ensures a one to one mapping between the lattice-decoded codeword and the network coded symbol. The four proposed schemes are titled lattice-decoded JCCNC (LD-JCCNC) with soft decoding scheme, LD-JCCNC with hard decoding, LDA-aided JCCNC (LDA-JCCNC) scheme and polar lattice-aided JCCNC (PLA-JCCNC) scheme. The proposed JCCNC schemes use fixed sum-product decoding algorithms for error correction codes over different finite field sizes, and therefore do not carry the overhead of requiring different decoding algorithms to be written for code over a different finite field size, corresponding to a modulation scheme of different order.

Additionally the proposed schemes offer the advantage of low decoding complexity compared to the standard XOR-JCCNC, by mitigating the excessive calculations of soft information on network coded symbols used to initialize the messages for the sum-product algorithm. The reduction in decoding complexity is particularly significant when the code is extended to larger finite fields.

In our proposed LD-JCCNC scheme, LDPC codes in $\mathrm{GF}(q)$ where q is prime, can be used for $q-$PAM and q^2-QAM modulation systems; LDPC codes in $\mathrm{GF}(q^2)$ can also be used for q^2-QAM modulation systems; while binary polar codes can be used for 2^m-PAM and 2^m-QAM modulation systems, where m is an even number. All our proposed JCCNC schemes can be applied to 2^m-PAM and 2^m-QAM constellations which are subsets of $q-$PAM and q^2-QAM constellations. Other soft decoding linear ECC codes can also be used if available.

1.3.2 Channel coded Multi-way Relay Systems

So far, the vast majority of channel coded PNC schemes have been proposed for two-way relay systems and to our knowledge no channel coded PNC scheme has been specifically proposed for general multi-way relay networks. In Chapter 5, we adapt the JCCNC schemes studied in Chapter 4, from the bidirectional relay channel to the multi-way relay channel. We further provide the methodology to apply them to general types of higher order PAM and QAM modulation systems. For the multi-way relay network where the relay is required to

decode an integer linear combination from more than two transmitting users, our proposed JCCNC schemes from Chapter 4 can be efficiently adapted to the multi-relay, without incurring significant additional complexity in the decoding process. Numerical results are used to evaluate their performances.

1.4 Lattice codes

The underlying code behind compute-and-forward, and the channel coded PNC schemes proposed in this work is lattice codes which contain a number of useful algebraic and statistical properties making it a powerful tool in wireless communication [64]. Understanding the fundamental principles of lattice codes is therefore a prerequisite for this work.

Lattice codes are the Euclidean space equivalent of linear codes and extend the linearity property from a finite field into the real field. A lattice code is usually constructed by taking a finite set of points of an infinite lattice contained within a bounded shaping region, with the set of lattice points satisfying the power constraint of the channel. It is known that the performance of a lattice code in the AWGN channel depends on the coding lattice as well as the shaping region.

It has been recognized in recent years that the coding and shaping of lattice codes are largely separable problems. Coding for the AWGN channel is tackled in depth in the works [65] through the concept of AWGN-coding good lattices, while the lattice shaping problem is solved by using quantization good lattices which can be obtained using Voronoi shaping [69] as well as lattice Gaussian shaping [66]. A coding good lattice affords good protection against Gaussian noise, while shaping good lattices are suitable for guaranteeing power constraints of the channel. Therefore lattices with good coding and shaping properties are considered ideal for reliable PNC schemes.

In [65] it is shown that Shannon's information capacity can be achieved for the power constraint AWGN channel using nested lattice codes by employing a coding-good fine lattice and a quantization good shaping lattices. In [67] capacity-achieving lattices constructed from non-binary LDPC codes were proposed. In [68] the authors proposed a new class of polar lattices from polar codes which is shown to achieve capacity for the power constraint Gaussian channel. Unlike all previous work in the area, the authors also provided an explicit low complexity construction for their capacity-achieving lattice code.

In recent years, lattice codes have found use in large number of applications in multi-terminal communications, where they are shown to achieve gains previously unobtainable using standard random codes. In particular lattices are used in wireless network coding under the compute-and-forward setting which is the focus of this work. Additionally lattices are also widely used in interference alignment, distributed source coding as well as information-theoretic security.

From a practical point of view lattices carry a range of advantages due to their simple structure and their range of useful statistical and algebraic properties. For example, lattices can be used to design higher dimensional constellations with high SNR efficiency, the performance of which can be defined by the coding gain and shaping gain. Multidimensional codes are based on the same lattice theory framework behind coset codes and therefore are naturally suited to coded modulation, where they can be used to send a fractional numbers of bits per two dimensions and support opportunistic secondary channels. The advantages of multidimensional constellations are well documented in [69].

Another key advantage of lattice codes is their low structural complexity. Many well known lattices have well defined constructions from binary linear code [64] and there exist efficient lattice decoding algorithms for several key types of lattices [70]. When used for network coding and computations over multiple access channels in general, lattice codes allow for fast efficient decoding of the computed linear functions at the relay nodes, achieving large gains previous unobtainable using standard random codes.

1.4.1 Lattice Properties

First we provide a summary of key lattice properties which are important to understand the topic:

- An n-dimensional lattice Λ is a set of point in R^n where if $s,t \in \Lambda$ then $s+t \in \Lambda$, and $-s \in \Lambda$.
- Lattice Λ can be expressed in term of its generator matrix $\mathrm{G} \in \mathrm{R}^{n \times n}$ so that
$$\Lambda = \left\{ \lambda = \mathrm{G}\, x : x \in \mathrm{Z}^n \right\}.$$
- If Λ is a lattice in R^n, around each lattice point x is its Voronoi region $\mathcal{V}(x)$. Any point in the region is at least as close to x as to any other lattice point:
$$\mathcal{V}(x) = \{ z \in \Re^n : \|x - z\| \le \|y - z\|, \text{ for all } y \in \Lambda \}.$$

The fundamental Voronoi region of a lattice is defined as the Voronoi region around the origin lattice point.

- A lattice decoder is essentially a lattice quantizer which maps the received vector x to the nearest lattice point λ in Euclidean distance:

$$Q_\Lambda(x) = \arg\min_{\lambda\in\Lambda} \|x-\lambda\|.$$

- A set of points in Euclidean space which quantises to a given lattice point is defined as the Voronoi region of that point.
- The modulo operation with respect to Λ for a nested lattice code gives the quantization error:

$$x \bmod \Lambda = x - Q_\Lambda(x)$$

- A lattice code C is the intersection of a lattice with a bounded shaping region S: $C = \Lambda \cap S$
- The sublattice Λ_s of a n-dimensional lattice Λ can be considered be a scaled version $\mathrm{M}\Lambda$ of Λ where M is a scaling factor. In other words the sublattice Λ_s is a subset of the points of the lattice Λ so that the sublattice is simply a larger scaled version of the lattice with the two lattices being lattices of the same type.
- A nested lattice code is a lattice code whose shaping region is the Voronoi region of a sublattice. The lattice used for coding is known as the fine lattice. The sublattice used for shaping is known as the coarse lattice.
- Let $\mathcal{V}$ and $\mathcal{V}_s$ denote the fundamental Voronoi regions of fine lattice Λ and its sublattice Λ_s respectively and the volumes of the two Voronoi regions by V and Vs respectively. The nested lattice code set can be defined as $\{C = \Lambda \bmod \Lambda_s\} = \{\Lambda \cap \mathcal{V}_s\}$
- For each $\mathrm{c} \in C$, the shifted lattice $\Lambda_\mathrm{c} = \mathrm{c} + \Lambda_s$ is called a coset of Λ_s relative to Λ.
- A nested lattice code can be constructed from a lattice partition of the fine lattice Λ by the sublattice Λ_s denoted by Λ/Λ_s. The order of partition defined by $|\Lambda/\Lambda_s|$ gives the size of the quotient group or the number of code points of Λ. These code points are the coset leaders (minimum norm points) of each coset of sublattice Λ_s. If we defined the sublattice to be the fine lattice scaled by factor M such that $\Lambda_s = \mathrm{M}\Lambda$, then the nested code contains $|C| = \mathrm{M}^n$ codewords where n is the lattice dimension.

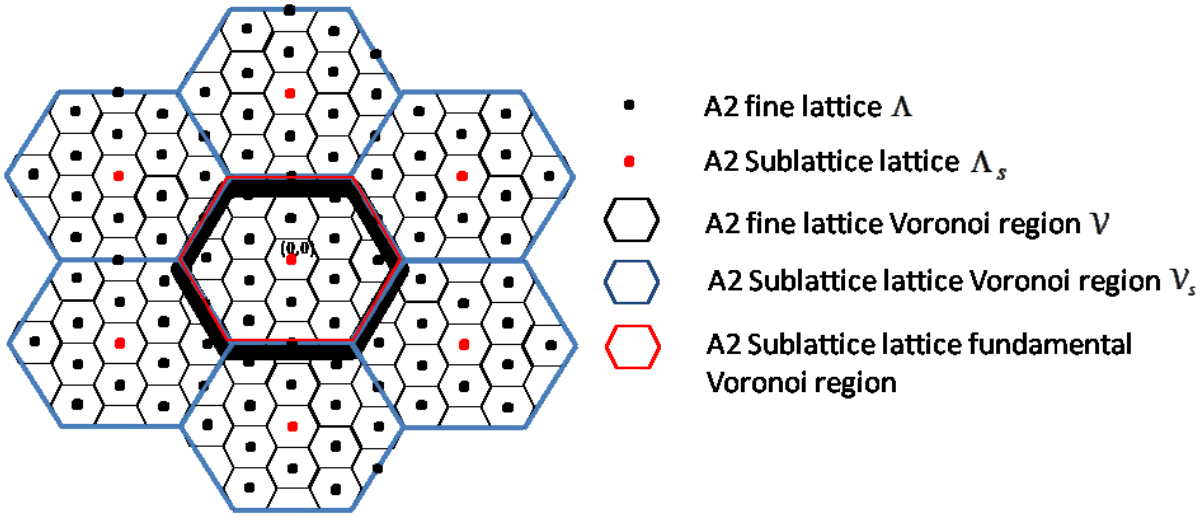


Figure 1-7: Example of a nested lattice code with the A2 hexagonal lattice

- The two principal geometrical parameters of Λ are the minimum squared distance $d^2_{\min}(\Lambda)$ between its points, and the fundamental volume (or volume of its Voronoi region) denoted by $V(\Lambda)$. These determine the fundamental coding gain of the lattice.
- The fundamental coding gain for an n-dimensional lattice is a measure of how much more effectively points are packed in the n-dimensional lattice than the packing in the integer Z^n lattice. It is the gain obtained by using a lattice Λ with a higher density than Z^n. It is defined as:

$$\gamma_c = \frac{d^2_{\min}(\Lambda)}{V(\Lambda)^{2/n}}.$$

- The boundary for the lattice code contributes shaping gain $\gamma_s(R)$ which is the gain in average energy obtained by using a *n*-dimensional constellation bounded by a more circular shaping region *R* instead of a constellation bounded by n-dimensional cube of the same volume. The shaping gain is the ratio between the normalized second moment of any *n*-dimensional cube (which is 1/12) and the normalized second moment of *R* expressed by $G(R)$,

$$\gamma_s(R) = \frac{1/12}{G(R)}.$$

- The maximum shape gain achieved by a *n*-dimensional region *R* is that of a sphere for which $\gamma_s(R) \to 1.53\text{db}$ as $n \to \infty$.
- The normalized second moment $\mathrm{G}(R)$ of a shaping region *R* captures the shaping gain of the region and indicates how good the lattice is as a vector quantizer:

$$G(R) = \frac{\frac{1}{n}\int_R x \cdot x dx}{V(R)^{1+\frac{2}{n}}}, \text{ where } x \in \Re^n.$$

1.4.2 Some Important Lattices

To use lattices and lattice codes in real applications, deep knowledge of various types of lattices is required. In this section, we identify some important lattices documented in the famous book on lattices authored by Conway and Sloane [64] as well as in the work of Forney [71], which give in-depth description of various lattices, lattice codes, and important lattice decoding algorithms. Below is a summary:

- Some important lattices have been identified, which include the A_2 lattice, D_4 lattice, E_8 lattice, the Barnes-Wall lattice Λ_{16}, and the Leech Lattice Λ_{24}. These are the best lattices in their respective dimensions in terms of densest packing, highest kissing number, thinnest covering and best quantizer.
- Construction A lattices: Let *C* be a binary linear code of length *n* defined in the GF(2), a lattice Λ can be built via construction A using the said code,

$$\Lambda = \mathrm{C} + 2Z^n = \bigcup_{x \in \mathrm{C}} (x + 2Z^n).$$

  Examples of lattices built from construction A include:

  - D_4 lattice: $\mathrm{D}_4 = (4,3) + 2Z^4$, where (4,3) is a binary parity check code. In other words, the lattice $2Z^4$ is shifted by 8 codewords of the (4,3) parity check code to form 8 cosets of the $2Z^4$ lattice whose union form the D_4 lattice.
  - E_8 lattice: $\mathrm{E}_8 = (8,4) + 2Z^8$, where (8,4) the is the extended (7,4) Hamming code.
  - Low-density lattice (LDA): lattice constructed from construction A and nonbinary LDPC code. Section 4.3.1 will give a more detailed description.

- Construction D lattices: Lattices which are constructed by using multi-layer nested binary linear codes. For example, the $\mathrm{N}=2^{n+1}$ dimension Barnes Walls lattices $\Lambda(0,n)$ can be constructed from Reed Muller codes. The construction formula is given by:

  For $n-r$ even: $\Lambda(r,n)=2^{(\mathrm{n}-\mathrm{r})/2}Z^{2\mathrm{N}}+\sum_{r+1\le r'\le n,\text{ given } n-r' \text{ odd}} 2^{(r'-r-1)/2}\mathrm{RM}(r',n+1)$

  For $n-r$ odd:

  $$\Lambda(r,n)=2^{(n-r)/2}Z^{2\mathrm{N}}+\sum_{r+1\le r'\le n,\text{ given } n-r' \text{ even}} 2^{(r'-r-1)/2}\mathrm{RM}(r',n+1)$$

  where RM(r,n) represents r-th order Reed Muller Code of length 2^n. For example, the 16-dimension Barnes Walls lattice $\Lambda(0,3)$ can be constructed using the formula:

  $$\Lambda_{16}=\Lambda(0,3)=4Z^{16}+2\,\mathrm{RM}(3,4)+\mathrm{RM}(1,4)=4Z^{16}+2(16,15,2)+(16,5,8)$$

- Polar lattice: This is a construction D lattice which is created by using a set of nested binary polar codes. Polar lattice has been shown to achieve better performance than Barnes Wall lattices [101][102]. Section 4.4 will give a more detailed of description.

DIMENSION	1	2	3	4	5	6	7	8	12	16	24
DENSEST PACKING	Z	A_2	A_3	D_4	D_5	E_6	E_7	E_8	K_{12}	Λ_{16}	Λ_{24}
HIGHEST KISSING NUMBER	Z 2	A_2 6	A_3 12	D_4 24	D_5 40	E_6 72	E_7 126	E_8 240	P_{12a} 840	Λ_{16} 4320	Λ_{24} 196560
THINNEST COVERING	Z	A_2	A_3^*	A_4^*	A_5^*	A_6^*	A_7^*	A_8^*	A_{12}^*	A_{16}^*	Λ_{24}
BEST QUANTIZER	Z	A_2	A_3^*	D_4	D_5^*	E_6^*	E_7^*	E_8	K_{12}	Λ_{16}	Λ_{24}

Figure 1-8: Table of the best lattices for various dimensions, source: [64], Chapter 4.

1.4.3 Voronoi Constellation

Voronoi constellations are a methodology to construct good lattice codes originally proposed by Forney as well as Conway and Sloane in [64] [69] [71] [75]. The structural properties of the Voronoi code provide two distinct and largely separable forms of gains: coding gain due to the fine lattice points which are used as codewords, and the shaping gain from the shaping region of the lattice code or the boundary which defines the constellation group. A nested

lattice code is a form of Voronoi code for which the shaping region is the fundamental Voronoi region of the coarse lattice.

Voronoi codes can be designed to have desirable characteristics. For example Voronoi constellations can be designed to have constituent 2D constellations with points from a desired grid that has small peak-to-average ratio. The ability to design good Voronoi codes with good symmetric properties and their scalability makes them well suited to coded modulation such as Trellis modulation and QAM modems.

Two general approaches are put forward in [71] on how to select points for Voronoi constellations or nested lattices, in particular in the case where there are more than one minimum norm points for each coset of the sublattice and only a limited number of constellation points are required. Maximally biased method takes all the minimum norm points of a coset and search for the point with the maximum coordinate in each dimension. The second method uses ties (i.e. the multiple minimum norm points for a single coset) as part of the constellation. Using this method maintains the symmetric properties of the constellation which aids more accurate carrier phase offset recovery at the receiver. Generic lattice decoding algorithms proposed in [74] can be used to decode the lattice codes efficiently.

A decoder for a Voronoi code created from a partition of fine lattice by sublattice Λ_1 / Λ, first quantises the received vector in N dimensional real space onto the nearest lattice point in the fine lattice Λ_1. Then, modulo operation with regards to the sublattice Λ is performed on the quantized pointed in order to map the data back to a point in the fundamental Voronoi region of the sublattice, so we obtain the received signal as a lattice codeword in the original finite lattice codework used during transmission. An example of a Voronoi code decoder is given in Figure 1-9 using a nested lattice code built from the A2 hexagonal lattice. The diagram illustrates the two stage operation of decoding to the nearest fine lattice point followed by modulo operation with respect to the sublattice, to map the decoded lattice point back to the original lattice codebook used for transmission.

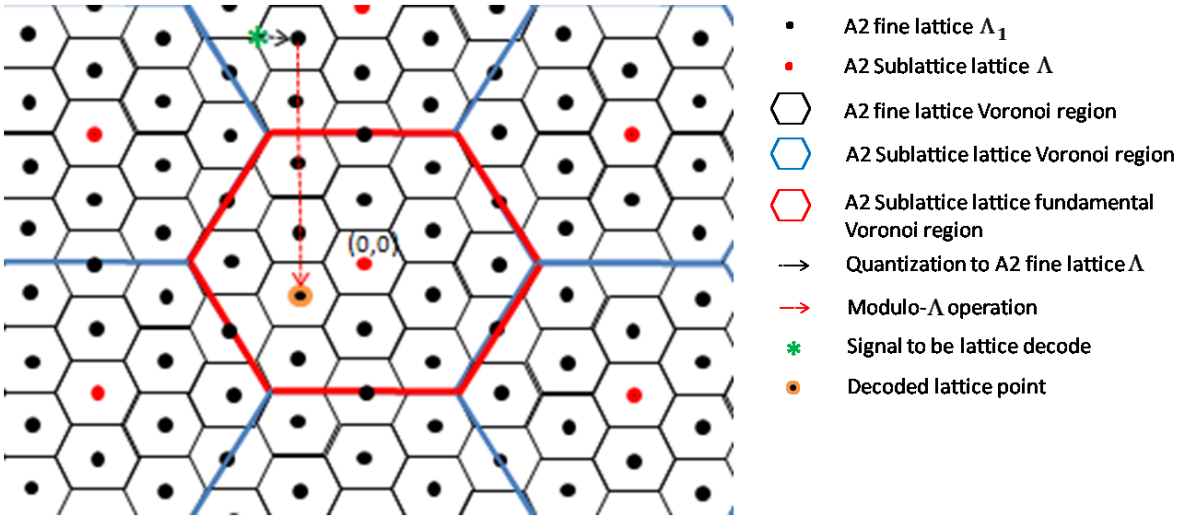


Figure 1-9: Illustration of a Voronoi code decoder for a nested lattice created from the A2 hexagonal lattice.

1.5 Thesis Organization

The thesis is organised as following: In Chapter 2, we propose efficient search algorithms for compute-and-forward with single receive antenna, and MIMO compute-and-forward, to find the optimal integer coefficients over the ring of Gaussian integers and Eisenstein integers. In Chapter 3, the achievable rates are derived for two-way relay PNC schemes over Rayleigh fading channel. A precoding scheme is then proposed for these two-way relay PNC schemes to improve their performance. In Chapter 4, using LDPC codes and polar codes, we propose three new joint channel coded network coding schemes for PAM and QAM two-way relay systems. We call these schemes lattice-decoded JCCNC (LD-JCCNC), low density construction A JCCNC (LDA-JCCNC), and polar lattice aided JCCNC (PLA-JCCNC) scheme. In Chapter 5, we adapt our three proposed JCCNC schemes to the multi-way relay network for the compute-and-forward scheme, and show how they can be used with general types of higher PAM and QAM modulation systems. Chapter 6 presents conclusions and future work.

2 Parameter Selection for Compute-and-Forward

In this chapter, we present our work in [34] on integer coefficient search for compute-and-forward (C-F). Selecting the optimal equation coefficient vector to maximise the overall computation rate of C-F is an important decoding step in the implementation of C-F scheme. Choosing the optimal integer coefficients for C-F has been shown to be a shortest vector problem (SVP). SVP is known to be NP hard in its general form. However, in this chapter, we propose algorithms to solve the special case of SVP in C-F in polynomial time.

The most basic approach to finding the optimal integer coefficients in C-F is by performing exhaustive search [23] over all integer vectors within a given range. Although exhaustive search guarantees to find the optimal integer coefficients, this approach is practically non-viable due to a search complexity that grows exponentially with number of transmitting users. In [23], the authors showed that finding the optimal integer coefficients in C-F is a Shortest Vector Problem (SVP), which is known to be NP hard in its general form. Therefore, lattice reduction based approximation algorithms have been proposed in [23] and [30] for the standard C-F scheme with single receive antenna, and [27] for the Integer-Forcing MIMO C-F receiver in [49]. These algorithms use the basis of the reduced lattice to find a short vector within an exponential factor of the shortest length vector. Due to the sub-optimality of the approach, using approximation algorithms such as the LLL [28] to find the optimal integer coefficients will result in significant computational rate loss in C-F that increases with the number of users.

A heuristic approach of finding integer coefficients called quantized exhaustive search (QES) was proposed in [31] for the phase-precoded C-F introduced in the same work. With this method the relay iteratively adjusts the phase and magnitude of the C-F scaling factor to make the scaled channel coefficients as close as possible to the optimal integer coefficients. However the authors do not explicitly give methods on how to select the size of the phase and magnitude step to guarantee the optimal solution hence this method is considered suboptimal. The complexity of this algorithm depends on the search step size for both phase and magnitude of the scaling factor, as well the absolute value of the maximum range for the scaling factor.

In a more recent work, the authors in [32] proposed an optimal integer coefficients search method for the case of real valued channels which shows that the SVP problem resulting from C-F can be solved exactly through an algorithm of low polynomial complexity in the number of transmitting users. Since we have to deal with complex valued gains for wireless channels in most real application scenarios, optimal coefficient search algorithms must be developed for complex valued channel cases.

The main contribution of the work in this chapter is the following: building on the work of [32], we extend their integer coefficient search algorithm for real valued channels to C-F over complex valued channels. We propose two algorithms with low polynomial complexity in relation to the number of transmitting users, to find the exact solution for the complex valued version of the SVP in C-F. Algorithm 1 finds the optimal coefficient vector for C-F with single receive antenna over the ring of Gaussian integers and the ring of Eisenstein integers. Algorithm 2 finds the optimal integer coefficients for MIMO C-F over the ring of Gaussian integers and the ring of Eisenstein integers. The crux of both algorithms is Theorem 1 which specifies an exact relationship for the optimal solution, relating the optimal coefficients to the underlying channel coefficients. Additionally, through Theorem 2, we derive the solution to the shortest vector problem of C-F based on the exact components of the underlying gram matrix. In comparison to the original algorithm given for C-F over real valued channel [32], we make necessary modifications to streamline our proposed algorithms in order to make it efficient for computations over the complex valued channels. The complexity of our proposed algorithms is examined in detail and simulation analysis is provided to compare the performance and efficiency of Algorithm 1 with other efficient integer coefficient search algorithms.

Throughout this chapter, we denote scalars by plain font and vectors by bold lower case font. All vectors in this thesis are row vectors. Bold capital letters represent matrices. For a matrix **D,** the notation D_{mn} refers to the element in row m and column n. When denoting indexed vectors, the term $\mathbf{a}_m$ indicates the m-th vector, while a_{mn} denotes the n-th element of the m-th vector. For an $L \times N$ matrix **D,** the notation $\mathbf{D}_\tau$ denotes a submatrix of **D** consisting of rows indexed in the set $\tau \subseteq \{1, \dots, L\}$. The vector d_τ denotes a subset of vector **d** in same manner**.**

2.1 Overview of Existing Coefficient Search Methods

In this section, we review in detail the main integer coefficient search methods for C-F proposed in literature and discuss their performance and complexity.

1). Exhaustive search of integer coefficients [23]: The optimal integer coefficients in C-F can be found through exhaustive search by checking all the non-zero integer vectors of length L in a sphere of radius given by the bound on the norm of integer coefficients in (1.7) for standard C-F and (1.17) for MIMO C-F. The complexity of exhaustive search is given by approximately $O(\Phi^{2L})$ where $\Phi = \sqrt{1 + P\|\mathbf{h}^2\|}$. Since the complexity increases exponentially with the number of transmitting users, this method has the highest computational complexity, although it guarantees the optimal solution.

2). Lattice reduction based algorithms focused around the LLL algorithm [23], [28], [30] for standard C-F: For the SVP in Integer-Forcing MIMO C-F, proposed lattice reduction based algorithms include Complex-LLL(CLLL) [27], [29], Hermite-Korkine-Zolotareff(HKZ) [27] and Minkowski reduction algorithms [27]. The underlying principle of the LLL algorithm, which has a complexity of $O(L^4 \log L)$, is to compute a reduced basis from the original basis of the lattice, where the first vector of the LLL-reduced basis approximates the shortest vector of the lattice. For L users the length of the LLL-approximated shortest vector is at most $2^{\frac{L-1}{2}}$ times the length of the shortest vector in the lattice. Therefore as L increases, the rate loss that occurs from using LLL for C-F becomes significant enough such that an optimal yet computationally efficient search method is required. The performance of using complex-LLL to approximate the optimal Gaussian integer coefficients for C-F is examined in Figure 2-1 for L=4. If we take shortest vector obtained using complex-LLL as the integer coefficients for C-F, then complex-LLL fails to find the exact optimal integer coefficients over 36% of the time. The resultant rate loss incurred due to the use of suboptimal integer coefficients is at least 5%. However for L=10, complex-LLL misses the optimal coefficient around 70% of the time, and incurs a C-F rate loss of at least 40% between 10dB and 20dB of SNR. Therefore, for a larger number of users, the rate loss using Complex-LLL to solve the SVP of C-F becomes significant enough such that an optimal, yet computationally efficient search method is required.

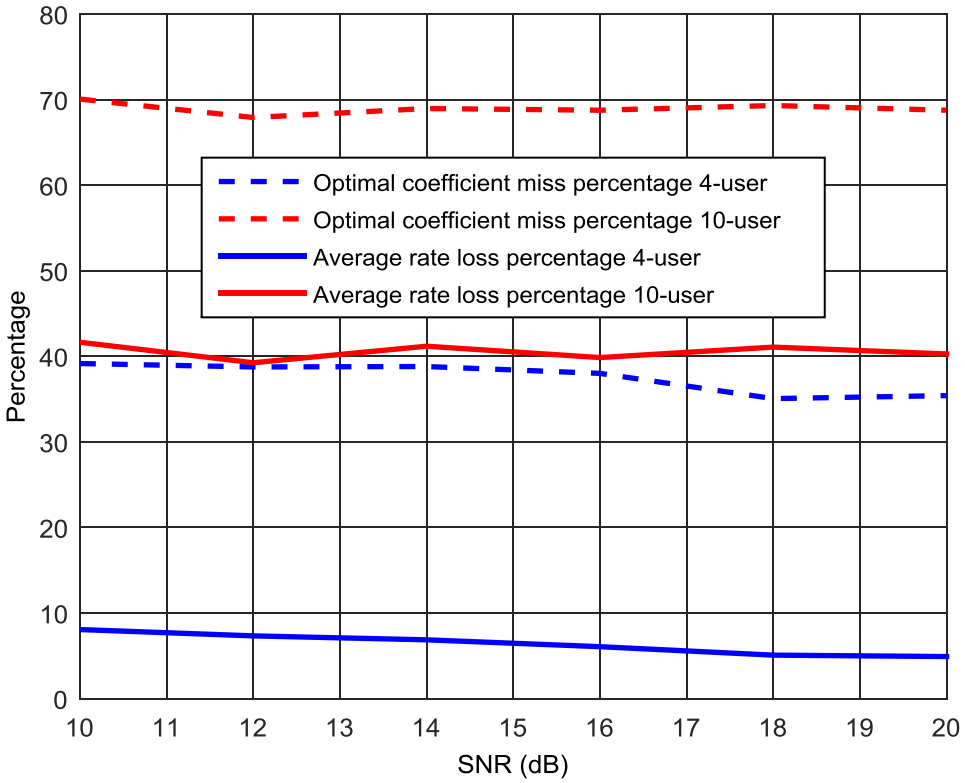


Figure 2-1: Compute-and-forward rate loss incurred using integer coefficients obtained via complex-LLL approximation, for L=4 and L=10

For SVP over MIMO C-F, the proposed HKZ and Minkowski reduction algorithms in [27] with complexity of $(2\pi e)^{L+O(\log 2L)}$ and $(5/4)^{2L^2}$ respectively, offer improved performances over CLLL at the cost of higher complexity for higher order MIMO channels.

3). Quantized exhaustive search [24] is a streamlined version of the standard exhaustive search method which is originally proposed for the phase precoded C-F, but is also relevantly applicable to standard C-F scheme. The basis of the method uses the inferred relationship between optimal integer coefficients vectors and channel coefficients,

$$\mathbf{a} = [\alpha \mathbf{h}]_{Z[j]} \tag{2-1}$$

whereby $\alpha \in C$. This relationship was not proven in [31] but made based on heuristic analysis of the C-F model. In quantized exhaustive search, the complex scaling factor α is broken down into its phase and magnitude components. The algorithm change the magnitude component from 0 to the maximum magnitude $|\alpha|_{\max}$ in steps of $|\alpha|_{\text{step}}$, while simultaneously changing the phase component of α from 0 to 90 in steps of d which has to be chosen as part of the search parameter. The α value taken at each iteration is used to

make an initial estimate of integer coefficient vector using (2-1), which is then used to calculate the corresponding MMSE coefficient. The estimated **a** along with its corresponding MMSE coefficient is then used to evaluate the effective noise term of C-F in (1.4) and the **a** that minimises the effective noise is taken as the optimal integer coefficients. The complexity of this algorithm is shown to be $O\left(90/d \times |\alpha|_{\max}/|\alpha|_{\text{step}}\right)$. Whether this algorithm can yield the optimal integer coefficients depends entirely on selecting d and $|\alpha|_{\text{step}}$ to be sufficiently small for a specific set of channel instances. Since the authors do not give an explicit method on how to select the exact search parameters to guarantee the optimal coefficients, this search method can be considered suboptimal.

4). An optimal integer coefficient search algorithm of polynomial complexity was proposed in [32] for C-F over real valued channels. The authors proved that the solution to minimization problem of (1-14) for integer coefficients, satisfies

$$\mathbf{a}^{\text{opt}} = [\alpha \mathbf{h}]_Z, \tag{2-2}$$

for $\alpha \in \mathrm{R}$, otherwise $\mathbf{a}^{\text{opt}}$ must be a standard unit norm vector. This is the fundamental theorem on which the optimal polynomial complexity algorithm in [32] is based around. Using (2-2), the authors showed that finding the optimal integer coefficients becomes a one-dimensional optimization problem where the cost function $f(\mathbf{a}(\alpha))$ to be minimised in (1-14) is a piece-wise constant function in $\mathbf{a}(\alpha)$ and therefore a piecewise constant function in α. The authors find a set of points of α that represent all intervals in $f(\mathbf{a}(\alpha))$ which is constant and checks the value of $f(\mathbf{a}(\alpha))$ over all points. This is done by first finding all the discontinuity points of $a_l(\alpha) = [\alpha h_l]$ for $l = 1, \ldots L$ which occur at points where αh_l is a half integer. Then, using the discontinuity points of $a_l(\alpha)$ the authors proceed to find all the α values for which $a_l(\alpha)$ is discontinuous, which are also the α values for which the function $f(\mathbf{a}(\alpha))$ is discontinuous. The α discontinuities points are then sorted in ascending order and the middle value between consecutive discontinuities are used to evaluate $f(\mathbf{a}(\alpha))$. The α value that minimises the target function therefore generates the optimal integer coefficients based on relationship (2-2). The complexity of this algorithm is given by $O\left(L^2\sqrt{1 + P\|\mathbf{h}\|^2}\right)$.

2.2 Proposed Optimal algorithm for Complex valued channel

Since the channel coefficients to be quantized as part of C-F decoding lies in C it is natural that we should first consider the case where the integer coefficients decoded at the relay belong to the ring of Gaussian integers. For a C-F system of L transmitting users sending to relays with a single receive antenna over complex channels $\mathbf{h} \in \mathrm{C}^L$, we first propose an algorithm that finds the optimal coefficient vector $\mathbf{a} \in \mathrm{Z}[j]^L$ for C-F over Gaussian integers. Additionally, since it is known that $\mathrm{Z}[\omega]$ quantizes C best among all imaginary quadratic integers, we also adapt our algorithm to find the optimal coefficients $\mathbf{a} \in \mathrm{Z}[j]^L$ for C-F over Eisenstein integers. We then propose a search algorithm for MIMO C-F of L transmitting users and k receive antennas at the relay, that finds the optimal coefficient vector in both $\mathbf{a} \in \mathrm{Z}[j]^L$ and $\mathbf{a} \in \mathrm{Z}[\omega]^L$. The basis of both algorithms is Theorem 1 which shows that SVP problem in its complex form in C-F has an exact solution.

{**Theorem 1**} For C-F system over complex ring R, where R denotes either the ring of Gaussian integers or the ring of Eisenstein integers, the solution to (1-18) satisfies,

$$\mathbf{a}^{\mathrm{opt}} = [\mathbf{bH}]_{\mathrm{R}} = \begin{cases} \mathbf{a}^{\mathrm{opt}} = [\mathbf{bH}]_{\mathrm{Z[j]}} \text{ for } \mathbf{a} \in \mathrm{Z[j]}, \\ \mathbf{a}^{\mathrm{opt}} = [\mathbf{bH}]_{\mathrm{Z}[\omega]} \text{ for } \mathbf{a} \in \mathrm{Z}[\omega], \end{cases} \tag{2-3}$$

for some $\mathbf{b} \in \mathrm{C}^k$, or $\mathbf{a}^{\mathrm{opt}}$ must be a unit norm vector in the ring R. The unit norm vector refers to any vector where every element of $\mathbf{a}^{\mathrm{opt}}$ is zero with the exception of one element, with the nonzero element taking the value of either $1, -1, j$ or $-j$ for the ring of Gaussian integers, and $1, -1, \pm\omega$ or $\pm\omega^2$ for the ring of Eisenstein integers.

Proof: We first give proof for the case $\mathbf{a} \in \mathrm{Z}[j]^L$. Consider the optimization problem of finding $\mathbf{a} \in \mathrm{Z}[j]^L \setminus \mathbf{a} = 0$ that maximizes the computational rate of MIMO C-F in (1-15). This is clearly equivalent to the minimization problem on the target cost function $f(\mathbf{a})$ in (1-18). Let $(\mathbf{b}^{\mathrm{opt}}, \mathbf{a}^{\mathrm{opt}})$ be the optimal solution and let $\mathbf{h}_n^T$ be the $n-\mathrm{th}$ column vector of $\mathbf{H} \in \mathrm{C}^{k \times L}$. The two possible cases are, for $n = 0,1,....,L$,

- $\forall n$: $[\mathbf{b}^{\mathrm{opt}} \mathbf{h}_n^T]_{\mathrm{Z}[j]} = 0$. In this case, $\mathbf{a}^{\mathrm{opt}}$ is a Gaussian integer vector of unit norm.
- $\exists n$: $[\mathbf{b}^{\mathrm{opt}} \mathbf{h}_n^T]_{\mathrm{Z}[j]} \neq 0$. In which case, $\mathbf{a}^{\mathrm{opt}} = [\mathbf{b}^{\mathrm{opt}} \mathbf{H}]_{\mathrm{Z}[j]}$.

Therefore the optimal solution takes on the form of $\mathbf{a}^{\text{opt}} = [\mathbf{bH}]_{Z[j]}$ or it will be a Gaussian integer vector of unit norm. The proof for Theorem 1 with Eisenstein integers can be deduced in a similar manner.

Additionally we can derive the solution to the shortest vector problem of compute-and-forward based on the exact components of the underlying gram matrix $\mathbf{M}$, provided that it is DP^k decomposable as first defined in [32]. A positive definite matrix is said to be DP^k if it can be written as $\mathbf{M} = \mathbf{D} - \mathbf{P}$ where $\mathbf{D}$ is a diagonal matrix and $\mathbf{P}$ is a positive semi-definite matrix of rank k. $\mathbf{P}$ can be written as $\mathbf{P} = \mathbf{V}^H\mathbf{V}$ where $\mathbf{V}$ is a $k \times L$ matrix. Based on the given definitions, we propose Theorem 2. The proof is provided in the Appendix.

{**Theorem 2**} For C-F over complex ring R, where R denotes either the ring of Gaussian integers or the ring of Eisenstein integers, the solution to $f(\mathbf{a}) = \mathbf{aMa}^H$ for a DP^k decomposable matrix $\mathbf{M} = \mathbf{D} - \mathbf{P} = (1 + P\|\mathbf{h}\|^2)\mathbf{I} - P\mathbf{h}^{\mathbf{T}}\mathbf{h}$ is given by

$$\mathbf{a}^{\text{opt}} = \left[\mathbf{xVD}^{-1}\right]_{\mathrm{R}} = \begin{cases} \mathbf{a}^{\text{opt}} = [\mathbf{xVD}^{-1}]_{Z[j]} & \text{for } \mathbf{a} \in Z[j], \\ \mathbf{a}^{\text{opt}} = [\mathbf{xVD}^{-1}]_{Z[\omega]} & \text{for } \mathbf{a} \in Z[\omega], \end{cases} \tag{2-4}$$

for some $\mathbf{x} \in \mathrm{C}^k$, or $\mathbf{a}^{\text{opt}}$ must be a unit norm vector in the ring R.

2.2.1 Proposed algorithm: C-F over Gaussian Integers

We first present our algorithm for C-F over Gaussian integers for the complex channel, when the relay has one receive antenna. Note that for single receive antenna case of $k = 1$, it can be deduced from Theorem 1 that the optimal coefficient vector will either take on the form of $\mathbf{a}^{\text{opt}} = [\alpha\mathbf{H}]_{Z[j]}$ for some $\alpha \in C$, or it will be a Gaussian integer vector of unit norm. Based on the relationship between optimal coefficient vector and α, we can likewise express the target minimization function in (1-14) as a function of α.

The first step in our search method is finding the set of values in α representing the discontinuity points of the continuous function $f(\mathbf{a}(\alpha))$ based on the discontinuity points of neighbouring Gaussian integer coefficients $\mathbf{a}(\alpha)$. This is possible since the values of α for which $\mathbf{a}(\alpha) \in Z[j]^L$ is discontinuous are also values for which $f(\mathbf{a}(\alpha))$ is discontinuous. Using the points of discontinuity for $f(\mathbf{a}(\alpha))$, we can then find a point to represent every

constant interval of $f(\mathbf{a}(\alpha))$, therefore allowing us to evaluate the specific value of α and the corresponding $\mathbf{a}(\alpha)$ that minimises $f(\mathbf{a}(\alpha))$. Note that $f(\mathbf{a}(\alpha))$ is a two dimensional piecewise linear function due to the real and complex components Gaussian integer variable.

From the minimization cost function in (1-14), it can be observed that the discontinuities in $f(\mathbf{a}(\alpha))$ occur when the Gaussian integer coefficients $\mathbf{a}(\alpha) \in \mathrm{Z}[j]^L$ change in value, or equivalently when its individual elements a_l change in value, for $l = 1,..L.$ Since we know from Theorem 1 that $a_l(\alpha) = [\alpha h_l]$, the discontinuities of $a_l(\alpha)$ occur when the real part of the imaginary part, or both components of αh_l are half integers. From the rounding relationship denoted in Theorem 1, we can see that the significance of the discontinuities between neighbouring Gaussian integers is that they represent the exact cut-off points between rounding to one Gaussian integer or to the other, and therefore they also represent the discontinuities of $f(\mathbf{a}(\alpha))$. The discontinuities of Gaussian integer vector $\mathbf{a}(\alpha)$ that we need to consider are therefore the intermediate points between neighbouring Gaussian integer points in the real and imaginary domain. Denoting discontinuities of $\mathbf{a}(\alpha)$ by φ, there are complex valued points of the form:

$$\varphi \in \Psi : \begin{cases} c + dj \ \text{ where } c \in \mathrm{Z},\ d - \dfrac{1}{2} \in \mathrm{Z}, \\ e + fj \ \text{ where } e - \dfrac{1}{2} \in \mathrm{Z},\ f \in \mathrm{Z}, \\ o + pj \ \text{ where } o - \dfrac{1}{2} \in \mathrm{Z},\ p - \dfrac{1}{2} \in \mathrm{Z}. \end{cases} \tag{2-5}$$

Given the constraint on the optimal coefficient vector in (1-7) we know that it is only necessary to check discontinuities of $\mathbf{a}(\alpha)$ within a sphere of radius slightly larger than the bound in (1-7), namely:

$$|\varphi| \le \lceil \Phi \rceil + \frac{1}{2}. \tag{2-6}$$

Using the relationship between $\mathbf{a}(\alpha)$ and α inferred in Theorem 1, along with the discontinuities of $\mathbf{a}(\alpha)$ that satisfy the bound in (2-6), we can deduce that the values of α for which $\mathbf{a}(\alpha)$ and $f(\mathbf{a}(\alpha))$ is discontinuous, are given by,

$$\Omega = \left\{ \frac{\varphi}{h_l} \,|\, l = 1,...,L,\ h_l \neq 0,\ |\varphi| \le \lceil \Phi \rceil + \frac{1}{2} \right\}. \tag{2-7}$$

The discontinuities of $f(\mathbf{a}(\alpha))$ given by set Ω therefore form the vertices of planes where $f(\mathbf{a}(\alpha))$ is locally constant. This can be illustrated by the three dimensional plot of the target function $f(\mathbf{a}(\alpha))$ against real and imaginary components of α, shown in Figure 2-2 where each locally constant part of the function is composed of polytopes. To evaluate the minimization problem completely, we must find a value of α to represent every constant part of the function.

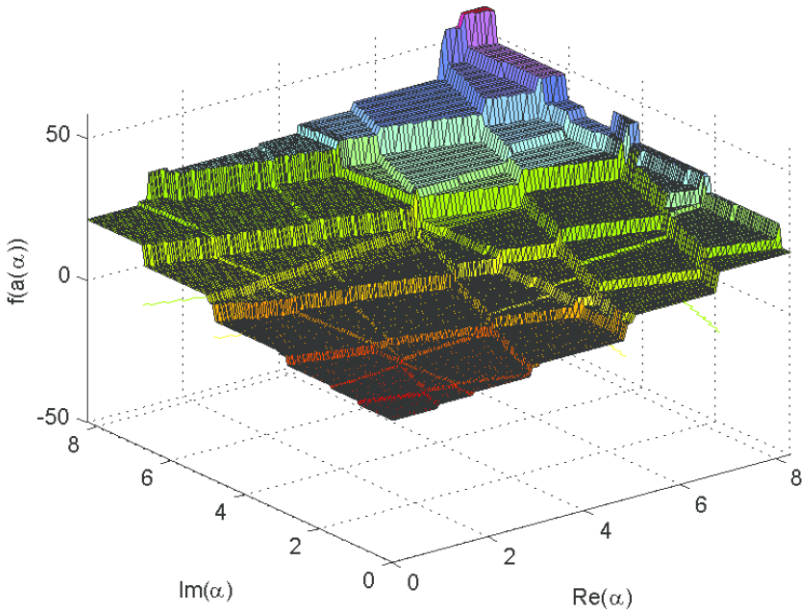


Figure 2-2: 3D plot of the optimization function in terms of α

Note that for the one dimensional piecewise constant function that occurs in the minimization problem for C-F over real valued channels, the authors in [32] proposed that the obtained discontinuities should be sorted and the average value of consecutive discontinues used to evaluate $f(\mathbf{a}(\alpha))$. If we were to translate this approach to the complex channel, then we need to find a point of α to represent each constant polytopes plane of $f(\mathbf{a}(\alpha))$ and evaluate $f(\mathbf{a}(\alpha))$ for that point. This optimization problem over multiple dimensions was tackled for the MIMO C-F model in [32] whereby when the number of receive antennas at the relay is more than one, the cost function to be minimised can also be viewed as a multi-dimensional piecewise linear function that can be solved by ensuring every constant polytopes region of $f(\mathbf{a}(\alpha))$ is checked. The authors in [32] proposed to find the average of every possible pair wise combinations of all the discontinuities in α which will guarantee that each constant region will have at least one representative. The problem with obtaining

average values in this manner is that it makes this part of the algorithm inefficient, resulting in an excessively high complexity depending on the total number discontinuities, which is determined by the bound in (2-6) and the number of transmitting users.

However, in our proposed algorithm we make an efficient simplification of the approach in [32] to significantly reduce computational complexity without sacrificing performance. In fact, it is sufficient to use the values of α for which $f(\mathbf{a}(\alpha))$ is discontinuous to directly evaluate the cost function in (1-14). For the real channel case, this is true because the piece-wise constant function will be open at one end and closed at the other, by the virtue of the rounding operation. When we evaluate $f(\mathbf{a}(\alpha))$ using its α discontinuities directly, there will be a singular elements of $\alpha\mathbf{h}$ that are exactly half integers. This means the value of $f(\mathbf{a}(\alpha))$ for that α will be calculated using the $\mathbf{a}(\alpha)$ obtained by rounding up $\alpha\mathbf{h}$ at its half integer components. Therefore the discontinuity points can be used to obtain the value of $f(\mathbf{a}(\alpha))$ for the particular interval which the discontinuity proceeds in the piecewise step function. Taking into account that $f(\mathbf{a}(\alpha))$ is zero when α is zero, no interval is unaccounted for.

Therefore for the complex channel case, we can use the discontinuities of $f(\mathbf{a}(\alpha))$ denoted by set Ω to directly evaluate $f(\mathbf{a}(\alpha))$. Note that the value of $\mathbf{a}(\alpha)$ calculated using Theorem 1 for $\alpha \in \Omega$ will assume the rounded up value of $\alpha\mathbf{h}$, where half integer values in the real and imaginary planes are both rounded upwards. In this way, every discontinuity in α represents a plane where $f(\mathbf{a}(\alpha))$ is constant. The minimization problem can be written as

$$\mathbf{a}^{\text{opt}} = \underset{\mathbf{a}\in Z[j]^L \setminus \mathbf{a}\neq 0}{\arg\min}\ f(\mathbf{a}) = \underset{\alpha\in\Omega}{\arg\min}\, f(\mathbf{a}(\alpha)) \tag{2-8}$$

From an implementation perspective, there are a number of ways to optimise the search algorithm. One way to do this is to efficiently generate Ψ in (2-5) and consequently Ω in (2-7). For small values of L, Ω can be easily generated by iterating over two loops, each loop running from $-(\lceil\Phi\rceil+0.5)$ to $\lceil\Phi\rceil+0.5$ incrementing in steps of 0.5, which represents loops over the real and imaginary part of the discontinuities. Even though an excess of points is generated for Ψ, this method is a computationally effective way of implementing this part of the algorithm for small number of users. The reason is that since every element of Ψ is used L times to generate elements of Ω, then for small L the excess of points generated for Ω is also small. For larger values of L, because using this imprecise method will lead to a larger amount of excess points of Ω, it might be computationally efficient to generate the

exact values in Ψ without any additional excess points. One to generate the exact points in Ψ is by iterating over the same two loops by adjusting the range of values searched in one loop based on the value selected in the other loop. For example, denote $\Re$ as the value selected by the loop generating real value part of the discontinuity, and denote $\Im$ as the value generate by the loop representing the imaginary values. We iterate the real value loop from $-(\lceil\Phi\rceil + 0.5)$ to $\lceil\Phi\rceil + 0.5$ incrementing in steps of 0.5, and for each value of the real loop $\Re$, we iterate the imaginary value loop from $-\left\lceil\sqrt{(\lceil\Phi\rceil + 0.5)^2 - \Re^2}\right\rceil$ to $\left\lceil\sqrt{(\lceil\Phi\rceil + 0.5)^2 - \Re^2}\right\rceil$ in steps of 0.5. Even though this leads to additional computational costs to generate Ψ, the overall complexity of generating Ω and subsequent operations of evaluating $\mathbf{a}(\alpha)$ will be lower for larger values of L.

Our proposed algorithm can also be made more efficient by recognising that only one quadrant of Ω needs to be evaluated to completely solve the minimization of $f(\mathbf{a}(\alpha))$. This is because discontinuities represented by Ω has a rotational symmetry of four, and Gaussian integer space has a rotational symmetry order of four with respect to the origin which means any Gaussian integer will become another Gaussian integer of the same modulus after rotation through the four quadrants. Therefore the values of components $\|\mathbf{a}\|$, $|\mathbf{a}\mathbf{h}^T|$ in (1-8) and the overall computational rate of C-F, do not change for any single element of Ω as it is rotated through each of the four quadrants. As a result, when evaluating $f(\mathbf{a}(\alpha))$ for elements of Ω, only discontinuities within a single quadrant of Ω needs to be checked. The proposed method is explicitly outlined in Algorithm 1.

2.2.2 Proposed Algorithm: C-F over Eisenstein Integers

C-F over rings of Eisenstein integers was studied extensively in [24] where the authors proved the existence of sequence of nested lattice pairs over Eisenstein integers whose coarse lattice are quantization-good and fine lattice AWGN coding-good. The authors showed that C-F over Eisenstein integers achieves superior performance in outage rate and better error correction performance than C-F over Gaussian integers, which is largely due to the fact Eisenstein integers quantizes the complex channel coefficients better than Gaussian integer. Naturally we also extend our novel integer coefficient search method to C-F over Eisenstein integers.

Eisenstein integers are complex numbers of the form $a+b\omega$ where $a,b\in Z$ and $\omega = -\frac{1}{2}+j\frac{\sqrt{3}}{2}$. In the case where the relay would like to compute linear combination of codewords with coefficients $\mathbf{a}(\alpha)\in Z[j]^L$ in Eisenstein integers, our proposed method can also be adapted to find the optimal integer coefficients in Eisenstein integers, effectively finding the exact solution for,

$$\mathbf{a}^{\text{opt}} = \underset{\mathbf{a}\in Z[\omega]^L\setminus\mathbf{a}\neq 0}{\arg\min}\ f(\mathbf{a}) = \underset{\mathbf{a}\in Z[\omega]^L\setminus\mathbf{a}\neq 0}{\arg\min}\left(\mathbf{a}\mathbf{M}\mathbf{a}^H\right). \tag{2-9}$$

The proposed optimal coefficient search algorithm for C-F over Eisenstein integers therefore follows the same structure as for Gaussian integers. First we calculate the discontinuities φ between neighbouring Eisenstein integers that satisfy the constraint,

$$|\varphi| \le \lceil \Phi \rceil + \frac{3}{4}. \tag{2-10}$$

The discontinuities between Eisenstein integers are intermediate points between one Eisenstein integer and a neighbouring Eisenstein integer that is closer to it in Euclidean distance than any other Eisenstein integer. The significance of the discontinuities is that they represent the exact cut-off point for lattice decoding to one Eisenstein integer or to a neighbouring Eisenstein integer. Therefore the discontinuities represent the transition points where the value of $f(\mathbf{a}(\alpha))$ changes. The discontinuities α between neighbouring Gaussian integer points that we must consider are,

$$\varphi\in\Psi:\begin{cases} c+dj & \text{where } c\in\frac{1}{4}+\frac{1}{2}\mathrm{Z},\ d\in\frac{\sqrt{3}}{4}+\frac{\sqrt{3}}{2}\mathrm{Z}, \\ e+fj & \text{where } e-\frac{1}{2}\in\mathrm{Z},\ f\in\frac{\sqrt{3}}{2}+\sqrt{3}\mathrm{Z}, \\ o+pj & \text{where } o-\frac{1}{2}\in Z,\ p\in\sqrt{3}Z. \end{cases} \tag{2-11}$$

Note that each of the three types of discontinuities in (2-11) can be generated by iterating over two loops, generating the real and imaginary components of the discontinuities. Based on (2-11), the range of each loop can be selected to be $\left(\lceil\Phi\rceil+\frac{3}{4}\right)$ to $-\left(\lceil\Phi\rceil+\frac{3}{4}\right)$.

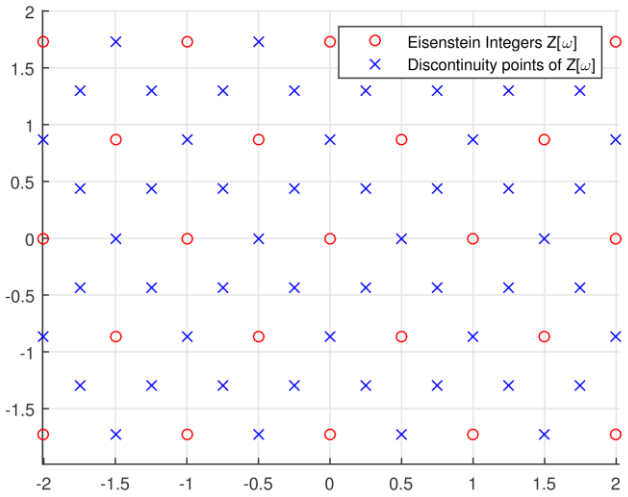


Figure 2-3: Discontinuity representatives of Eisenstein integer points

Using the discontinuities of $\mathbf{a}(\alpha) \in Z[\omega]^L$ denoted by φ and the relationship specified in Theorem 1, we can then obtain the values of α for which $f(\mathbf{a}(\alpha))$ is discontinuous as,

$$\Omega = \left\{ \frac{\varphi}{h_l} \mid l = 1,\ldots,L,\ h_l \neq 0,\ |\varphi| \leq \lceil \Phi \rceil + \frac{3}{4} \right\}. \tag{2-12}$$

Using each discontinuity point in Ω we calculate a corresponding Eisenstein integer coefficient vector using Theorem 1, taking the optimal Eisenstein integer coefficient to be the $\mathbf{a}(\alpha) \in Z[\omega]^L$ that minimises the target cost function in (2-9). A number of efficient A_2 hexagonal lattice decoders have been proposed in [70] and [76], which can be used to perform $[\cdot]_{Z[\omega]}$, the quantization operation over Eisenstein integers. Since the efficiency of the A_2 decoder will have a significant impact on the overall search speed of our algorithm, the most efficient A_2 decoder should be used. In [76], a very efficient A_2 decoder was proposed based on breaking down the A_2 lattice as a union of a rectangular lattice $B = \{u + vj;\ u \in Z,\ v \in \sqrt{3}Z\}$ and its coset $B + \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$ giving the lattice,

$$A_2 = B \cup \left(B + \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \right).$$

Let $x \in \mathrm{R}^2$ be the input to the decoder. Let $Q_{B(x)}$ be the closest point of x in B. To decode x to the nearest point in the A_2, first perform $Q_{B(x)}$ to decode to the closest point in the rectangular lattice B, and perform,

$$Q_B\left(x-\left(\tfrac{1}{2},\tfrac{\sqrt{3}}{2}\right)\right)+\left(\tfrac{1}{2},\tfrac{\sqrt{3}}{2}\right),$$

to decode to the nearest point on its coset $B+\left(\frac{1}{2},\frac{\sqrt{3}}{2}\right)$. For the two vectors decoded, the one with the minimum Euclidean distance to x is the output for the A_2 decoder. In the case of a tie, lattice point with the largest norm is selected. The proposed search method over Eisenstein integers is included as part of the Algorithm 1.

Algorithm 1 Optimal Coefficient Search Algorithm for C-F over Complex Channel

Input: Channel vector $\mathbf{h}$, transmission power P

Output: Optimal coefficient vector $\mathbf{a}^{\text{opt}}$

Initialization

1: $\mathbf{M}=\left(1+P\|\mathbf{h}\|^2\right)\cdot\mathbf{I}-P\cdot\mathbf{h}^H\mathbf{h}$

2: $f(\mathbf{a})=\mathbf{a}\mathbf{M}\mathbf{a}^H$

3: $f_{\min}=\infty$

4: $\Phi=\sqrt{1+P\|\mathbf{h}\|^2}$

5: For $\mathbf{a}\in Z[j]$: Define set Ψ as in (2-5) , where any $\varphi\in\Psi$ satisfies $\|\varphi\|\le\lceil\Phi\rceil+\frac{1}{2}$.

For $\mathbf{a}\in Z[\omega]$: Define set Ψ as in (2-11) where any $\varphi\in\Psi$ satisfies $\|\varphi\|\le\lceil\Phi\rceil+\frac{1}{4}$.

6: Define Gaussian integer decoder $[\cdot]_{Z[\omega]}$, or 2D integer lattice decoder

7: Define Eisenstein integer A_2 decoder function $[\cdot]_{Z[\omega]}$ as in section 2.2.2

Phase 1 – Obtain discontinuities of $f(\mathbf{a}(\alpha))$

8: **for** all $l\in\{1,...,L\}$ and $h_l\neq 0$ **do**

9: **for** all $\varphi\in\Psi$ **do**

10: Calculate $\mathbf{a}=\left[\frac{\varphi}{h_l}\mathbf{h}\right]_{Z[j]}$ for $\mathbf{a}\in Z[j]$ or $\mathbf{a}=\left[\frac{\varphi}{h_l}\mathbf{h}\right]_{Z[\omega]}$ for $\mathbf{a}\in Z[\omega]$

11: Calculate $f(\mathbf{a})=\mathbf{a}\mathbf{M}\mathbf{a}^H$

12: if $f(\mathbf{a})<f_{\min}$ and $\mathbf{a}$ is not an all zero vector **then**

13: Set $\mathbf{a}^{\text{opt}} = \mathbf{a}$

14: Set $f_{\min} = f(\mathbf{a})$

15: **end if**

16: **end for**

17: **end for**

Phase 2 –Check unit norm vectors

18: **for** all $\mathbf{a} \in \left\{ \mathrm{Z}[j]^L \text{ for search in } \mathrm{Z}[j], \text{ or } \mathrm{Z}[\omega]^L \text{ for search in } \mathrm{Z}[\omega] \right\}$ where $\|\mathbf{a}\| = 1$ **do**

19: if $f(\mathbf{a}) < f_{\min}$ **then**

20: Set $\mathbf{a}^{\text{opt}} = \mathbf{a}$

21: **end if**

22: **end for**

23: **return** $\mathbf{a}^{\text{opt}} = \mathbf{a}$

2.2.3 Proposed Algorithm for MIMO C-F over Complex channels

Using Theorem 1, we adapt our proposed search algorithm to MIMO C-F setting over complex channels, where the minimization function in (1-18) can be again represented as a multidimensional function with constant polytope regions that must be checked. However, for k number of receive antennas, each vertex is the intersection of k linearly independent hyperplanes of the form $c_n = \mathbf{b}\mathbf{h}_n^T$ where $c_n \in \Psi$ is the Gaussian integer discontinuity as defined in (2-5). Note that for optimal coefficient vector search over Eisenstein integers, simply set $c_n \in \Psi$ as in (2-11). Therefore to obtain all the vertex points of the cost function $f(\mathbf{a}(\alpha))$, choose any set $\tau \subseteq \{1,...,L\}$ for which $|\tau| = k$, with $\mathbf{H}_\tau$ full ranked, and solve $\mathbf{c} = \mathbf{b}\mathbf{H}_\tau$ for $\mathbf{b}$ where the individual components of vector $\mathbf{c}$ take values from the set of discontinuities φ in (2-5) and (2-11) for search over Gaussian integers and Eisenstein integers respectively.

Based on the range for optimal coefficient vector for MIMO C-F given by (1-17), it is clear that only vertices within a specific range have to be considered,

$$|\mathbf{c}| = \begin{cases} \left(\lceil \Phi \rceil + \frac{1}{2}\right)\mathbf{1} & \text{for search over } \mathbf{a} \in Z[j] \\ \left(\lceil \Phi \rceil + \frac{3}{4}\right)\mathbf{1} & \text{for search over } \mathbf{a} \in Z[\omega]. \end{cases} \tag{2-13}$$

Therefore for C-F over Gaussian integers, the discontinuities of $f(\mathbf{a}(\alpha))$ that must be checked are all points in the set Ω_τ:

$$\begin{aligned} \Omega_\tau &= \left\{ \mathbf{c}(\mathbf{H}_\tau)^{-1} \mid |\mathbf{c}| \le \left(\lceil \Phi \rceil + \frac{1}{2}\right)\mathbf{1},\ \mathbf{c} \in \Psi^k \right\}. \\ &\tau \subseteq \{1,\ldots,L\},\ |\tau| = k,\ \mathbf{H}_\tau \text{ full rank} \\ \Omega &= \bigcup_{\tau \subseteq \{1,\ldots,L\}, |\tau| = k} \Omega_\tau \end{aligned} \tag{2-14}$$

For optimal search over Eisenstein integers, the relevant bound in (2-10) should be used instead to evaluate the relevant discontinuities of interest. Finally using Theorem 1, each vertices obtained in Ω_τ can be used directly to calculate C-F coefficient vector either over Gaussian integers or Eisenstein integers. This way the coefficient vectors can be used to evaluate cost function $f(\mathbf{a}(\alpha))$ without missing any potential minimal regions. This search method over MIMO C -F is outlined explicitly in Algorithm 2.

Algorithm 2 Optimal coefficient search for MIMO C-F over Complex channels

Input: Channel matrix $\mathbf{H}$, transmission power P, Gram matrix $\mathbf{M} = \mathbf{V}\mathbf{D}\mathbf{V}^H$

Output: Optimal coefficient vector $\mathbf{a}^{\text{opt}}$

Initialization

1: $\lambda_{\max}$ is the maximum singular value of $\mathbf{H}$

2: $f_{\min} = \infty$

3: $\Phi = \sqrt{1 + P\lambda_{\max}}$

4:

For $\mathbf{a} \in Z[j]$: Define set Ψ as in (2-5) where any $\varphi \in \Psi$ satisfies $\|\varphi\| \le \lceil \Phi \rceil + \frac{1}{2}$.

For $\mathbf{a} \in Z[j]$: Define set Ψ as in (2-11) where any $\varphi \in \Psi$ satisfies $\|\varphi\| \le \lceil \Phi \rceil + \frac{1}{4}$

5: Define Gaussian integer decoder $[\cdot]_{Z[\omega]}$ or 2D integer lattice decoder

6: Define Eisenstein integer A_2 decoder function $[\cdot]_{Z[\omega]}$ as in section 2.2.2

Phase 1 – Obtain discontinuities of $f(\mathbf{a}(\alpha))$

7: **for** all $\tau \in \{1,...,L\}$, $|\tau| = k$, $(\mathbf{H})_\tau$ full rank **do**

8: **for** all $\mathbf{c} \in \Psi^k$, $|\mathbf{c}| = \begin{cases} \left(\lceil \Phi \rceil + \frac{1}{2}\right)\mathbf{1} & \text{for search over } \mathbf{a} \in Z[j] \\ \left(\lceil \Phi \rceil + \frac{3}{4}\right)\mathbf{1} & \text{for search over } \mathbf{a} \in Z[\omega] \end{cases}$

9: Calculate $\mathbf{a} = [\mathbf{c}(\mathbf{H}_\tau)^{-1}\mathbf{H}]_{Z[j]}$ for $\mathbf{a} \in Z[j]$ or,

Calculate $\mathbf{a} = [\mathbf{c}(\mathbf{H}_\tau)^{-1}\mathbf{H}]_{Z[\omega]}$ for $\mathbf{a} \in Z[\omega]$

10: **if** $f(\mathbf{a}) = \mathbf{a}\mathbf{M}\mathbf{a}^H < f_{\min}$ and $\mathbf{a}$ is not a zero vector **then**

11: Set $\mathbf{a}^{\text{opt}} = \mathbf{a}$

12: Set $f_{\min} = f(\mathbf{a})$

13: **end if**

14: **end for**

15: **end for**

Phase 2 –Check unit norm vectors

16: **for** all $\mathbf{a} \in \left\{ Z[j]^L \text{ for search in } Z[j], \text{ or } Z[\omega]^L \text{ for search in } Z[\omega] \right\}$ where $\|\mathbf{a}\| = 1$ **do**

17: **if** $f(\mathbf{a}) < f_{\min}$ **then**

18: Set $\mathbf{a}^{\text{opt}} = \mathbf{a}$

19: **end if**

20: **end for**

21: **return** $\mathbf{a}^{\text{opt}} = \mathbf{a}$

2.3 Complexity Analysis

In this section, the complexity of Algorithm 1 and Algorithm 2 are analysed. For Algorithm 1, the complexity of the algorithm over both Gaussian integers and Eisenstein integers are

provided. For Algorithm 2 over MIMO C-F, the complexity of search over Gaussian integers is given while its complexity over Eisenstein integers can be deduced in a similar manner.

To find the optimal coefficients over Gaussian integers, the first part of the algorithm consists of finding all the α values for which $f(\mathbf{a}(\alpha))$ is continuous, the complexity of which depends on the number of discontinuities of $\mathbf{a}(\alpha)$ that lie within a sphere of radius $\lceil\Phi\rceil+0.5$ which is a form of Gauss circle problem. We can suggest a loose upperbound for the number of discontinuities to be calculated using the ratio of area of a sphere of radius $\lceil\Phi\rceil+0.5$, relative to a square with sides of length $2\lceil\Phi\rceil+1$. Within the bound defined in (2-6), in each axis, there are $2\lceil\Phi\rceil$ number of integers and $2\lceil\Phi\rceil+2$ number of half integers. Therefore within a square of sides $2\lceil\Phi\rceil+1$ centred at the origin, there are a total of $12\lceil\Phi\rceil^2+12\lceil\Phi\rceil+4$ discontinuities of $\mathbf{a}(\alpha)$ of the form given in (2-5). The area ratio between a sphere of radius $\lceil\Phi\rceil+0.5$ and a square of sides $2\lceil\Phi\rceil+1$ is given by $\pi/4$. Therefore we can deduce the upperbound on the total number of discontinuities of $\mathbf{a}(\alpha)$ to be searched as,

$$\lceil\pi(3\lceil\Phi\rceil^2+3\lceil\Phi\rceil+1)\rceil \tag{2-15}$$

Taking into account the number of transmitting users L, the total number points of α for which $f(\mathbf{a}(\alpha))$ is discontinuous is given by

$$L\lceil\pi(3\lceil\Phi\rceil^2+3\lceil\Phi\rceil+1)\rceil \tag{2-16}$$

Since we only need to check discontinuities in one of the quadrants of the set Ω to evaluate $f(\mathbf{a}(\alpha))$ completely, the complexity is quartered

$$0.25L\lceil\pi(3\lceil\Phi\rceil^2+3\lceil\Phi\rceil+1)\rceil \tag{2-17}$$

Finally to enumerate the value of $f(\mathbf{a})=\mathbf{aMa}^H$, L number of calculations are performed to obtain the summation resulting from the matrix multiplication. The overall complexity of the optimal coefficient search algorithm for the complex channel case is therefore given by

$$0.25L^2\lceil\pi(3\lceil\Phi\rceil^2+3\lceil\Phi\rceil+1)\rceil \tag{2-18}$$

The complexity of Algorithm 1 for finding the optimal C-F coefficients over Eisenstein integers can characterised by three main parts. The complexity of the first part derives from obtaining all the discontinuities of $f(\mathbf{a}(\alpha))$ defined in (2-11). Following the same method of analysis for Gaussian integers, for L users it can be shown that the number of discontinuities of $f(\mathbf{a}(\alpha))$ is $O(L\Phi^2)$. In the second part of the algorithm, each α discontinuity is used to

obtain a corresponding $\mathbf{a}(\alpha)$ via the A_2 decoder which means the speed of the overall algorithm is also highly dependent on the efficiency of the A_2. Finally the value of $f(\mathbf{a}(\alpha))$ can be calculated in $O(L\Phi)$. The overall complexity of the algorithm is therefore given by $O(L^2\Phi^2)$. In comparison to Algorithm 1 over Gaussian integers, search over Eisenstein integers runs slightly slower in part due to more discontinuities, and also due to the use of the A_2 decoder to generate coefficients.

For complexity of Algorithm 2 for MIMO C-F, there are approximately $(\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k$ discontinuities of $\mathbf{a}(\alpha) \in \mathrm{Z}[j]^L$ satisfying bound (2-6). For each of these discontinuities, $\mathbf{c}(\mathbf{H}_\tau)^{-\mathbf{1}}$ is calculated $P(L,k)$ times to obtain all the discontinuities of $f(\mathbf{a}(\alpha))$ in α,

$$\binom{L}{k}(\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k. \tag{2-19}$$

This expression can be upperbounded by,

$$\frac{L^k}{k!}(\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k = \frac{(\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k}{k!} \tag{2-20}$$

.

Each discontinuity of $f(\mathbf{a}(\alpha))$ is used to calculate $\mathbf{a} = [\mathbf{c}(\mathbf{H}_\tau)^{-\mathbf{1}}\mathbf{H}]_{\mathrm{Z}[j]}$ and evaluate $f(\mathbf{a}(\alpha))$, both operations can be calculated at $O(kL)$. Therefore the complexity of Algorithm 2 can be denoted by,

$$O\left(kL\frac{(L\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k}{k!}\right) = O\left(\frac{(L^{k+1}\lceil\pi(3\lceil\Phi\rceil^2 + 3\lceil\Phi\rceil + 1)\rceil)^k}{(k-1)!}\right). \tag{2-21}$$

In [32], it is shown that the term Φ is upperbounded by a polynomial function of L. Therefore the complexity of Algorithm 2 for MIMO C-F is of polynomial complexity in L. From an implementation perspective, the search speed of Algorithm 2 will be slower than Algorithm 1 for two main seasons. Firstly with Algorithm 2 the complexity of evaluating total number of permutations for discontinuities $\mathbf{c} \in \Psi^k$ that satisfy the bound $|\mathbf{c}| \leq \left(\lceil\Phi\rceil + \frac{1}{2}\right)\mathbf{1}$ is much higher than for Algorithm 1, where only discontinuities $\varphi \in \Psi$ within the bound $|\varphi| \leq \lceil\Phi\rceil + \frac{3}{4}$ needs to be evaluated. Additionally, the matrix inversion operation $(\mathbf{H}_\tau)^{-\mathbf{1}}$ which must be performed for each set of discontinuities also increases runtime significantly for each iteration in comparison to Algorithm 1 where only a division by a complex variable is required to obtain discontinuities of $f(\mathbf{a}(\alpha))$. It can additionally be

observed that the complexity of Algorithm 1 and Algorithm 2 are expressed as a function of the SNR. The reason is that the complexity for both algorithms depends on the number of discontinuities of $f(\mathbf{a}(\alpha))$ to be checked for the minimization process which is dependent on the maximum bound for integer equation coefficients $\mathbf{a}(\alpha)$, taken to be equation (2-6) for standard C-F, and equation (2-13) for MIMO C-F. Furthermore, the maximum bound for integer equations is defined as a function of SNR. Therefore, it can be observed that the complexity of Algorithms 1 and 2 can be expressed as a function of the SNR.

In Table 2-1 the complexities of the two proposed algorithms are listed, along with the best performing existing equation coefficient search algorithms in literature.

Table 2-1 Comparison of complexities of search algorithms for C-F

C-F Search Options	**Complexity**
Exhaustive Search	$O(\Phi^{2L})$, where $\Phi = \sqrt{1 + P\|\mathbf{h}^2\|}$, and L is number of users
QES	$O\left(90/d \times \|\alpha\|_{\max}/\|\alpha\|_{\text{step}}\right)$, d and $\|\alpha\|_{\text{step}}$ are the step sizes in phase and magnitude respectively, $\|\alpha\|_{\max}$ is the maximum magnitude range of the scaling factor searched
Complex-LLL	$O(L^4 \log L)$
Algorithm 1-Standard C-F	$O(L^2\Phi^2)$
Algorithm 2-MIMO C-F	$O\left(\frac{\left(L^{k+1}\lceil\pi(3\lceil\Phi\rceil^2+3\lceil\Phi\rceil+1)\rceil\right)^k}{(k-1)!}\right)$, where k is the number of receiver antennas on the relay

2.4 Numerical Results

We verify the performance of our proposed Algorithm 1 for standard C-F over the ring of Gaussian integers by running simulations to examine its performance and efficiency. All the numerical results are obtained using MATLAB on an Intel Core(TM) i5 CPU @2.2GHz.

In Figure 2-4 we compare the efficiency of Algorithm 1 against the exhaustive search method for C-F over Gaussian integers, with two transmitting users and single receive antenna. The wireless channel between each user and the relay node is assumed to be complex Gaussian channel. 5000 channel instances are generated for each SNR. The search

duration for each channel instance is accumulated and the total CPU runtime is compared between Algorithm 1 and the exhaustive search method. The ratio of CPU runtime for exhaustive search over the Algorithm 1 is given in Figure 2-4. It can be observed that as SNR increases, our proposed search algorithm runs much faster than the exhaustive search method. This is consistent with theory as for two users, the complexity of our proposed Algorithm 1 increases at the rate of $O(\Phi^2)$ as the SNR increases, while the exhaustive search algorithm increases at the rate $O(\Phi^4)$.

In Figure 2-5 and Figure 2-7, the average C-F rates of Algorithm 1, QES and CLLL are compared for $L=4$ and $L=8$ respectively. The computational rate is taken as rate per user which is obtained by averaging the rate for 5000 channel instances. For QES, simulations with three different sets of search step parameters are performed. The step sizes are selected to ensure the achievable rate is at least as good as CLLL, which is taken as the baseline comparison. From Figure 2-5 and Figure 2-7 it can be seen that Algorithm 1 produces the optimal rate while QES simulations approach optimal rate for finer step size selections. Figure 2-6 and Figure 2-8 compares the efficiency of the three search algorithms, Algorithm 1, CLLL and 3 versions of QES with different search steps for $L=4$ and $L=8$ respectively, comparing their CPU runtime against SNR.

In terms of performance against complexity trade-off comparison, we focus on Algorithm 1 and QES which are the two most competitive search algorithms investigated in this work. For $L=4$ we observe QES can achieve within 5% of the optimal rates at a much lower CPU runtime cost than the proposed Algorithm 1, especially for high SNR values. QES (MagnitudeStep = 0.1, PhaseStep = 5°) provides near optimal rate performance to Algorithm 1 at slightly higher computational cost than Algorithm 1 at lower SNR values. At higher SNR, the efficiency of QES becomes more distinctive. For lower SNR values of the QES (MagnitudeStep = 0.1, PhaseStep = 5°) plot, it is worth noting that even when step sizes are set to a minute level that pushes QES runtime to exceed the runtime of Algorithm 1, the QES will still return a noticeable amount of suboptimal coefficient vectors which is reflected in the slight gap from optimal rates. Another observation we can make based on the data in Figure 2-6 is that as we fine tune the QES step size, the runtime increases significantly, yet the improvement in C-F rate diminishes considerably as QES approaches optimality. This can be observed by comparing the rate and CPU time for QES (MagnitudeStep = 0.5, PhaseStep = 5°) and QES (MagnitudeStep = 0.1, PhaseStep = 5°) where the improvement in rate

performance is approximately 5% across all SNR values, yet the CPU time increased by over 100% at 10dB and over 300% at 20dB.

For $L=8$, by comparing the plots for QES (MagnitudeStep = 0.5, PhaseStep = 5°) against plot Algorithm 1, for SNR of 10dB it can be seen that Algorithm 1 gives approximately 30% improvement in at rate at the cost of 5 times the runtime of QES. At higher SNR values, this trade-off becomes much worse, since the complexity of our proposed method increases exponentially with SNR which is consistent with our analysis of its complexity at $O(L^2\Phi^2)$. Nevertheless, even using minute QES step values that result in very long runtime, QES still returned non-optimal coefficient vectors for a large number of channel instances, as it does for the $L=4$ simulations. QES (MagnitudeStep = 0.1, PhaseStep = 5°) provides near optimal rate performance at slightly lower complexity than Algorithm 1 for smaller SNR values. At higher SNR, the better efficiency of QES becomes distinctively larger. Comparing QES (MagnitudeStep = 0.5, PhaseStep = 5°) with QES (MagnitudeStep = 0.5, PhaseStep = 1°) we note that runtime drastically increases with no noticeable change in rate, which against suggests there is a threshold on the extent altering magnitude or angle step can efficiently improve the performance of the QES for specific sets of channel parameters. Using the numerical data available, we have found that for certain SNR values it is not possible for QES to return 100% optimal coefficient vectors, even when its runtime exceeds that of Algorithm 1. Therefore, we conclude that while QES is an efficient and competitive search algorithm which can approach optimality at a much smaller computational cost than Algorithm 1, it cannot guarantee to return the exact optimal coefficients for random channel instances, even when search steps are set to miniscule values.

Additionally, the performance of the QES depends entirely on selecting a good set of search step parameters in advance to allow the equation coefficient search to be performed optimally. This is the main drawback of QES, since search step parameters that guarantee optimal search performance for one set of channel instances, user numbers and SNR range may not do so for a different set. This is because performing an optimal search relies on an evaluation of all the constant polytope regions in the cost function $f(\mathbf{a}(\alpha))$ shown in Figure 2-2. However, the polytope regions increase in density as L increases and the vertices of the polytope regions change as the channel instances change. Furthermore, as SNR increases, the overall range of α that must be evaluated for $f(\mathbf{a}(\alpha))$ to ensure all the constant regions are evaluated naturally increases. To ensure optimal rate is achieved, the search step parameters of QES must be checked and re-selected each time the search conditions change. This

requires the optimal parameters to be obtained in advance by exhaustively testing and simulating the search performances of different step size parameters, which adds additional complexity to the QES method.

Figure 2-9 illustrates the rate performance of each search method for C-F as L increases while Figure 2-10 shows the corresponding CPU runtime. Note that QES plots in Figure 2-9 and Fig. Assumes a fixed magnitude and phase step size which was selected to ensure QES performed at least on par with CLLL. We observe that computational rate per user of C-F naturally decreases as L increases due to increased interference between users. The performance of CLLL and QES both deteriorate with increasing L in a similar manner. This is consistent with theory as the performance of CLLL is known to deteriorate at an exponential factor of as L increase. The performance of QES deteriorates with L for a fixed search step since this parameter should be adjusted to be gradually smaller with increasing L in order to maintain optimality of search. How to adaptively select search step parameters to bring about optimal performance for the QES is still an open problem. However, due to the close relationship between QES and Algorithm 1, it can be conjectured that the QES search steps required to ensure optimal performance depends on the angle and Euclidean distance between the two α discontinuities of $f(\mathbf{a}(\alpha))$ with the minimum Euclidean distance between them.

Figure 2-10 shows the CPU time of the search algorithms as L increases for SNR of 10dB. The runtime of Algorithm 1 increases exponentially with L as predicted by its theoretical complexity of $O(L^2\Phi^2)$. QES shows gradual increase in runtime with increasing L due to the minimization function term (1-14) which must be evaluated at a complexity of $O(L)$ for each $\mathbf{a}(\alpha)$ obtained in the iterative search. Finally, even though CLLL has a theoretical complexity of $O(L^4 \mathrm{Log} L)$, based on actual runtime performance, it is still very efficient because of the low computation complexity of each round of calculation.

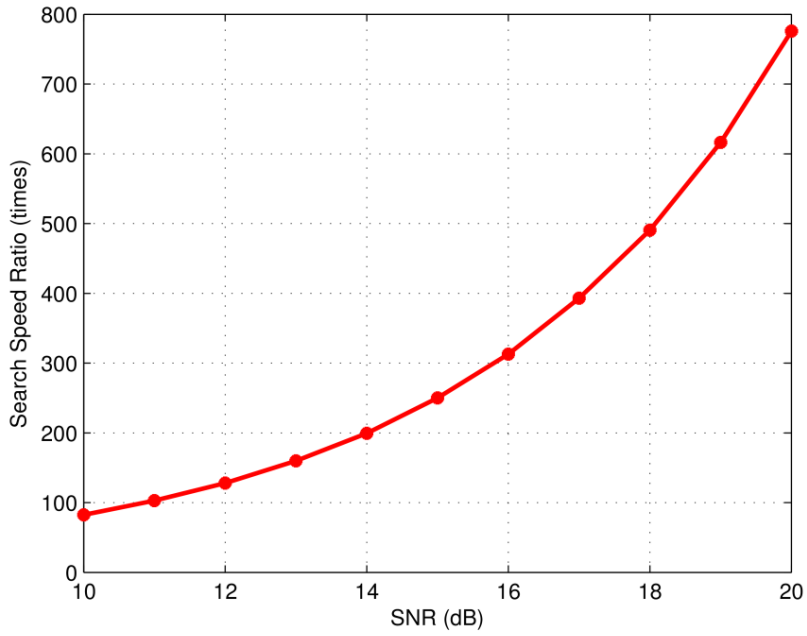


Figure 2-4: Runtime performance ratio (Exhaustive Search CPU/Algorithm1 CPU) against SNR for L=2

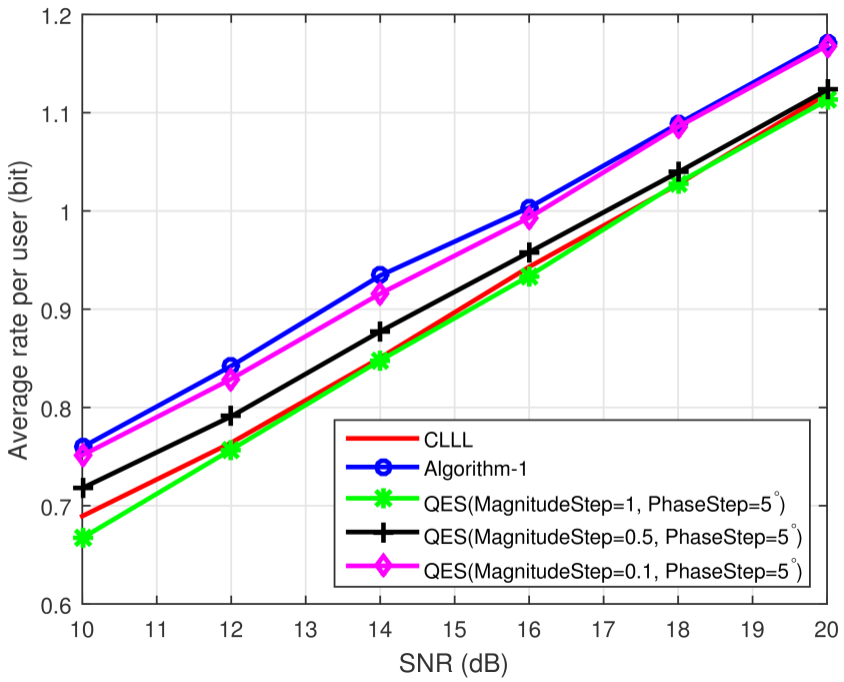


Figure 2-5: Rate performance comparison for $L = 4$

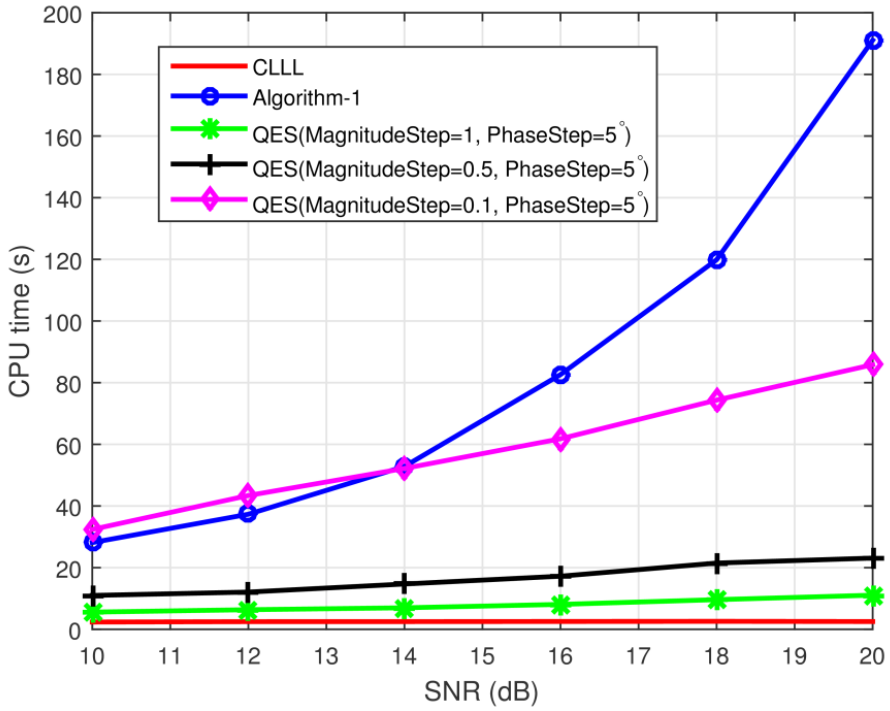


Figure 2-6: CPU time comparison for L=4

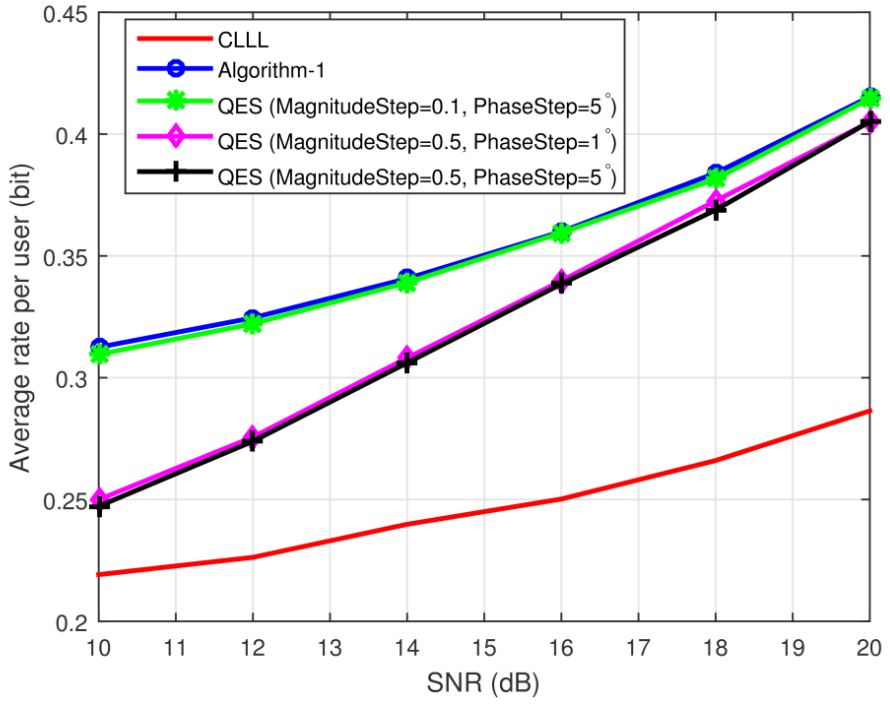


Figure 2-7: Rate performance comparison for L=8

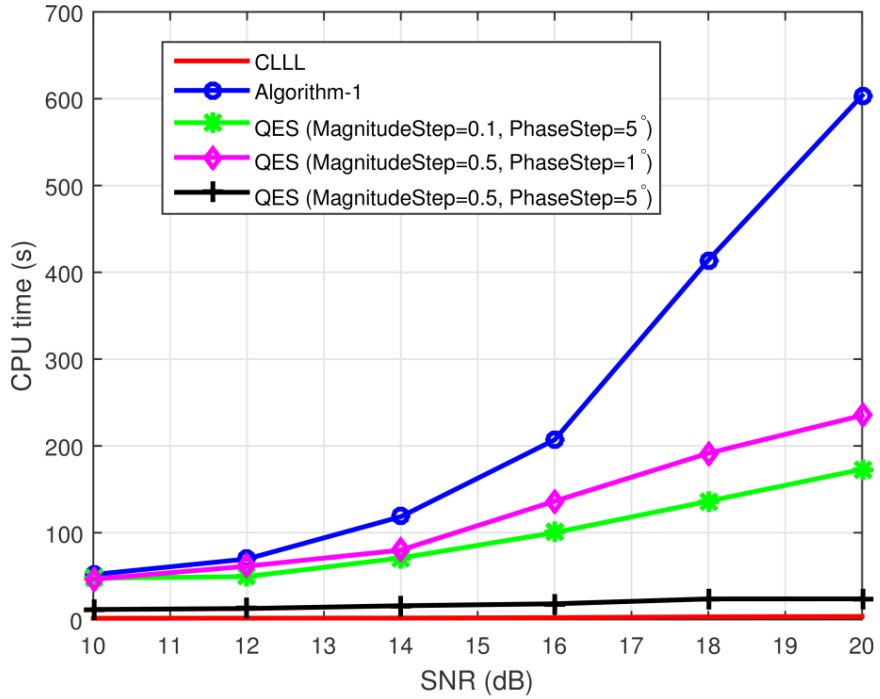


Figure 2-8: CPU time comparison for L=8

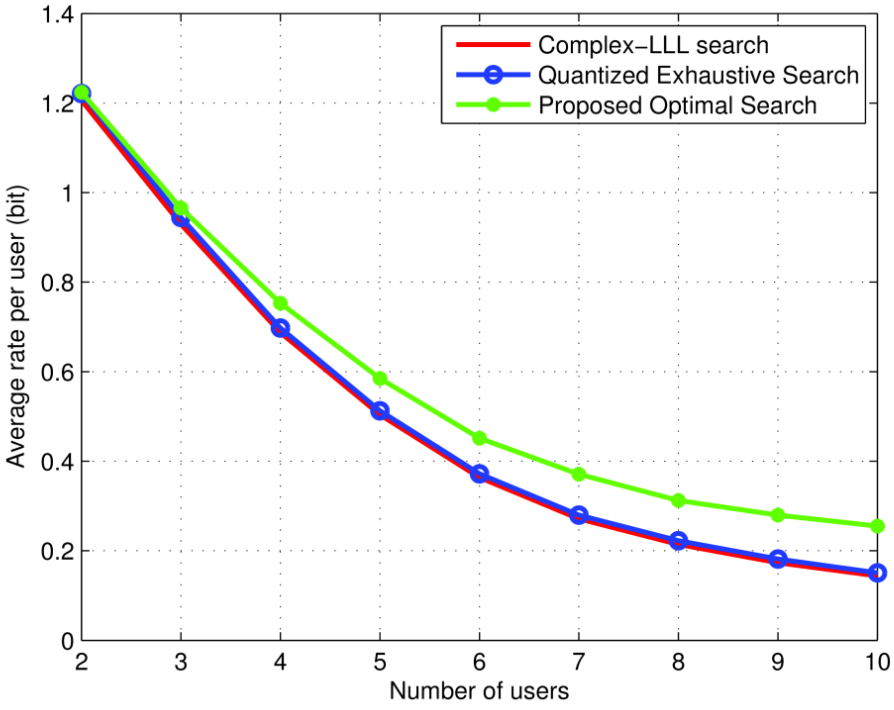


Figure 2-9: Rate performance against transmitting user L, SNR=10dB

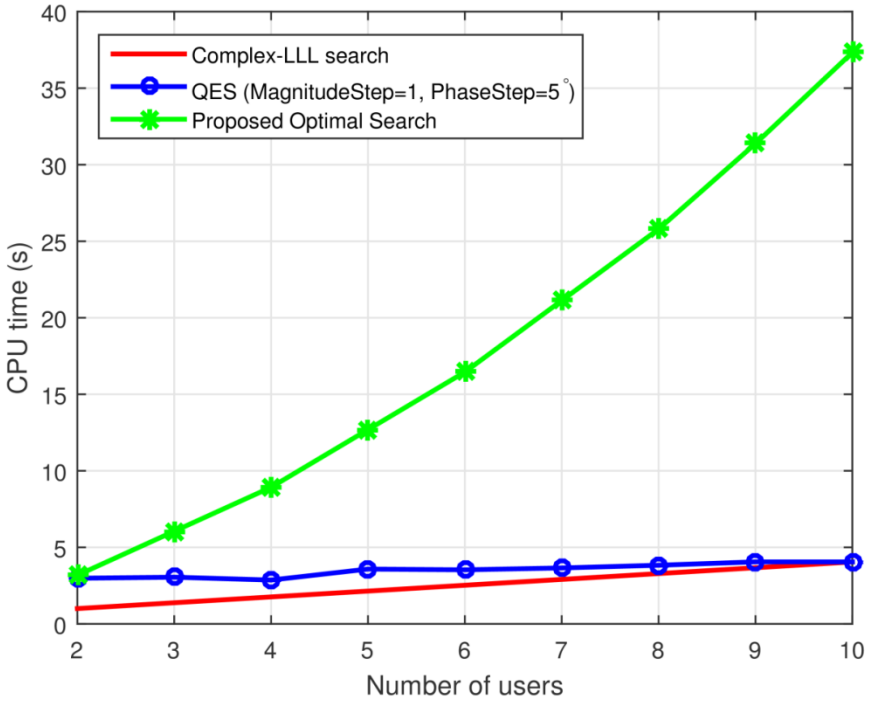


Figure 2-10: CPU time against transmitting user L, SNR=10dB

3 Achievable rate of PNC schemes for the Two-way Relay

In this chapter, the concept of physical layer network coding is revisited for wireless two-way relay channel. The schemes revisited include compute-and-forward, amplify-and-forward, compress-and-forward, denoise-and-forward, and decode-and-forward, with the four relaying strategies forming most of the proposed cooperative communication schemes to date. So far, the achievable rates of these PNC schemes for two-way relay over the AWGN channel are summarized in [1] and [2]. The achievable rates of the same PNC schemes over Rayleigh fading channel are summarized in chapter 2 of [16], and [15].

In this chapter, we first review the achievable rates for selected two-way relay PNC schemes over AWGN and Rayleigh fading channel. In Section 3.4, we optimise the achievable rates of selected PNC schemes, under the condition of equal data rate across all channel states by applying channel inversion precoding schemes to the PNC schemes. We derive the end-to-end achievable rates of the selected PNC schemes over Rayleigh fading channel when channel inversion precoding is performed for the uplink, with additional power usage accounted for in the derived rates using a precoded power adjustment factor. We show that of all the studied PNC schemes, C-F gains the most in performance after channel inversion precoding. Furthermore, since channel inversion precoding is inefficient for poor channel conditions, in Section 3.5 we propose a power efficient channel aware precoding scheme based on C-F which achieves higher theoretical rate as well as attaining superior symbol error rate performance compared to standard C-F and channel inversion precoded C-F.

3.1 PNC Schemes for Two-Way Relay

The two-way relay channel, also known as the bi-directional relay channel, consists of two source nodes communicating with each other in a bi-directional manner via a relay node. The traditional relaying protocols, such as the routing strategy, require four channel uses (e.g., four timeslots in time division multiple access (TDMA)) to exchange the data of the two source nodes whereas a two-way relaying PNC scheme requires less than four channel uses to achieve bi-directional communication between the two source nodes. In a typical two phase PNC scheme, the first phase is referred to as the multiple access (MAC) phase, and the

second phase is referred to as the broadcast phase. The term PNC refers to any network coding technique where the overall communication takes place over two phases, the multiple access phase, and broadcast phase In the MAC phase, both source nodes transmit their messages to the relay node which decodes them. The two-way relay transmit and receive structure can be modelled as below in Figure 3-1.

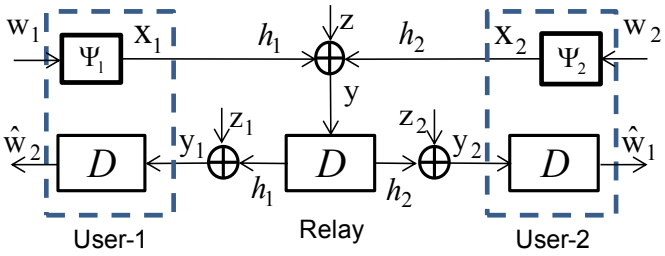


Figure 3-1 Model of two way relay channel

w_1, $w_2 \in \mathbb{F}$ are the source data from node 1 and node 2 respectively. The Source data are mapped to constellation symbols x_1 and x_2 via mapping $\Psi: \mathbb{F} \to \mathbb{C}$ at their respective nodes. The received signal at relay R during the uplink is:

$$Y_R = h_1 x_1 + h_2 x_2 + z \,, \tag{3-1}$$

where h_1, h_2 are complex valued channel coefficients experienced by transmission from node 1 and node 2 to the relay node respectively, z represents complex AWGN noise with variance $\sigma^2 = \mathrm{N}$.

During the broadcast phase, the relay node generates a function of transmitted codeword from the two user nodes. The function depends on the type of network coding scheme being used. In this chapter, the PNC schemes investigated in detail are amplify-and-forward, compress-and-forward, decode-and-forward, and compute-and-forward. The computed function is broadcasted back to the user nodes. The received signal at source nodes can be expressed as:

$$Y_i = h_i x_R + z_i \quad \text{for } i \in \{1,2\} \tag{3-2}$$

where z_i is complex AWGN noise with variance $\sigma^2 = \mathrm{N}$ and h_1 is complex valued channel coefficients from relay node to the user nodes. Each user node decodes the function of transmitted codeword and extracts the other user codeword using self-information. For ease of analysis, we assume our system is time-synchronous. We also assume channel reciprocity for MAC and broadcast phases, i.e. $h_{i\to R} = h_{R\to i}$, $i \in \{1,2\}$.

Decode-and-forward: In decode-and-forward relaying scheme, the relay perform joint

decoding of individual codeword from each user which is then re-encoded for the broadcast phase. For the two way relay, the uplink phase corresponds to a multiple access channel (MAC). During phase the user terminals transmit independent messages to a relay node over a shared channel. It is known that as long as the rates R_{1R} and R_{2R} i.e. rates from users to relay, are in the MAC capacity region, then reliable communication can be made between the two user nodes and the relay. The relay is ultimately interference-limited as the number of transmit sources increases.

Amplify-and-forward: In this scheme, the relay simply acts as a repeater. Amplify-and-forward provides a simple "analogue-to-analogue" interface in which the relay node scales the received signal in the MAC phase and transmits the scaled signal to the source nodes during the broadcast phase. Each source node, knowing their own information, can subtract it before decoding. This scheme can improve network throughput and alleviate bottlenecks in network by allowing a relay node to pass on signal even if it cannot decode. However, noise accumulates with each retransmission as no decoding is performed at the relay nodes.

Compress and forward: In the compress-and-forward scheme, instead of using "analogue-to-analogue" interface, an "analogue-to-digital" interface is used. The relay compresses the received signal and sends the compression index to the receiver. Similar to amplify-and-forward, as no decoding is performed at the relay nodes, noise accumulates as messages traverse the network.

Denoise and forward: In denoise-and forward-scheme, the individual messages are not decoded at the relay. Instead, the arithmetic sum of signals from the source nodes is decoded, and then sent to the source nodes in broadcast phase after power scaling in order to maintain the same transmit power.

Compute-and-forward: This scheme exploits the multiple access channel interference for cooperative gains. Using nested lattice codes as its underlying codebook, the relay decodes linear combination of transmitted messages from the source nodes. Application of compute-and-forward scheme to two-way relay is a special case.

Below we review the achievable rate for each relaying scheme over AWGN channel and derive the achievable rate over fading channel, assuming lattice codes and lattice encoding

and decoding are used. The following assumptions have been made to derive the rate of various PNC schemes:

- The AWG noises at the relay and the two source nodes have the same power, i.e. $N_1 = N_2 = N$.
- The average transmit power of each user and the relay is restricted to be P in the cases without precoding. Symmetric rate is assumed for the two users.
- The channel coefficients between the source nodes and the relay node remains the same during the transmit and receive period. In the case of fading channel, complex-valued Gaussian fading is assumed.
- Cut-set bound [77] is used throughout the paper in deriving the achievable rate for various PNC schemes. Specifically the average rate per user per timeslot is given by
  - $$R_{PNC} = \frac{1}{users} \cdot \frac{1}{timeslots} \cdot (R_{12} + R_{21}) \quad (3\text{-}3)$$
- Where R_{12} and R_{21} are the end to end rates from user to user defined by:
  - $$R_{12}, R_{21} = \min\{R_{up-link}, R_{downlink}\} \quad (3\text{-}4)$$
- $R_{up-link}$ and $R_{downlink}$ are the uplink and downlink achievable rates from user to relay and then relay to the second user respectively. $R_{up-link}$ depends on the PNC scheme in use and $R_{downlink}$ is Shannon's information capacity.

3.2 Achievable rate of PNC Schemes over AWGN Channel

The achievable rate of different relaying schemes for the two way relay AWGN channel can be found in [1][21], which is summarized below. The derivations in the two papers make the assumption that lattice decoding is applied at the relay for all listed PNC schemes:

Upper bound achievable rate: The upper bound achievable rate per- user per timeslot is the point-to-point complex-valued AWGN channel capacity given by:

$$R_{UB} = \frac{1}{2}\log\left(1 + \frac{P}{N}\right) \quad (3\text{-}5)$$

Four timeslots routing strategy: In this relaying strategy, in the first timeslot, User-1 transmits and the relay decodes codeword x_1. In the second timeslot, User-2 transmits and the relay decodes codeword x_2. In the third timeslot, the relay transmits the decoded

codeword x_1 and User-2 decodes. In the fourth timeslot, the relay transmits the decoded codeword x_2 and User-1 decodes. Since each transmitted message experiences two point-to-point decoding processes in two timeslots, the achievable rate per-user of the four timeslots routing strategy becomes:

$$R_{4T_Routing} = \frac{1}{4}\log\left(1+\frac{P}{N}\right) \tag{3-6}$$

Three timeslots decode-and-forward: In this relaying strategy, in the first timeslot, User-1 transmits and the relay decodes codeword x_1. In the second timeslot, User-2 transmits and the relay decodes codeword x_2. Then the relay performs modulo-sum to the two decoded codewords and broadcasts $x_1 \oplus x_2$ to both users in the third timeslot. Both users decode the combined codeword and then obtain the wanted codeword by modulo-subtracting their own transmitted codewords. Since the two transmitted messages experience two point-to-point decoding processes in three timeslots, the capacity per-user of the three timeslots decode-and-forward scheme becomes:

$$R_{3T_DF} = \frac{1}{3}\log\left(1+\frac{P}{N}\right) \tag{3-7}$$

Two timeslots decode-and-forward: In the first timeslot both users transmit simultaneously and the relay decodes individual codeword x_1 and x_2. Then the relay broadcasts the modulo-sum of the decoded codewords $x_1 \oplus x_2$ to both users in the second timeslot, and both users decode the combined codeword and then obtain the wanted codeword by modulo-subtracting their own transmitted codewords. In this PNC scheme, the decoding at the source nodes in the second timeslot has the upper bound capacity, thus the overall achievable rate is determined by the decoding at the relay node in the first timeslot. The relay node aims to decode the codeword transmitted by user 1, while treating the second user's message as interference. After the first codeword is successfully decoded, it is used to decode user 2's message via successive cancellation decoding. Based on the bound on the sum rate of the two transmitted messages given by the Gaussian multiple access channel capacity region, the achievable rate per-user of the decode-and-forward PNC scheme is:

$$R_{DF} = \frac{1}{4}\log\left(1 + \frac{2P}{N}\right) \tag{3-8}$$

Amplify-and-forward: In the first timeslot, both users transmit simultaneously, thus the signal received at the relay becomes $\mathrm{y} = \mathrm{x}_1 + \mathrm{x}_2 + \mathrm{z}$ having a power of $2P + N$. Since each transmitter has a power constraint of P, the received signal has to be scaled down by a factor of $\sqrt{P/(2P+N)}$ before being forwarded to both users. In the second timeslot, the relay simply broadcast the scaled received signal to both users. The signals received by the two users are:

$$\mathrm{y}_1 = \sqrt{\frac{P}{2P+N}}(x_1 + x_2 + \mathrm{z}) + \mathrm{z}_1 \tag{3-9}$$

$$\mathrm{y}_2 = \sqrt{\frac{P}{2P+N}}(x_1 + x_2 + \mathrm{z}) + \mathrm{z}_2 \tag{3-10}$$

Each user subtracts its own transmitted codeword from the received signal before decoding the codeword sent from the other user. The SNR at the receiver of each source node is:

$$SNR = \frac{\frac{P}{2P+N} \cdot P}{\frac{P}{2P+N} \cdot N + N} = \frac{P}{N}\frac{P}{3P+N} \tag{3-11}$$

Therefore, the achievable rate per-user of the amplify-and-forward PNC scheme is:

$$R_{AF} = \log\left(1 + \frac{P}{N}\frac{P}{3P+N}\right) \tag{3-12}$$

Note that compress-and-forward PNC scheme is a digital version of amplify-and-forward scheme therefore they have the same achievable rate.

Denoise-and-forward: The process is similar to that of amplifier-and-forward. The difference is that at the relay, the arithmetic sum $\mathrm{y} = x_1 + x_2$ of the signals from the source nodes is decoded. Since each transmitter has a power constraint of P, the received signal has to be scaled down by a factor of $1/\sqrt{2}$ before being forwarded to both users. The signals received by the two users are:

$$\mathrm{y}_1 = \sqrt{\frac{1}{2}}(x_1 + x_2) + \mathrm{z}_1 \tag{3-13}$$

$$y_2 = \sqrt{\frac{1}{2}}(x_1 + x_2) + z_2 \tag{3-14}$$

Each user subtracts its own transmitted codeword from the received signal before decoding the codeword sent from the other user. The SNR at the receiver of each source node is:

$$SNR = \frac{\frac{1}{2} \cdot P}{N} = \frac{P}{N} \frac{P}{3P + N} \tag{3-15}$$

Therefore, the achievable rate per-user of the denoise-and-forward PNC scheme is:

$$R_{DNF} = \log\left(1 + \frac{P}{2N}\right) \tag{3-16}$$

Compute-and-forward PNC scheme: In the first timeslot both users transmit simultaneously and the relay decodes an integer linear combination of the codewords transmitted from the two users. For AWGN channels, the channel coefficients are integers $h_1 = h_2 = 1$, therefore the need to search for optimal integer coefficients is bypassed. The MMSE scaling factor becomes:

$$\alpha_{MMSE} = \frac{2P}{N + 2P} \tag{3-17}$$

And the effective noise becomes:

$$N_{eff}(\alpha, a_m) = N|\alpha|^2 + 2P(1 - \alpha)^2 = \frac{2PN}{N + 2P} \tag{3-18}$$

Therefore the achievable rate of compute-and-forward is:

$$R_{CF} = \log\left(\frac{P}{N_{eff}}\right) = \log(0.5 + SNR) \tag{3-19}$$

The achievable rates for all the relaying schemes are plotted below in Figure 3-2.

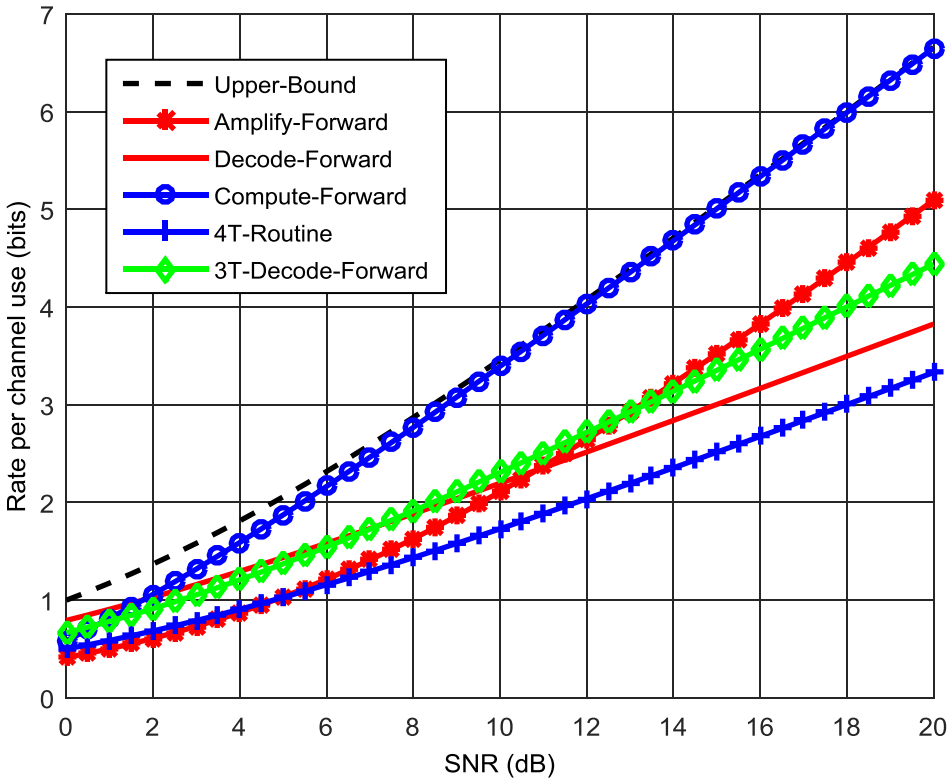


Figure 3-2: Achievable rates of relaying schemes over AWGN channels

3.3 Achievable rate of PNC Schemes over Fading Channel

The upper bound achievable rate*:* The upper bound achievable rate can be considered in the scenario that the signal transmitted from each user arrives at the other user via the relay node without interrupt each other. Thus the achievable rate per-user per channel use is given by:

$$R_{UB} = \frac{1}{4}\left(\log\left(1+\frac{P|h_1|^2}{N}\right)+\log\left(1+\frac{P|h_2|^2}{N}\right)\right) \tag{3-20}$$

Four timeslots routing strategy*:* Again, the rate at which each user can decode the transmitted message is determined by the channel with the worst fading in the path it takes first from the user to relay and then from relay to the other user. Summing the rate for both users over 4 timeslots, the achievable rate per-user per channel use becomes:

$$R_{4T_Routing} = \frac{1}{4}\min\left(\log\left(1+\frac{P|h_1|^2}{N}\right), \log\left(1+\frac{P|h_2|^2}{N}\right)\right) \quad (3\text{-}21)$$

Three timeslots decode-and-forward: The message sent from each user has to be decoded correctly at the relay, and the modulo summation of the messages send from the relay also has to be decoded correctly by each user. The rate at which each user can decode modulo summation and obtain the other user's message is therefore limited by the worst rate of the three point to point decoding, i.e. the two point to point decoding of individual messages at the relay, as well as the decoding of modulo summation broadcasted to the users. Thus the achievable rate per-user per channel use becomes:

$$R_{3T_DF} = \frac{1}{3}\min\left(\log\left(1+\frac{P|h_1|^2}{N}\right), \log\left(1+\frac{P|h_2|^2}{N}\right)\right) \quad (3\text{-}22)$$

***Two timeslots decode-and-forward*:** The message sent from each user has to be decoded correctly at the relay, and the modulo summation of the messages sent from the relay also has be decoded correctly by each user. For the two timeslot decode-and-forward, the two messages transmitted from the users are decoded at MAC rate at the relay given by R_1 and R_2. The relay then broadcast the modulo summation of the two messages to the two users which decodes at rates which are upper bounded by R_3 and R_4 respectively. Totaling the rate for both users over two timeslots, the achievable rate per-user per channel use becomes:

$$R_{DF} = \frac{1}{4}\left(\min(R_1, R_4) + \min(R_2, R_3)\right) \quad (3\text{-}23)$$

where $R_1 = \log\left(1+\frac{P|h_1|^2}{N}\right)$, $R_2 = \log\left(1+\frac{P|h_2|^2}{+N}\right)$, $R_3 = \log\left(1+\frac{P|h_1|^2}{N}\right)$,

$$R_4 = \log\left(1+\frac{P|h_2|^2}{N}\right),\ R_1 + R_2 \le \log\left(1+\frac{P|h_1|^2 + P|h_2|^2}{\sigma^2}\right)$$

***Amplify-and-forward*:** In the first timeslot, the signal received at the relay is $y = h_1x_1 + h_2x_2 + z$ having a power of $P(|h_1|^2 + |h_2|^2) + N$. Since each transmitter has a power constraint of P, the received signal has to be scaled by a factor of

$\gamma = \sqrt{P/(P(|h_1|^2 + |h_2|^2) + N)}$ before being broadcast to both users. In the second timeslot, the signals received by the two users from the relay are:

$$y_1 = \gamma(h_1 x_1 + h_2 x_2 + z)h_1 + z_1 \tag{3-24}$$

$$y_2 = \gamma(h_1 x_1 + h_2 x_2 + z)h_2 + z_2 \tag{3-25}$$

from which each user can subtract its own signal and obtain a corrupted version of the signal transmitted by the other user. The SNR at the receiver of each source node becomes:

$$SNR_1 = \frac{\gamma^2 \cdot P \cdot |h_1 h_2|^2}{\gamma^2 \cdot N|h_1|^2 + N} \tag{3-26}$$

$$SNR_2 = \frac{\gamma^2 \cdot P \cdot |h_1 h_2|^2}{\gamma^2 \cdot N|h_2|^2 + N} \tag{3-27}$$

Therefore, the achievable rate per-user per channel use can be written as:

$$R_{AF} = \frac{1}{4}\left(\log(1 + SNR_1) + \log(1 + SNR_2)\right) \tag{3-28}$$

Note that since the channel gains h_1 and h_2 are not integers, lattice decoding cannot be used to decode $h_1 x_1 + h_2 x_2$. The channel gains also have to be known by the source nodes.

***Denoise-and-forward*:** In the MAC phase, the signal decoded at the relay is $y = h_1 x_1 + h_2 x_2$ having a power of $P(|h_1|^2 + |h_2|^2)$. Since each transmitter has a power constraint of P, the received signal has to be scaled by a factor of $\gamma = \sqrt{1/(|h_1|^2 + |h_2|^2)}$ before being broadcast to both users. In the broadcast phase, the signals received by the two users from the relay are:

$$y_1 = \gamma(h_1 x_1 + h_2 x_2)h_1 + z_1 \ , \tag{3-29}$$

$$y_2 = \gamma(h_1 x_1 + h_2 x_2)h_2 + z_2 , \tag{3-30}$$

from which each user can subtract its own signal and obtain a corrupted version of the signal transmitted by the other user. The SNR at the receiver of each source node becomes:

$$SNR_1 = \frac{\gamma^2 \cdot P \cdot |h_2|^2}{N} \tag{3-31}$$

$$SNR_2 = \frac{\gamma^2 \cdot P \cdot |h_1|^2}{N} \tag{3-32}$$

Therefore, the achievable rate per-user per channel use can be written as:

$$R_{DNF} = \frac{1}{4}\left(\log\left(1 + \frac{\gamma^2 \cdot P \cdot |h_2|^2}{N} \right) + \log\left(1 + \frac{\gamma^2 \cdot P \cdot |h_1|^2}{N} \right) \right) \tag{3-33}$$

Note that since the channel gains h_1 and h_2 are not integers, lattice decoding cannot be used to decode $h_1 x_1 + h_2 x_2$. The channel gains also have to be known by the source nodes.

***Compute-and-forward*:** Considering the obtainable rates at the relay during the MAC phase and then at user nodes during the broadcast phase, from the cut-set bound, the achievable rate per-user per channel use can be written as:

$$R_{CF} = \frac{1}{4}(R_1 + R_2) \tag{3-34}$$

$$R_1 = \min\left(R_{\text{CF_Phase1}}, \log\left(1 + \frac{P|h_1|^2}{N} \right) \right) \tag{3-35}$$

$$R_2 = \min\left(R_{\text{CF_Phase1}}, \log\left(1 + \frac{P|h_2|^2}{N} \right) \right) \tag{3-36}$$

$$R_{\text{CF_Phase1}} = \frac{1}{2}\log\left(\left(\left\| a^{\text{opt}} \right\|^2 - \frac{P\left| a^{\text{opt}} h^H \right|^2}{N + P\|h\|^2} \right)^{-1} \right) \tag{3-37}$$

Equation (3-34) is the achievable rate at the relay node using compute-and-forward PNC scheme over Rayleigh fading channel derived in [20], where $\mathrm{a}^{\text{opt}} = [a_1, a_2]$ is the optimal integer coefficient vector which is dependent on the channel coefficient vector $\mathrm{h} = [h_1, h_2]$. With the variances of the channel gains to be $\sigma_{h_1}^2 = \sigma_{h_2}^2 = 1$, the average achievable rates of all the relaying schemes are plotted in Figure 3-3.

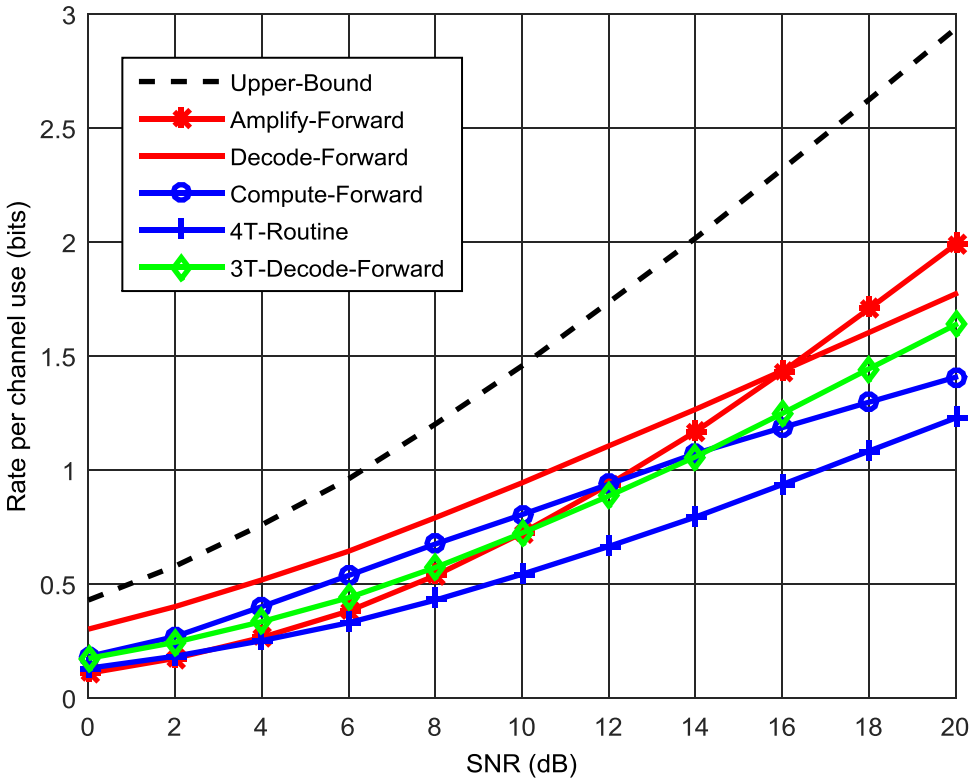


Figure 3-3: Achievable rates of relaying schemes over fading channel without precoding.

3.4 Achievable rate of channel inversion precoded PNC Schemes over Rayleigh Fading Channel

It is known from [40] that with channel knowledge available at the transmitter, waterfilling scheme for power and rate adaptation can be employed to maximize the capacity of a communication system over fading channels. It is shown that this technique results in the highest capacity but incurs a significant system delay. For correlated fading the optimality of waterfilling scheme is most profound, achieving both a large capacity increase, as well as lower complexity over non-adaptive schemes with receiver channel state information. For independent and identically distributed (i.i.d) fading, the gain in capacity and reduction in complexity is far less significant.

An alternative method is to perform non-power restricted channel inversion precoding at the transmitter so that the channel becomes effectively as an AWGN channel. Although channel inversion precoding is inefficient and results in capacity loss, particularly for severe fading conditions, the scheme offers the advantage of low complexity and minimum delay

since it allows the encoding and decoding to be designed for the AWGN channel, independent of fading statistics. Additionally it allows the same data rate to be maintained at all channel states.

. In this section, we aim to optimise the achievable rates of selected PNC schemes, under the condition of equal data rate across all channel states by applying channel inversion precoding schemes to the PNC schemes. One important feature of two-way relay channel is that channel fading can be compensated completely at the transmitter for the relay receiver using channel inverse precoding, which is not possible for a general relay network of multiple source nodes and multiple relay nodes, as each transmitter cannot precode for multiple channel instances to multiple relay nodes. For the same reason, channel inverse precoding cannot be used at the relay transmitter in the broadcast phase for the two-way relay, unless we assume symmetric channels from relay to user nodes. Assuming the instant transmit power of each transmitter can be very high to avoid saturation, precoding can be performed at the transmitters of the two source nodes to make sure that the signal received at the relay remains to be $\mathrm{y} = x_1 + x_2 + \mathrm{z}$, which is the same as in the case of AWGN channel. Through the use of pilot symbols in transmission packets, the relay can estimate the channels h_1 and h_2, and broadcast this information back to the user nodes. Assuming slow fading channel where channel conditions remain constant between MAC and broadcast phase, the user nodes can use this information to precode symbols in the next packet transmitted to the relay, namely User-1 with $1/h_1$ and the signal of User-2 with $1/h_2$. This also means the transmit power of the two source nodes becomes $P/|h_1|^2$ and $P/|h_2|^2$ respectively. Without restricting the transmit power of the source nodes and the relay node, the achievable rates of various PNC schemes when precoding is used at user transmitter but not at the relay are derived in this section. First we define a total power variation factor α given by,

$$\alpha = \frac{P/|h_1|^2 + P/|h_2|^2}{2P} = \frac{|h_1|^2 + |h_2|^2}{2|h_1|^2|h_2|^2} \tag{3-38}$$

This power variation factor represents the average additional power increase required at source nodes to compensate channel fading and allow the network coding scheme to achieve AWGN rate as if the channel were AWGN channel. Note precoding is only done by user transmitters for MAC phase and not at the relay. Precoding removes fading in the uplink at the expense of additional transmission power (represented by power variation factor α), which is included in the rate derivation. The reasoning is that an increased transmit power of

αP is required so a signal of power P is received at the receiver. Therefore for fair comparison with non-precoded case, we make assumption that the average transmitter power after precoding is P, and therefore the received signal power will be P/α. Taking the variation of the transmit power caused by the precoding into account, the achievable rates of various PNC schemes under precoding are derived.

The upper bound achievable rate: The upper bound achievable rate per-user per channel use becomes:

$$R_{UB} = \frac{1}{4}\left(R_1 + R_2\right) \tag{3-39}$$

where,

$$R_1 = \min\left(\log\left(1+\frac{P}{\alpha N}\right), \log\left(1+\frac{P|h_1|^2}{\alpha N}\right)\right) \quad \text{and} \quad R_2 = \min\left(\log\left(1+\frac{P}{\alpha N}\right), \log\left(1+\frac{P|h_2|^2}{\alpha N}\right)\right)$$

Three timeslots decode-and-forward: The rates are derived by using the same reasoning as for the un-precoded case. With the correct precoding at user nodes, the relay can decode each user's message at AWGN point to point rate. As precoding cannot be performed at the relay for the broadcast phase, the channel fading during the broadcast phase again also acts as the limit for the decodable rate at each user, represented by R_1 and R_2. The achievable rate per-user per channel use becomes:

$$R_{3T_DF} = \frac{1}{6}\left(R_1 + R_2\right) \tag{3-40}$$

where,

$$R_1 = \min\left(\log\left(1+\frac{P}{\alpha N}\right), \log\left(1+\frac{P|h_1|^2}{\alpha N}\right)\right) \quad \text{and} \quad R_2 = \min\left(\log\left(1+\frac{P}{\alpha N}\right), \log\left(1+\frac{P|h_2|^2}{\alpha N}\right)\right)$$

***Two timeslots decode-and-forward*:** The achievable rate per-user per channel use becomes:

$$R_{DF} = \frac{1}{4}\left(R_1 + R_2\right) \tag{3-41}$$

where,

$$R_1 = \min\left(\frac{1}{2}\log\left(1+\frac{2P}{\alpha N}\right), \log\left(1+\frac{P|h_1|^2}{N}\right)\right) \quad \text{and} \quad R_2 = \min\left(\frac{1}{2}\log\left(1+\frac{2P}{\alpha N}\right), \log\left(1+\frac{P|h_2|^2}{N}\right)\right)$$

***Amplify-and-forward**:* In the first timeslot, the signal received at the relay is $y = x_1 + x_2 + z$. In the second timeslot, the signals received by the two users from the relay are:

$$y_1 = (x_1 + x_2 + z)h_1 + z_1 \tag{3-42}$$

$$y_2 = (x_1 + x_2 + z)h_2 + z_2 \tag{3-43}$$

from which each user can subtract its own signal and obtain a corrupted version of the signal transmitted by the other user. The SNR at the receiver of each user becomes:

$$SNR_1 = \frac{|h_1|^2 \cdot P/\alpha}{N|h_1|^2 + N} \qquad SNR_2 = \frac{|h_2|^2 \cdot P/\alpha}{N|h_2|^2 + N} \tag{3-44}$$

Therefore, the achievable rate per-user per channel use becomes:

$$R_{AF} = \frac{1}{4}\left(\log(1+SNR_1) + \log(1+SNR_2)\right) \tag{3-45}$$

***Compute-and-forward*:** The achievable rate per-user per channel use can be written as:

$$R_{CF} = \frac{1}{4}(R_1 + R_2) \tag{3-46}$$

where,

$$R_1 = \min\left(R_{CF_Phase1}, \log\left(1+\frac{P|h_1|^2}{\alpha N}\right)\right), \; R_2 = \min\left(R_{CF_Phase1}, \log\left(1+\frac{P|h_2|^2}{\alpha N}\right)\right), \tag{3-47}$$

and,

$$R_{CF_Phase1} = \log\left(0.5+\frac{P}{\alpha N}\right),$$

is the achievable rate by the compute-and-forward PNC scheme over AWGN channel at the relay node as given in (3-19). With the variances of the channel gains to be $\sigma_{h_1}^2 = \sigma_{h_2}^2 = 1$, the average achievable rate of all the relaying schemes for 1000 channel instances are plotted below in Figure 3-4.

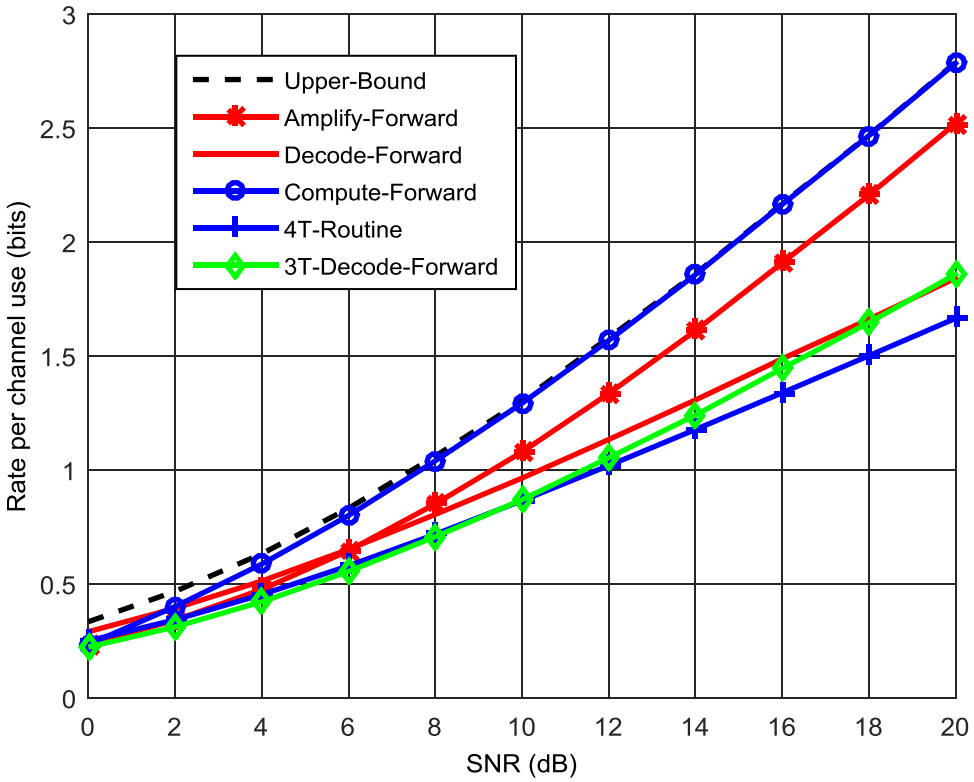


Figure 3-4: Achievable rates of relaying schemes over fading channel with precoding

From Figure 3-4, it can be seen that using the proposed precoding scheme, compute-and-forward has the best rate of all the relaying schemes and has a higher rate compared with that without precoding. With precoding it is possible to obtain a received signal of $y = x_1 + x_2 + z$ at relay. The highly improved performance of compute-and-forward when compared to the compute-and-forward without precoding, is due to the resulting AWGN channel computation observed at the relay. Since the fading coefficients are removed by the channel inversion precoding, the scheme does not suffer performance degradation caused by the Diophantine trade-off between channel approximation and amplification of original noise, when the C-F scaling factor is applied to the received signal. By using channel inversion with C-F, it is possible to significantly improve the C-F, at the cost of increasing transmitter power by a factor of α. Based on the achievable rate defined in equation, and the comparison the theoretical achievable rate for precoded compute-and-forward over TWRC against other PNC schemes with precoding as shown in Figure 3-4, it can be concluded that for a large number of sampled channel instances the power penalty of increasing the transmit power by a factor

of α channel inversion precoding is a small price to pay in exchange for the significantly improved performance.

Another advantage of using compute-and-forward over the two way relay channel is that each user or destination node only needs to decode one unknown message, hence only one integer linear equation needs to be decoded at the relay, which can always be provided by the relay node. Also note that compute-and-forward cannot be used for a general relay network of multiple source nodes and multiple relay nodes over AWGN channel. This is because the relay nodes need to provide the destination nodes with enough independent integer linear equations.

3.5 Achievable rate of Channel-aware Precoding for Compute-and-Forward over Fading Channel

In Section 3.4, we found that for all the PNC schemes investigated, channel inversion precoding results in the largest performance gain for the C-F scheme. However one application issue of the channel inversion precoding method in general, is that in a real wireless transmitter, whether a base station or a user terminal, the transmit power has an upper limit and cannot be arbitrarily high. Thus when the channel fading is severe, precoding cannot be efficiently used without incurring large loss in achievable rate. To address this problem, we propose a selective precoding transmission scheme for C-F which we call C-F with channel aware precoding. The transmission scheme selects between two different modes of transmission based on the quality of the two uplink two way relay channels. C-F with channel inversion precoding is performed for uplink channels above a predefined threshold on the depth of channel fading, while TDMA transmission is used when only one of the two uplink channels is above the same predefined threshold. Assuming the average transmit signal power is P, and an instant transmit power limit is P_{Limit}, the transmission scheme can be described by the following modes of operation:

Scenario 1): If both channels are in good condition, i.e., $P/|h_1|^2 \le P_{Limit}$ and $P/|h_2|^2 \le P_{Limit}$, both users precode their signals and compute-and-forward scheme is applied.

Scenario 2): If both channels are in bad condition, i.e., $P/|h_1|^2 > P_{Limit}$ and $P/|h_2|^2 > P_{Limit}$, neither user transmits. By doing so, the transmit power is also saved.

Scenario 3): If one of the channels is in good condition, i.e., $P/|h_1|^2 \le P_{Limit}$ or $P/|h_2|^2 \le P_{Limit}$, only the user with good channel condition performs precoding and transmit, the other user does not transmit. By doing so, the transmitting user does not get interfered from the other user thus can have a upper limit of single user rate. The average achievable rate per-user per timeslot can be expressed by:

$$R_{CF} = 0.5\left(p_1 \log\left(0.5 + \frac{P}{\alpha N}\right) + \quad p_2 \log\left(1 + \frac{P|h_1|^2}{N}\right) + p_3 \log\left(1 + \frac{P|h_2|^2}{N}\right)\right) \tag{3-48}$$

where p_1 is the probability of both h_1 and h_2 are good; p_2 is the probability of only h_1 is good; and p_3 is the probability of only h_2 is good. Note that rate loss incurred due to non-transmission in the case when both channels are poor, is factored into theoretical rate equation 3-48. For comparison purpose, the average achievable rates per-user per timeslot are plotted below in Figure 3-5, where the variances of the channel gains are set to be, $\sigma_{h_1}^2 = \sigma_{h_2}^2 = 1$, and the rate per channel use against SNR, is plotted using the average rate for 10000 channel instances.

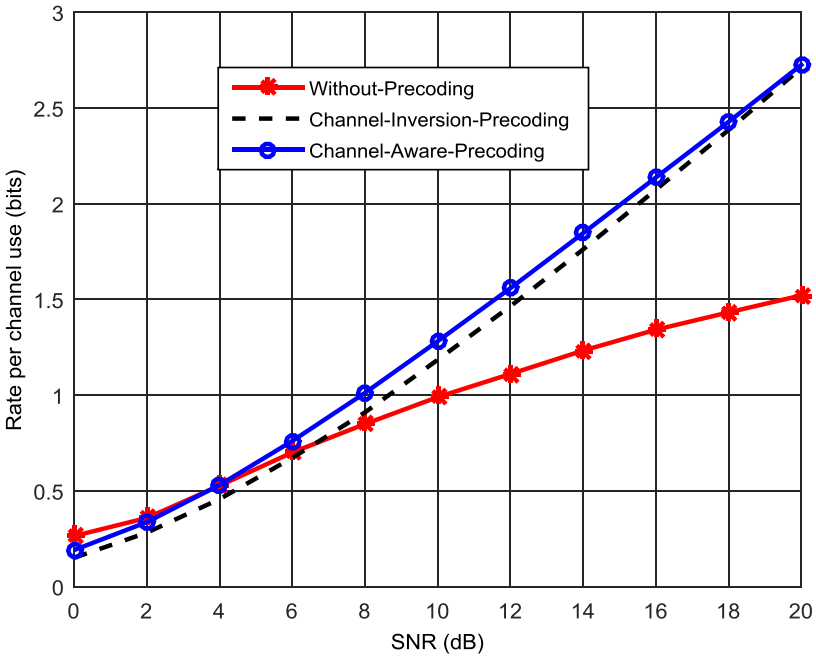


Figure 3-5: Achievable rates of relaying schemes over Rayleigh fading channel with channel aware precoding.

From Figure 3-5, it can be seen that with channel aware precoding, the achievable rate of compute-and-forward is superior to both C-F with channel inversion precoding and C-F without precoding when the SNR exceeds 4 dB. It can be observed that for low SNR values, C-F without precoding attains superior theoretical rates than both channel inversion precoding and channel aware precoding. This is because the power cost of precoding is fixed for all SNRs, since the additional power only depends on the channel instances. Therefore at low SNR values, the cost of precoding due to additional power expenditure at the transmitters exceeds the benefit of increased rate attained by precoding the channel fading instances. However, at medium to high SNR, the additional power cost is more than compensated by the superior performance of the precoded C-F schemes.

Simulation verification:

A real case communication system simulation was carried out to see whether the error performance with the proposed channel aware precoding scheme is indeed better than that of without precoding and with unrestricted precoding over Rayleigh fading channel. In the simulation, Z^2 lattice code of 2 points per-dimension is employed in the two source nodes, which is equivalent to 4-QAM constellation. At the relay node, lattice decoding is used to decode the integer linear combination of the codewords sent from the two users. The error rates are plotted in Figure 3-6.

For C-F with channel inversion precoding, the symbol error rate shows a drop-off characteristic of symbol error rate performance over AWGN channel. Therefore the curve for channel inversion precoding is effectively the SER performance curve shifted by a quantity along the SNR axis in the positive direction to account for the additional power usage. Similarly for channel aware precoding C-F scheme, the symbol error rate is adjusted by an adjustment factor that reflects the additional power used for the precoding scheme. The symbol error rate count for the channel aware precoding scheme is tallied by detecting for error in the network coded symbol when C-F transmission is used and incrementing the error count by one for such error instances; and detecting for error in the transmitted symbol when TDMA transmission is used, and incrementing the error count by 0.5 for such types of errors.

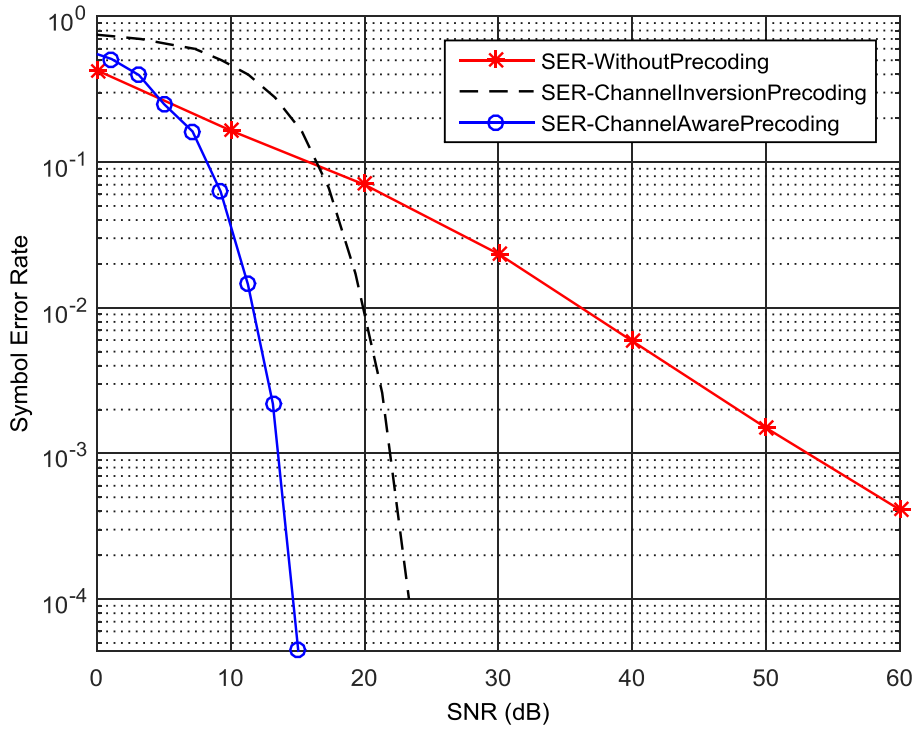


Figure 3-6: Symbol error rates of compute-and-forward over fading channel with and without the proposed channel precoding schemes.

From Figure 3-6, it can be seen that when the SNR is larger than a certain value (4 dB for 4-QAM in the simulation), the performance of C-F with channel aware precoding is far better than C-F without precoding. This corresponds to the theoretical rates plotted in Figure 3-5 which shows the two precoding-based schemes outperforming C-F without precoding for high SNR. At low SNR, the better performance of C-F without precoding compared to the two precoding-based schemes can be attributed to the fixed power penalty cost for performing precoding, for all SNR values, as explained for the theoretical rate curves in Figure 3-5. From Figure 3-6, it can be observed that the performance of C-F with channel inversion precoding is only better than C-F without precoding when the SNR is larger than about 16.5 dB. Comparing the two precoding-based C-F schemes, channel aware precoding displays an earlier roll-off than channel inversion precoding. This is because channel inversion precoding incurs a much higher power penalty by performing precoding under all channel conditions, rather than selectively precoding and transmitting over only good quality channels, as done for the channel aware precoding scheme. Note that the performance gap between C-F with channel inversion precoding and C-F with channel aware precoding is also determined by power-limit set for the transmitters for the channel aware precoding scheme.

4 New JCCNC Schemes for Two-way Relay Systems

In this chapter, we investigate the application of channel coding to PNC schemes over the two way relay channel and propose low complexity joint channel coded network coding schemes (JCCNC) that can be efficiently used with higher order PAM and QAM modulations. .

In literature two main paradigms of channel coded network coding were outlined for the two way relay channel. One approach studied in [2], [3] and [62] is the XOR-mapped channel coded network coding scheme, which we shall refer to as XOR-JCCNC. In this scheme, the relay jointly decodes the XOR of the channel coded messages by first performing PNC mapping before performing standard point-to-point channel decoding. The PNC mapping consists of decoding the received signal at the relay to a superimposed constellation symbol, and then mapping the superimposed signal combination to a network coded symbol by obtaining soft information on the network coded symbol. The soft information on network coded symbols is then fed into a point to point decoder which decodes a network coded symbol.

The second main approach to implement channel coded network coding scheme, studied in [7] and [62] comprises directly decoding the superimposed constellation based on a tanner graph reconstructed for the superimposed encoder, and performing PNC mapping to map decoded superimposed signal to network coded symbols. This scheme is commonly known as the arithmetic-sum channel coded network coding (CCNC) scheme.

One drawback of the XOR-JCCNC scheme is that information is lost when mapping from the superimposed symbol to the network coded symbol, which leads to a worse symbol error rate performance compared to arithmetic sum CCNC.

Furthermore a disadvantage of the arithmetic-sum CCNC scheme is that the tanner graph of the superimposed encoder will need to be reconstructed for code over different finite fields, and a new sum-product algorithm will need to be designed based on the superimposed encoder for different finite fields. Therefore, different arithmetic-sum CCNC decoding algorithms must be used for codes over different finite fields. This introduces huge complexity into the design and implementation since a real communication system may adaptively select the transmitting modulation order based on the channel conditions. There is

clearly demand for low complexity CCNC schemes that can be used with higher order modulations with decoding algorithms can be easily adapted for high order modulations without requiring significant redesigning.

In this chapter, four novel channel coded network coding schemes based on lattice decoding are proposed, and presented in the following order: Lattice-decoded JCCNC (LD-JCCNC) with soft information, LD-JCCNC with hard decoding, Low density construction A JCCNC (LDA-JCCNC) and Polar lattice aided JCCNC (PLA-JCCNC). Each scheme offers specific benefits over the two main JCCNC approaches described in literature.

The key advantage over XOR-JCCNC, offered by the proposed LDA and PLA schemes is a reduced decoding complexity, due to fewer number of a posteriori probability calculations required for initializing the messages for the belief propagation decoder. For higher modulation order, or systems using code over a larger finite field, the achievable reduction in decoding complexity over XOR-JCCNC increases significantly, making the proposed LDA-JCCNC and PLA-JCCNC schemes particularly suitable for higher order modulations

The proposed lattice-decoded JCCNC with hard-decoding, employing hard decoded values as input into sum-product algorithm offers the advantage over standard XOR-JCCNC, of not requiring channel noise information for the channel decoding of the network coded symbols. Therefore the performance of the LD-JCCNC is not affected by the misestimation of the channel noise. Additionally, the scheme benefits from reduced complexity associated with lattice decoding.

The four proposed channel coded network coding schemes can be used with general PAM and QAM modulation systems that satisfy certain criteria. Additionally, the proposed schemes offer the advantage of employing fixed sum-product decoding algorithms for code over different finite field sizes, and therefore do not carry the overhead of requiring different decoding algorithms to be written for modulation order or code over different alphabet sizes.

In this chapter, we first introduce LDPC code and Polar code to be used in channel coded two-way relay systems. Then we present the details of our four newly proposed JCCNC schemes. We present the schemes over the AWGN channel although they can similarly be adapted to the Rayleigh fading channel. For our two-way relay channel models, we assume that perfect power control, precoding and synchronization are applied so that the channels between user nodes and relay have unit gain.

4.1 Review of LDPC Code

4.1.1 LDPC Code

Interest in iterative decoding algorithms led to the rediscovery of the low-density parity-check (LDPC) codes. LDPC codes were invented by Gallager in 1962 [79][80] but were not pursued due to implementation complexity until the last decade. The bit error rate (BER) performance of the normally designed LDPC codes is close to that of the turbo codes. Modified versions of the original scheme, implemented as irregular LDPC codes are shown to perform even better than the turbo codes when operating over GF(q), with q equal to 4, 8 and 16 [81].

4.1.1.1 Binary LDPC Code

4.1.1.1.1 LDPC Codes Construction

LDPC codes are linear block codes and thus the set of all codewords, $\mathbf{v} \in C$, spans the null space of a $M \times N$ parity check matrix **H,**

$$\mathbf{Hv} = 0, \quad \forall \mathbf{v} \in C \tag{4-1}$$

The parity check matrix **H** for LDPC codes is a sparse binary matrix. Each row of **H** corresponds to a parity check and an element H_{mn} indicates that the data symbol n participates in the parity check m. In a block of N code bits or symbols, there are M redundant parity symbols and the code rate R is given by,

$$R = (N - M)/N. \tag{4-2}$$

Designing a LDPC code is to select the parity check matrix. In LDPC code design, the elements of **H** are chosen to satisfy a desired row and column weight profile, where the row and column weights are defined as the number of 1s in a given row and column respectively [82]. In a *regular* LDPC code, all rows and columns have a uniform row weight and column weight respectively. If the row and column weights are not uniform, the LDPC code is said to be *irregular*. Regular LDPC codes generally have inferior BER performance to the irregular LDPC codes. In general, we can always design LDPC codes as systematic codes.

According to the construction method used for generating the corresponding sparse parity check matrix **H**, LDPC codes can be classified as random or structured. In general, random LDPC codes show a slightly better BER performance than that of structured LDPC codes although structured LDPC codes are less complex to encode than random LDPC codes. LDPC codes can also be represented using a bipartite graph [83], where one set of nodes

represents the parity check constraints and the other set of nodes represents the data symbols or variables as illustrated in Figure 4-1.

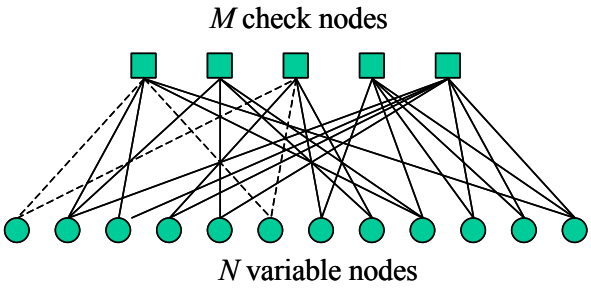


Figure 4-1: Bipartite graph representation of an LDPC code. A short cycle of length 4 has been highlighted.

The parity check matrix is the incidence matrix of the graph where a variable node v_n, corresponding to column n in **H,** is connected to check node z_m, corresponding to row m in **H**, if the entry H_{mn} in **H** is "1". As will be mentioned in the decoding section, for good decoding performance, it is important that the length of the *cycles* in the graph representation is as long as possible, which is also the task of the LDPC code design and optimisation.

4.1.1.1.2 LDPC codes encoding

The general procedure of LDPC code encoding is:

a) Generate the sparse parity check matrix **H** under certain constraints based on a given design rule.
b) Encode the information bits in some way using the matrix **H**.

In general, we consider only systematic codes, in which the information bits are directly represented by a subset of the N code bits. Let's partition the matrix **H** into a $M \times M$ left part **A**, and a $M \times (N-M)$ right part **B**. Rearrange columns if necessary to make **A** non-singular. Partition a codeword $\boldsymbol{v}$ in the same way into M check bits **c**, and N-M information bits **u**. The parity check equation, **Hv**=0, becomes,

$$[\mathbf{A}\,|\,\mathbf{B}]\begin{bmatrix}\mathbf{c}\\ \mathbf{u}\end{bmatrix} = 0\,. \tag{4-3}$$

From the above, we have

$$\mathbf{c} = \mathbf{A}^{-1}\mathbf{B}\mathbf{u} \tag{4-4}$$

where the $M \times (N-M)$ matrix $\mathbf{A}^{-1}\mathbf{B}$ is the generator matrix and can be pre-computed. When column rearranging happens in making a non-singular matrix **A** (this is the usual case),

a vector of indices is needed to remember the reordering, which is used to map the information bits to the correct position in the code word.

4.1.1.1.3 LDPC codes decoding

The algorithm used for decoding LDPC codes is known as the Belief Propagation (BP) algorithm (also known as the sum-product algorithm (SPA)) [79][80][82]. For good decoding performance with this algorithm, it is important that the length of the *cycles* in the graph representation is as long as possible [82][83], which is the task of the LDPC code design and optimization. Short cycles, such as the length-4 cycle illustrated in Figure 4-1, degrade the performance of the message passing algorithm. However, the degrading effect of the short cycles diminishes as the code length increases, and is significantly reduced if the code length is larger than 2000.

The BP algorithm is a kind of message passing algorithm which involves iterations in decoding. The algorithm determines the *a posteriori* probability of each information bit as a function of the received signal, the code information expressed as parity equations, and the channel characteristics.

4.1.1.1.3.1 The a priori probability of coded bits

Before the iterative decoding starts, the *a priori* probability of each code bit needs to be determined by the decoder. Assuming the coded bits are digitally modulated using a QAM or a PSK modulation scheme before being transmitted over the wireless channel, then the soft output from the QAM or PSK demodulation, i.e., the reliability information (i.e. the likelihood ratio or log-likelihood ratio) of each code bit, are just what we need here. We denote this *a priori* probability of the *n*-th code bit as γ_n and $\overline{\gamma}_n$ to represent likelihood ratio (LR) and log likelihood ratio (LLR) respectively.

4.1.1.1.3.2 Decoding procedure

For the convenience of describing the BP decoding algorithm, we introduce the following definitions. The set of code bits *n* that participate in parity check *m* is defined as $\mathcal{N}(m) = \{n : H_{m,n} = 1\}$, and the set of parity checks *m* in which code bit *n* participates as $\mathcal{M}(n) = \{m : H_{m,n} = 1\}$. The set $\mathcal{N}(m)$ with code bit *n* excluded is denoted by $\mathcal{N}(m) \setminus n$, and the set $\mathcal{M}(n)$ with parity check m excluded by $\mathcal{M}(n) \setminus m$. With these definitions, the iterative BP decoding algorithm can be described below.

Input: *LR or LLR soft input information for the N code bits* $\{\gamma_1, \gamma_2, ..., \gamma_N\}$ *or* $\{\bar{\gamma}_1, \bar{\gamma}_2, ..., \bar{\gamma}_N\}$

.

Output: *Hard decision estimation* $\hat{v} = \{\hat{v}_1, \hat{v}_2, ..., \hat{v}_N\}$.

1 Initialization

Compute the a priori LR information for each code bit v_n: $\gamma_n = \frac{p_n^0}{p_n^1}$, and for each $(m,n) \in \{(i,j) \mid H_{i,j} = 1\}$, compute

$$\alpha_{m,n} = \frac{\gamma_{m,n} - 1}{\gamma_{m,n} + 1} = \frac{\gamma_n - 1}{\gamma_n + 1} \tag{4-5}$$

2 Iterative decoding

(a) *Horizontal (or check node computation) step*: For each m, n, compute

$$\beta_{m,n} = \frac{1+\alpha}{1-\alpha} = \frac{1 + \prod_{n' \in \mathcal{N}(m) \backslash n} \alpha_{m,n'}}{1 - \prod_{n' \in \mathcal{N}(m) \backslash n} \alpha_{m,n'}} \tag{4-6}$$

(b) *Vertical (or variable node computation) step*: For each m, n, compute

$$\alpha_{m,n} = \frac{\gamma_{m,n} - 1}{\gamma_{m,n} + 1} = \frac{\gamma_n \cdot \prod_{m' \in \mathcal{M}(n) \backslash m} \beta_{m',n} - 1}{\gamma_n \cdot \prod_{m' \in \mathcal{M}(n) \backslash m} \beta_{m',n} - 1} \tag{4-7}$$

where $\gamma_{m,n} = \gamma_n \cdot \prod_{m' \in \mathrm{M}(n) \backslash m} \beta_{m',n}$. Update the a posteriori LR information λ_n for each code bit v_n as:

$$\lambda_n = \gamma_n \cdot \prod_{m \in \mathrm{M}(n)} \beta_{m,n} \tag{4-8}$$

(c) *Decision step*:

(i) Perform hard decision on $(\lambda_1, ..., \lambda_N)$ to obtain $\hat{v} = \{\hat{v}_1, \hat{v}_2, ..., \hat{v}_N\}$ such that $\hat{v}_n = 0$ if $\lambda_n \geq 1$ and $\hat{v}_n = 1$ if $\lambda_n < 1$;

(ii) If $\boldsymbol{H} \cdot \hat{v} = 0$ then algorithm terminates, else go to step 2 (a) until the pre-set maximum number of iterations have been reached.

The variables $\alpha_{m,n}$ and $\beta_{m,n}$ in the above BP algorithm are referred to as the *extrinsic* LR information. The $\alpha_{m,n}$ is the soft information delivered from variable node to check node and $\beta_{m,n}$ is the soft information delivered from check node to variable node.

4.1.1.2 Non-binary LDPC Code

One of limitations of binary LDPC is that the capacity approaching performance can be achieved only for randomly constructed codes of very large block length. This can be improved by extending LDPC codes over finite fields of GF(q), q>2, It has been shown that non-binary LDPC code of moderate length can achieve better practical performance than binary LDPC code of large block length [84].

4.1.1.2.1 Construction of Non-binary LDPC code

The construction method for non-binary LDPC codes is similar to that for binary LDPC codes, which we do not repeat here.

4.1.1.2.2 Decoding of Non-binary LDPC Code

The BP algorithm is also used for non-binary LDPC code decoding [84]. The decoding process can be best described by using message passing on the bipartite graph representing the parity check matrix of the LDPC code, which is shown below in Figure 4-2.

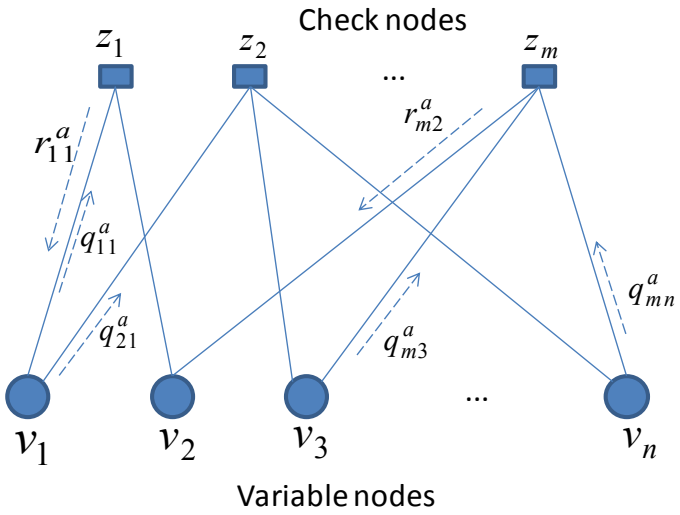


Figure 4-2: Message passing on the bipartite graph representing a parity check matrix

For each iteration, given the received signal and channel properties, the posterior probability of the value of each variable node is calculated. The algorithm can be observed clearly on a bipartite graph of variable node v_n and check node z_m based on H matrix. During each iteration of decoding, each variable node v_n sends messages q_{mn}^a to each check node z_m connected to it. The q_{mn}^a messages approximate the variable node's belief that v_n has value of *a,* based on the r_{mn}^avalues it has received from the other check nodes connected to it. Additionally each check node z_m send message r_{mn}^a to each variable node connected to it. The quantity r_{mn}^a approximate the probability of check node z_m being satisfied if variable node is assumed to be fixed at *a,* and taking into account messages received from all its other connected variable nodes given by probabilities $\{q_{mn'}^a : n' \in \mathcal{N}(m) \backslash n\}$.

After each iteration, using the updates q_{mn}^a and r_{mn}^a, tentative decoding is performed. The decoding algorithm is repeated until the decoded codeword produces a syndrome of 0 for which the decoding would be declared a success. Alternatively if successful decoding does not occur after a set number of iterations, decoding is declared a failure.

Given f_n^a, the probability $v_n = a$ according to the channel model, $n = 1, \dots, N$, where v_n can take any value $a \in \mathrm{GF}(q), i = 0, \dots q-1$, the decoding process can be described below.

Initialization: First set values of q_{mn}^a to f_n^a, the probability $v_n = a$ according to the channel model.

Updating $\boldsymbol{r_{mn}^a}$: New value of $r_{mn}^a$is obtained via:

$$r_{mn}^a = \sum_{\boldsymbol{x}: x_n = a} P(z_m | \boldsymbol{v}) \prod_{j \in \mathcal{N}(m) \backslash n} q_{m_j}^{x_j} \tag{4-9}$$

The $P(z_m|\boldsymbol{v}) \in \{0,1)$ depending on whether v satisfies check *m.*

The $\boldsymbol{R}$ matrix can be obtained efficiently by taking the partial sums of a parity checks as states in a Markov chain, where the transition probability determined by the relevant $\boldsymbol{Q}$ values. The forward and backward probabilities can be calculated using probabilities given by the $\boldsymbol{Q}$ messages as below:

$$F_{mn}(a) = P\left(\sum_{j<n} H_{mj}\, v_j = a\right) \tag{4-10}$$

$$B_{mn}(a) = P\left(\sum_{j>n} H_{mj}\, v_j = a\right) \tag{4-11}$$

$$r_{mn}^a = \sum_{s,t: H_{mn} a + s + t = z_i} F_{mn}(s)\, B_{mn}(t) \tag{4-12}$$

Updating $\boldsymbol{q_{mn}^a}$: The messages q_{mn}^a which is sent from the variable node *n* to check *m* should approximate the variable node's belief that v_n has value of *a,* based on the r_{mn}^a values it has received from the other check nodes connected to it.

To update q_{mn}^{a}, the product of all other check nodes' votes for state a, weighted by their prior probability f_n^a, is calculated . For variable node v_n the updated outgoing message to check node z_m for each value *a* is given as:

$$q_{mn}^{a} = \alpha_{mn} f_n^a \prod_{j \in \mathcal{M}(n) \backslash m} r_{jn}^{a}, \tag{4-13}$$

where term $\alpha_{mn}$is chosen so that $\sum_{a=1}^{q} q_{mn}^{a} = 1$. Again forward and backward algorithm can be used to find the update. Once the Q and R updates are obtained, tentative decoding is performed where for each $n \in \{1, \ldots N\}$ and possible state a, the term v_n is calculated:

$$\hat{v}_n = \text{argmax}_a \, f_n^a \prod_{j \in \mathcal{M}(n) \backslash m} r_{jn}^{a} \tag{4-14}$$

The decoding procedure will repeat and R and Q will be repeatedly updated until a valid decoding of the syndrome occurs, in which case the syndrome equation $H\hat{v}_n = z$ is satisfied. If successful decoding does not occur after a set number of iterations, then failure of decoding is declared.

4.1.1.2.3 Complexity of Non-binary LDPC decoding

It was shown by Mackay that LDPC codes can approach Shannon's limit given sufficiently high **H** column weight are selected and given sufficiently long block length. But increasing the column weight leads to increase in number of cycles in the bipartite graph which reduces performance of the iterative decoding method. When decoding over $\text{GF}(q = 2^p)$, where p is the number bits per code symbol, the mean column weight t of the equivalent binary matrix $\mathbf{H}_2$ of q-ary matrix $\mathbf{H}_q$ is increased, and the bipartite graph remains the same. However the complexity of the algorithm is increased to $O(q^2/p)$ [85].

4.2 The Lattice-Decoded (LD)-JCCNC Schemes

Before we describe the details of the new JCCNC scheme, we introduce some background knowledge and give an important definition, and some useful examples.

With regard to the ECC code, assuming the mathematical operations are within the finite field for which the linear ECC code is defined, a linear combination of two linear ECC codewords will still be a codeword [86]. With regard to lattice codes, assuming lattice decoding is used (that is, modulo operation is performed after linear summation), a linear combination of two lattice codes is still a lattice code. We assume that there is a one-to-one mapping between the lattice code and the channel code used at each node. Below we introduce the following definition:

Definition 1: Consider a noise free channel coded two-way relay network, where the message to be transmitted at each user node is channel coded and then mapped to a lattice code (constellation). For any pair of codewords sent by the two source nodes during uplink, we define the channel code and lattice code as being **compatible**, if a unique codeword can be recovered at the relay node with modulo lattice decoding, and the recovered codeword is equal to the linear combination of the two transmitted codewords in Galois field.

The importance of selecting channel code (ECC) and lattice code that are compatible in our proposed JCCNC schemes for the TWRC, is that it ensures a one to one mapping relationship between the output of the lattice decoder at the relay and the network coded symbols Note that Definition 1 can be naturally extended to general multi-way relay systems. Based on the definition of channel code and lattice code compatibility, we propose two conditions where the channel code and lattice code compatibility holds true.

Condition 1: A linear channel code and lattice code are compatible if the channel code operates in the finite field GF(q), while the lattice code has a cardinality q, where q is a prime.

Condition 2: A linear channel code and a lattice code are compatible if the lattice code has a cardinality of q^m, while the channel code operates in the extension field $\mathrm{GF}(q^m)$ of $\mathrm{GF}(q)$, where m is a positive integer and q is prime..

These two conditions can be easily understood because for a prime q, arithmetic operation between elements in GF(q) is equivalent to the modulo q operation in the finite field. Below we give examples to explain our definitions of channel code and lattice code compatibility.

Example 1: A BPSK two-way relay system with linear channel code in GF(2) is channel code and lattice code compatible. This is because the linear combination of codewords is performed with Mod-2 addition, and the lattice decoding at the relay node can also be translated to Mod-2 addition. Therefore the linear combination of any pairs of codewords sent from the two source nodes can be uniquely recovered at the relay node when the channel is noise free.

Example 2: A 4-QAM two-way relay system with linear channel code in GF(4) is channel code and lattice code compatible. This is because lattice decoding of 4-QAM constellation signal is based on GF(2) in each of the two dimensions.

Example 3: A q-PAM two-way relay system with linear channel code in GF(q), where q is prime, is channel code and lattice code compatible. This is obvious according to condition 1.

Example 4: A $q^2-\text{QAM}$ two-way relay system with linear channel code in GF(q), where q is prime, is channel code and lattice code compatible. This is because lattice decoding of $q^2-\text{QAM}$ constellation signal is based on GF(q) in each of the two dimensions.

Example 5: A 4-PAM two-way relay system with linear channel code in GF(2) is not channel code and lattice code compatible. This is because a linear combination of codewords is performed with Mod-2 operation, but the lattice decoding at the relay node is performed with Mod-4 operation. Therefore after lattice decoding, the linear combination of codewords cannot be recovered.

Example 6: A 4-PAM two-way relay system with linear channel code in GF(4) is not channel code and lattice code compatible. This is because a linear combination of codewords is performed with GF(4) operation, but the lattice decoding at the relay node is performed with Mod-4 operation. Thus after lattice decoding, the linear combination of codewords cannot be recovered.

In our newly proposed LD-JCCNC and LDA-JCCNC schemes, we require that the selected channel codes and the lattice codes be compatible.

4.2.1 LD-JCCNC Scheme for q-PAM Two-way relay Systems

In this JCCNC scheme, we use unconventional q-PAM constellation schemes where q is a prime number. We use the same LDPC code on GF(q) for the two source nodes (same LDPC code means the same parity check matrix and generator matrix). The codeword element to PAM signal mapping uses $x = c - (q-1)/2$, where c represents codeword elements in GF(q). This constellation mapping guarantees that the channel code and lattice code are

compatible for the channel coded two-way relay systems. As an example, below we present the details of our decoding scheme on a 2-PAM (i.e., BPSK) two-way relay system.

Below we denote c_A and c_B as the codeword elements of source node A and source node B. The term c_{AB} is the linear addition of c_A and c_B on GF(q). And x_A and x_B are the q-PAM constellation mapping points of c_A and c_B. We also denote x_{AB} as the linear superposition of the two constellation symbols sent from the two source nodes, y_R as the signal received at the relay node including the noise, and $\bar{c}_{AB}$ is the codeword element recovered at the relay node after lattice decoding. The codeword element at the relay can be recovered by lattice decoding using the following equation, where round function $[\cdot]$ rounds the number to the nearest integer with half integers rounded upwards, and mod represents modulo operation:

$$\bar{c}_{AB} = [y_R + (q-1)] \bmod q \tag{4-15}$$

All the possible combinations of codeword elements and the corresponding signals are given below in Table 4-1 for a noise free two-way relay system. In the table, we can see that the codeword elements recovered at the relay after lattice decoding are exactly the same as the modulo addition of the two codeword elements from the two source nodes. Therefore this BPSK two-way relay system is channel code and lattice code compatible.

Table 4-1: Codeword elements and signal mappings of a noise free BPSK two-way relay system

i	c_A	c_B	c_{AB}	x_A	x_B	$x_{AB} = X_{AB}(i)$	$\bar{c}_{AB}$
0	0	0	0	-0.5	-0.5	-1	0
1	1	0	1	0.5	-0.5	0	1
2	0	1	1	-0.5	0.5	0	1
3	1	1	0	0.5	0.5	1	0

In this channel coded BPSK two-way relay system, we can assume all the $q^2 = 4$ signal combinations listed in Table 4-1 have the same probability of occurrence. If we denote the signal combinations as $X_{AB}(i), \quad i = 0,1,\ldots,q^2$, and the probability of their occurrences as $P\{X_{AB}(i)\}$, we have: $P\{X_{AB}(i)\} = 1/q^2$. If the noise at the relay node is zero mean Gaussian with variance σ_n^2, the probability for y_R given $X_{AB}(i)$ is sent can be written as:

$$P\{y_R \mid X_{AB}(i)\} = \frac{1}{\sigma_n \sqrt{2\pi}} \exp\left(\frac{-(y_R - X_{AB}(i))^2}{2\sigma_n^2} \right) \tag{4-16}$$

Thus from Bayes's Theorem, the probability that signal $X_{AB}(i)$ is sent given received signal y_R can be written as:

$$\begin{aligned} P_i &= P\{X_{AB}(i) \mid y_R\} \\ &= P\{y_R \mid X_{AB}(i)\} \frac{P\{X_{AB}(i)\}}{P\{y_R\}} = P\{y_R \mid X_{AB}(i)\} \frac{1}{B} \end{aligned} \tag{4-17}$$

where the constant $B = q^2 \cdot P\{y_R\}$ can be calculated to normalize the probabilities P_i so that their sum is 1.

In our proposed JCCNC scheme, the aim is to decode at the relay, the linear addition in GF(q) of the two codewords sent from the two source nodes. For the message passing algorithm used to decode the LDPC code, we can use the following options to obtain a soft-input for the decoder.

4.2.1.1 Standard XOR-JCCNC Scheme

For standard XOR-network coded JCCNC scheme presented in [2], [3] and [62], the a posteriori probability for each possible network coded symbol, is calculated using the a posteriori probabilities of all superimposed signal combinations which yield that code symbol when lattice decoding is applied to the superimposed signal combination. For instance, for the example given in Table 4-1, the soft input into the BP decoder is given as follows,

$$\begin{aligned} &P\{c_{AB} = 0 \mid y_R\} \\ &= P\{X_{AB}(0) \mid y_R\} + P\{X_{AB}(3) \mid y_R\} = P_0 + P_3 \end{aligned} \tag{4-18}$$

$$\begin{aligned} &P\{c_{AB} = 1 \mid y_R\} \\ &= P\{X_{AB}(1) \mid y_R\} + P\{X_{AB}(2) \mid y_R\} = P_1 + P_2 \end{aligned} \tag{4-19}$$

4.2.1.2 Proposed LD-JCCNC with Soft Information (Option-1)

This is one of the two decoded options of our new LD-JCCNC scheme. In this method, modulo operation is performed first for the received signal at the relay node to obtain $\hat{y}_R = y_R \bmod \Lambda$ so that $\hat{y}_R$ will be within the Voronoi region of the coarse lattice that defines the lattice code (i.e. the q-PAM constellation points). The $\bmod \Lambda$ operation can be performed in the following steps:

$$\hat{c}_{AB} = (y_R + (q-1)) \bmod q \tag{4-20}$$

After the above operation, $\hat{c}_{AB}$ will be in the range $0 \le \hat{c}_{AB} < q$. The following operation is performed to fold $\hat{c}_{AB}$ to within $0 \le \hat{c}_{AB} \le q-1$:

$$\hat{c}_{AB} = q - \hat{c}_{AB}, \quad if\ \hat{c}_{AB} > q - 0.5 \tag{4-21}$$

$$\hat{c}_{AB} = q - 2 + (q - \hat{c}_{AB}), \quad if\ q-1 \le \hat{c}_{AB} \le q - 0.5 \tag{4-22}$$

Then $\hat{c}_{AB}$ is mapped back to the signal space by:

$$\hat{y}_R = \hat{c}_{AB} - (q-1)/2 \tag{4-23}$$

Note that after the modulo operation, the AWG noise can still roughly be treated as normal distributed if the signal to noise ratio is high enough (e.g., SNR>2dB). Therefore, for the q possible values of codeword element combination, the a priori probabilities can be written as:

$$P\{\hat{y}_R \mid c_{AB} = c\} = \frac{1}{\sigma_n \sqrt{2\pi}} \exp\left(\frac{-(\hat{y}_R - x(c))^2}{2\sigma_n^2} \right) \tag{4-24}$$

where $x(c)$ is the constellation mapping point of $c \in GF(q)$. Thus from Bayes's Theorem, the soft input to the binary LDPC code decoder can be calculated by:

$$\begin{aligned} P\{c_{AB} = c \mid \hat{y}_R\} &= P\{\hat{y}_R \mid c_{AB} = c\} \frac{P\{c_{AB} = c\}}{P\{\hat{y}_R\}} \\ &= P\{\hat{y}_R \mid c_{AB} = c\} \frac{1}{B}; \quad c \in \mathrm{GF}(q) \end{aligned} \tag{4-25}$$

where the constant $B = q \cdot P\{\hat{y}_R\}$ can be calculated to normalize the probabilities so that their sum is 1. As useful information is lost after the modulo operation, we expect the performance of LD-JCCNC-Option-1 to be worse than the standard JCCNC scheme.

4.2.1.3 Proposed LD-JCCNC with Hard Decoding (Option-2)

This is the second option of our LD-JCCNC scheme, where lattice decoding is performed first at the relay node to obtain $\bar{c}_{AB} = \bar{c}$, where $\bar{c}$ is the decoded result. Then we use the following values as the soft input to the LDPC code decoder:

$$P\{\bar{c}_{AB} = \bar{c} \mid y_R\} = 1 \tag{4-26}$$

$$P\{\bar{c}_{AB} = c \mid y_R\} = 0, \quad c \in \{GF(q) \setminus \bar{c}\} \tag{4-27}$$

Since useful information will be lost with hard lattice decoding, we expect the performance of JCCNC-Option-2 will be worse than both the standard JCCNC-3 method and LDJCCNC-Option-1. However, this method is simpler and channel information σ_n^2 is not required,

which also means its decoding performance is not affected by misestimation of the channel information.

4.2.2 LD-JCCNC Scheme for q^2-QAM Two-way relay Systems

In this JCCNC scheme, we use unconventional $q^2-\text{QAM}$ constellation schemes where q is a prime number. We use the same LDPC code for the two source nodes (same LDPC code means the same parity check matrix and generator matrix). We have two options to choose the LDPC codes to be used: we can either use LDPC codes in GF(q) or use LDPC codes in GF(q^2). Due to the page limit, here we only describe the option of using LDPC codes in $GF(q^2)$.

The mapping between GF(q^2) symbols and q^2-QAM constellation points is performed in this way: first we convert a GF(q^2) symbol to a GF(q) symbol pair, denoted by $s_1 s_0$, then we use $(s_1-(q-1)/2,\ s_0-(q-1)/2)$ as its constellation mapping point. This constellation mapping guarantees that the channel code and lattice code are compatible for the channel coded two-way relay systems. As an example, below we present the details of our decoding scheme on a GF(4) channel coded 4-QAM two-way relay system. The mapping between GF(4) symbols and 4-QAM constellation points is given in Table 4-2.

Table 4-2: 4-QAM constellation mapping for GF(4) symbols

GF(4) symbol	GF(2) symbol pair	4-QAM constellation
0	00	(-0.5,-0.5)
1	01	(-0.5, 0.5)
2	10	(0.5, -0.5)
3	11	(0.5, 0.5)

Below we denote c_A and c_B as the codeword elements of source node A and source node B in GF(q^2) with $s_{A1}s_{A0}$ and $s_{B1}s_{B0}$ being their corresponding GF(q) symbol pairs. c_{AB} is the linear addition of c_A and c_B in GF(q^2). x_A and x_B are the q^2-QAM constellation mapping points of c_A and c_B. We also denote x_{AB} as the superposition of the two symbols sent from the two source nodes, and $\bar{c}_{AB}$ is the codeword element recovered at the relay node after lattice decoding with $\bar{s}_{AB1}\bar{s}_{AB0}$ being its corresponding symbol pair. The

codeword element at the relay can be recovered by lattice decoding using the following mapping equations, where round function $[\cdot]$ rounds the number to the nearest integer, $\mathrm{Re}(\cdot)$ function takes the real part of the complex number, $\mathrm{Im}(\cdot)$ function takes the imaginary part of the complex number, and mod represents modulo operation:

$$\bar{s}_{AB1} = [\mathrm{Re}(y_R) + (q-1)] \bmod q \tag{4-28}$$

$$\bar{s}_{AB0} = [\mathrm{Im}(y_R) + (q-1)] \bmod q \tag{4-29}$$

$$\bar{c}_{AB} = q \cdot \bar{s}_{AB1} + \bar{s}_{AB0} \tag{4-30}$$

To make it easy to fill out the table of signal combination and codeword addition, GF(4) addition table is given below in Table 4-3.

Table 4-3: GF(4) addition table

GF(4) addition	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

Based on Table 4-3 and the above lattice decoding rule, all the possible combinations of codeword elements and the corresponding signals are given below in Table 4-4 for a noise free two-way relay system.

Table 4-4: Codeword elements and signal mappings of a noise free 4-QAM two-way relay system

i	c_A	c_B	c_{AB}	x_A	x_B	$x_{AB} = X_{AB}(i)$	$\bar{s}_{AB1}\ \bar{s}_{AB0}$ / $\bar{c}_{AB}$
0	0	0	0	-0.5,-0.5	-0.5, -0.5	(-1, -1)	00/0
1	1	0	1	-0.5, 0.5	-0.5, -0.5	(-1, 0)	01/1
2	2	0	2	0.5, -0.5	-0.5, -0.5	(0, -1)	10/2
3	3	0	3	0.5, 0.5	-0.5, -0.5	(0, 0)	11/3
4	0	1	1	-0.5,-0.5	-0.5, 0.5	(-1, 0)	01/1
5	1	1	0	-0.5, 0.5	-0.5, 0.5	(-1, 1)	00/0
6	2	1	3	0.5, -0.5	-0.5, 0.5	(0, 0)	11/3
7	3	1	2	0.5, 0.5	-0.5, 0.5	(0, 1)	10/2
8	0	2	2	-0.5,-0.5	0.5, -0.5	(0, -1)	10/2
9	1	2	3	-0.5, 0.5	0.5, -0.5	(0, 0)	11/3
10	2	2	0	0.5, -0.5	0.5, -0.5	(1, -1)	00/0

11	3	2	1	0.5, 0.5	0.5, -0.5	(1, 0)	01/1
12	0	3	3	-0.5,-0.5	0.5, 0.5	(0, 0)	11/3
13	1	3	2	-0.5, 0.5	0.5, 0.5	(0, 1)	10/2
14	2	3	1	0.5, -0.5	0.5, 0.5	(1, 0)	01/1
15	3	3	0	0.5, 0.5	0.5, 0.5	(1, 1)	00/0

In this channel coded 4-QAM two-way relay system, we can assume all the 16 signal combinations listed in Table 4-4 have the same probability of occurrence. If we denote the signal combinations as $X_{AB}(i),\ \ i=0,1,..15$, and the probability of their occurrences as $P\{X_{AB}(i)\}$, we have: $P\{X_{AB}(i)\}=1/q^4=1/16$. If the noise at the relay node is zero mean complex Gaussian with variance σ_n^2, the probability for y_R given signal combination $X_{AB}(i)$ can be written as:

$$P\{y_R \mid X_{AB}(i)\}=\frac{1}{\pi\sigma_n^2}\exp\left(\frac{-\left|y_R-X_{AB}(i)\right|^2}{\sigma_n^2}\right) \tag{4-31}$$

Thus from Bayes' Theorem, the probability that signal $X_{AB}(i)$ is sent given received signal y_R can be written as:

$$\begin{aligned} P_i &= P\{X_{AB}(i) \mid y_R\} \\ &= P\{y_R \mid X_{AB}(i)\}\frac{P\{X_{AB}(i)\}}{P\{y_R\}} = P\{y_R \mid X_{AB}(i)\}\frac{1}{B} \end{aligned} \tag{4-32}$$

where the constant $B=q^4\cdot P\{y_R\}$ can be calculated to normalize the probabilities P_i so that their sum is 1. For the message passing algorithm used to decode the LDPC code, we can use the following options to obtain a soft-input for the decoder.

4.2.2.1 Standard JCCNC Scheme

In the standard JCCNC scheme presented in [56], the a posteriori probability for linear combination of transmitted codewords, is calculated using the a posteriori probabilities of all signal combinations which yield that codeword when lattice decoding is applied. For the example given in Table 4-4, the soft input into the BP decoder is given as follows,

$$P\{c_{AB}=0 \mid y_R\}=P_0+P_5+P_{10}+P_{15} \tag{4-33}$$

$$P\{c_{AB}=1 \mid y_R\}=P_1+P_4+P_{11}+P_{14} \tag{4-34}$$

$$P\{c_{AB}=2 \mid y_R\}=P_2+P_7+P_8+P_{13} \tag{4-35}$$

$$P\{c_{AB}=3 \mid y_R\}=P_3+P_6+P_9+P_{12} \tag{4-36}$$

4.2.2.2 LD-JCCNC with Soft Information (Option-1)

In this method, modulo operation is performed first for the received signal at the relay node to obtain $\hat{y}_R = y_R \bmod \Lambda$ so that $\hat{y}_R$ will be within the Voronoi region of the coarse lattice that defines the lattice code (i.e. the q^2-QAM constellation points). The $\bmod \Lambda$ operation can be performed in the following steps:

$$\hat{s}_{AB1} = (\mathrm{Re}(y_R) + (q-1)) \bmod q \tag{4-37}$$

$$\hat{s}_{AB0} = (\mathrm{Im}(y_R) + (q-1)) \bmod q \tag{4-38}$$

After the above operation, $\hat{s}_{AB0}$ and $\hat{s}_{AB1}$ will be in the range $0 \le \hat{s}_{AB0}, \hat{s}_{AB1} < q$. The following operation is performed to fold $\hat{s}_{AB0}$ and $\hat{s}_{AB1}$ to within $0 \le \hat{s}_{AB0}, \hat{s}_{AB1} \le q-1$:

$$\hat{s}_{ABi} = q - \hat{s}_{ABi}, \quad if\ \hat{s}_{ABi} > q-0.5; \quad i=0,1 \tag{4-39}$$

$$\hat{s}_{ABi} = q-2+(q-\hat{s}_{ABi}), \quad if\ q-0.5 \le \hat{s}_{ABi} < q; \quad i=0,1 \tag{4-40}$$

Then $\hat{s}_{AB0}$ and $\hat{s}_{AB1}$ are mapped back to the two dimensional signal space by

$$\hat{y}_R = (\hat{s}_{AB1} - (q-1)/2,\ \hat{s}_{AB0} - (q-1)/2) \tag{4-41}$$

Note that after the modulo operation, the AWG noise can still roughly be treated as normal distributed if the signal to noise ratio is high enough (e.g., SNR>2dB). Therefore, for the q^2 possible values of codeword element combination, the a priori probabilities can be written as:

$$P\{\hat{y}_R \mid c_{AB}=c\} = \frac{1}{\pi\sigma_n^2}\exp\left(\frac{-|\hat{y}_R - x(c)|^2}{\sigma_n^2}\right) \tag{4-42}$$

where $x(c)$ is the constellation mapping point of $c \in GF(q^2)$.

Thus from Bayes' Theorem, the soft input to the LDPC code decoder can be calculated by:

$$\begin{aligned} P\{c_{AB}=c \mid \hat{y}_R\} &= P\{\hat{y}_R \mid c_{AB}=c\}\frac{P\{c_{AB}=c\}}{P\{\hat{y}_R\}} \\ &= P\{\hat{y}_R \mid c_{AB}=c\}\frac{1}{B}; \quad c \in GF(q^2) \end{aligned} \tag{4-43}$$

where the constant $B = q^2 \cdot P\{y_R\}$ can be calculated to normalize the probabilities so that their sum is 1.

4.2.2.3 LD-JCCNC with Hard Decoding (Option-2)

In this method, lattice decoding is performed first at the relay node to obtain $\bar{c}_{AB} = \bar{c}$, where $\bar{c}$ is the decoding result. Then we use the following values as the soft input to the LDPC code decoder:

$$P\{\bar{c}_{AB} = \bar{c} \mid y_R\} = 1 \tag{4-44}$$

$$P\{\bar{c}_{AB} = c \mid y_R\} = 0, \quad c \in \{\mathrm{GF}(q^2) \setminus \bar{c}\} \tag{4-45}$$

4.2.3 Complexity Analysis

Here we present a complexity analysis of the two proposed LD-JCCNC schemes compared with the standard XOR-JCCNC scheme. Since the proposed LD-JCCNC schemes and standard XOR-JCCNC both employ non-binary LDPC codes in GF(q), BP algorithm with complexity O($q^2N\log N$) is used in all three JCCNC schemes. The difference in computational complexity between proposed LD-JCCNC schemes and the standard XOR-JCCNC, lies in the number of a priori probability calculations required to initialise the messages for the BP algorithm for decoding the network coded symbols. For each signal combination X_{AB}, the a priori probability calculation involves an exponentiation operation, such as equation (4-24) for PAM modulation and (4-42) for QAM modulation. For the standard XOR-JCCNC scheme with code over GF(q) employing q-PAM modulation, $2q$-1 number of a priori probability calculations are required for each codeword element, based on the *2q-1* unique signal combination values X_{AB}. For the standard XOR-JCCNC scheme with q^2-QAM modulation, $(2q\text{-}1)^2$ number of unique a priori probability calculations are required for the $(2q\text{-}1)^2$ unique signal combination values X_{AB} for each codeword element. Note that the relationship between the modulation order and the number of unique a priori probability calculations required for XOR-JCCNC over q-PAM and q^2-QAM modulations can be observed from the examples given in table 4-1 and table 4-4 respectively. It should be emphasised that the complexity on the number of a priori probabilities to be calculated only depends on the number of unique signal combinations values X_{AB} and not the total number of possible signal combinations.

For our proposed LD-JCCNC scheme with soft information (Option-1), a number of q such a priori probability calculations are required on the q-PAM modulation systems for each codeword element; and a number of q^2 such calculations are required on the q^2-QAM modulation systems for each codeword element. For our proposed LD-JCCNC with hard

decisions (Option-2), lattice decoding is used to determine the soft information into the BP decoder for the LDPC code. This means the $2q$-1 number of a priori probability calculations of equation (4-42) for the q-PAM scheme; and the $(2q\text{-}1)^2$ number of calculations of equation (4-24) for the q^2-QAM scheme, are not required. Instead, the soft-information input into the Belief propagation decoder is determined using efficient low complexity lattice decoder, that decodes the received signal to the nearest lattice point in a one dimension integer Z lattice or a two dimensional Z^2 lattice, for schemes over q-PAM and q^2-QAM respectively. Due to the efficient nature of lattice decoding, LD-JCCNC Option 2 benefits from a reduced complexity compared to XOR-JCCNC and LD-JCCNC Option 1. The complexity of the BP algorithm for non-binary LDPC code over GF(q) is O($q^2N\log N$) for codeword length of N.

4.2.4 Simulation Verification

To check the performance of our newly proposed LD-JCCNC scheme, simulations are performed for binary channel coded BPSK and GF(4) channel coded 4-QAM two-way relay systems. For the BPSK system the message block length is 500 bits and the codeword length is 1000 bits. For the 4-QAM system the message block length is 400 bits and the codeword length is 1200 bits. 20 iterations are performed by the BP decoding algorithm. The linear addition of the codewords sent from the two source nodes is decoded at the relay node. Simulations are also performed with the noise power being misestimated to see its effect on the performance. The results are shown below in Figure 4-3 through to Figure 4-7. From the results, it can be seen that when the noise power is perfectly known (in reality this is normally not possible due to the interference), the performance of our new LD-JCCNC schemes are worse than that of the standard JCCNC scheme. However, as discussed in Section 4.2.4, the complexity of the two LD-JCCNC schemes are lower than the standard JCCNC. When the noise power is misestimated, the performance of the XOR JCCNC scheme becomes much worse, but the performance of the LD-JCCNC scheme Option-1 is less affected and the performance of the LD-JCCNC scheme Option-2 is not affected at all.

Figure 4-4 shows that the performance of two LD-JCCNC schemes are much better than the standard JCCNC scheme for the BPSK system when the noise power is misestimated by 6 dB. Figure 4-7 shows that the performance of two LD-JCCNC scheme is much better than the standard JCCNC scheme for the 4-QAM system when the noise power is misestimated by 8 dB. This is consistent with theory, since LD-JCCNC option 2 does not require channel noise

information in its decoding of network coded symbols and is unaffected by misestimation of channel noise information.

It can be noted that the performance of LD-JCCNC Option-1 performs better than LD-JCCNC Option 2. This is logical since LD-JCCNC Option-1 initialises the Belief propagation decoder with q number of a priori probabilities corresponding to q constellation points, while Option 4 uses the hard decision from the latter decoding to initialise the Belief propagation decoding. Therefore, more useful information is lost in determining the probabilistic input for the BP decoder for JCCNC option 2.

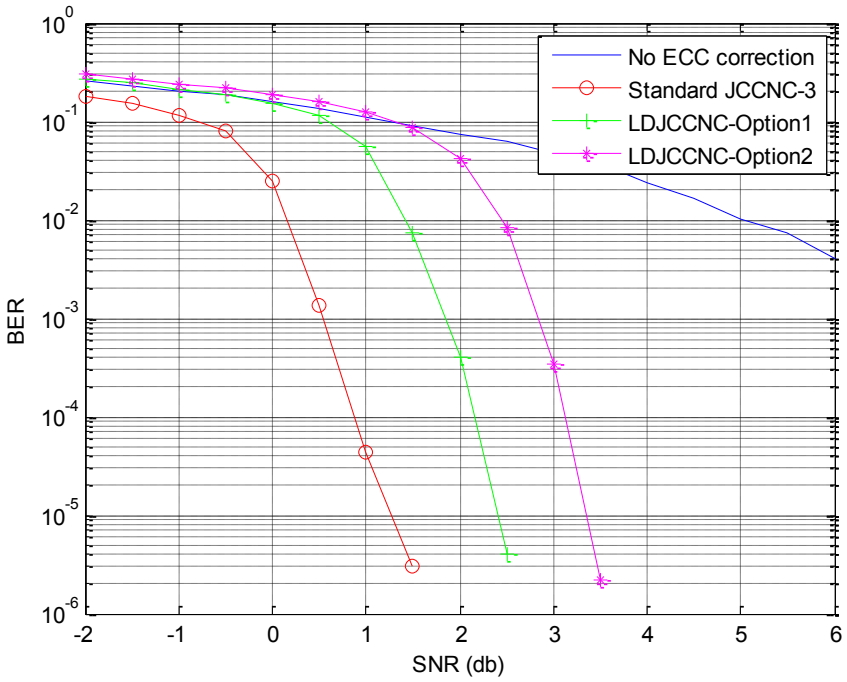


Figure 4-3: BPSK, BER performance with perfectly known channel information

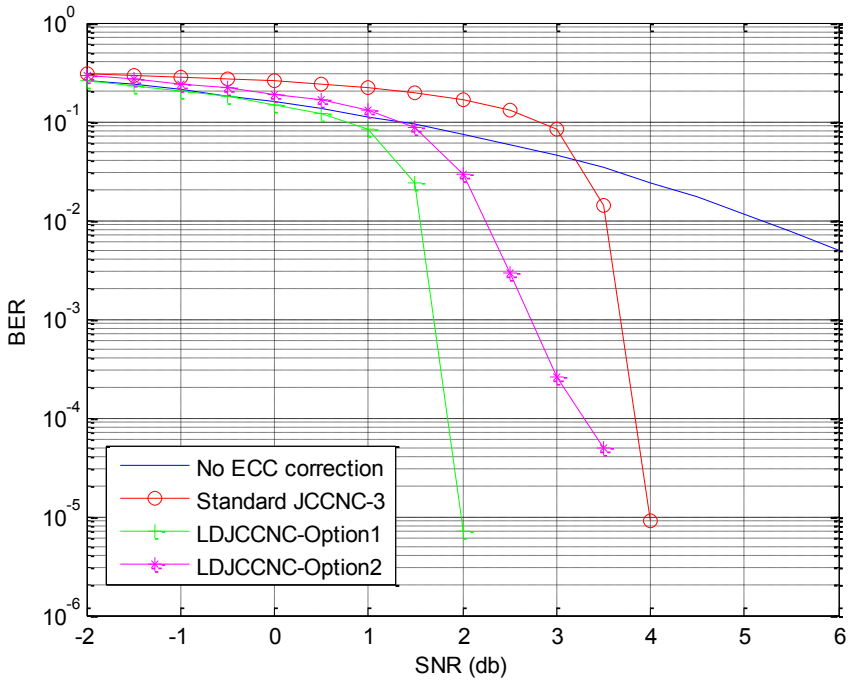


Figure 4-4: BPSK, BER performance with noise power misestimated by 6 dB

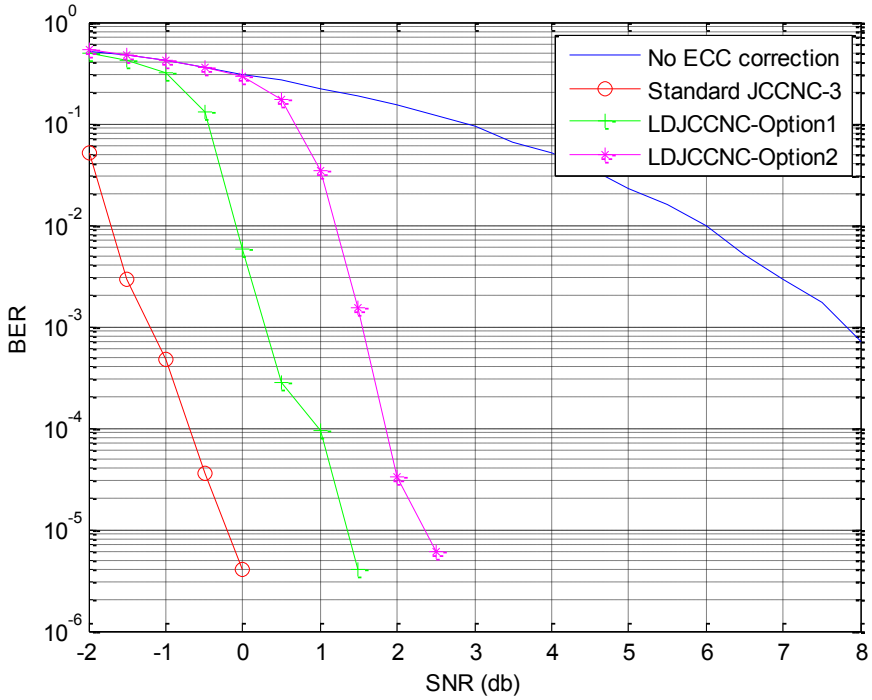


Figure 4-5: 4-QAM, BER performance with perfectly known channel information

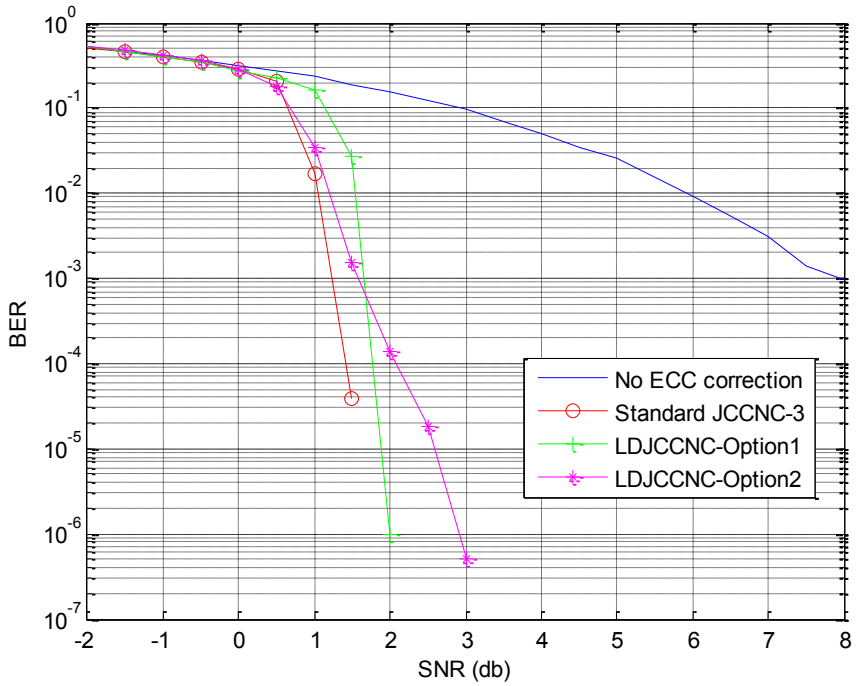


Figure 4-6: 4-QAM, BER performance with noise power misestimated by 6 dB

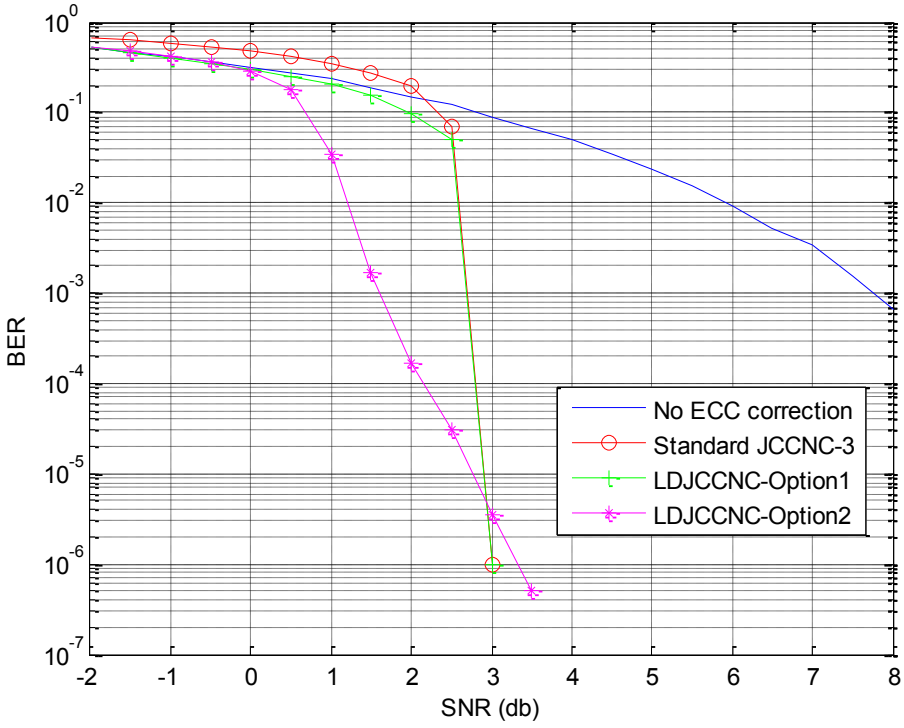


Figure 4-7: 4-QAM, BER performance with noise power misestimated by 8 dB\

4.2.5 Summary

In this section, we proposed a new joint channel coding network coding scheme for two-way relay systems based on lattice decoding. This scheme has a reduced complexity compared to the standard XOR channel coded network coding scheme for the two way relay and can be conveniently applied to general types PAM and QAM modulation systems. Compared with the existing channel coded PNC schemes for two-way relay, it has reasonably good decoding performance that is not affected by misestimate of channel information.

4.3 LDA-Aided JCCNC Scheme

In this section a new joint channel coded network coding (JCCNC) scheme is proposed based on low density construction A (LDA) lattices for the two-way relay channel, which we call LDA-lattice aided JCCNC (LDA-JCCNC). In LDA-JCCNC, the channel coding and network coding operations are integrated through the underlying LDA lattice. The arithmetic addition of the channel coded signals from the two source nodes is directly decoded using the LDA lattice decoding algorithm, and mapped to modulo addition codewords. It is shown that the proposed scheme has a lower complexity and yet performs almost on par with one of the best performing JCCNC schemes, the standard modulo sum JCCNC. We also give methodology on how our LDA-JCCNC scheme can be applied to PAM and QAM modulation systems, over Gaussian integers and Eisenstein integers, in contrast with most existing JCCNC schemes, which are restricted to set types of modulation frameworks. Note that although lattice encoding and decoding is applied at the user and relay nodes respectively, this JCCNC scheme is still considered a finite field channel coded PNC scheme, as there is a one to one mapping between the finite values in the lattice codebook employed and the PAM or QAM modulation symbols used. For our channel models, we assume that that perfect power control, precoding and synchronization are applied so that the channels between user nodes and relay have unit gain.

4.3.1 LDA Lattice

Error correction codes have been used to build good performance lattices via construction A or construction D [87][88]. The performance of the lattice built in this way is normally determined by the performance of the underlining error correction code. The LDA lattice has already been shown to have excellent performance [67]. However in this section, our main

interest is the decoding algorithm of the LDA lattices, as it is going to be reused in the decoding of the two-way relay systems.

In this section, we consider LDA lattices built from non-binary LDPC code in GF(p), where p is a prime [67][89]. Three types of LDA lattices are considered: integer LDA lattice, Gaussian integer lattice, and Eisenstein integer LDA lattice. The three types of LDA lattices can be expressed by:

$$\Lambda = \Phi(C) + pZ^n \tag{4-46}$$

$$\Lambda = \Phi(C) + \phi Z[i]^n \tag{4-47}$$

$$\Lambda = \Phi(C) + \pi Z[\omega]^n \tag{4-48}$$

where C is a LDPC code, p is a prime number, ϕ is a prime ideal of $Z[i]$ whose norm is a prime number, π is an Eisenstein prime whose norm is a prime number and also restricted to be congruent to 1 mod 3. Φ is the embedding of C into Z^n, $Z[i]^n$, or $Z[\omega]^n$ depending on the type of lattices.

Below we describe the decoding algorithm for integer LDA lattice, which can be easily generalised for the decoding of Gaussian integer LDA lattice and Eisenstein integer LDA lattice. The factor graph of an integer LDA lattice is shown below in Figure 4-8 over an AWGN channel, where $(c_1, c_2, \cdots, c_n) \in C$ is a codeword, $(z_1, z_2, \cdots, z_n)$ are integers, $(x_1, x_2, \cdots, x_n)$ is a lattice point, n_o is AWGN noise, and $(y_1, y_2, \cdots, y_n)$ are the noisy channel output.

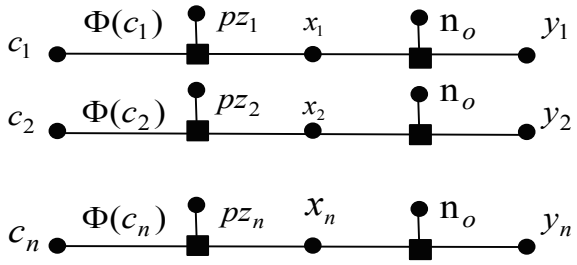


Figure 4-8: Factor graph of an integer LDA lattice

The constraints and relations are:

- $y_i \sim N(x_i, \sigma_n^2)$, $i = 1,2,\cdots,n$.
- $x_i = \Phi(c_i) + pz_i$, where $c_i \in \mathrm{GF}(p)$, $z_i \in Z$.
- The embedding Φ of GF(p) into the Euclidean space. The isomorphism mapping is simply defined as $\Phi(c_i) = c_i - (p-1)/2$ with $p \neq 2$.

For decoding, the maximum likelihood (ML) decoder gives the best performance. However for high dimensional LDA lattices this is impossible because of the high complexity. Fortunately a suboptimal algorithm, the BP decoding algorithm, which is used for LDPC code decoding, can also be used for LDA lattice decoding. Below are the steps of the decoding algorithm for integer LDA lattices:

- ***Initialisation***: First, determine a integer decoding set w_i which contains all the integers between the interval $[y_i - m\sigma_n, y_i + m\sigma_n]$. The value of parameter m should be large enough so that the probability of the transmitted x_i being outside of the interval is negligible. For example, with $m = 6.467$, this probability is less than 10^{-10}. Next, compute the a priori probability $P(y_i \mid x_i)$ using:

$$P(y_i \mid x_i) = \frac{1}{\sigma_n \sqrt{2\pi}} \exp(-\frac{(y_i - x_i)^2}{2\sigma_n^2}), \forall x_i \in w_i \tag{4-49}$$

  Then, compute the intrinsic a posteriori probability $P(c_i \mid y_i)$:

$$P(c_i \mid y_i) = \sum_{x_i \in Z \mid x_i \equiv c_1} P(y_i \mid x_i) \approx \sum_{x_i \in w_i \mid x_i \equiv c_1} P(y_i \mid x_i) \tag{4-50}$$

- ***Iteration*:** Apply the BP algorithm with input $P(c_i \mid y_i)$ to compute the extrinsic soft output $P(c_i \mid C, y \setminus y_i)$.
- ***Final decision***: For every $x_i \in w_i$, compute the a posteriori probability using:

$$APP(x_i) = P(y_i \mid x_i) \cdot \pi(x_i) \cdot P(c_i \mid C, y \setminus \{y_i\}) \tag{4-51}$$

  where $\pi(x_i) = 1/\mu_i(y_i, c_i)$ and $\mu_i(y_i, c_i)$ is the number of integers in w_i satisfying $x_i \equiv c_i \bmod p$. The x_i with the maximum $APP(x_i)$ value is the decoding result.

4.3.2 LDA-JCCNC Scheme for PAM and QAM Two-way Relay Systems

Our proposed LDA-JCCNC scheme works based on the following fundamental principles:

- With regard to a lattice, the arithmetic addition of two lattice points is still a lattice point [104].
- Given the construction of LDA lattices from LDPC code, we know that the arithmetic sum of two LDPC codeword elements (drawn from the same LDPC codebook) will be a lattice point in the LDA lattice constructed from that LDPC code.
- For mathematical operations over a Galois Field GF(p), the modulo addition of two LDPC codewords is still a codeword if p is a prime number

The above conditions match with the conditions of channel code and lattice code compatibility which we introduced in Section 4.2. Therefore, assuming message bits are encoded to LDA lattice codewords from the same lattice codebook at the user nodes, then the relay obtain the XOR of information bits $s_{\mathrm{AB}} = s_{\mathrm{A}} \oplus s_{\mathrm{B}}$ from $y = x_A + x_B + n_0$ through the use of LDA lattice decoding. This is possible since the LDA lattice code is compatible with the channel code, which means there exists a direct one to one PNC mapping between the output of the LDA decoder, the estimate of $x_{\mathrm{AB}} = x_{\mathrm{A}} + x_{\mathrm{B}}$, and the modulo sum of channel coded codewords, $c_{\mathrm{AB}} = c_{\mathrm{A}} \oplus c_{\mathrm{B}}$. The desired term $s_{\mathrm{AB}} = s_{\mathrm{A}} \oplus s_{\mathrm{B}}$, is naturally contained within the term $c_{\mathrm{A}} \oplus c_{\mathrm{B}}$ in the information bits portion of the codeblock. This scheme effectively performs both channel decoding and network coding through one single LDA decoding operation at the relay, as the PNC mapping from $x_{\mathrm{A}} + x_{\mathrm{B}}$ to $c_{\mathrm{A}} \oplus c_{\mathrm{B}}$ is a straightforward one using a simple lookup table. A schematic of the proposed LDA JCCNC is given in Figure 4-9.

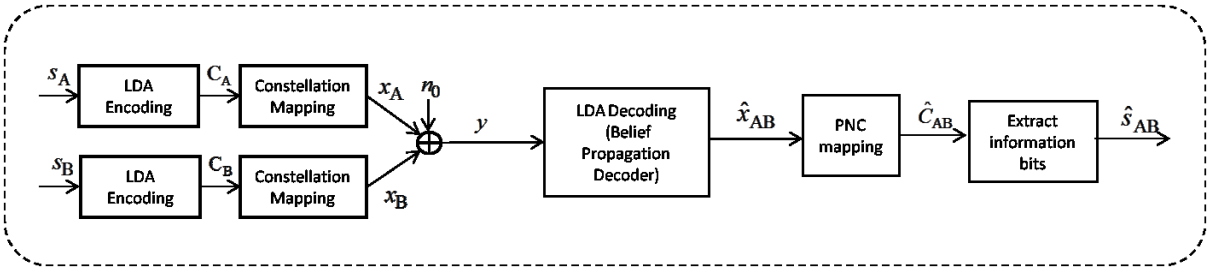


Figure 4-9: LDA lattice JCCNC scheme for the two-way relay

For performance comparison, we will compare our LDA-JCCNC scheme with the standard modulo-sum JCCNC scheme in section 4.2.1.1. For each case, unconventional p-PAM and p-QAM constellation schemes are used, where p is a prime number.

4.3.2.1 PAM and QAM Constellation Schemes

For the PAM constellation schemes to be used, the signal mapping is simply $x = c_i - (p-1)/2$, where c_i denotes a codeword element in GF(p). After PAM constellation mapping, a LDPC codeword is mapped to an integer LDA lattice point.

For the QAM constellation schemes to be used, the essential requirement is an isomorphism between the modulo operations in integer domain Z and complex number domain C. So far only two subrings of C, Gaussian integers and Eisenstein integers, have been identified to have the isomorphism.

Gaussian integers are defined to be the set $Z[i] = \{a + bi : a, b \in Z\}$. Eisenstein integers are defined to be the set $Z[\omega] = \{a + b\omega : a, b \in Z\}$ where $\omega = (-1 + i\sqrt{3})/2$. We use the representative sets of Gaussian integers and Eisenstein integers as the QAM constellation points. These sets of representatives are defined and determined by the prime ideals. The prime ideals of Gaussian integers are the Gaussian integers whose norms are prime numbers. For example, $1+2i$, $2+3i$, $4+5i$, $5+6i$ are all prime ideals of Gaussian integers, which give QAM constellation sizes of 5, 13, 41, and 61. The typical constellation shape over Gaussian integers is shown below in Figure 4-10, which is centred square.

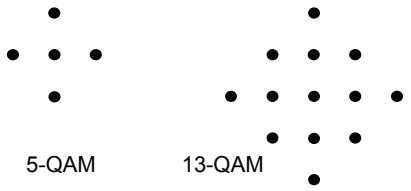


Figure 4-10: Constellation shape over Gaussian integers

The prime ideals of Eisenstein integers are the Eisenstein integers whose norms are prime numbers and congruent to 1 mod 3. For example, $1+3\omega$, $2+5\omega$, $3+7\omega$, $5+9\omega$ are prime ideals of Eisenstein integers, which give QAM constellation sizes of 7 , 19, 37, and 61. The typical constellation shape over Eisenstein integers is shown below in Figure 4-11, which is centred hexagonal.

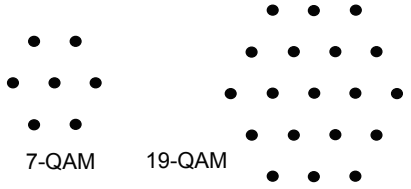


Figure 4-11: Constellation shape over Eisenstein integers

As examples, the details of the decoding methods are presented below for the two JCCNC schemes on 5-PAM and 5-QAM two-way relay system. In the following, we represent c_A and c_B as the codeword elements of source node A and source node B. $c_{AB} = c_A \oplus c_B$ is the mod-p addition of c_A and c_B.The term x_A and x_B are the p-PAM or p-QAM constellation mapping points of c_A and c_B. We also represent x_{AB} as the superposition (i.e., arithmetic addition) of the two symbols from the two source nodes, y_R as the signal received at the relay node including the noise, and $\bar{c}_{AB}$ as the codeword element decoded at the relay node.

The constellation mapping and all the possible combinations of codeword elements and the corresponding signals are given below in Table 4-5 and Table 4-6 respectively for 5-PAM and 5-QAM modulation schemes.

Table 4-5: Codeword elements and signal mappings of a 5-PAM two-way relay system

i	c_A	c_B	$c_{AB} = C_{AB}(i)$	x_A	x_B	$x_{AB} = X_{AB}(i)$
0	0	0	0	-2	-2	-4
1	1	0	1	-1	-2	-3
2	2	0	2	0	-2	-2
3	3	0	3	1	-2	-1
4	4	0	4	2	-2	0
5	0	1	1	-2	-1	-3
6	1	1	2	-1	-1	-2
7	2	1	3	0	-1	-1
8	3	1	4	1	-1	0
9	4	1	0	2	-1	1
10	0	2	2	-2	0	-2
11	1	2	3	-1	0	-1
12	2	2	4	0	0	0
13	3	2	0	1	0	1
14	4	2	1	2	0	2
15	0	3	3	-2	1	-1
16	1	3	4	-1	1	0
17	2	3	0	0	1	1
18	3	3	1	1	1	2
19	4	3	2	2	1	3
20	0	4	4	-2	2	0
21	1	4	0	-1	2	1
22	2	4	1	0	2	2
23	3	4	2	1	2	3
24	4	4	3	2	2	4

Table 4-6: Codeword elements and signal mappings of a 5-QAM two-way relay system

i	c_A	c_B	$c_{AB} = C_{AB}(i)$	x_A	x_B	$x_{AB} = X_{AB}(i)$
0	0	0	0	0, 0	0, 0	(0, 0)
1	1	0	1	1, 0	0, 0	(1, 0)
2	2	0	2	0, 1	0, 0	(0, 1)
3	3	0	3	0, -1	0, 0	(0, -1)
4	4	0	4	-1, 0	0, 0	(-1, 0)
5	0	1	1	0, 0	1, 0	(1, 0)
6	1	1	2	1, 0	1, 0	(2, 0)
7	2	1	3	0, 1	1, 0	(1, 1)
8	3	1	4	0, -1	1, 0	(1, -1)
9	4	1	0	-1, 0	1, 0	(0, 0)
10	0	2	2	0, 0	0, 1	(0, 1)
11	1	2	3	1, 0	0, 1	(1, 1)
12	2	2	4	0, 1	0, 1	(0, 2)
13	3	2	0	0, -1	0, 1	(0, 0)
14	4	2	1	-1, 0	0, 1	(-1, 1)
15	0	3	3	0, 0	0, -1	(0, -1)
16	1	3	4	1, 0	0, -1	(1, -1)
17	2	3	0	0, 1	0, -1	(0, 0)
18	3	3	1	0, -1	0, -1	(0, -2)
19	4	3	2	-1, 0	0, -1	(-1, -1)
20	0	4	4	0, 0	-1, 0	(-1, 0)
21	1	4	0	1, 0	-1, 0	(0, 0)
22	2	4	1	0, 1	-1, 0	(-1, 1)
23	3	4	2	0, -1	-1, 0	(-1, -1)
24	4	4	3	-1, 0	-1, 0	(-2, 0)

In this channel coded 5-PAM and 5-QAM two-way relay system, it can be assumed that all the $p^2 = 25$ signal combinations listed in Table 4-5 and Table 4-6 have the same probability of occurrence.

4.3.2.2 The Standard JCCNC Scheme

The standard JCCNC scheme has been described in [92] as JCCNC-3. For the 5-PAM and 5-QAM constellation schemes, we have the soft input to the LDPC code decoder given by the following equations based on Table 4-5 or Table 4-6:

$$P\{c_{AB}=0 \mid y_R\} = P_0 + P_9 + P_{13} + P_{17} + P_{21} \tag{4-52}$$

$$P\{c_{AB}=1 \mid y_R\} = P_1 + P_5 + P_{14} + P_{18} + P_{22} \tag{4-53}$$

$$P\{c_{AB}=2 \mid y_R\} = P_2 + P_6 + P_{10} + P_{19} + P_{23} \tag{4-54}$$

$$P\{c_{AB}=3 \mid y_R\} = P_3 + P_7 + P_{11} + P_{15} + P_{24} \tag{4-55}$$

$$P\{c_{AB}=4 \mid y_R\} = P_4 + P_8 + P_{12} + P_{16} + P_{20} \tag{4-56}$$

4.3.2.3 LDA Aided JCCNC Scheme

Based on the analysis in sections 4.3.1 and 4.3.2.1, and the fundamental principles given at the start of section 4.3.2, the arithmetic addition x_{AB} of the signals from the two source nodes can be decoded directly from the noisy signal y_R by using the LDA lattice decoding algorithm described in section 4.3.1. After the decoding, c_{AB} can be obtained simply by using the isomorphism mapping between c_{AB} and x_{AB}.

4.3.3 Complexity Analysis

For LDA-JCCNC, the modulations are designed using prime integers over the ring of Gaussian integers and Eisenstein integers. It can be shown that the constellations designed from Gaussian prime integers are of size given by $p=2n(n+1)+1$, where n is a positive integer greater than 1. While for LDA-JCCNC over Eisenstein integer modulation, the modulation orders which can be used are given by $p=3n(n+1)+1$.

The difference between the computational complexity of the standard XOR-JCCNC scheme and our proposed LDA-JCCNC scheme lies in the number of a priori probability calculations required for the signal combination X_{AB} at the relay. For each signal combination, the probability calculation involves an exponentiation operation. For the XOR-JCCNC scheme with p-PAM modulation, $2p$-1 unique a priori probability calculations must be made. For the standard XOR-JCCNC scheme with p-QAM modulation, we know that the modulation order p determines whether the constellation should be designed from Gaussian integers or Eisenstein integers. For standard XOR-JCCNC with p-QAM modulation designed

over the ring of Gaussian integers, it can be derived that $p=4n(2n+1)+1$ number of a priori probability calculations must be made. For standard XOR-JCCNC with p-QAM designed over the ring of Eisenstein integers, the number of a priori probability calculations to be performed for XOR-JCCNC is $p=6n(2n+1)+1$. For LDA-JCCNC scheme, the number of such calculations given in equation (4-49) is determined by the number of integers (or Gaussian integers and Eisenstein integers) within the decoding range. From equation (4-50), it can be seen that the calculation of intrinsic a posteriori probability of network coded symbols is based on modulo operation. Therefore only the p integer (or Gaussian integer or Eisenstein integer) points around the received signal provide useful a priori information for the decoder. Thus, for PAM modulation, the computational complexity of LDA-JCCNC scheme is about a half of that of the standard JCCNC scheme with regard to calculating the a priori probability for the decoder. For QAM modulation, the complexity reduction of LDA-JCCNC scheme from the standard JCCNC scheme is even greater, which can be seen below in Table 4-7 where a number of typical constellations are listed.

Table 4-7: Complexity comparison

QAM constellation points	**Number of calculations for the a priori probability**	
	Standard JCCNC-3	***LDA-JCCNC***
5	13	5
7	19	7
13	41	13
19	61	19
25	85	25
37	127	37
41	145	41
61	217	61

In summary, the computational complexity of LDA-JCCNC scheme is lower than that of the standard XOR-JCCNC scheme.

4.3.4 Simulation Verification

To check the performance of the proposed LDA-JCCNC schemes, simulations are performed for GF(5) channel coded 5-PAM and 5-QAM two-way relay systems with message block length of 500 and codeword length of 1000. 40 iterations are performed by the BP decoding algorithm. The results are shown below in Figure 4-12 and Figure 4-13 for 5-PAM and 5-QAM respectively.

From the simulation results, it can be seen that on the 5-PAM, the performance of LDA-JCCNC scheme is about 0.3dB worse than the standard JCCNC scheme. On the 5-QAM, the difference is about 1.2dB.

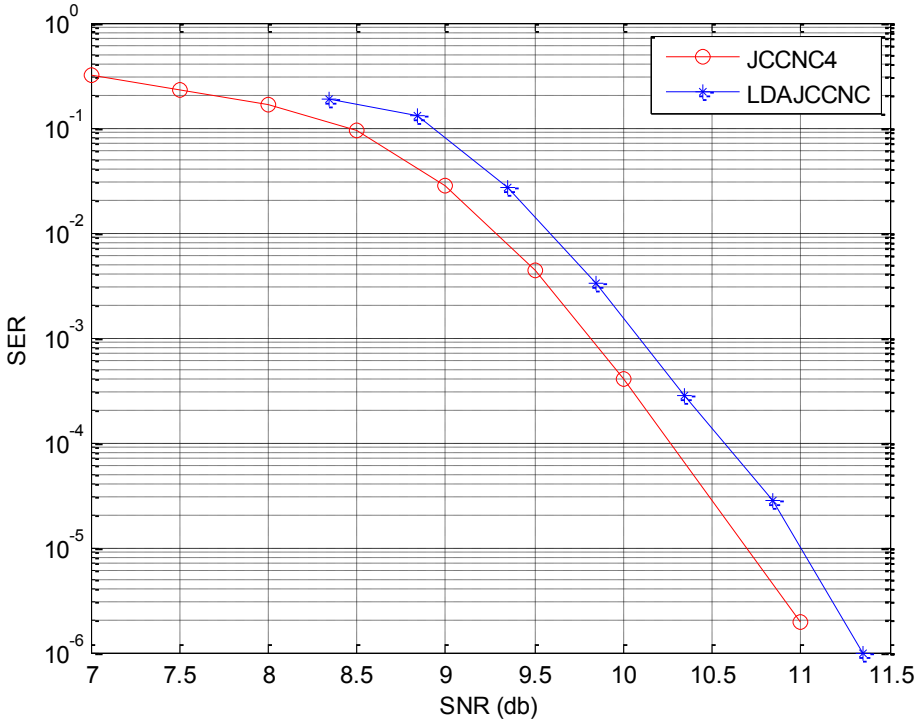


Figure 4-12: SER performance on 5-PAM

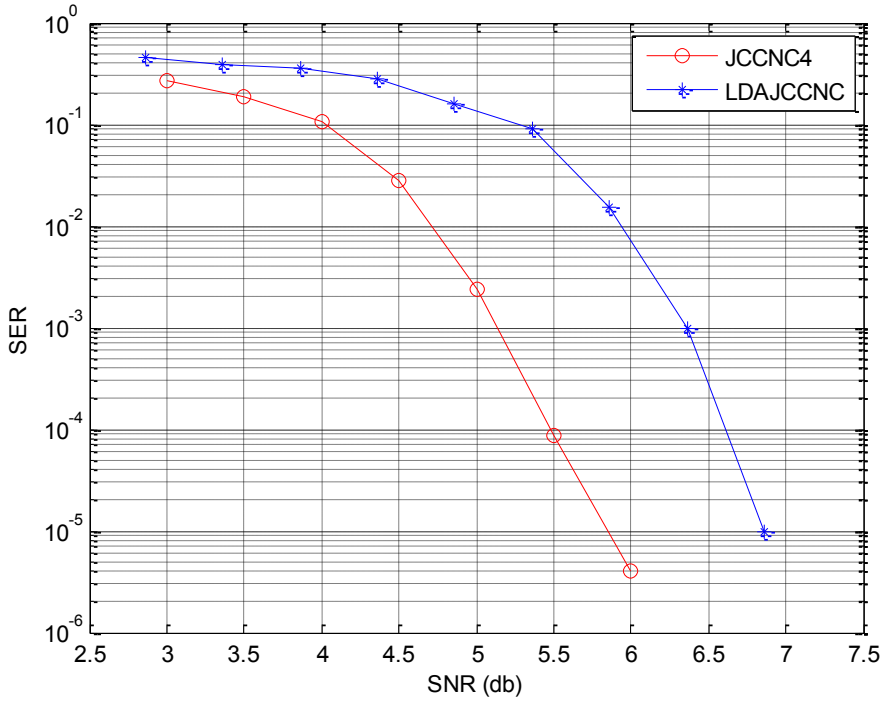


Figure 4-13: SER performance on 5-QAM

Looking at the BP decoding algorithm for LDA lattice, it can be understood why the LDA-JCCNC scheme has worse performance. This is because in the LDA decoding algorithm, the intrinsic information fed to the BP decoder is calculated in (4-51) on several points around the received signal based on mod-p operation.

4.3.5 Summary

In this section, a new form of integrated JCCNC scheme is proposed based on LDA lattices. A modulation methodology is provided over complex number domains that allow the standard JCCNC scheme and our new LDA-JCCNC scheme to be used with another two types of general QAM modulation systems. Our new LDA-JCCNC scheme shows competent performance in comparison with the standard JCCNC scheme, while operating at lower computational complexity cost.

4.4 Polar Lattice Aided JCCNC Scheme

Before we discuss our proposed channel coded network coding scheme with polar lattice, we provide a brief overview on the construction and decoding of both the polar lattice and its underlying code, the polar code.

4.4.1 Polar Code

Introduced by Arıkan in 2009, polar codes provide the first deterministic construction of capacity-achieving codes for binary memoryless symmetric (BMS) channels [94]. A brief but detailed review of polar codes has been given in [95]. In polar codes, rather than attacking the coding problem directly, by certain splitting and combining operations, the polarization approach follows an information-theoretic route to manipulate N identical independent copies of binary-input channel W to form a new set of binary-input channels, which have capacities either near 0 or near 1, except for a fraction that vanishes as N becomes large. After such polarized channels are obtained, polar encoding consists of transmitting information at full rate over channels that are near perfect (used channels) and fixing the inputs of the remaining channels (unused channels) to zero. It was shown that polar codes constructed in this way could achieve capacity with encoding and decoding methods of complexity $O(N \log N)$.

The basic binary polar code is a linear code defined for any block length $N = 2^n$ in terms of a generator matrix,

$$\mathbf{G}_N = \mathbf{F}^{\otimes n}, \ \mathbf{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \tag{4-57}$$

where $\mathbf{F}^{\otimes n}$ represents the nth Kronecker power of $\mathbf{F}$. In polar coding, one encodes a data block $\mathbf{u} = (u_1, u_2, \cdots, u_N)$ into a codeword $\mathbf{x} = (x_1, x_2, \cdots, x_N)$ via transformation $\mathbf{x} = \mathbf{u}\mathbf{G}_N$. For a rate K/N polar code, $N - K$ elements of $\mathbf{u}$ are fixed to zero and the rest K elements are the information bits to be encoded. This effectively reduces $\mathbf{G}_N$ to a $K \times N$ matrix. For example, for a $(N, K) = (8, 5)$ polar code, you may fix u_2, u_4, u_6 to zero and obtain the 5×8 generator matrix

$$\mathbf{G}_{5,8} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

from $\mathbf{G}_8 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$

The problem of polar code design is to determine which set of $N-K$ elements to freeze in order to achieve the best possible performance under Successive-Cancellation (SC) decoding on a given channel. It turns out that the solution to this problem is channel dependent. Therefore, in general there is no fixed set of elements that are guaranteed to work well for all channels of a given capacity.

A full solution for polar code design problem is given in papers [98]-[100] with complexity $O(N)$. With this development, polar codes became the first provably capacity-achieving class of codes with polynomial-time complexity algorithms for code construction, encoding, and decoding.

At current stage, the performance of finite length polar codes is not as good as the state-of-the-art due to the suboptimal nature of the standard SC decoding algorithm and the relatively weak minimum distance properties of these codes. However, polar codes continue to attract interests for potential applications for a few reasons. Firstly, while the performance of the basic polar code is not as good as the state-of-the-art at short to practical block length, they are still good enough to be a viable alternative with enhancements. Secondly, polar codes have the unique property that their code rate can be adjusted from 0 to 1 without changing the encoder and decoder. Thirdly, polar codes have a recursive structure, which makes them highly suitable for implementation in hardware.

For polar code decoding, in addition to SC decoder, belief propagation (BP) decoder produces similar performance but has the potential to achieve higher throughputs [96]. So far the best performance decoder is list decoder with CRC [97], which achieves near ML performance.

4.4.2 Polar Lattice

Polar lattice was proposed by Yan and Ling [101][102]. Polar lattices are generated by applying Construction D lattice construction to a set of nested polar codes. Since polar

lattices are as explicit as polar codes, their construction is equally efficient. Compared with the other existing lattices, polar lattices have the properties of AWGN-goodness and low complexity, namely, the sphere bound can be achieved asymptotically with multistage decoding.

As described in [105], polar codes are subcodes of the full Reed-Muller code. Therefore the way of construction of Barnes-Wall lattices can be followed to construct lattices from polar codes. If $\mathbf{G}$ denotes the generator matrix of the full Reed-Muller code RM(n,n), where the dimension of the generator matrix is $N = 2^n$, the generator matrix of RM(r,n) is selected from the rows of $\mathbf{G}$ with Hamming weights $\geq 2^{n-r}$, while the generator matrix of polar code Polar(r,n) is selected according to channel polarization. The difference in the selection of rows is the main reason why polar codes perform better than Reed-Muller codes.

The construction of polar lattices is given in [68][101][102], which follows the multilevel construction of AWGN-good lattices described in [106]. Let $\Lambda_1/\Lambda_2 \cdots \Lambda_r/\Lambda_{r+1}$ be an n-dimensional lattice partition chain. In the multilevel construction of lattices, for each partition $\Lambda_k/\Lambda_{k+1}, (1 \leq k \leq r)$, a code C_k over Λ_k/Λ_{k+1} selects a sequence of coset representatives a_k for the cosets of Λ_{k+1}. The symbols from each partition are summed to form a multilevel code. Given r levels of polar codes are used, then the set of nested binary polar codes for each level is represented by C_k, for $1 \leq k \leq r$, where $C_1 \subseteq C_2 \subseteq \cdots \subseteq C_r$.

So far only binary partition chains are used to form polar lattices. The one-dimensional lattice partition chain $\mathrm{Z}/2\mathrm{Z}/\cdots/2^r\mathrm{Z}$ can be used to form one-dimensional polar lattices, while the two-dimensional lattice partition chain $\mathrm{Z}^2/R\mathrm{Z}^2/\cdots/R^r\mathrm{Z}^2$ can be used to form two-dimensional polar lattices, where,

$$R = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

is a rotation operator . As explicitly outlined in [68], to construct a multilevel polar lattice, a set of nested binary polar codes are required to design the coded levels, in addition to an uncoded level designed from basic integer lattice. Given C_k has block length N with l_k number of information bits for $1 \leq k \leq r$. Let $\mathrm{g}_1, \mathrm{g}_2, \ldots \mathrm{g}_N$ be a basis such that $\mathrm{g}_1, \mathrm{g}_2, \ldots \mathrm{g}_{l_k}$ span C_k . The one-dimensional polar lattice L constructed from r coded levels can be represented by,

$$L = \left\{ \sum_{k=1}^{r} 2^{k-1} \sum_{i=1}^{l_k} u_k^i \mathrm{g}_i \ + 2^r \mathrm{Z}^N \mid \ u_k^i \in \{0,1\} \right\} \tag{4-58}$$

This multilevel lattice coding scheme is essentially a coset coding scheme, where the information bits is encoded to a lattice point in a cosets of $2^r \mathrm{Z}^N$. In a practical system with power constraint, we can choose the coset leaders as the codebook for transmission. For example, given a polar lattice with 2 coded levels (r=2) and an uncoded 4Z integer lattice, the cosets $4\mathrm{Z} + 1$ is the same as $4\mathrm{Z} + 4001$, and both yield the same result if we apply cubic shaping to any lattice point in the two cosets via modulo 4 operation. Therefore naturally we can choose coset representatives $\{0,1,2,3\}$ as codewords for use with a 4-PAM modulation scheme. Note the subtle but important difference between the multilevel coset coding lattice scheme and standard multilevel codes. For the same r=2 example, for multilevel codes, the encoded information for each polar code level is strictly binary, so that the encoded point is directly the set $\{0,1,2,3\}$. The encoded result is exactly the codeword and not a coset. This type of coding does not share a lattice structure.

Returning to our multilevel polar lattice coding scheme, for the case where r is an even number, the two-dimensional polar lattice L can be presented by,

$$L = \left\{ \sum_{k=1}^{r} 2^{k-1} \sum_{i=1}^{l_k} u_k^i \mathrm{g}_i \otimes R^{k-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \left(2^{r/2} Z^2\right)^N \mid \ u_k^i \in \{0,1\} \right\} \tag{4-59}$$

There are several key considerations in the design of polar lattices, in particular the number of component polar code levels, and the capacity of each level according to the target error probability for a given noise variance. Once we know which levels are required and the capacity for each level, we can construct polar codes to achieve capacity for each level. As shown in [68], the $\mathrm{Z}/2\mathrm{Z}$ channel is in fact a binary modulo 2 AWGN, while $2\mathrm{Z}/4\mathrm{Z}$ and $4\mathrm{Z}/8\mathrm{Z}$ channels are also variations of the modulo 2 AWGN channel with noise variance reduced by a factor of 4 each time we move to a higher partition level. Let's define $f(x)$ as the probability density function of Gaussian noise, $x \in \mathrm{R}^n$,

$$f_{\sigma^2}(x) = (2\pi\sigma^2)^{\frac{n}{2}} e^{-\frac{\|x\|^2}{\sigma^2}}, \tag{4-60}$$

After mod 2 operation, the PDF of the noise becomes,

$$f_{2\mathrm{Z},\sigma^2}(x') = \sum_{i\in 2\mathrm{Z}} f_{\sigma^2}(x'+i), \qquad x \in [1,-1] \tag{4-61}$$

The capacity of mod- Λ channel is given as,

$$\mathrm{C}(\Lambda_1,\sigma^2) = \log V(\Lambda) - h(\Lambda_1,\sigma^2), \tag{4-62}$$

where $V(\Lambda)$ denotes the Voronoi region of Λ and $h(\Lambda_1,\sigma^2)$ is the differential entropy of the mod Λ aliased noise,

$$h(\Lambda_1,\sigma^2) = -\int_{V(\Lambda)} f_{\Lambda,\sigma^2}(x') \log f_{\Lambda,\sigma^2}(x')dx \tag{4-63}$$

The capacity of the $\mathrm{C}(\mathrm{Z}/2\mathrm{Z})$ is given as,

$$\mathrm{C}(\mathrm{Z}/2\mathrm{Z}) = \mathrm{C}(2\mathrm{Z},\sigma^2) - \mathrm{C}(\mathrm{Z},\sigma^2). \tag{4-64}$$

The channel capacity of the modulo 2 Binary AWGN channel is calculated for each lattice partition level and plotted in Figure 4-14 .

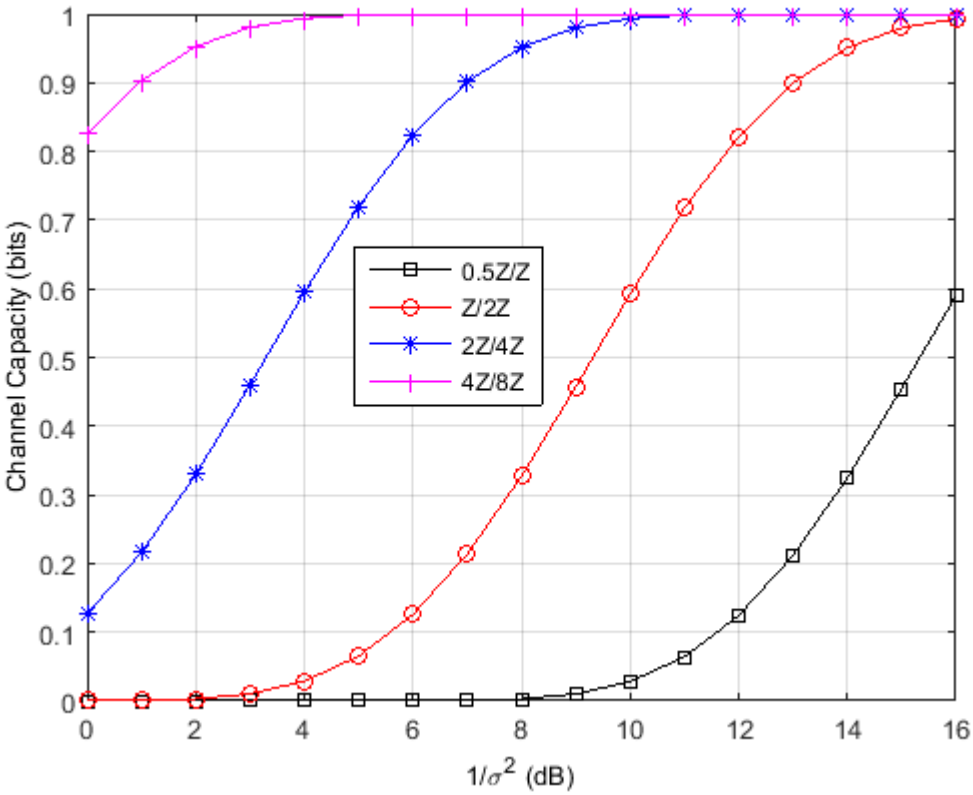


Figure 4-14: Channel capacity for each level of partition chains

From the capacity plot, the number of required levels can be selected using σ as the reference parameter. This is because for a target error probability, σ can be calculated for each level and fully represents the channel. In [68] it is shown that the levels to be selected are those that can achieve target error probability for a reference σ, with a rate not too close to 0 or 1. Once we determined which partition levels will be used, the aim is then to find for each of these binary mod 2 channels, a polar code with rate that approaches the calculated capacity at target error probability.

As polar lattice has multilevel construction, multistage decoder is used for its decoding, where either BP decoder or SC decoder can be used for decoding on each level [68][101][102]. Since polar lattice uses multistage decoding, the overall error probability of $P_e(L, \sigma^2)$ of a polar lattice is upperbounded by the sum of error probability at individual levels. Denote the block error of polar code for each coded level as $P_B(C_l, \sigma^2)$ for $1 \le l \le r$, and denote the block error of the uncoded Z^N lattice level as $P_B(\Lambda_{r+1}^N, \sigma^2)$, then,

$$P_e(L,\sigma^2) \le \sum_{l=1}^{r} P_B(C_l,\sigma^2) + P_B(\Lambda_{r+1}^N,\sigma^2), \tag{4-65}$$

where the error probability of Λ_{r+1}^N depends on the PDF of the noise over its Voronoi region $\nu(\Lambda_{r+1}^N)$,

$$P_B(\Lambda_{r+1}^N,\sigma^2) = 1 - \int_{\nu(\Lambda_{r+1}^N)} f_{\sigma^2}(x)\,dx \tag{4-66}$$

The overall decoding complexity of the multi-level polar lattice, can be given by $O((r+1)N\log N)$ where r is the number of coded levels. To design the component polar codes, the equal-error-probability rule [108] is used. For a polar lattice constructed from r polar code levels and a single uncoded integer lattice level, if the total target error probability is P_e, then the target error probability of each level is $P_e/(r+1)$. The work in this thesis focuses on the application of readily designed polar lattices to network coding. For more details on how to design capacity achieving polar lattices the reader is referred to [68]. Figure 4-15 below shows an encoding and decoding chain for a one-dimensional two-level polar lattice.

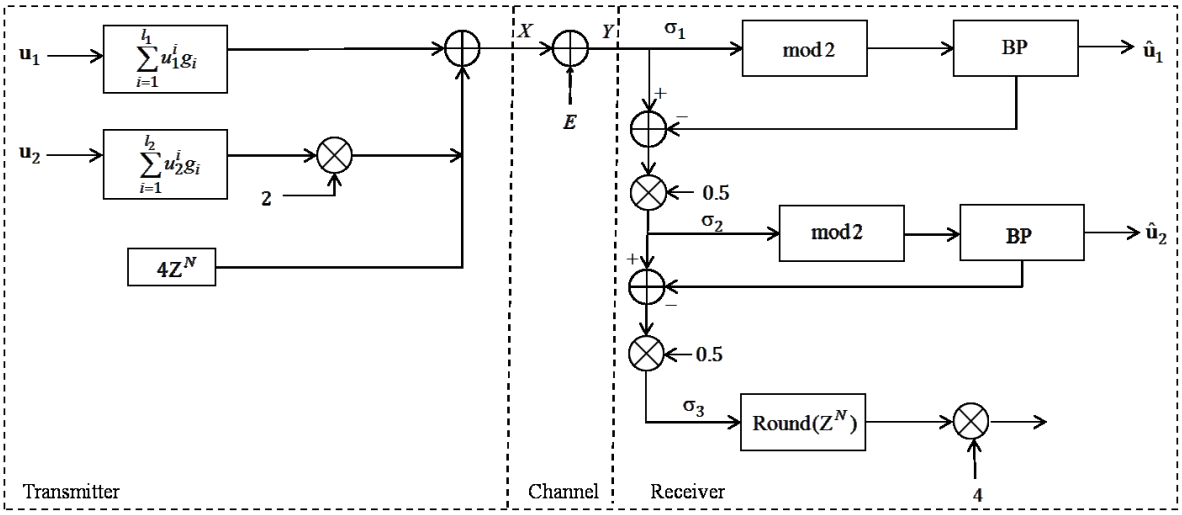


Figure 4-15: Encoding and decoding chain for a one-dimensional two-level polar lattice

Figure 4-16 below shows an encoding and decoding chain for a two-dimensional four-level polar lattice.

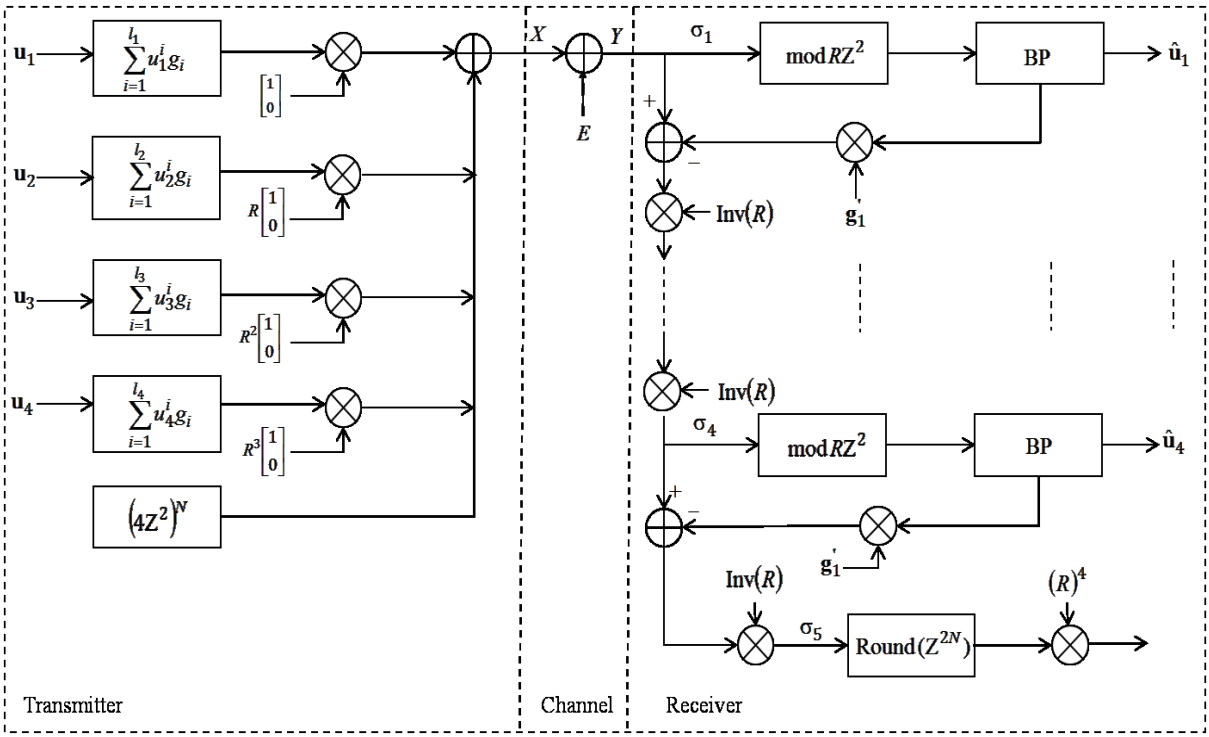


Figure 4-16: Encoding and decoding chain for a two-dimensional four-level polar lattice

In the following section, we apply polar lattice codes to two-way relay system and propose the polar lattice aided JCCNC (PLA-JCCNC) scheme.

4.4.3 PLA-JCCNC Scheme for PAM and QAM Two-way Relay Systems

One feature of polar lattices is that if we assume basic cubic shaping on the lattice, they can be used with PAM and square QAM constellation schemes. While binary polar codes can only be applied to BPSK and QPSK modulation on relay systems due to the issue of channel code and lattice code compatibility, polar lattices constructed from multilevel polar codes can be designed to be used in conjunction with higher order modulation systems. In this section, we will explain how one-dimensional polar lattice codes can be used to form PAM constellation schemes and two-dimensional polar lattice codes can be used to form square QAM constellation schemes

Our proposed PLA-JCCNC scheme for two-way relay is based on the same coset encoding principles of the polar lattice. First we design the polar lattice based on a target error probability by selecting the required number of polar code levels and then constructing

polar codes to achieve the capacity for each level. Assuming the same pre-designed polar lattice is used at each node, information bits at each user node are first encoded to a lattice point in a coset of the basic integer lattice, denoted as $2^r\mathrm{Z}^N$ in (4-58). Next we apply cubic shaping with regards to the integer lattice on the encoded lattice point to obtain a codeword in the codebook selected for transmission Note that this method is equivalent to encoding the information bits to a coset representative of each coset of $2^r\mathrm{Z}^N$. The encoded codeword from each user is then mapped to a modulation symbol before transmission over wireless channel.

At the relay node the polar lattice decoder is employed to decode a lattice codeword representing the linear combination of the transmitted polar lattice codewords from the two users. The polar lattice decoder naturally performs channel decoding to decode a function of transmitted information bits for each polar code level. The channel decoded information bits are then re-encoded to form the decoded lattice codeword representing the linear combination of transmitted lattice codewords. The decoded lattice codeword is broadcasted to the two source nodes during the broadcast phase, where each user decodes the transmitted lattice codeword and subtracts its own original lattice codeword to retrieve the other user's lattice codeword. Since the information bits and lattice codewords are at one-to-one mapping, the decoded information bits of the other user can be easily retrieved. An example of our proposed PLA-JCCNC scheme using two-level one-dimensional polar lattices is shown in Figure 4-17.

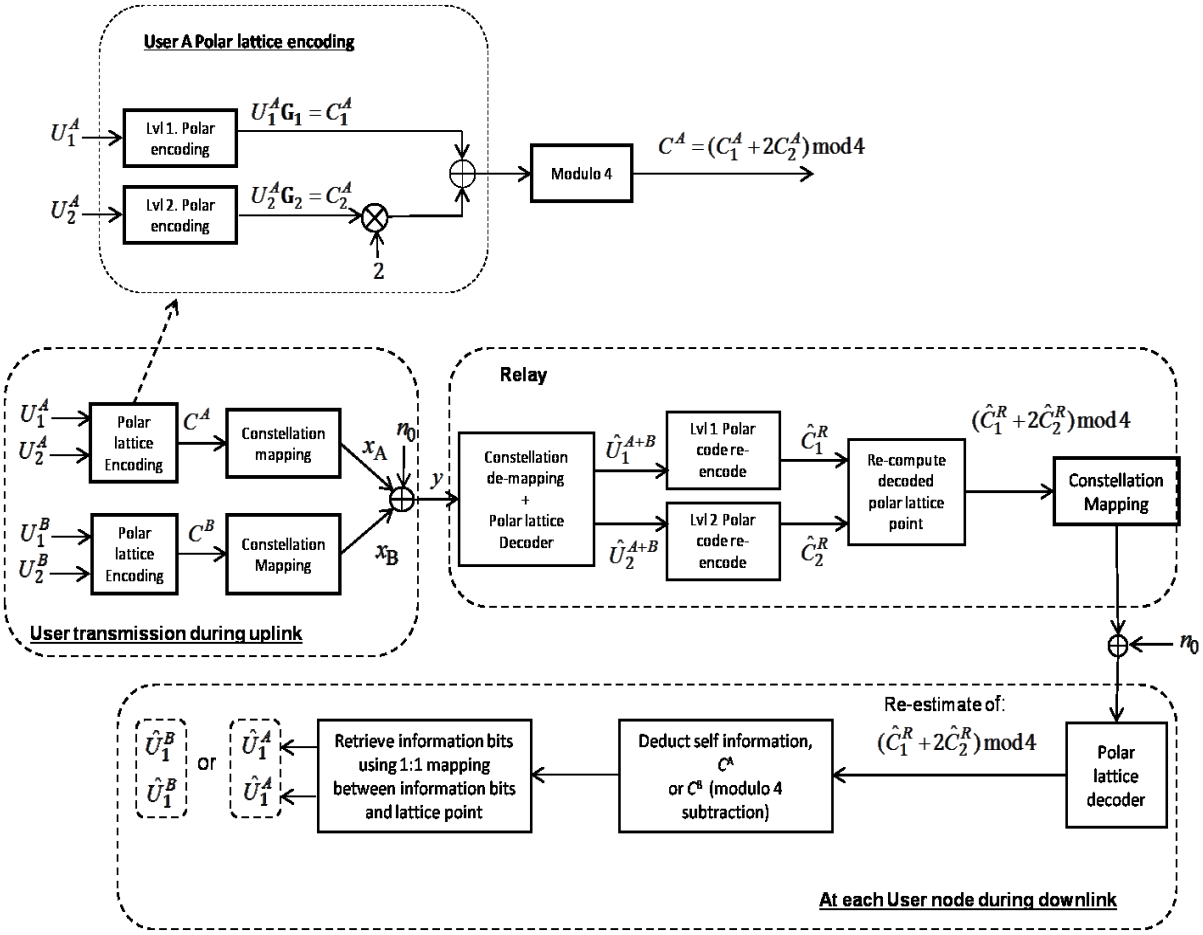


Figure 4-17: Example of polar lattice aided JCCNC scheme for two-way relay channel

4.4.3.1 PLA-JCCNC Scheme on PAM Constellation Systems

It can be envisaged that PAM constellations can be designed from one-dimensional polar lattices, such as the one outlined in Figure 4-15. In this section, we describe how to apply the PLA-JCCNC scheme to 2^r-PAM two-way relay systems, where r is the number of polar code levels used for the polar lattice.

In the PLA-JCCNC for two-way relay, both users employ the same polar lattice code and therefore the same component polar codes. Denote U_k^l as the vector containing the information bits of user l for the level k polar code of the polar lattice, and define $\mathbf{G}_k$ as the generator matrix of level k polar code. Let $U_k^l\mathbf{G}_k = C_k^l$, $(1 \le k \le r)$ denote the generated term by the k-th level polar code sent by user l. Taking into account all r component code levels of the polar lattice, the polar lattice codeword transmitted from user l is therefore $C^l = \left(C_1^l + 2C_2^l + \cdots + 2^{r-1}C_r^l\right) \bmod 2^r$.

To map the lattice codeword to constellation, in general, 2^r-PAM constellation can be formed by a one-dimensional r-level polar lattice codeword via simple shifting, $\left(C_1^l + 2C_2^l + \cdots\right.$

$+2^{r-1}C_r^l)\text{mod}2^r - (2^r - 1)/2$. For example, the 4-PAM constellation can be formed by a one-dimensional two-level polar lattice code through the process $(C_1 + 2C_2)\text{mod}4 - 3/2$. The 2^r-PAM signals transmitted from user A and user B can be represented by X_A and X_B respectively:

$$X_A = (C_1^A + 2C_2^A + \cdots + 2^{r-1}C_r^A)\text{mod}2^r - (2^r - 1)/2 \quad (4\text{-}67)$$

$$X_B = (C_1^B + 2C_2^B + \cdots + 2^{r-1}C_r^B)\text{mod}2^r - (2^r - 1)/2 \quad (4\text{-}68)$$

The signal received at the relay node is given by:

$$\begin{aligned} Y_R &= X_A + X_B + z_R \\ &= \left(C_1^A + 2C_2^A + \cdots + 2^{r-1}C_r^A\right)\text{mod}2^r + (C_1^B + 2C_2^B + \cdots + 2^{r-1}C_r^B)\text{mod}2^r - (2^r - 1) + z_R \end{aligned}$$

Since the received signal is a noisy linear combination of modulation symbols, it must be mapped back to lattice domain before lattice decoding can be performed. This de-mapping is done by simply shifting the modulation signal by $2^r - 1$. Then, the remaining term is modulo-2^r operated to obtain a lattice point in the original codebook. Given the addition property of modular arithmetic, we can re-write the modulo-operated terms as,

$$\begin{aligned} &\left(\left(C_1^A + 2C_2^A + \cdots + 2^{r-1}C_r^A\right)\text{mod}2^r + (C_1^B + 2C_2^B + \cdots + 2^{r-1}C_r^B)\text{mod}2^r + z_R\text{mod}2^r\right)\text{mod}2^r \\ &= \left(C_1^A + C_1^B + 2\left(C_2^A + C_2^B\right) + \cdots 2^{r-1}(C_r^A + C_r^B) + \text{z}_R\right)\text{mod}2^r \\ &= (C_1^R + 2C_2^R + \cdots 2^{r-1}C_r^R + z_R)\text{mod}2^r \end{aligned}$$

where z_R denotes the noise vector. The shifted and modulo operated signal is then fed into the polar lattice decoder, which is a multistage successive cancellation decoder with BP decoding performed for each level of component polar codes. The aim of the lattice decoder at the relay node is to decode for each of the r component polar code levels, a binary representative of the component $C_k^A + C_k^B$, $(1 \le k \le r)$ which we shall denote by $\widehat{U}_r^{A+B}$. The decoded information for each level can be used to re-compute the decoded polar lattice point that represents the linear combination of transmitted polar lattice points. The decoding steps are the same as those of the polar lattice decoder as shown in Figure 4-15 except modulo operation with regards to cubic shaping is initially performed on the received signal before it is fed into the polar lattice decoder. The decoding steps for PLA-JCCNC at the relay can be summarized as below:

1. Add $2^r - 1$ to the received signal Y_R
2. Perform mod2^r operation (i.e., Z-lattice code decoding) on the received signal at the relay node: $Y = (Y_R + 2^r - 1)\text{mod}2^r$
3. Perform modulo 2 operation on signal Y then apply BP decoder to decode binary information bits representing the level 1 component $C_1^A + C_1^B$, which we shall denote

by $\widehat{U}_1^{A+B}$ for level-1 polar code with BP decoder. The decoded term contributed by level-1 polar code is given by $\hat{C}_1^R = \widehat{U}_1^{A+B}\mathbf{G}_1$.

4. Using successive cancellation, deduct the decoded term $\hat{C}_1^R = \widehat{U}_1^{A+B}\mathbf{G}_1$ from signal *Y*. Then scale the signal by 0.5. This part can be viewed as the reverse of steps took to construct the lattice. Next, feed the signal into BP decoder to decode the information bits $\widehat{U}_2^{A+B}$ for level-2 polar code. The decoded information bits are the binary representative of the level 2 component $C_2^A + C_2^B$. The decoded term for this polar code level is therefore $\hat{C}_2^R = \widehat{U}_2^{A+B}\mathbf{G}_2$.
5. Continue with successive cancellation and BP decoding until level-*r* information bits $\widehat{U}_r^{A+B}$ have been decoded.
6. Re-compute the decoded polar lattice point using $\left(\widehat{U}_1^{A+B}\mathbf{G}_1 + 2\widehat{U}_2^{A+B}\mathbf{G}_2 + \cdots 2^{r-1}\widehat{U}_r^{A+B}\mathbf{G}_r\right)\mathrm{mod}2^r$. The decoded information bits are therefore re-encoded through the reapplication of polar lattice encoding, and ready for the broadcast phase.

4.4.3.2 PLA-JCCNC Scheme on QAM Constellation Systems

QAM constellations can be naturally designed from the two-dimensional polar lattices outlined in Figure 4-16. In this section, we describe how to adapt our PLA-JCCNC scheme to 2^r-square QAM two-way relay systems, where *r* is the number of coded levels in the lattice. Note that in this case *r* must be an even positive number. Denote U_k^l as the vector containing the information bits of user *l* for the level *k* polar code, and $\mathbf{G}_k$ is the generator matrix of level *k* polar code. Let $U_k^l\mathbf{G}_k = C_k^l$, $(1 \le k \le r)$ denote the generated term by the *k*-th level polar code of the polar lattice point sent by user *l*. The two-dimensional polar lattice codewords transmitted from user *l* can be expressed as,

$$C^l = \left[C_1^l \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2^l \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r^l \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2}.$$

Note that the Kronecker product generates two rows of data from the original polar encoded data, which representing the values in the two dimensions of the lattice. The summation between matrix terms is done on a row by row basis so that elements of the same dimension are added together. Finally the $\bmod 2^{r/2}$ operation is done on an element by element basis. Therefore C^l is a $2 \times N$ matrix with the two rows containing the values in two dimensions of *N* number of lattice points. In general a square 2^r-QAM constellation can be naturally formed by a two-dimensional *r*-level polar lattice code, through the application of cubic shaping via modulo operation, and then shifting the lattice code to center it around the zero point,

$$\left[C_1 \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2 \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2} - \begin{bmatrix}(2^{r/2}-1)/2\\(2^{r/2}-1)/2\end{bmatrix},$$

Note the above encoding method produces two streams of data, for the real and imaginary components of QAM symbols. For example, the 4-QAM constellation can be formed by a two-dimensional two-level polar lattice code,

$$\left[C_1 \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2 \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2 - \begin{bmatrix}1/2\\1/2\end{bmatrix},,$$

Therefore the square 2^r-QAM signals transmitted from user-*A* and user-*B* can be represented by X_A and X_B respectively, each containing two parallel streams of data,

$$X_A = \left[C_1^A \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2^A \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r^A \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2} - \begin{bmatrix}(2^{r/2}-1)/2\\(2^{r/2}-1)/2\end{bmatrix} \quad (4\text{-}69)$$

$$X_B = \left[C_1^B \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2^B \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r^B \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2} - \begin{bmatrix}(2^{r/2}-1)/2\\(2^{r/2}-1)/2\end{bmatrix} \quad (4\text{-}70)$$

The signal received at the relay node is given by:

$$Y_R = X_A + X_B + z_R \quad (4\text{-}71)$$

where z_R denotes the noise vector on both dimensions. As with the case for PAM modulation, the relay will first adjust the signal to move it from constellation domain to lattice domain. The two rows of signal data are combined into one complex valued data stream and fed into the BP decoder. The aim of the polar lattice decoder at the relay node is to decode for each polar code level r, a binary representative of the component $C_k^A + C_k^B$, for $(1 \le k \le r)$, which we shall denote by $\widehat{U}_r^{A+B}$. The decoded information for each polar code level can then be used to re-compute the overall decoded polar lattice term. The decoding steps are almost the same as those of the polar lattice decoder as shown in Figure 4-16 except modulo operation is performed with regards to the cubic shaping, on the received signal before input into the polar lattice decoder. The output of each polar code level from the BP decoder is finally used to re-compute the decoded polar lattice points. The decoding steps is summarized in detail below:

1. Add $2^{r/2} - 1$ to received signal Y_R, in both dimension.
2. Perform modulo $2^{r/2}$ operation in both dimensions, effectively decoding to the cubic shaping region of the Z^2 lattice:

$$Y = \left[Y_R - \begin{bmatrix}2^{r/2}-1\\2^{r/2}-1\end{bmatrix}\right] \bmod 2^{r/2}$$

3. Perform modulo $R\mathrm{Z}^2$ operation on signal Y and feed signal into BP decoder to decode information bits representing the level 1 component $C_1^A + C_1^B$, denoted by $\widehat{U}_1^{A+B}$ for level-1 polar code
4. Using successive cancellation, deduct the decoded term of level 1 polar code $\hat{C}_1^R = \widehat{U}_1^{A+B}\mathbf{G}_1$ from signal Y. Multiply by R^{-1} and input signal into BP decoder to decode the information bits $\widehat{U}_2^{A+B}$ for level 2 polar code, where the decoded information bits are the binary representative of the level 2 component $C_2^A + C_2^B$. The decoded term for this polar code level is therefore $\hat{C}_2^R = \widehat{U}_2^{A+B}\mathbf{G}_2$.
5. Continue with successive cancellation and BP decoding until level-r information bits $\widehat{U}_r^{A+B}$ has been decoded.
6. Re-compute the decoded polar lattice point $\hat{C}^R$ using,

$$\hat{C}^R = \left[\hat{C}_1^R \otimes \begin{bmatrix}1\\0\end{bmatrix} + \hat{C}_2^R \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] \cdots + \hat{C}_r^R \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2},$$

Note that this re-computation regenerates a $2 \times N$ matrix of decoded lattice points, the two rows carrying information for the two dimensions of the lattice. This process can be viewed as re-application of channel code for the broadcast phase.

4.4.4 Simulation Verification

In this section, we check the performance of our PLA-JCCNC scheme on PAM and QAM two-way relay systems. A 4-PAM constellation example and a 16-QAM constellation example are used. In the 4-PAM and 16-QAM two-way relay systems, we use the one-dimensional and two-dimensional polar lattice design examples given in [68] to form the 4-PAM and 16-QAM systems. Polar code length of 1024 is used for both examples. The decoding performance is measured at the relay and block length error rate is calculated. At the relay node, the error rate probability is calculated based on the error criteria on the transmitted polar lattice point. For one-dimensional polar lattice, assuming that user A and user B each selects a lattice codewords $C^A = \left(C_1^A + 2C_2^A + \cdots + 2^{r-1}C_r^A\right) \bmod 2^r$ and $C^B = (C_1^B + 2C_2^B + \cdots + 2^{r-1}C_r^B) \bmod 2^r$ respectively, the error decision criteria is set as $P_e = \Pr\left(C^R \neq \hat{C}^R\right)$ where $C^R = (C^A + C^B) \bmod 2^r$, and $\hat{C}^R = \left(\hat{C}_1^R + 2\hat{C}_2^R \cdots + 2^{r-1}\hat{C}_r^R\right) \bmod 2^r$, where the estimates $\hat{C}_k^R$, $(1 \leq k \leq r)$, are the decoded outputs of the BP decoder for each level of polar codes at the relay. For two-dimensional polar lattice, assuming that user A and user B each selects lattice codewords,

$$C^A = \left[C_1^A \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2^A \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r^A \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2}$$

$$C^B = \left[C_1^B \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2^B \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + \cdots C_r^B \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2}$$

the error decision criteria is set as $P_e = \Pr(C^R \neq \hat{C}^R)$ where $C^R = (C^A + C^B) \bmod 2^{r/2}$, and

$$\hat{C}^R = \left[\hat{C}_1^R \otimes \begin{bmatrix}1\\0\end{bmatrix} + \hat{C}_2^R \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] \cdots + \hat{C}_r^R \otimes \left[R^{r-1}\begin{bmatrix}1\\0\end{bmatrix}\right]\right] \bmod 2^{r/2},$$

where the estimates $\hat{C}_k^R$, $(1 \leq k \leq r)$, are the decoded outputs of the BP decoder for each level of polar codes at the relay. Note since C^A and C^Beach consists of two rows of data for values in the two dimensions of the lattice. As a result, the addition $C^A + C^B$ refers to separate addition between the components of each dimension for C^A and C^B.

For the two-level one-dimensional polar lattice, the target error probability is chosen as $P_e(L, \tilde{\sigma}^2) = 10^{-5}$. The bottom level is a base Z^N lattice decoder, therefore we select $\sigma_3 = 0.0845$ for a target error probability of $\frac{1}{3} \cdot 10^{-5}$, Then for the middle level, we select $\sigma_2 = 2\sigma_3 = 0.1690$, which gives a channel capacity of $\mathrm{C}(\mathrm{Z}/2\mathrm{Z}, \sigma_2) = 0.9874$ for the middle level. Finally we select $\sigma_1 = 2\sigma_2 = 0.3380$ for a channel capacity of $\mathrm{C}(2\mathrm{Z}/4\mathrm{Z}, \sigma_2) = 0.5145$ for the top level. The aim is then to find the two polar codes with rates close to the two calculated capacities at each level with block error probabilities less than $\frac{1}{3} \cdot 10^{-5}$ over each binary input mod-2 AWGN channels.

With $N = 1024$, for $P_e(C_1, \sigma_1^2) \approx \frac{1}{3} \cdot 10^{-5}$ the level-1 polar code obtained has rate $\frac{K_1}{N} = 0.23$, while the level-2 polar code has a rate of $\frac{K_2}{N} = 0.9$ for $P_e(C_2, \sigma_2^2) \approx \frac{1}{3} \cdot 10^{-5}$. The sum rate of the component polar codes is, $R_C = 0.23 + 0.9$, giving an average code rate of 0.5650. To compare with the standard JCCNC scheme with LDPC code rate of 0.5, we adjust the two-level component polar code rates to be $\frac{K_1}{N} = 0.22$ and $\frac{K_2}{N} = 0.78$. For the four level two-dimensional polar lattice,

$$L = C_1 \otimes \begin{bmatrix}1\\0\end{bmatrix} + C_2 \otimes \left[R\begin{bmatrix}1\\0\end{bmatrix}\right] + C_3 \otimes \left[R^2\begin{bmatrix}1\\0\end{bmatrix} + C_4 \otimes \left[R^3\begin{bmatrix}1\\0\end{bmatrix}\right]\right] + (4\mathrm{Z}^2)^N \qquad (4\text{-}72)$$

$$= C_1 \otimes \mathbf{g_1} + C_2 \otimes \mathbf{g_2} + C_3 \otimes \mathbf{g_3} + C_4 \otimes \mathbf{g_4} + (4\mathrm{Z}^2)^N$$

where $\mathbf{g}_1' = (1,0)$, $\mathbf{g}_2' = (1,1)$, $\mathbf{g}_3' = (2,0)$, $\mathbf{g}_4' = (2,2)$ each level is equivalent to a binary-input mod$R\mathrm{Z}^2$ channel with $\sigma_k = \sqrt{2}\sigma_{k+1}$ for $1 \leq k \leq 4$. At the decoder, the channel likelihood ratio is calculated as part of the coset decoding problem. The probability density function of the modRZ^2 operated Gaussian noise is given by:

$$f_{\tilde{\sigma},RZ^2}(w) = \sum_{\lambda \in RZ^2} f_{\tilde{\sigma}}(w+\lambda), \quad w \in \Re(RZ^2) \tag{4-73}$$

The likelihood ratio is hence:

$$\mathrm{LR} = \frac{f_{\tilde{\sigma},RZ^2}(w)}{f_{\tilde{\sigma},RZ^2}(w-(1,0))} \tag{4-74}$$

With an overall target error probability $P_e(L,\tilde{\sigma}^2) = 10^{-5}$, we select $\sigma_5 \approx 0.083$ to ensure the error probability of the bottom level is $P_e(L,\tilde{\sigma}^2) = \frac{1}{5}\cdot 10^{-5}$. Then we select $\sigma_4 = \sqrt{2}\sigma_5 = 0.117$, $\sigma_3 = \sqrt{2}\sigma_4 = 0.166$, $\sigma_2 = \sqrt{2}\sigma_3 = 0.235$, $\sigma_1 = \sqrt{2}\sigma_2 = 0.332$. The component code rates to obtain error probability of $\frac{1}{5}\cdot 10^{-5}$ at each level are given by 0.07, 0.40, 0.825, and 0.981 respectively, which gives an average code rate of 0.569.

In the simulation, block error rate at the relay node is counted in this way: if there is a decoding error for any component code in one block transmission, a block error is counted. In Figure 4-18 the BLER performance of 4-PAM PLA-JCCNC is compared with standard JCCNC 3-PAM system and standard JCCNC 5-PAM system.

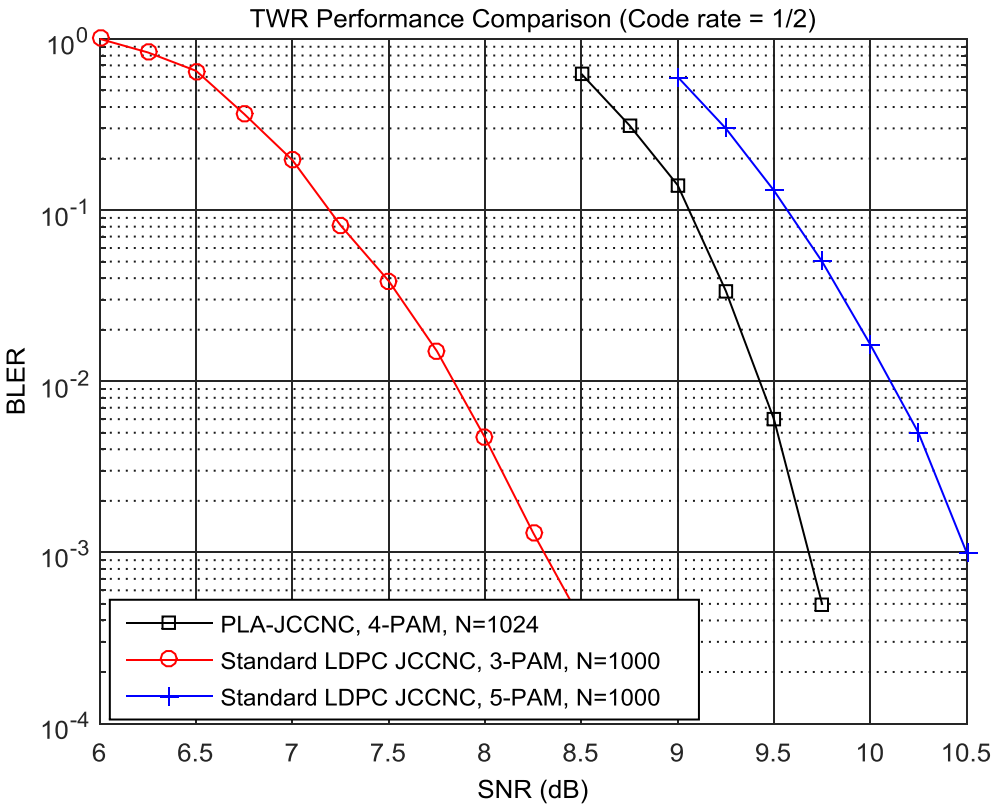


Figure 4-18: BLER performance of 4-PAM PLA-JCCNC two-way relay system

Since standard JCCNC schemes with LDPC codes requires the code to be drawn from prime finite field, 3-PAM and 5-PAM standard JCCNC schemes are the closest fair modulation schemes that we can use for comparison with our 4-PAM PLA-JCCNC. From Figure 4-18, it can be seen that our PLA-JCCNC scheme has a similar error rate performance as the standard LDPC JCCNC scheme. Any difference in error rates between the 3 schemes is largely down to a larger or smaller constellation alphabet being used for the PLA-JCCNC scheme.

The BLER performance of 16-QAM PLA-JCCNC two-way relay system is shown in Figure 4-19. It can be seen that the performance is very good and consistent with that of the 4-PAM PLA-JCCNC two-way relay system.

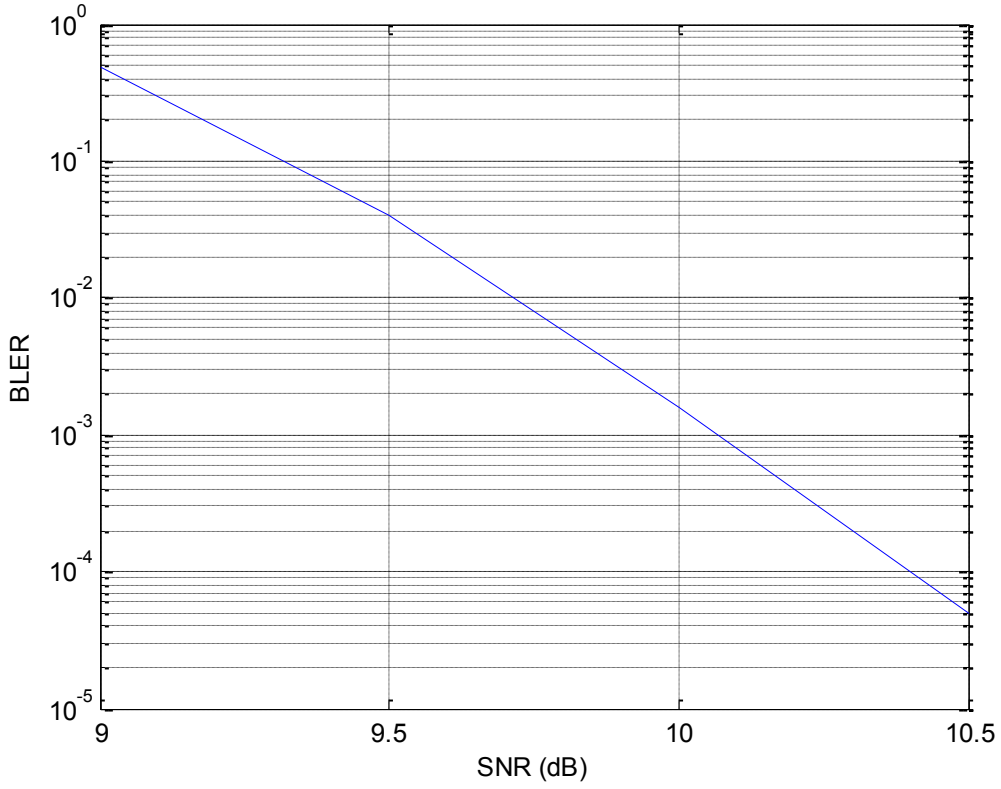


Figure 4-19: BLER performance of 16-QAM PLA-JCCNC two-way relay system

4.4.5 Complexity Analysis

The complexity of the belief propagation decoding of PLA-JCCNC over 2^r PAM and 2^r QAM is given by $rN\text{Log}N$ where r is the number of levels and N is the block length. The number of a priori probabilities that must be calculated to initialize the BP algorithm is given by 2^r, since two a priori probabilities must be calculated to initialize the BP decoder for each

binary polar code level. It can be noted that with LDA-JCCNC and PLA-JCCNC, the number of a priori probabilities to be calculated for each codeword dimension to initialise the BP algorithm, is the same as the modulation order.

In comparison to XOR-JCCNC, PLA-JCCNC JCCNC offers the same complexity reduction as LDA-JCCNC in reducing the number of a priori probability calculations required to initialize the BP algorithm. Since the number of a prior probabilities to be calculated for standard XOR-JCCNC increases in an exponential manner with modulation order for XOR-JCCNC, the resulting complexity reduction offer by PLA-JCCNC is more profound for higher order modulations.

4.4.6 Summary

The properties of the PLA-JCCNC scheme for two-way relay system can be summarised in the following:

- Encoding and decoding of the PLA-JCCNC scheme is a few times faster than the standard LDPC JCCNC scheme and LDA-JCCNC scheme due to the embedded polar codes being used.
- The PLA-JCCNC scheme can be applied to 2^r-PAM and square 2^r-QAM constellation systems if the binary polar codes are used to design the polar lattice. Note that it is also possible to design PLA-JCCNC schemes for irregular PAM and QAM modulation systems if non-binary polar codes over prime finite fields are available.
- The error rate performance of the PLA-JCCNC scheme on two-way relay system depends on the performance of the underlining polar codes. Simulation results show that the PLA-JCCNC scheme has the same level of error rate performance as the standard JCCNC scheme with LDPC codes.

5 JCCNC Compute-and-Forward Scheme for Multi-way Relay

In Chapter 4, the XOR-JCCNC scheme commonly studied in literature was reviewed and several new JCCNC schemes were proposed all for the two-way relay network to tackle complexity and scalability issues of existing JCCNC schemes for higher modulation orders. In this chapter, we extend the proposed JCCNC schemes to the general multi-way relay network by using the JCCNCs with the compute-and-forward relaying scheme. We show how the proposed schemes can be applied to general types of PAM and QAM modulation systems. Additionally we compare the performance and complexity of the proposed schemes against the standard XOR-JCCNC over the multi-way relay network.

A key difference between the implementation of the proposed JCCNC schemes for the two way relay channel and implementation for the multi-way relay network, is that while our proposed JCCNC schemes can be used for TWRC over AWGN channel, the same schemes cannot be applied to multi-way relay network over AWGN channel. This is because for TWRC, each user can obtain the other user's message using self-information. However, this is not possible for a multi-way relay network consisting of multiple users, multiple relays and an end destination tasked with receiving multiple computations from all the relays. This is because the constant channel gains in the uplink phase prevent the end destination from recovering the individual messages from multiple independent linear equations forwarded from all the relays.

For the TWR network over fading channel, one suitable option is to apply channel inversion precoding at transmitting user nodes, so the detection problem for the proposed JCCNC schemes only needs to be considered for the AWGN channel. At the cost of additional transmit power, the decoding complexity is reduced at the relay since it is not required to decode a set of integer equation coefficients. Therefore during the downlink phase, each user can decode the other user's symbol by using self-information.

However, for the multi-way relay network, assuming that each node and each relay only employs a single antenna, and assuming that all users transmit their codewords simultaneously, then channel inversion precoding cannot be used to simultaneously mitigate the effects of fading between each user and all the relay nodes during uplink. Therefore, the C-F relaying strategy cannot be used for the multi-way relay network over AWGN channel,

as the decoded equation coefficients will be unity, making it impossible for the original transmitted messages to be decoded. Instead, the proposed JCCNC schemes can only be used over fading channel, using the C-F decoding strategy. By manipulating the noisy linear received signal, each relay in the multi-way relay network decodes a set of integer equation coefficients that corresponds to an integer linear combination of the transmitted coded symbols. An end destination, on obtaining the decoded integer linear combinations and integer equation coefficient values from all the network relays, can solve transmitted symbols from each transmitting user.

So far, all the investigation for C-F has been performed without the use of error correction codes, while the channel coded network coding schemes available in literature have all been proposed for the TWRC. In this chapter, we first show how the standard XOR JCCNC and our four proposed JCCNC schemes can be adapted for use over the multi-way relay. We then analyse the additional complexity introduced in the JCCNC schemes when the relay decodes a linear computation of messages from multiple (more than 2) users. We show that the low complexity of our proposed schemes compared to the standard XOR-JCCNC makes them suited for the multi-way relay network.

Furthermore, in this chapter, we explicitly outline how the proposed JCCNC schemes may be used with various modulation schemes over various finite fields. In particular, for code over GF(q), the adaption of our JCCNC schemes to q-PAM, and q-QAM are given. And for code over GF(q^2), adaptation of the two proposed LD-JCCNC schemes to q^2-QAM. For PLA-JCCNC, r level binary codes are used for 2^r QAM.

Figure 5-1 below gives a multi-way relay system with both error correction coding (ECC) and lattice coding over fading channel, assuming L source nodes and M relay nodes. The message element of source node l is represented by e_l $(l = 1,2,\cdots,L)$ while the ECC codeword element is represented by c_l $(l = 1,2,\cdots,L)$. x_l $(l = 1,2,\cdots,L)$ represent signal after lattice coding (i.e. modulation constellation mapping). h_{ml} represent channel gain between source node l and relay node m. At relay node m, the signal received is,

$$y_m = \sum_{l=1}^{L} h_{ml} x_l + z_m,$$

where z_m is white Gaussian noise.

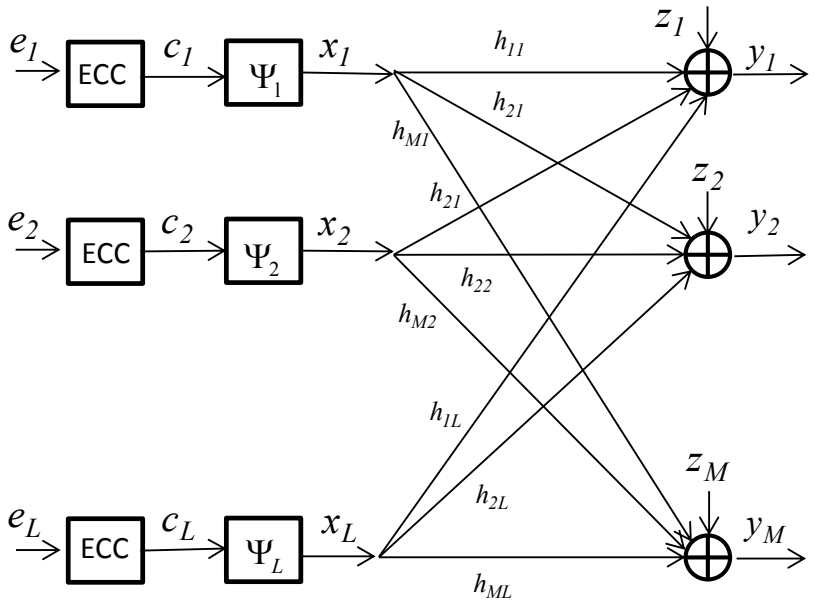


Figure 5-1: A multi-way relay with ECC and lattice coding

Throughout this chapter, the following notations are used: $[\cdot]$ rounds a real number to the nearest integer, or rounds the complex number to the nearest Gaussian integer or Eisenstein integer depending on which domain is used. $\mathrm{Re}(\cdot)$ takes the real part of a complex number, $\mathrm{Im}(\cdot)$ takes the imaginary part of a complex number, and $\bmod$ represents modulo operation.

5.1 JCCNC Schemes for Compute-and-Forward

In Chapter 4, the channel code and lattice code compatibility is introduced, which determines which modulation schemes can be used in our JCCNC schemes. For instance, it has been known that q-PAM and $q^2-\mathrm{QAM}$ modulation schemes with linear ECC codes in GF(q) can be used in JCCNC schemes, where q is prime. It is also proposed that the QAM modulation schemes formed by the representative sets of Gaussian integers and Eisenstein integers can also be used in JCCNC schemes.

Note that the channel gains must be real-valued in order to apply our JCCNC schemes to the multi-way network for $q^2-\mathrm{QAM}$ modulation systems. This is because for complex-valued channel gains, a way has not been found to recover the linear combination of codewords after lattice decoding. However there is no restriction for the QAM modulation schemes formed by the representative sets of Gaussian integers and Eisenstein integers.

For the multi-way JCCNC schemes proposed in this chapter, compatibility of the selected channel code and the lattice code is required. Before describing our JCCNC schemes, a useful equation from the compute-and-forward decoding process [20][24] is reintroduced:

$$\begin{aligned} y_{Rm} &= \alpha y_m = \alpha\left(\sum_{l=1}^{L} h_{ml}x_l + z_m\right) \\ &= \sum_{l=1}^{L} a_{ml}x_l + \sum_{l=1}^{L}\left(\alpha h_{ml} - a_{ml}\right)x_l + \alpha z_m \\ &= \sum_{l=1}^{L} a_{ml}x_l + n_{eff} \end{aligned} \tag{5-1}$$

where y_{Rm} is scaled output at relay node m, α is the optimal compute-and-forward scaling factor, a_{ml} is the optimal integer coefficient (can be real integer, complex Gaussian integer, or complex Eisenstein integer), and n_{eff} is the effective noise of compute-and-forward,

$$n_{eff} = \sum_{l=1}^{L}\left(\alpha h_{ml} - a_{ml}\right)x_l + \alpha z_{\mathrm{m}} \tag{5-2}$$

5.1.1 JCCNC Schemes for *q*-PAM Systems

With these JCCNC schemes, q-PAM modulations are used, where q is prime. For PAM modulation systems, the channel gains can be assumed to be real-valued, or only the real part of the complex channel gain is considered. The PAM signal mapping generates $x = c - (q-1)/2$, where c denotes codeword elements in GF(q). The details of the decoding process are presented below.

Denote $c_1, c_2, \ldots, c_L$ as the codeword elements in GF(q) of the L source nodes that are sent at the same time instant. Let's denote,

$$c_{Rm} = \sum_{l=1}^{L} a_{ml}c_l\,, \tag{5-3}$$

as the integer linear combination of $c_1, c_2, \ldots, c_L$ in GF(q) at relay node m, $(m = 1, 2, \cdots M)$, which has a total of q^L possible combinations, denoted by $C_R(k)$, where $k = 1,2,\cdots, q^L$. Denote $x_1, x_2, \ldots, x_L$ as the q-PAM constellation points of $c_1, c_2, \ldots, c_L$. Let the expression,

$$x_{Rm} = \sum_{l=1}^{L} a_{ml}x_l\,, \tag{5-4}$$

denote at relay m, the weighted arithmetic addition of the constellation points of the L source nodes. This term also has q^L number of combinations, denoted by $X_R(k)$, where $k = 1,2,\cdots,q^L$. Note that $C_R(k)$ and $X_R(k)$ correspond to the same combination of codeword elements thus have a one-to-one mapping relationship. In our decoding schemes at the relay node, the aim is to decode integer linear combination of the codewords from all the source nodes. Based on the proposed schemes in Chapter 4, we propose 5 channel coded network coding schemes for q-PAM multi-way relay network based on the following decoding options at the relay.

5.1.1.1 Option-1 (Standard XOR-JCCNC Scheme):

Option 1 is an adaptation of the standard XOR-JCCNC [2], [3] [62] for the TWR channel to the multi-way relay setting. For the TWR channel, the network coded symbols are the XOR of transmitted code symbols, and are decoded from the superposition of the transmitted constellation symbols. For XOR-JCCNC over the multi-way relay network, using the C-F paradigm at the relay, the network coded symbols are a weighted sum in GF(q) of transmitted codewords, and are decoded from an integer linear combination of the transmitted constellation symbols, as will be described in detail below.

Denoting the signal combinations at any relay node as $X_R(k)$, $k = 0,1,\ldots,q^L$, and their probability denoted as $P\{X_R(k)\}$, we will have: $P\{X_R(k)\} = 1/q^L$. Given $X_R(k)$ is sent, the probability for y_{Rm} can be written as:

$$P\{y_{Rm} \mid X_R(k)\} = \frac{1}{\sqrt{2\pi n_{eff}}} \exp\left(\frac{-(y_{Rm} - X_R(k))^2}{2n_{eff}}\right), \tag{5-5}$$

where n_{eff} is the effective noise power. From Bayes' theorem, given received signal y_{Rm}, the probability of $X_R(k)$ being sent is given by:

$$\begin{aligned} P_k &= P\{X_R(k) \mid y_{Rm}\} \\ &= P\{y_{Rm} \mid X_R(k)\}\frac{P\{X_R(k)\}}{P\{y_{Rm}\}} = P\{y_{Rm} \mid X_R(k)\}\frac{1}{B}, \end{aligned} \tag{5-6}$$

where $B = q^L \cdot P\{y_{Rm}\}$ can be used as a normalization factor. The input to the BP decoder is given by the equation below, where $c \in \mathrm{GF}(q)$

$$P\{c_{Rm} = c \mid y_{Rm}\} = \sum_{C_R(i)=c} P_i \tag{5-7}$$

This is effectively a multi-way relay version of the standard lookup table method for the two-way relay covered in Section 4.2.1.1, where soft input to the BP decoder $P\{c_{Rm}=c \mid y_{Rm}\}$ is based on the sum of all $P\{X_R(k) \mid y_{Rm}\}$ that corresponds to a certain c. The difference is for the multi-way relay case, the target codeword to be decoded is not an XOR of transmitted codewords, but a weighted sum in GF(q) of transmitted codewords from all transmitting users, where the weight is in fact the compute-and-forward integer coefficients. Therefore, as with the TWR channel case, the complexity of the XOR-JCCNC for the multi-way relay network depends on the size of the lookup table, which is dependent on the finite field size of the ECC code. The difference is that the complexity of XOR-JCCNC for the multi-way relay network is further dependent on the number of transmitting users which determines the number of unique weighted signal combinations $X_R(k)$ received at the relay. As illustrated by equation (5-5), (5-6) and (5-7), the a priori probabilities for the weighted signal combination is used to determine the a posteriori probabilities for the network coded symbols. In other words, the number of a priori probability calculations required will increase with more transmitting users.

5.1.1.2 Option-2 (LDA Aided JCCNC Scheme):

Similar to the LDA-JCCNC scheme for two-way relay system described in section 4.3, in this LDA-JCCNC scheme for multi-way relay systems, the weighted arithmetic sum x_{Rm} of the L signals is decoded first directly from the scaled noisy signal y_{Rm} by using the LDA lattice decoding algorithm described in Chapter 4. Then the integer linear combination c_{Rm} of codeword elements from the L source nodes can be obtained simply by using the isomorphism mapping between c_{Rm} and x_{Rm}. The performance of this LDA-JCCNC scheme is expected to be worse than that of the Option-1, since modulo operation is used in LDA lattice decoding, which loses information. However, the computational complexity of this scheme is lower.

5.1.1.3 Option-3 (LD-JCCNC, decoder input calculated after modulo operation):

Similar to the case for two-way relay, in this JCCNC scheme, lattice modulo operation is carried out for the scaled channel output at the relay to get $\tilde{y}_{Rm}=(\alpha_{MMSE}y_m) \bmod \Lambda$. The modulo operation can be carried out in the steps below:

$$s_m = (\alpha_{MMSE} y_m + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml}) \bmod q \,. \tag{5-8}$$

s_m is within the range $0 \le s_m < q$. The operation below is carried out to fold s_m into the range $0 \le s_m \le q-1$:

$$s_m = q - s_m, \quad if\ q - 0.5 < s_m < q \tag{5-9}$$

$$s_m = q - 2 + (q - s_m), \quad if\ q - 1 \le s_m \le q - 0.5 \tag{5-10}$$

Then s_m is transformed back with q-PAM mapping:

$$\tilde{y}_{Rm} = s_m - (q-1)/2 \,. \tag{5-11}$$

The next step is to calculate the a-priori probabilities for the q possible combinations of codeword elements:

$$P\{\tilde{y}_{Rm} \mid c_m = c\} = \frac{1}{\sqrt{2\pi n_{eff}}} \exp\left(\frac{-(\tilde{y}_{Rm} - x(c))^2}{2n_{eff}} \right) \tag{5-12}$$

where n_{eff} is the effective noise power, and $x(c)$ is the q-PAM constellation point of $c \in \mathrm{GF}(q)$. From Bayes' Theorem, the input to the LDPC decoder can be obtained:

$$\begin{aligned} P\{c_{Rm} = c \mid \tilde{y}_{Rm}\} &= P\{\tilde{y}_{Rm} \mid c_{Rm} = c\} \frac{P\{c_{Rm} = c\}}{P\{\tilde{y}_{Rm}\}} \\ &= P\{\tilde{y}_{Rm} \mid c_{Rm} = c\} \frac{1}{B}; \quad c \in \mathrm{GF}(q), \end{aligned} \tag{5-13}$$

where $B = q \cdot P\{\tilde{y}_{Rm}\}$ is a normalization factor. The performance of this JCCNC scheme is also expected to be worse than that of Option-1 because of the modulo operation. However, the computational complexity of this scheme is lower since only a number of q a-priori probabilities need to be calculated instead of the q^L number of calculations in Option-1.

5.1.1.4 Option-4 (LD-JCCNC, decoder input decided after hard lattice decoding):

If $\bar{c}_{Rm}$ denotes the codeword element decoded at relay node m after hard lattice decoding is performed on scaled channel output $y_{Rm} = \alpha_{MMSE} y_m$, it can be obtained by:

$$\bar{c}_{Rm} = \left[y_{Rm} + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml} \right] \bmod q \tag{5-14}$$

Then the values below are used as the input to the LDPC code decoder:

$$P\{c_{Rm} = \bar{c}_{Rm} \mid y_{Rm}\} = 1 \tag{5-15}$$

$$P\{c_{Rm} = c \mid y_{Rm}\} = 0, \quad c \in \{\mathrm{GF}(q) \setminus \bar{c}_{Rm}\} \tag{5-16}$$

As useful information is lost in hard lattice decoding, Option-4 is expected to perform worse than Option-1 and Option-2. However, the computational complexity of Option-4 is much lower and the channel noise information n_{eff} is not required.

5.1.1.5 Option-5 (PLA-JCCNC Scheme)

Similar to the PLA-JCCNC scheme for PAM two-way relay systems described in section 4.4.3.1, in this PLA-JCCNC scheme for multi-way relay systems, the aim of the decoder at the relay nodes is to decode an integer linear combination of transmitted polar lattice codewords from the weighted arithmetic sum of the PAM signals from the *L* users. Assuming the polar lattice codeword selected for user *l* is,

$$C^l = (C_1^l + 2C_2^l + \cdots + 2^{r-1}C_r^l) \bmod 2^r, \tag{5-17}$$

and the $2^r - \mathrm{PAM}$ signal transmitted from user *l* be represented by:

$$X_l = (C_1^l + 2C_2^l + \cdots + 2^{r-1}C_r^l) \bmod 2^r - (2^r - 1)/2. \tag{5-18}$$

The signal received at relay node *m* can be written as:

$$Y_m = \sum_{l=1}^{L} h_{ml} X_l + z_m, \tag{5-19}$$

where z_m is the noise vector at relay node *m*, and h_{ml} is the channel coefficient from user *l* to relay *m*. Under the compute-and-forward scheme, the aim of relay *m* is to decode an integer linear combination of transmitted lattice points,

$$C^{Rm} = \sum_{l=1}^{L} a_{ml} C^l, \tag{5-20}$$

where a_{ml} is the optimal compute-and-forward integer coefficient. The operation at the relay is to shift the signal from constellation domain to lattice domain. This is done by adding,

$$0.5\left(2^r - 1\right)\sum_{l=1}^{L} h_{ml}$$

to Y_m. The resultant signal can then be decoded as standard for compute-and-forward network coding. The noisy signal is first scaled by MMSE coefficient α, and then modulo 2^r operation is applied with regards to the cubic shaping of the lattice codebook. The shifted, scaled and modulo operated signal at relay node *m* can be written as:

$$
\begin{aligned}
Y_{Rm} &= \left(\alpha \left(Y_m + 0.5\left(2^r - 1\right)\sum_{l=1}^{L} h_{ml} \right) \right) \bmod 2^r = \left(\sum_{l=1}^{L} \alpha h_{ml} C^l + \alpha z_m \right) \bmod 2^r \\
&= \left(\sum_{l=1}^{L} a_{ml} C^l + \sum_{l=1}^{L} \left(\alpha h_{ml} - a_{ml}\right) C^l + \alpha z_m \right) \bmod 2^r \\
&= \left(\sum_{l=1}^{L} a_{ml} \left(C_1^l + 2C_2^l + \cdots + 2^{r-1} C_r^l \right) \bmod 2^r + n_{eff} \right) \bmod 2^r \\
&= \left(C_1^{Rm} + 2C_2^{Rm} + \cdots 2^{r-1} C_r^{Rm} + n_{eff} \right) \bmod 2^r
\end{aligned}
\tag{5-21}
$$

where C_k^{Rm} is defined as,

$$
C_k^{Rm} = \sum_{l=1}^{L} a_{ml} C_k^l, \left(1 \le k \le r\right). \tag{5-22}
$$

Note the important difference between C^{Rm} in (5-20) and C_k^{Rm} in (5-22). While C^{Rm} is an integer linear combination of the transmitted polar lattice points, C_k^{Rm} is an integer linear combination of the lattice components from each user, for the *k*-th polar code level of the one-dimensional polar lattice employed at relay node *m*. The term n_{eff} is the effective noise vector of compute-and-forward that includes amplification of original Gaussian noise and the noise due to errors in approximation of arbitrary valued channels using integer values.

$$
n_{eff} = \sum_{l=1}^{L} \left(\alpha h_{ml} - a_{ml}\right) C^l + \alpha \mathrm{z_m} \tag{5-23}
$$

Following the normal decoding method of compute-and-forward relaying scheme, the integer coefficients a_{ml} must be determined first using methods proposed in Section 2. Then using the same PLA-JCCNC decoding scheme described in Section 4.4.3.1, an integer linear combination of transmitted polar lattice codeword can be decoded. The polar lattice decoder serves to decode the scaled and modulo operated signal to the nearest polar lattice point in the original polar lattice codebook.

Note that the BP decoder for each level of polar lattice will now decode binary information bits representing the component C_k^{Rm} for the *k* coded levels. The decoded information $\hat{C}_k^{Rm}$ for all $1 \le k \le r$ polar code levels of the polar lattice, can then be used to re-compute the decoded polar lattice codeword representing an integer linear combination of transmitted polar lattice codewords. The decoded polar lattice point can be explicitly expressed as, $\hat{C}^{Rm} = (\hat{C}_1^{Rm} + 2\hat{C}_2^{Rm} + + \cdots 2^{r-1}\hat{C}_r^{Rm}) \bmod 2^r$. At relay, the error rate

probability is calculated based on the error criteria $P_e = \Pr(C^{Rm} \neq \hat{C}^{Rm})$ where C^{Rm} is defined in (5-20).

5.1.2 JCCNC schemes for q^2 QAM relay systems

Note that to use these JCCNC schemes for $q^2-\text{QAM}$ relay systems the channel gains must be real-valued. In these JCCNC schemes, unconventional $q^2-\text{QAM}$ constellation schemes are used, where q is prime. We have two options to choose LDPC codes: LDPC codes in GF(q) or LDPC codes in GF(q^2). Here we only describe LDPC codes in GF(q^2).

The mapping from GF(q^2) symbols to $q^2-\text{QAM}$ constellation points is done in this method: a GF(q^2) symbol is converted to a GF(q) symbol pair first, represented by $s_1 s_0$, then the constellation mapping point is obtained as $(s_1-(q-1)/2,\ s_0-(q-1)/2)$. The details of the decoding process is as follows. Denote $c_1, c_2, \ldots, c_L$ as the code-word elements of all the source nodes in GF(q^2) with $s_{11}s_{10}, s_{21}s_{20}\cdots, s_{L1}s_{L0}$ being their corresponding GF(q) symbol pairs. The term,

$$c_{Rm} = \sum_{l=1}^{L} a_{ml} c_l ,$$

is the integer linear combination of $c_1, c_2, \ldots, c_L$ in GF(q^2) at relay node m ($m = 1, 2, \cdots M$),which has q^{2L} number of combinations, denoted by $C_R(k)$, where $k = 1,2,\cdots, q^{2L}$. Denote $x_1, x_2, \ldots, x_L$ as the $q^2-\text{QAM}$ constellation mapping points of $c_1, c_2, \ldots, c_L$. Denote,

$$x_{Rm} = \sum_{l=1}^{L} a_{ml} x_l$$

as the weighted arithmetic addition of the constellation mapping points of the L source nodes, which also has q^{2L} number of combinations, denoted by $X_R(k)$, where $k = 1,2,\cdots, q^{2L}$. Note that $C_R(k)$ and $X_R(k)$ correspond to the same combination of codeword elements thus have a one-to-one mapping relationship. In our decoding schemes at the relay node, the aim is to decode integer linear combination of the codewords from all the source nodes. Based on the proposed schemes in Chapter 4, we propose 5 channel coded network coding

schemes for $q^2-\text{QAM}$ multi-way relay network based on the following decoding options at the relay.

5.1.2.1 Option-1 (Standard JCCNC Scheme):

With the signal combinations at any relay node being denoted as $X_R(k),\ \ k=0,1,\ldots,q^{2L}$, and their probabilities denoted as $P\{X_R(k)\}$, we have: $P\{X_R(k)\}=1/q^{2L}$. Given $X_R(k)$ is sent, the probability for y_{Rm} can be written as:

$$P\{y_{Rm}\mid X_R(k)\}=\frac{1}{\pi n_{eff}}\exp\left(\frac{-\left|y_{Rm}-X_R(k)\right|^2}{n_{eff}}\right) \tag{5-24}$$

where n_{eff} is the effective noise power. Using Bayes' theorem, given received signal y_{Rm}, the probability of $X_R(k)$ being sent is given by:

$$\begin{aligned} P_k &= P\{X_R(k)\mid y_{Rm}\} \\ &= P\{y_{Rm}\mid X_R(k)\}\frac{P\{X_R(k)\}}{P\{y_{Rm}\}}=P\{y_{Rm}\mid X_R(k)\}\frac{1}{B} \end{aligned} \tag{5-25}$$

where $B=q^{2L}\cdot P\{y_{Rm}\}$ is a normalization factor. The input to the BP decoder is given by the equation below, where $c\in \text{GF}(q^2)$:

$$P\{c_{Rm}=c\mid y_{Rm}\}=\sum_{C_R(i)=c}P_i \tag{5-26}$$

Once again, the optimal compute-and-forward coefficients, and a lookup table of all the transmitted codeword combinations, must be determined to evaluate the probabilistic input into the BP decoder.

5.1.2.2 Option-2 (LDA Aided JCCNC Scheme):

The proposed LDA aided JCCNC scheme cannot be used for $q^2-\text{QAM}$ multi-way relay systems since q^2 is not a prime.

5.1.2.3 Option-3 (LD-JCCNC, decoder input calculated after modulo operation):

In this JCCNC scheme, mod-q operation in each dimension of the QAM symbol, is carried out for the scaled channel output at the relay to obtain $\tilde{y}_{Rm}=(\alpha_{MMSE}y_m)\bmod\Lambda$. The modulo operation can be performed in the following steps:

$$s_{m1} = (\mathrm{Re}(\alpha_{MMSE} y_m) + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml}) \bmod q \tag{5-27}$$

$$s_{m0} = (\mathrm{Im}(\alpha_{MMSE} y_m) + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml}) \bmod q \tag{5-28}$$

s_{m0} and s_{m1} are within range $0 \le s_{m0}, s_{m1} < q$. The operation below is carried out to make s_{m0} and s_{m1} in the range $0 \le s_{m0}, s_{m1} \le q-1$:

$$s_{mi} = q - s_{mi}, \quad if\ q - 0.5 \le s_{mi} < q; \quad i = 0,1 \tag{5-29}$$

$$s_{mi} = q - 2 + (q - s_{mi}), \quad if\ q - 1 \le s_{mi} \le q - 0.5; \quad i = 0,1 \tag{5-30}$$

Then s_{m0} and s_{m1} are transferred back to constellation point given by:

$$\tilde{y}_m = (s_{m1} - (q-1)/2,\ s_{m0} - (q-1)/2) \tag{5-31}$$

The next step is to calculate the a-priori probabilities for the q^2 possible combinations of codeword elements:

$$P\{\tilde{y}_m \mid c_{AB} = c\} = \frac{1}{\pi n_{eff}} \exp\left(\frac{-|\tilde{y}_m - x(c)|^2}{n_{eff}} \right) \tag{5-32}$$

where $x(c)$ is $q^2 - \mathrm{QAM}$ constellation mapping point of $c \in \mathrm{GF}(q^2)$. From Bayes' theorem, the LDPC decoder input can be obtained:

$$\begin{aligned} P\{c_{AB} = c \mid \tilde{y}_m\} &= P\{\tilde{y}_m \mid c_{AB} = c\} \frac{P\{c_{AB} = c\}}{P\{\tilde{y}_m\}} \\ &= P\{\tilde{y}_m \mid c_{AB} = c\} \frac{1}{B}; \quad c \in \mathrm{GF}(q^2) \end{aligned} \tag{5-33}$$

where $B = q^2 \cdot P\{\tilde{y}_m\}$ is a normalization factor.

5.1.2.4 Option-4 (LD-JCCNC, decoder input decided after hard lattice decoding):

Denote $\bar{c}_{Rm}$ as the code-word element decoded at relay node after hard lattice-decoding on the scaled channel output $y_{Rm} = \alpha_{MMSE} y_m$, and $\bar{s}_{m1}\bar{s}_{m0}$ as the corresponding symbol pair. The code-word element at relay node can be obtained by hard lattice decoding using equations below:

$$\bar{s}_{m1} = \left[\mathrm{Re}(\alpha_{MMSE} y_m) + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml} \right] \bmod q \tag{5-34}$$

$$\bar{s}_{m0} = \left[\mathrm{Im}(\alpha_{MMSE} y_m) + 0.5 \cdot (q-1) \cdot \sum_{l=1}^{L} a_{ml} \right] \bmod q \tag{5-35}$$

$$\bar{c}_{Rm} = q \cdot \bar{s}_{m1} + \bar{s}_{m0} \tag{5-36}$$

Then the values below are used as the input to the LDPC code decoder:

$$P\{c_{Rm} = \bar{c}_{Rm} \mid y_{Rm}\} = 1 \tag{5-37}$$

$$P\{c_{Rm} = c \mid y_{Rm}\} = 0, \quad c \in \{\mathrm{GF}(q^2) \setminus \bar{c}_{Rm}\} \tag{5-38}$$

5.1.2.5 Option-5 (PLA-JCCNC Scheme)

Similar to the PLA-JCCNC scheme for QAM two-way relay systems described in section 4.4.3.2, in this PLA-JCCNC scheme for multi-way relay systems, the aim of the decoder at the relay nodes is to decode an integer linear combination of transmitted polar lattice codewords from the weighted arithmetic sum of the QAM signals from the L users. Assuming the polar lattice codeword selected for user l is,

$$C^l = \left(C_1^l \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + C_2^l \otimes \left(R \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + \cdots + C_r^l \otimes \left(R^{r-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \right) \bmod 2^{r/2}. \tag{5-39}$$

Let the $2^r - \mathrm{QAM}$ signal transmitted from user l be represented by:

$$X_l = C_1^l \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + C_2^l \otimes \left(R \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + \cdots + C_r^l \otimes \left(R^{r-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) - \begin{bmatrix} (2^{r/2}-1)/2 \\ (2^{r/2}-1)/2 \end{bmatrix} \tag{5-40}$$

The signal received at relay node m can be written as:

$$Y_m = \sum_{l=1}^{L} h_{ml} X_l + z_m \tag{5-41}$$

From this noisy linear signal, relay m aims to decode an integer linear combination of transmitted lattice points,

$$C^{Rm} = \sum_{l=1}^{L} a_{ml} C^l, \tag{5-42}$$

This is achieved by first shifting signal from constellation domain to signal domain. Optimal compute-and-forward scaling factor is applied to move the noisy signal as close to an integer linear combination as possible. Modulo operation with respect to cubic shaping is applied. The shifted, scaled and modulo operated signal at relay node m can be written as Y_{Rm} where,

$$
\begin{aligned}
Y_{Rm} &= \left(\alpha \left(Y_m + 0.5 \begin{bmatrix} (2^{r/2}-1)/2 \\ (2^{r/2}-1)/2 \end{bmatrix} \sum_{l=1}^{L} h_{ml} \right) \right) \bmod 2^{r/2} = \left(\sum_{l=1}^{L} \alpha h_{ml} C^l + \alpha z_m \right) \bmod 2^{r/2} \\
&= \left(\sum_{l=1}^{L} a_{ml} C^l + \sum_{l=1}^{L} (\alpha h_{ml} - a_{ml}) C^l + \alpha z_m \right) \bmod 2^{r/2} \\
&= \left(\sum_{l=1}^{L} a_{ml} \left(C_1^l \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + C_2^l \otimes \left(R \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + \cdots + C_r^l \otimes \left(R^{r-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \right) + n_{eff} \right) \bmod 2^{r/2} \\
&= \left(C_1^{Rm} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + C_2^{Rm} \otimes \left(R \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + \cdots C_r^{Rm} \otimes \left(R^{r-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + n_{eff} \right) \bmod 2^{r/2}
\end{aligned}
\tag{5-43}
$$

The term z_m contains the noise vectors for both dimensions. And the term,

$$
C_k^{Rm} = \sum_{l=1}^{L} a_{ml} C_k^l, (1 \le k \le r), \tag{5-44}
$$

is an integer linear combination of the lattice components from each user, for the *k*-th polar code level of the two-dimensional polar lattice. The term n_{eff} contains the compute-and-forward effective noise vectors for both dimensions including:

$$
n_{eff} = \sum_{l=1}^{L} (\alpha h_{ml} - a_{ml}) C^l + \alpha z_{\mathrm{m}} \tag{5-45}
$$

Finally the processed signal Y_{Rm} is feed into the polar lattice decoder to decode a polar lattice point to represent an integer linear combination of transmitted polar points. Each level of multistage decoder decodes information bits representing the component of C_k^{Rm}. The term $\hat{C}_r^{Rm}$ for each polar coded level of the lattice can then be used to re-compute the decoded polar lattice codeword representing an integer linear combination of transmitted polar lattice codewords. The error decision rule is taken as $P_e = \Pr(C^{Rm} \neq \hat{C}^{Rm})$ where,

$$
\hat{C}^{Rm} = \left(\hat{C}_1^{Rm} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \hat{C}_2^{Rm} \otimes \left(R \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) + \cdots \hat{C}_r^{Rm} \otimes \left(R^{r-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \right) \bmod 2^{r/2}.
$$

5.1.3 JCCNC Schemes for *q*-QAM Systems

With these JCCNC schemes, *q*-QAM modulations are used, where *q* is prime and QAM constellation points are formed by the representative sets of Gaussian integers and Eisenstein integers, as described in section 4.3.2.1. The codeword elements in GF(*q*) and QAM

constellation points have a one-to-one mapping: $x = \phi(c)$, where c denotes a code-word element in GF(q). Below the details of the decoding process are presented.

Denote $c_1, c_2, \ldots, c_L$ as the code-word elements of the L source nodes that are sent at the same time instant. Denote $x_1, x_2, \ldots, x_L$ as the q-QAM constellation points of $c_1, c_2, \ldots, c_L$. The term,

$$x_{Rm} = \sum_{l=1}^{L} a_{ml} x_l \ ,$$

is the weighted arithmetic addition of the constellation mapping points of the L source nodes, which has a number of q^L combinations, denoted by $X_R(k)$, where $k = 1,2,\cdots,q^L$. Since a_{ml} are complex valued, the codeword elements at a relay node cannot simply be expressed as the linear combination of codeword elements from the L source nodes. However, it can be obtained by using $c_{Rm} = \phi^{-1}(x_{Rm} \bmod \pi)$ according to the one-to-one mapping relationship between GF(q) values and the q-QAM constellation points, where π is the prime ideal of Gaussian integer or Eisenstein integer that forms the QAM constellation points, $\bmod \pi$ operation is given by,

$$y_{Rm} \bmod \pi = y_{Rm} - \pi \left\lfloor {y_{Rm}\pi^* \big/ \pi\pi^*} \right\rfloor \tag{5-46}$$

The term c_{Rm} also has a number of q^L combinations of $c_1, c_2, \ldots, c_L$, denoted by $C_R(k)$. Note that $C_R(k)$ and $X_R(k)$ correspond to the same combination of codeword elements thus have a one-to-one mapping relationship.

In our decoding schemes at the relay node, the aim is to decode integer linear combination of the code words from all the source nodes. Based on the proposed schemes in Chapter 4, we propose 5 channel coded network coding schemes using q-QAM for the multi-way relay network based on the following decoding options at the relay.

5.1.3.1 Option-1 (Standard JCCNC Scheme):

With the signal combinations at any relay node being denoted as $X_R(k),\ k = 0,1,\ldots,q^L$, and their probabilities denoted as $P\{X_R(k)\}$, we have: $P\{X_R(k)\} = 1/q^L$. Given $X_R(k)$ is sent, the probability for y_{Rm} can be written as:

$$P\{y_{Rm} \mid X_R(k)\} = \frac{1}{\pi n_{eff}} \exp\left(\frac{-|y_{Rm} - X_R(k)|^2}{n_{eff}} \right) \tag{5-47}$$

From Bayes' theorem, given received signal y_{Rm}, the probability of signal $X_R(k)$ being sent is given by:

$$\begin{aligned} P_k &= P\{X_R(k) \mid y_{Rm}\} \\ &= P\{y_{Rm} \mid X_R(k)\} \frac{P\{X_R(k)\}}{P\{y_{Rm}\}} = P\{y_{Rm} \mid X_R(k)\} \frac{1}{B} \end{aligned} \tag{5-48}$$

where $B = q^L \cdot P\{y_{Rm}\}$ is a normalization factor. The input to the BP decoder is given by the equation below, where $c \in \mathrm{GF}(q)$:

$$P\{c_{Rm} = c \mid y_{Rm}\} = \sum_{C_R(i)=c} P_i \tag{5-49}$$

5.1.3.2 Option-2 (LDA Aided JCCNC Scheme):

Similar to the LDA-JCCNC scheme for two-way relay described in section 4.3, in this LDA-JCCNC scheme for multi-way relay systems, the weighted arithmetic sum x_{Rm} of the signals from the L source nodes is decoded first directly from the scaled noisy signal y_{Rm} by using the LDA lattice decoding algorithm described in section 4.3.1. Then the integer linear combination c_{Rm} of code-word elements can be obtained simply by using the isomorphism mapping between c_{Rm} and x_{Rm}.

5.1.3.3 Option-3 (LD-JCCNC, decoder input calculated after modulo operation):

This JCCNC scheme cannot be applied to this q-QAM multi-way relay system.

5.1.3.4 Option-4 (LD-JCCNC, decoder input decided after hard lattice decoding):

If $\bar{c}_{Rm}$ is denoted as the code-word element decoded at relay node m after hard lattice-decoding on scaled channel output $y_{Rm} = \alpha_{MMSE} y_m$, it can be obtained by using the following equation, where π is the prime ideal of Gaussian integer or Eisenstein integer that forms the q-QAM constellation points:

$$\bar{c}_{Rm} = \phi^{-1}(y_{Rm} \bmod \pi) \tag{5-50}$$

The modulo operation in Gaussian integer domain or Eisenstein integer domain can be performed with the equation below, where [·] rounds the complex number to the nearest Gaussian integer or Eisenstein integer depending on which domain the QAM constellation is formed from:

$$y_{Rm} \bmod \pi = y_{Rm} - \pi \left\lfloor {y_{Rm}\pi^* \big/ \pi\pi^*} \right\rceil \tag{5-51}$$

Then the values below are used as the input to the LDPC code decoder:

$$P\{c_{Rm} = \bar{c}_{Rm} \mid y_{Rm}\} = 1 \tag{5-52}$$

$$P\{c_{Rm} = c \mid y_{Rm}\} = 0, \quad c \in \{GF(q) \setminus \bar{c}_{Rm}\} \tag{5-53}$$

5.1.3.5 Option-5 (PLA-JCCNC Scheme)

Since the PLA-JCCNC scheme can only be used on square QAM system, it cannot be used on q-QAM multi-way relay systems.

5.2 Simulation and Complexity Analysis

To verify the performance of all the proposed JCCNC schemes for multi-way relay systems, simulations are performed for relay systems of two source nodes using BPSK, 3-PAM, 4-QAM, and 5-QAM modulation schemes. The LDPC code has a message block length of 200 and a codeword length of 600 for the 4-QAM relay system, while for the relay systems using the other modulation schemes the LDPC codes have a data block size of 500 and a code-word size of 1000. 10 iterations are performed for LDPC codes decoding using belief propagation decoding algorithm. For the PLA-JCCNC scheme, the polar codes used are the same as those in section 5, which have a code length of 1024 and average rate of 0.5. Integer linear-combination of code-words from all the source nodes is decoded at the relay-node. The obtained block error rates (BLER) are plotted from Figure 5-2 to Figure 5-5.

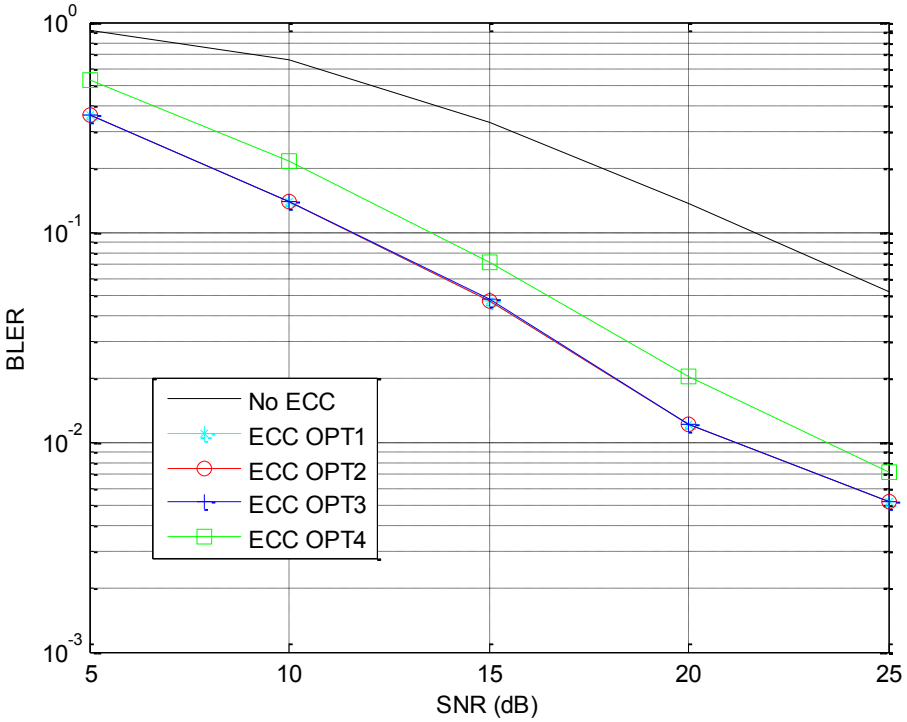


Figure 5-2: Error performance of compute-and-forward for BPSK

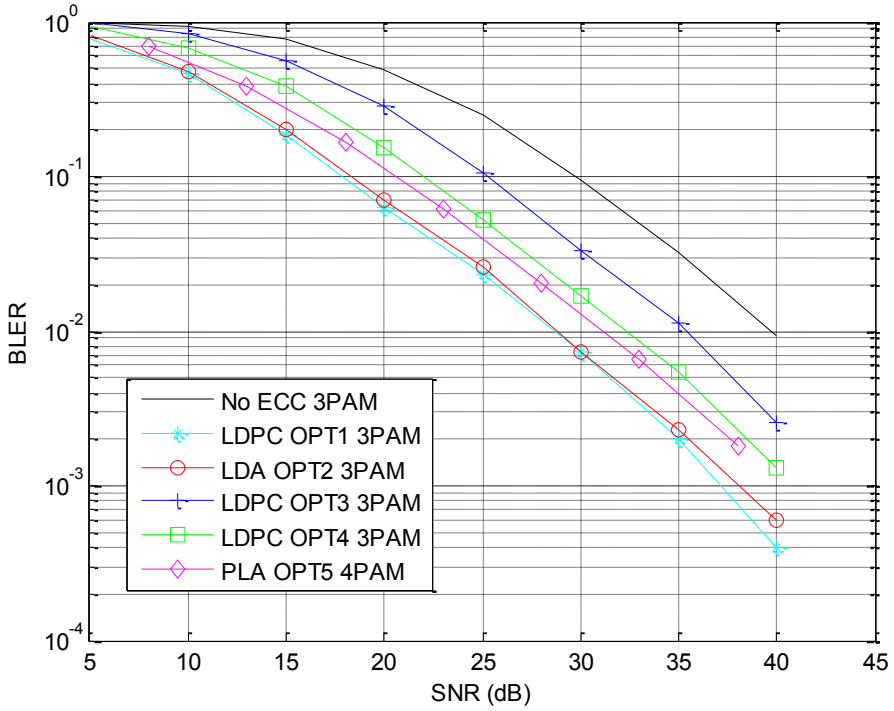


Figure 5-3: Error performance of compute-and-forward for 3-PAM and 4-PAM

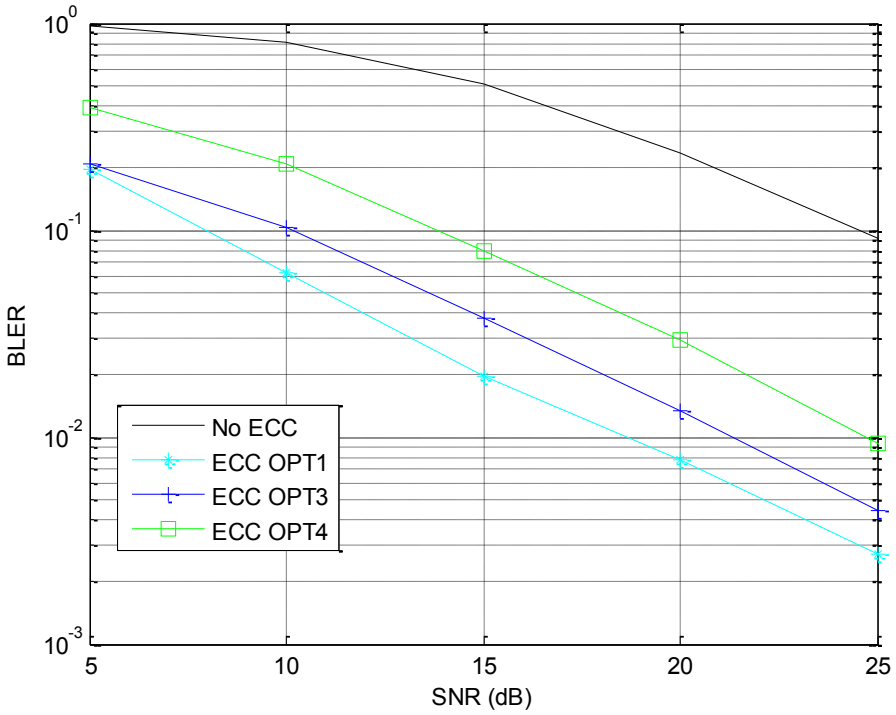


Figure 5-4: Error performance of compute-and-forward for 4-QAM

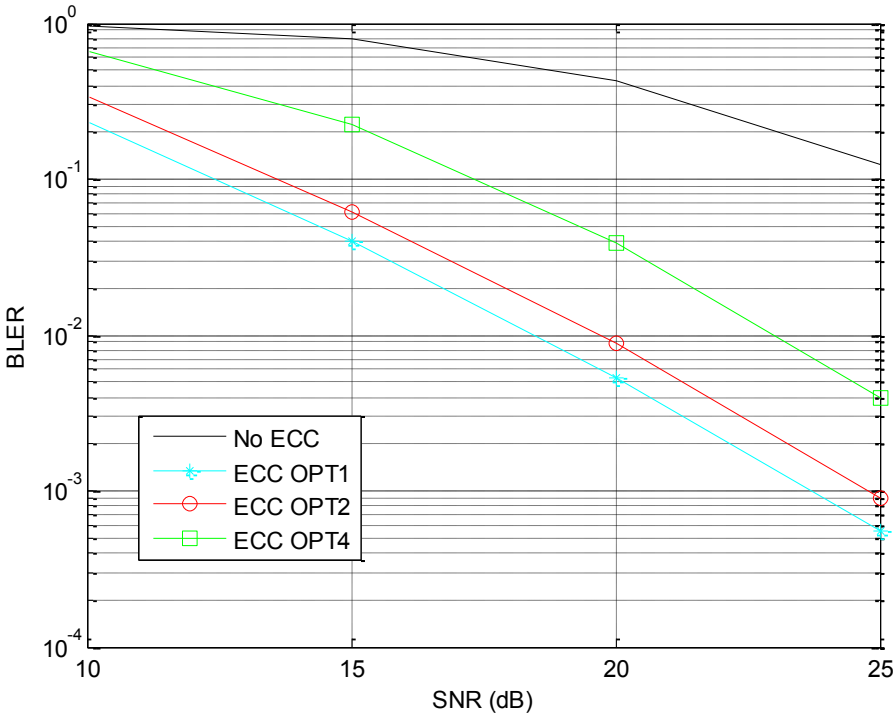


Figure 5-5: Error performance of compute-and-forward for 5-QAM

From the performance plots, the following points can be observed. The proposed JCCNC schemes greatly improve relay system performance compared with that without error

correction being used at relay nodes. For the simulated JCCNC schemes, Option-1, the XOR JCCNC yields the best performance. This is expected since useful channel information is lost more or less in our proposed JCCNC schemes, where modulo operation performed for lattice decoding results in lost information.

For relay systems with 3-PAM, the performance of Option 2 is very close to Option-1. Option 2 (LDA-JCCNC) performs slightly worse than XOR-JCCNC (Option 1). This is in part due to the loss of information incurred when the a priori probability of each coded symbol is determined based on the summation of the a priori probabilities of a selection of lattice points, as illustrated by equation (4-50). The a priori probability of each network coded symbol is calculated based on the sum of a priori probabilities of all the lattice points which after modulo operation, generate the said network coded symbol.

For q-PAM and q^2-QAM modulation schemes, the performance of Option-3 (LD-JCCNC with soft decoding) is superior to that of Option 4 (LD-JCCNC with hard decoding). The better performance of Option 3 can be attributed to calculating q number of priori probabilities corresponding to the q constellation points, while Option 4 essentially uses the hard decision from the lattice decoder to initialise the BP decoding. Therefore, more useful information is lost in Option 4 in determining the probabilistic input for the BP decoder.

The PLA-JCCNC scheme with 4-PAM achieves a similar level of error rate performance to LDA-JCCNC scheme over 3-PAM. The LDA-JCCNC performs slightly better, at the cost of a higher decoding complexity $O(q^2N \log N)$ in comparison to the PLA-JCCNC's decoding complexity of $O(rN \log N)$. Due to restrictions on the types of modulation that can be used for the two schemes, a direct fair comparison is not possible.

For JCCNC schemes over the multi-way relay network, additional complexity is incurred compared to the same JCCNC schemes over the TWRC, since the complexity of search algorithms for the optimal C-F equation coefficients is directly related to the number of transmitting users, as studied in detail in Chapter 2. For the JCCNC schemes over the TWRC, when channel state information is available, channel inversion precoding at transmitters can be used to reduce the encoding and decoding to that over AWGN channels. Doing so mitigates the need to perform C-F at the relay and search for optimal C-F equation coefficients, therefore effectively reducing the complexity of the schemes over the TWRC.

For XOR-JCCNC over the multi-way relay network, the number of calculations of a priori probability of unique values of signal combinations X_{AB} naturally increases with the number of transmitting users, since more transmitting users results in a larger number of permutations for signal combination X_{AB}. As an example, for L user XOR-JCCNC employing p-PAM, Lq-1 number of unique a priori calculations for X_{AB} are made for initialising the messages for the Belief Propagation decoder. For LDA-JCCNC and PLA-JCCNC, the number of unique a priori calculations required for X_{AB} is the same as the modulation order.

There are several reasons why a number of JCCNC schemes were proposed. Firstly certain JCCNC schemes can only be used with error correction over Galois fields of certain sizes. For example, binary polar codes are used in PLA-JCCNC scheme, while LDPC codes in GF(q) and GF(q^2) are used in the two LD-JCCNC schemes and LDA-JCCNC, where q must be prime. As a result of the differences in lattice construction (construction D versus construction A) and requirements on the underlying code, each of the proposed types of JCCNC schemes can only be adapted to specific types of modulation schemes. For example, the PLA-JCCNC schemes can be used with PAM and square QAM modulations of order 2^r, where r is the number of polar code levels in the polar lattice employed. Meanwhile, the proposed LD-JCCNC schemes and LDA-JCCNC scheme can only be adapted to q-PAM, q-QAM and q^2-QAM. Naturally, the selection of JCCNC schemes also depends on the performance and complexity trade-off for each scheme. PLA-JCCNC and LDA-JCCNC have similar levels of performances, but PLA-JCCNC has a lower complexity. The proposed LD-JCCNC with hard decoding has the worst performance of all the proposed schemes, but has the lowest complexity as well as benefiting from a decoding scheme that does not require channel noise information. It can be concluded that each proposed JCCNC scheme offers unique advantages and drawbacks, and a JCCNC scheme can be selected based on the requirement of the communication network.

5.3 Summary

In this chapter, we showed how the five JCCNC schemes investigated in Chapter 4 can be adapted to general multi-way relay networks using the compute-and-forward relaying scheme. We showed how these JCCNC schemes can be applied to different types of PAM and QAM modulation systems. It is shown that modulation type and order will restrict the types of channel coded network coding schemes which can be used. A suitable JCCNC scheme can be selected based on the required finite field size and the characteristics of

modulation constellation required. In the case of Polar lattice aided-JCCNC, the number of component code levels can be selected based on the desired modulation size.

We further investigated the practical performance of each JCCNC scheme and analysed their complexity under the multi-way relay setting. We showed that the proposed JCCNC schemes all have comparatively low complexity compared to the standard XOR-JCCNC, since the complexity of lattice decoding employed for the proposed JCCNC schemes at the relay is independent of the number of transmitting users. Therefore, the four proposed JCCNC schemes are particularly suited for higher order modulation. However, as discussed in detail in Chapter 2, the complexity of finding the optimal equation coefficients for C-F increases with the number of users, the relationship of which depends on which one of the methods outlined in Chapter 2 is used.

6 Conclusions and Future Work

6.1 Conclusions

In this work, we proposed ways improve the performance of physical-layer network coding schemes using lattice codes. For compute-and-forward scheme, we proposed algorithms in low polynomial complexity to optimally solve the shortest vector problem that occurs in C-F over complex channels, for both single receive antenna case and MIMO C-F. The algorithms obtains the optimal integer coefficients for C-F over Gaussian integers as well as Eisenstein integers. We derived the computational complexity for our proposed algorithms through which our algorithms are shown to have low polynomial complexity in transmitting user number. Additionally we use simulation results to verify the efficiency of our algorithm and the gap performance gap between our optimal search algorithm and suboptimal algorithms such as quantized exhaustive search, and complex-LLL. It is shown that our proposed search algorithm for standard compute-and-forward can achieve optimal performance at competent computational costs.

For the two-way relay network, we investigated the performances of physical layer network coding schemes and derived their end to end achievable rates over Rayleigh fading channel, with or without precoding. We also proposed a channel-aware precoding scheme based on compute-and-forward to optimise its performance over Rayleigh fading channel. The proposed scheme demonstrates superior performance to other PNC schemes which is verified through theoretical analysis and simulations.

Following our investigation of PNC schemes, we proposed a number of novel channel coded PNC schemes for both two-way relay and multi-way relay networks. We started by investigating channel coded PNC for the two-way relay network, where we introduced the concept of channel code and lattice code compatibility which allows network coding and channel coding to be integrated in their operations through the use of lattice decoding at the relay. As the most effective forms of channel coded PNC in contemporary literature are based around LDPC codes, we started by investigating LDPC codes with lattice decoding at the relay to improve the throughput of PNC schemes for the two-way relay network. We found that the suitable codes for this purpose are linear codes including binary codes and non-binary codes, among which non-binary LDPC codes are shown to be the best fit for lattice based PNC schemes. In our investigation, we found almost all of the existing channel coded network coding schemes proposed in literature are applied to BPSK, QPSK, 3-PSK, and 5-

PSK modulation systems. Consequently, we proposed a new JCCNC schemes based on lattice decoding that can be applied to general PAM and QAM modulation systems. We call these schemes lattice-decoded joint channel coded network coding (LD-JCNC). In our new schemes, binary LDPC code is used for BPSK and QPSK modulation systems; non-binary LDPC code in $\mathrm{GF}(q)$, where q is prime, is used for QPSK modulation systems; and non-binary LDPC code in $\mathrm{GF}(q)$, can be used for $q-\mathrm{PAM}$ and q^2 modulation systems. In relation to common channel coded network coding schemes available in literature, our proposed LD-JCCNC scheme has much reduced complexity.

Continuing on from PNC schemes based on LDPC codes, we proposed a new JCCNC scheme based on low density construction A (LDA) lattices for the two-way relay channel, which we call LDA-lattice aided JCCNC (LDA-JCCNC). Our scheme is a channel coded version of arithmetic-sum and-forward network coding, with channel coding and network-coding mapping operations integrated through the use of the lattice decoder. We also provide a clear methodology on how our LDA-JCCNC scheme can be applied to general PAM and QAM modulation systems, over Gaussian integers and Eisenstein integers. This is in contrast with most existing JCCNC schemes, which are restricted to set types of modulation frameworks. Our proposed schemes are shown to have a much lower complexity and yet perform almost on par with the best existing channel coded PNC schemes in literature.

Another high performance lattice which we applied for the purpose of channel coding is the polar lattice, for our polar lattice aided JCCNC scheme for the two-way relay network. As with our LDA-JCCNC scheme, we showed explicitly how polar lattice codes can be used with general QAM and PAM modulation systems in our channel coded PNC scheme, and examined their performance and complexity trade-off in relation to other channel coded network coding schemes. Unlike single level polar codes which can only be used with BPSK and QPSK in PNC, our adaption of multi-level polar lattices to JCCNC is applicable to higher order modulation systems. At the same time, channel decoding and network coding is integrated through direct lattice decoding of the arithmetic sum function of the transmitted lattice codewords.

Finally in the last section of our work, we generalised our proposed JCCNC schemes to the compute-and-forward over multi-way relay network employing PAM and QAM modulation. We show that this adaption is not a trivial one as a number of design considerations have to be taken into account.

6.2 Future Work

There are a number of areas where our work could be extended and improved. Starting with our optimal equation coefficient search algorithm for compute-and-forward, so far our algorithms are adapted to find the optimal coefficients over the ring of Gaussian integers as well as Eisenstein integers. A natural extension would be to extend our search algorithm to compute-and-forward over general algebraic rings. This would be particular useful in conjunction with the work in [107] where the authors proposed an adaptive compute-and-forward scheme that adaptively chooses the best rings of imaginary quadratic integers to quantize each complex valued channel coefficients, thus achieving better performance than by working strictly with fixed rings.

For the proposed Polar Lattice aided Channel coded network coding scheme, so far the scheme is applicable to 2^r-PAM and 2^r-QAM, as each of the r layers of the multilevel polar lattice is composed of binary Polar codes. A future work for this area is the potential to use polar lattices constructed from non-binary polar codes for this channel coded scheme. If non-binary polar codes is used for each coded level of the polar lattice, a greater variety of PAM and QAM modulation with better shaping properties can be used with our proposed scheme. For example, we can choose non-binary Polar codes over prime finite fields to construct a polar lattice-aided channel coded network coding scheme over the ring of Eisenstein integers, which can be used with QAM constellation formed by coset leaders of Eisenstein integer partitions.

Bibliography

[1] B. Nazer and M. Gastpar, “Reliable physical layer network coding,” Proceedings of the IEEE, vol. 99, no. 3, pp. 438–460, 2011.

[2] P. Popovski and H. Yomo, “Physical network coding in two-way wireless relay channels,” in Proc. IEEE Int. Conf. Commun. (ICC), 2007, pp. 707–712.

[3] M.Wilson, K. Narayanan, H. Pster, and A. Sprintson, Joint physical layer coding and network coding for bidirectional relaying," Information Theory, IEEE Transactions on, vol. 56, pp. 5641 - 5654, nov. 2010. 5

[4] S. Zhang, S.-C. Liew, and P. Lam, On the synchronization of physical-layer network coding," in Information Theory Workshop, 2006. ITW '06 Chengdu. IEEE, pp. 404 - 408, oct. 2006.

[5] D. Gündüz, E. Tuncel, and J. Nayak, “Rate regions for the separated two-way relay channel,” in Proc. 46th Ann. Allerton Conf., Sep. 2008, pp. 1333–1340.

[6] A. S. Avestimehr, A. Sezgin, and D. N. C. Tse, “Capacity of the two-way relay channel within a constant gap,” European Transactions on Telecommunications, vol. 21, no. 4, pp. 363–374, Jun. 2010.

[7] S. Zhang and S. C. Liew, “Channel coding and decoding in a relay system operated with physical-layer network coding,” IEEE J. Sel. Areas Commun., vol. 27, no. 5, pp. 788–796, 2009.

[8] J. Liu, M. Tao, Y. Xu, and X. Wang, “Superimposed xor: A new physical layer network coding scheme for two-way relay channels,” in Proc. IEEE GLOBECOM, Honolulu, Hawaii, Nov. 2009, pp. 1–6.

[9] Q. Zhou, Y. Li, F. Lau, and B. Vucetic, “Decode-and-forward two-way relaying with network coding and opportunistic relay selection,” IEEE Transactions on Communications, vol. 58, no. 11, pp. 3070–3076, Nov. 2010.

[10] I.-J. Baik and S.-Y. Chung, “Network coding for two-way relay channels using lattices,” in Proc. IEEE Int. Conf. Commun. (ICC), Beijing, China, May 2008, pp. 3898–3902.

[11] W. Nam, S.-Y. Chung, and Y. H.Lee, “Capacity of the Gaussian Two-Way relay channel to within 1/2 bit,” IEEE Transactions on Information Theory, vol. 56, no. 11, pp. 5488–5494, 2010.

[12] P. Zhong and M. Vu, “Partial decode-forward coding schemes for the Gaussian two-way relay channel,” in Proc. IEEE Int. Conf. Commun. (ICC), Ottawa, Canada, Jun. 2012.

[13] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, “Network information flow,” IEEE Trans. Inf. Theory, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[14] S.-Y. R. Li, R. W. Yeung, and N. Cai, “Linear network coding,” IEEE Transactions on Information Theory, vol. 49, no. 2, pp. 371–381, Feb. 2003.

[15] B. Rankov and A. Wittneben. “Spectral efficient signaling for half-duplex relay channels. In Proc. of Asilomar Conference on Signals, Systems and Computers (ACSSC), Asilomar, CA, November 2005.

[16] S. Smirani, (2014), Physical-layer network coding for two way relay channels, Universite Paris Sud - Paris XI, PhD thesis, available online at <https://tel.archives-ouvertes.fr/tel-00962333 >

[17] S. Zhang, S. Liew, and P. Lam, “Hot topic: Physical-layer network coding,” in Proc. ACM Int. Conf. Mobile Comput. Netw., Los Angeles, CA, pp. 358–365, Sep. 2006.

[18] P. Popovski and H. Yomo, “The anti-packets can increase the achievable throughput of a wireless multi-hop network,” in Proc. IEEE Int. Conf. Commun., Istanbul, Turkey, Jun. 2006, pp. 3885–3890.

[19] B. Nazer and M. Gastpar, “Computing over multiple-access channels with connections to wireless network coding,” in Proc. IEEE Int. Symp. Inf. Theory, Seattle, WA, Jul. 2006, pp. 1354–1358.

[20] B. Nazer and M. Gastpar, “Compute-and-forward: Harnessing interference through structured codes,” IEEE Transactions on Information Theory, vol. 57, no. 10, pp. 6463–6486, October 2011.

[21] M. Gastpar, B. Nazer, “Algebraic Structure in Network Information Theory,” tutorial, ISIT 2011.

[22] F. Chen, D. Silva, and F. R. Kschischang, “Design Criteria for Lattice Network Coding,” 45th Annual Conference on Information Sciences and Systems (CISS), 2011.

[23] F. Chen, D. Silva, and F. R. Kschischang, “An algebraic approach to physical-layer network coding,” submitted to IEEE Transactions on Information Theory, 2012.

[24] N. Tunali, K. Narayanan, J. Boutros, and Y. Huang, “Lattices over Eisenstein integers for compute-and-forward,” 50th Annual Allerton Conference on Communication, Control, and Computing, Allerton, 2012.

[25] Q. T. Sun, J. Yuan, and K. W. Shum, “ Lattice Network codes Based on Eisenstein Integers,” in IEEE Transactions on Communications, vol.61, no.7, pp. 2713-2725, July 2013

[26] Y.-C. Huang, K. Narayanan and P.-C. Wang, “Adaptive Compute-and-Forward with Lattice Codes Over Algebraic Integers,” Submitted to IEEE Trans. Inf. Theory, available online at http://arxiv.org/pdf/1501.07740.pdf

[27] A. Sakzad, J. Harshan and E. Viterbo, “Integer-Forcing MIMO Linear Receivers Based on Lattice Reduction,”in IEEE Transactions on Wireless Communications, vol. 12, no. 10, pp. 4905-4915, October 2013.

[28] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, “Factoring polynomials with rational coefficients,” Mathematische Annalen, vol. 261, no 4, pp. 515-534, 1982.

[29] Y. H. Gan, C. Ling, and W. H. Mow, “Complex lattice reduction algorithm for low-complexity full-diversity MIMO detection, in IEEE Transactions on Signal Processing, vol. 57, pp. 27012710, February 2009.

[30] A. Sakzad, E. Viterbo, Y. Hong, and J. J. Boutros, “On the ergodic rate for compute-and-forward,” in Proc. of International Symposium on Network Coding (NetCod 2012), MIT University, Boston, MA, USA, pp. 131-136, 2012.

[31] A.Sakzad, E. Viterbo, J. J. Boutros, and Y. Hong, “Phase Precoding for the Compute-and-Forward Protocol,” Submitted to IEEE Transactions on Information Theory, available online at htt://arxiv.org/abs/1404.4157

[32] S. Sahraei and M. Gastpar, “Compute-and-Forward: Finding the Best Equation,” in IEEE 52nd Annual Allerton Conference on Communication, Control, and Computing, October 2014.

[33] U. Niesen and P. Whiting, “The degrees of freedom of the Compute-and-Forward”, IEEE Transactions on Information Theory, vol. 58, no. 8, pp.5214-5232, 2012.

[34] W. Liu and C. Ling, “Efficient integer coefficient search for compute-and-forward,” IEEE Transactions on Wireless Communications, vol. 15, no. 12, pp. 8039-8050, Dec 2016.

[35] J. C. Belfiore, “Lattice codes for the compute-and-forward protocol: The flatness factor,” in Information Theory Workshop (ITW), 2011 IEEE, pp. 1–4.

[36] Ali Osmane, Jean-Claude Belfiore "The Compute-and-Forward Protocol: Implementation and Practical Aspects," Submitted, 2011, can be downloaded from <http://arxiv.org/abs/1107.0300>.

[37] A. Mejri, G. Rekaya-Ben Othman, and J. C. Belfiore, "Lattice decoding for the compute-and-forward protocol," in Communications and Networking (ComNet), 2012 Third International Conference on, pp. 1–8.

[38] B. Nazer, "Successive compute-and-forward," in Proceedings of the 22nd Biennial International Zurich Seminar on Communication (IZS 2012).

[39] Y. Huang, N. Tunali, and K. Narayanan, "A compute-and-forward scheme for gaussian bi-directional relaying with inter-symbol interference," Communications, IEEE Transactions on, vol. PP, no. 99, pp. 1–9, 2013.

[40] A. Goldsmith and P. Varaiya, "Capacity of fading channels with channel side information," IEEE Transactions on Information Theory, vol. 43, no. 6, pp. 1986-1992, Nov. 1997.

[41] B. Hern and K. Narayanan, "Multilevel coding schemes for compute-and-forward," in Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on, pp. 1713–1717.

[42] O. Ordentlich, Z. Jiening, U. Erez, M. Gastpar, and B. Nazer, "Practical code design for compute-and-forward," in Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on, pp. 1876–1880.N. E. Tunali and K. R. Narayanan, "Concatenated signal codes with applications to compute and forward," in Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE, pp. 1–5.

[43] G. Wang, W. Xiang, and J. Yuan, "Outage performance for compute-and-forward in generalized multi-way relay channels," Communications Letters, IEEE, vol. 16, no. 12, pp. 2099–2102, 2012.

[44] Lili Wei,Wen Chen, "Efficient Compute-and-Forward Network Codes Search for Two-Way Relay Channel," Communications Letters, IEEE, vol. 16, no. 8, pp. 1204-1207, 2012.

[45] Tao Huang, Jinhong Yuan, and Jing Li, "Analysis of Compute-and-Forward with QPSK in Two-way Relay Fading Channels," Communications Theory Workshop (AusCTW), 2013, Australian, pp. 75-80.

[46] Jinming Wen; Baojian Zhou; Wai Ho Mow; Xiao-Wen Chang, “Compute-and-forward protocol design based on improved sphere decoding,” IEEE International Conference on Communications (ICC), pp. 1631-1636, 2015.

[47] A. Sakzad, J. Harshan, E. Viterbo, “Integer-Forcing Linear Receivers Based on Lattice Reduction Algorithms”, to IEEE Trans. on Inform. Theory, Sept, 2012

[48] Asma Mejri; Ghaya Rekaya-Ben Othman, “Efficient Decoding Algorithms for the Compute-and-Forward Strategy,” IEEE Transactions on Communications, vol. 63, pp. 2475-2485, 2015.

[49] J. Zhan, U. Erez, M. Gastpar, and B. Nazer, “Mimo compute-and-forward,” in Information Theory, ISIT 2009, IEEE International Symposium on, pp. 2848–2852.

[50] J. Zhan, B. Nazer, U. Erez and M. Gastpar, “Integer-Forcing Linear Receivers,” in IEEE Transactions on Information Theory, vol. 60, no. 12, pp. 7661-7685, December 2014.

[51] Mohsen Hejazi, Masoumeh Nasiri-Kenari, “Simplified Compute-and-Forward and Its Performance Analysis,” Submitted, 2013, can be downloaded from <http://arxiv.org/abs/1304.4428>.

[52] Shahab Ghasemi-Goojani and Hamid Behroozi, “Nested Lattice Codes for Gaussian Two-Way Relay Channels,” Submitted, 2013, can be downloaded from < http://arxiv.org/abs/1301.6291 >

[53] D. Wubben and Y. Lang, "Generalized sum-product algorithm for joint channel decoding and physical-layer network coding in two-way relay systems," in IEEE Globecom, Miami, USA, Dec. 2010.

[54] D. Wubbn, “Joint Channel Decoding and Physical-Layer Network Coding in Two-Way QPSK Relay Systems by a Generalized Sum-Product Algorithm,” 7th International Symposium on Wireless Communication Systems (ISWCS), 2010.

[55] Y. Lang, D. Wubben, and K.D. Kammeyer, “An Improved Physical Layer Network Coding Scheme for Two-Way Relay Systems,” in Proc. International ITG Workshop on Smart Antennas (WSA), Bremen, Germany, Feb. 2010.

[56] Y. Lang and D. Wubben, “Generalized Joint Channel Coding and Physical Network Coding for Two-way Relay Systems,” in IEEE Proc. Vehicular Technology Conference (VTC), Taipei, Taiwan, May 2010.

[57] H. J. Yang, Y. Choi, and J. Chun, “Modified high-order PAMs for binary coded physical-layer network coding,” IEEE Commun. Lett., vol. 14, pp. 689–691, Aug. 2010.

[58] X. S. Zhou, L.-L. Xie, and X. Shen, “Low-density parity-check codes for two-way relay channels,” in Proc. IEEE Veh. Technol. Conf. – Fall, pp. 1–5, 2010.

[59] “Pairwise Check Decoding for LDPC Coded Two-Way Relay Fading Channels,” 2010.

[60] P. Popovski and T. Koike-Akino, “Coded bidirectional relaying in wireless networks,” in New Directions in Wireless Communications Research, pp. 291–316. Springer US, 2009.

[61] S. Pfletschinger, “A practical physical-layer network coding scheme for the uplink of the two-way relay channel,” in Asilomar Conference on Signals, Systems, and Computers (ASILOMAR), Pacific Grove, CA, U.S.A., Nov., 2011.

[62] S. Zhang and S.-C. Liew, and L. Lu, “Physical-layer network coding: Tutorial, survey, and beyond,” in Physical Communication on "Network Coding and Its Application to Wireless Communications", pp 4-42, vol 6, March 2013.

[63] Z. Faraji-Dana, P. Mitran, “On Non-Binary Constellations for Channel-Coded Physical-Layer Network Coding,” IEEE Transactions on Wireless Communications, vol. 12 , pp. 312-319, 2013.

[64] J. Conway and N. Sloane, Sphere Packings, Lattices and Groups, Springer-Verlag, 1993.

[65] U. Erez and R. Zamir, “Achieving 1/2 log(1 +SNR) on the AWGN channel with lattice encoding and decoding,” IEEE Transactions on Information Theory, vol. 50, no. 22, pp. 2293–2314, Oct. 2004.

[66] C. Ling and J.-C. Belfiore, “Achieving AWGN channel capacity with lattice Gaussian coding,” IEEE Transactions on Information Theory, vol. 60, no. 10, pp. 5918–5929, Oct. 2014.

[67] N. Pietro, J. Boutros, G. Zémor, and L. Brunel. “Integer low-density lattices based on construction A,” IEEE Information Theory Workshop Proceedings (ITW), pp. 422–426, 2012.

[68] Y. Yan, L. Liu, C. Ling, and X. Wu., “Construction of Capacity-Achieving Lattice Codes-Polar Lattices,” https://arxiv.org/abs/1411.0187, 28th Sept., 2015.

[69] G. D. Forney, Jr., and L.-F. Wei, “Multidimensional constellations-Part I: Introduction, figures of merit, and generalized cross constellations,” IEEE J. Select. Areas Comm, pp. 877-892.

[70] J. H. Conway and N. J. A. Sloane, “A fast encoding method for lattice codes and quantizers,” IEEE Trans. Inform Theory, vol. IT-19 pp. 820-824, 1983.

[71] G JR Forney, “Coset Codes-Part II: Binary Lattices and Related Codes,” IEEE Transactions on Information Theory, pp. 55-67, Sept. 1999.

[72] G. Ungerboeck, “Channel coding with multilevel/phase signals,” IEEE Transactions on Information Theory, pp. 55-67, Jan. 1982.

[73] G. Ungerboeck, “Trellis coded modulation with redundant signal sets part I and II,” IEEE Communications Magazine, pp. 5-21, Feb. 1987.

[74] Conway and N. J. A. Sloane, “Fast quantizing and decoding algorithms for lattice quantizers and codes,” IEEE Transactions on Information Theory, vol. IT-28, pp. 227-232, 1982.

[75] J. H. Conway and N. J. A. Sloane, "Voronoi regions of lattices, second moments of polytopes, and quantization,” IEEE Transactions on Information Theory, vol. IT-2, pp. 211-226, 1982.

[76] W. H. Mow, “Fast decoding of the hexagonal lattice with applications to power efficient multi-level modulation systems”, Proc. 1992 ICCS/ISITA Communications on the Move, pp.370-373.

[77] T. M. Cover and J. A. Thomas, Elements of Information Theory, New York: Wiley, 2006.

[78] W, Liu, “Compute-and-Forward for Two-Way Relay,” 9th IEEE/IET International Symposium on CSNDSP, July, 2014.

[79] R. Gallager, “Low-density parity-check codes,” IRE Transactions on Information Theory, vol. IT-8, pp.21-28, Jan. 1962.

[80] R. Gallager, “Low-density parity-check codes,” M.I.T Press, 1963.

[81] M. Davey, Error-Correction Using Low-Density Parity-Check Codes, PhD Thesis, University of Cambridge, Cambridge, United Kingdom, 1999.

[82] D. MacKay, and R. Neal, “Near Shannon limit performance of low density parity check codes,” Electron. Lett., vol. 33, no. 6, March 13, 1997.

[83] R. M. Tanner, “A recursive approach to low complexity codes,” IEEE Transactions on Information Theory, vol. IT-42, pp.533-547, 1981.

[84] M. Davey and D.J.C. MacKay, “Low Density Parity Check Codes over GF(q),” IEEE Commun. Lett., vol. 2, pp. 165-167, June 1998.

[85] H. Song and J.R. Cruz, “Reduced-Complexity Decoding of Q-ary LDPC Codes for Magnetic Recording,” IEEE Trans. Magn., vol. 39, pp. 1081-1087, Mar. 2003.

[86] Shu Lin, Daniel J. Costello, *Error Control Coding*, 2nd Edition, Pearson Prentice Hall, 2004

[87] J. Harshan, E. Viterbo, and J.-C. Belore, “Construction of Barnes-Wall lattices from linear codes over rings,” Proc. IEEE Int. Symp. Inform. Theory, Cambridge, MA, pp. 3110-3114, July 1-6, 2012.

[88] W. Kositwattanarerk and F. Oggier, “On Construction D and related constructions of lattices from linear codes,” in Int. Workshop on Coding and Cryptography (WCC), 2013.

[89] N. Tunali, K. Narayanan, and H. Pfister, “Spatially-coupled low density lattices based on construction A with applications to Compute-and-Forward,” IEEE Information Theory Workshop Proceedings (ITW), pages 1–5, 2013.

[90] F. Chen, D. Silva, and F. R. Kschischang “Blind Compute-and-Forward,” Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on, pp. 403-407.

[91] W. Liu, “Physical Layer Network Coding Schemes for Two-Way Relay,” 9th IEEE/IET International Symposium on CSNDSP, July, 2014.

[92] W. Liu, “A new channel coding network coding scheme for two-way relay,” IEEE Science and Information Conference (SAI), Proceedings, pp.989-996, July, 2015.

[93] W. Liu, C. Ling, “LDA-Lattice Aided Network Coding for Two-Way Relay” IEEE Science and Information Conference (SAI), July, 2016.

[94] E. Arıkan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels,” IEEE Trans. Inform. Theory, vol. 55, pp. 3051–3073, 2009.

[95] E. Ankan, N. Hassan; M. Lentmaier; G. Montorsi, and J. Sayir, “Challenges and some new directions in channel coding,” IEEE Journal of Communications and Networks, vol. 17, pp. 328-338, Aug., 2015.

[96] Y. S. Park, Y. Tao, S. Sun, and Z. Zhang, “A 4.68gb/s belief propagation polar decoder with bit-splitting register file,” in VLSI Circuits Digest of Technical Papers, 2014 Symposium on, pp. 1–2, June 2014.

[97] I. Tal and A. Vardy, “List decoding of polar codes,” in Proc. IEEE Int. Symp. Inform. Theory (ISIT), (St. Petersburg, Russia), pp. 1–5, IEEE, 2011.

[98] R. Mori and T. Tanaka, “Performance of polar codes with the construction using density evolution,” IEEE Commun. Lett., vol. 13, pp. 519–521, July 2009.

[99] Tal and A. Vardy, “How to construct polar codes,” IEEE Transactions on Information Theory, vol. 59, pp. 6562–6582, Oct 2013.

[100] R. Pedarsani, S. H. Hassani, I. Tal, and I. Telatar, “On the construction of polar codes,” in Proc. IEEE Int. Symp. Inform. Theory (ISIT), (St. Petersburg, Russia), pp. 11–15, IEEE, 2011.

[101] Y. Yan and C. Ling, “A Construction of lattices from polar codes,” Information Theory Workshop (ITW), 2012 IEEE, pp. 124-128.

[102] Y. Yan, C. Ling, and X. Wu, “Polar lattices: Where Arıkan meets Forney,” in IEEE International Symposium on Information Theory (ISIT), 2013, pp. 1292–1296.

[103] P.J. Schreier and L. L. Scharf, “Complex-Valued Signal Processing: The Proper Way to Deal with Impropriety,” IEEE Transactions on Signal Processing, vol. 59, issue 11, pp. 5101-5125, 2011.

[104] B. Nazer, and M. Gastpar, “The case for structured random codes in network communication theorems,” in Proc. Inform. Theory Workshop (ITW), CA, Sept. 2007

[105] E. Arıkan, “A performance comparison of polar codes and reed-muller codes,” IEEE Comm. Lett., vol. 12, no. 6, pp. 447 –449, June 2008.

[106] G. D. Forney Jr., M. Trott, and S.-Y. Chung, “Sphere-bound-achieving coset codes and multilevel coset codes,” IEEE Transactions on Information Theory, vol. 46, no. 3, pp. 820–850, May 2000.

[107] Y-C. Huang, K. Narayanan, P-C. Wang, “Adaptive Compute-and-Forward with Lattice Codes over Algebraic Integers,” IEEE International Symposium on Information Theory (ISIT), 2012.

[108] U. Wachsmann, R. Fischer, and J. Huber, “Multilevel codes: Theoretical concepts and practical design rules,” IEEE Transactions on Information Theory, vol. 45, no. 5, pp. 1361–1391, July 1999.

Appendix

Proof of Theorem 2

The proof for Theorem 2 consists of two parts, and follows the same structure as the proof for the real valued channel case in [32]. We first prove Theorem 2 for the case of C-F search over ring of Gaussian Integers. Then we adapt the proof to C-F over ring of Eisenstein integers.

Using the definition of DP^k decomposable matrix originally proposed in [32] for complex valued channels and complex valued integer coefficients, the cost function $f(\mathbf{a}) = \mathbf{a}\mathbf{M}\mathbf{a}^H$ can be written as

$$f(\mathbf{a}) = \sum_{m=1}^{L}(D_{mm} - P_{mm})a_m a_m^* - \sum_{m=1}^{L}\sum_{n=1,n\neq i}^{L} P_{mn}a_m a_n^* \tag{1}$$

Using Wirtinger calculus [103], we know that for a real-valued function $f(a) \equiv g(a, a^*)$ with a complex variable a, at the minimum or maximum point, the following condition holds

$$\frac{\partial f}{\partial a} = \frac{\partial f}{\partial a^*} = 0 \tag{2}$$

Suppose that the optimal value for all the individual elements of a_m^{opt} are known except for one element a_n. By taking partial derivative with respect to a_n^*, the optimal complex value of a_n can be taken to be equal to,

$$\frac{\sum_{m=1,m\neq n}^{L} P_{mn}a_m^{opt}}{D_{nn} - P_{nn}} \tag{3}$$

Note that $f(a)$ is a convex elliptic paraboloid in a_n, since $D_{nn} - P_{nn} = M_{nn}$ is the diagonal element of a positive definite matrix. Therefore the optimal Gaussian integer value for a_n is the closest Gaussian integer to its optimal complex value. Taking the closest Gaussian integer to the complex valued solution, we have,

$$a_n^{opt} = \left\lfloor \frac{\sum_{m=1,\, m\neq n}^{L} P_{mn}a_m^{opt}}{D_{nn} - P_{nn}} \right\rceil_{Z[j]} \tag{4}$$

The one circumstance where the above expression fails is when in the optimal vector, we have $a_m^{opt} = 0,\ m = 1...L,\ m \neq n$ as this would result in the all zero vectors $\mathbf{a}^{\text{opt}} = 0$ when substituting the terms into (4). In this particular scenario, note that the ensuing cost function $f(\mathbf{a}^{opt}) = M_{nn} a_n^{\text{opt}} (a_n^{\text{opt}})^*$ is minimised when $\left| a_n^{\text{opt}} \right| = 1$ which occurs when a_n^{opt} is an Gaussian integer unit of the form $j, -j,\ 1$ or 1. Therefore an examination of all the length L Gaussian integer vectors of unit norm will have to be checked on top of the proposed algorithm for finding $\mathbf{a}^{\text{opt}}$. From equation (4), we can deduce four inequalities,

$$\text{Re}\left(a_n^{opt} + \frac{1}{2} \right) \geq \text{Re}\left(\frac{\sum_{m=1,\, m \neq n}^{L} P_{mn} a_m^{opt}}{D_{nn} - P_{nn}} \right) \tag{5}$$

$$\text{Re}\left(a_n^{opt} - \frac{1}{2} \right) \leq \text{Re}\left(\frac{\sum_{m=1,\, m \neq n}^{L} P_{mn} a_m^{opt}}{D_{nn} - P_{nn}} \right) \tag{6}$$

$$\text{Im}\left(a_n^{opt} + \frac{1}{2} j \right) \geq \text{Im}\left(\frac{\sum_{m=1,\, m \neq n}^{L} P_{mn} a_m^{opt}}{D_{nn} - P_{nn}} \right) \tag{7}$$

$$\text{Im}\left(a_n^{opt} - \frac{1}{2} j \right) \leq \text{Im}\left(\frac{\sum_{m=1,\, m \neq n}^{L} P_{mn} a_m^{opt}}{D_{nn} - P_{nn}} \right) \tag{8}$$

First we work with equation (5). The key recurring assumptions in the derivation that follows is that the term $D_{nn} - P_{nn} = M_{nn}$ and its individual parts are real valued by definition. We multiply both sides by the term $D_{nn} - P_{nn} = M_{nn}$ and add the term $\text{Re}(a_n^{\text{opt}} P_{nn})$ to both sides,

$$\text{Re}\left(\left(a_n^{opt} + \frac{1}{2} \right) D_{nn} \right) \geq \text{Re}\left(\sum_{m=1}^{L} P_{mn} a_m^{opt} + \frac{1}{2} P_{nn} \right). \tag{9}$$

Dropping the non-negative term $\text{Re}\left(\frac{1}{2} P_{nn} \right)$ we obtain,

$$\text{Re}\left(\left(a_n^{opt} + \frac{1}{2} \right) D_{nn} \right) \geq \text{Re}\left(\sum_{m=1}^{L} P_{mn} a_m^{opt} \right). \tag{10}$$

Inequality (10) can be shown to be strict using the following reasoning. In (9), when $P_{nn} > 0$

We know that inequality (40) holds strictly. The case we have to consider is when $P_{jj} = 0$. Using the fact that $\mathbf{P}$ is positive semi-definite, we know that if $P_{nn} = 0$ then $P_{nn} = 0$ for $m = 1,...,L$ and the inequality becomes,

$$\mathrm{Re}\left(a_j^{opt} + \frac{1}{2} \right) D_{jj} > 0 \tag{11}$$

Given that $\mathrm{Re}(a_n^{opt})$ is integer and $D_{nn} - P_{nn} > 0$, then the inequality (10) is shown to be strict,

$$\mathrm{Re}\left(\left(a_n^{opt} + \frac{1}{2} \right) D_{nn} \right) > \mathrm{Re}\left(\sum_{m=1}^{L} P_{mn} a_m^{opt} \right) \tag{12}$$

Rewriting (12)

$$\mathrm{Re}\left(\left(a_n^{opt} + \frac{1}{2} \right) \right) > \frac{\mathrm{Re}\left(\sum_{m=1}^{L} P_{mn} a_m^{opt} \right)}{D_{nn}} \quad , \quad n = 1,..,L \tag{13}$$

Rewriting the inequality in vector format and substituting $\mathbf{P} = \mathbf{V}^H \mathbf{V}$,

$$\mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} + \frac{1}{2}\mathbf{1} \right) \right) > \mathrm{Re}\left(\mathbf{a}^{\mathrm{opt}} \mathbf{P}\, \mathbf{D}^{-1} \right) = \mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} \mathbf{V}^H \right) \mathbf{V}\, \mathbf{D}^{-1} \right) \tag{14}$$

Now taking (6) and applying the same steps used to derive (14) we can derive the inequality,

$$\mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} - \frac{1}{2}\mathbf{1} \right) \right) < \mathrm{Re}\left(\mathbf{a}^{\mathrm{opt}} \mathbf{P}\, \mathbf{D}^{-1} \right) = \mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} \mathbf{V}^H \right) \mathbf{V}\, \mathbf{D}^{-1} \right) \tag{15}$$

Let's define $\mathbf{x} = \mathbf{a}^{\mathrm{opt}} \mathbf{V}^H$. Combining inequalities (14) and (15), we obtain

$$\mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} - \frac{1}{2}\mathbf{1} \right) \right) < \mathrm{Re}\left(\mathbf{x}\mathbf{V}\mathbf{D}^{-1} \right) < \mathrm{Re}\left(\left(\mathbf{a}^{\mathrm{opt}} + \frac{1}{2}\mathbf{1} \right) \right) \tag{16}$$

Now we work with inequalities (7) and (8) dealing with imaginary components of the closest Gaussian integer solution (4), using the same steps we used to derive (14) and (15). Namely, with (7) and (8), we multiply both sides by $D_{nn} - P_{nn}$ and then add $\mathrm{Im}(a_n^{\mathrm{opt}} P_{nn})$ to both sides. The term $\mathrm{Im}\left(\frac{1}{2} P_{nn}\right)$ equals to zero and consequently, the ensuing inequality can be shown to be strict. Rearranging and rewriting the inequality, we are then able to derive inequalities (17) and (18) from (7) and (8) respectively,

$$\mathrm{Im}\left(\left(\mathbf{a}^{opt} + \frac{1}{2}\mathbf{1} \right) \right) > \mathrm{Im}\left(\mathbf{x}\mathbf{P}\, \mathbf{D}^{-1} \right) = \mathrm{Im}\left(\left(\mathbf{a}^{\mathrm{opt}} \mathbf{V}^H \right) \mathbf{V}\, \mathbf{D}^{-1} \right) \tag{17}$$

$$\mathrm{Im}\left(\left(\mathbf{a}^{opt}-\frac{1}{2}\mathbf{1}\right)\right)<\mathrm{Im}\left(\mathbf{xP}\,\mathbf{D}^{-1}\right)=\mathrm{Im}\left(\left(\mathbf{a}^{\mathrm{opt}}\mathbf{V}^{H}\right)\mathbf{V}\,\mathbf{D}^{-1}\right) \tag{18}$$

Inequalities (17) and (18) can be combined to obtain,

$$\mathrm{Im}\left(\left(\mathbf{a}^{opt}-\frac{1}{2}\mathbf{i1}\right)\right)<\mathrm{Im}\left(\mathbf{xVD}^{-1}\right)<\mathrm{Im}\left(\left(\mathbf{a}^{\mathrm{opt}}+\frac{1}{2}\mathbf{i1}\right)\right) \tag{19}$$

Finally based on (16) and (19), we can deduce that:

$$\mathbf{a}^{\mathrm{opt}}=[\mathbf{xVD}^{-1}]_{Z[j]} \tag{20}$$

This concludes proof of Theorem 1 for C-F over ring of Gaussian integers. The proof for Theorem 2 over the ring of Eisenstein integers follows the same structure as the proof for Gaussian integers. After performing differentiation on the cost function $f(a,a^{*})$ in (1) and finding the optimal complex value for a_n, we can take the closest Eisenstein integer to the optimal complex valued solution to be,

$$a_n^{\mathrm{opt}}=\left\lfloor\frac{\sum_{m=1,\,m\neq n}^{L}P_{mn}a_m^{\mathrm{opt}}}{D_{nn}-P_{nn}}\right\rceil_{Z[\omega]} \tag{21}$$

The only scenario where the above expression fails is when in the optimal vector, we have $a_m^{opt}=0,\ m=1...L,\ m\neq n$ as this would result in the all zero vectors $\mathbf{a}^{\mathrm{opt}}=0$ when substituting the terms into (21). Therefore, as in the Gaussian integer case, an examination of all length L Eisenstein integer vectors of unit norm will also have to be checked individually.

Since every Eisenstein integer in the complex domain can be written in the form of $a+b\omega$ where $a,b\in\mathrm{Z}$ and $=-\frac{1}{2}+j\frac{\sqrt{3}}{2}$, we can express any complex valued point $c+dj$, as a point $u+v\omega$ in a new domain, where $u,v\in\mathrm{R}$, are more precisely given by $u=c+\frac{d}{\sqrt{3}}$ and $v=\frac{2d}{\sqrt{3}}$. By expressing the optimal complex valued a_n^{opt} in $u+v\omega$ form, and working with the structure of the A_2 lattice, we can deduce that the nearest Eisenstein integer to any complex point denoted equivalently in $u+v\omega$ form, is one from the following four neighbouring Eisenstein integers $\lfloor u\rfloor+\lfloor v\rfloor\omega$, $\lfloor u\rfloor+\lceil v\rceil\omega$, $\lceil u\rceil+\lfloor v\rfloor\omega$, $\lceil u\rceil+\lceil v\rceil\omega$, with the smallest Euclidean distance in real and imaginary domain from the point $u+v\omega$. Hence for the optimal Eisenstein integer solution in (21) written in written in $u+v\omega$, by working with the real-valued u component of a_n^{opt} denoted by P_{Re}, and the v component of a_n^{opt} denoted by P_{ω}, we can derive a set of inequalities to represent each of the four possible Eisenstein

integer points that maybe represent the optimal Eisenstein integer solution, and show that Theorem 2 holds when the optimal Eisenstein integers solution takes on any of the four forms of Eisenstein integers, and therefore confirming that Theorem 2 holds for the optimal Eisenstein integer solution to the SVP in C-F over Eisenstein integers in general.

First we work with the possibility that the closest Eisenstein integer to the point $u+v\omega$ is given by $\lfloor u \rfloor + \lceil v \rceil \omega$. Following the same structure of derivation as for the Gaussian integer case, but instead working with the real and ω components instead of real and imaginary components two resulting bounds we can obtain are,

$$\mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}}\right) \le \mathrm{P}_{\mathrm{Re}}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}} + \mathbf{1}\right) \tag{22}$$

$$\mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}}\right) \le \mathrm{P}_{\omega}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}} + \omega\mathbf{1}\right) \tag{23}$$

Next we work with the possibility that the optimal Eisenstein integer solution a_n^{opt} is given by the form $\lfloor u \rfloor + \lceil v \rceil \omega$. From (21) we can obtain the bounds (24), (25) which shows that in the case where the A_2 decoder decodes to $\lfloor u \rfloor + \lceil v \rceil \omega$, the optimal Eisenstein integer solution a_n^{opt} again takes the form in Theorem 2.

$$\mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}}\right) \le \mathrm{P}_{\mathrm{Re}}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}} + \mathbf{1}\right) \tag{24}$$

$$\mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}} - \omega\mathbf{1}\right) \le \mathrm{P}_{\omega}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}}\right) \tag{25}$$

Now in the case where the closest Eisenstein integer to the point $u+v\omega$ is given by $\lceil u \rceil + \lfloor v \rfloor \omega$, we obtain the bounds,

$$\mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}} - \mathbf{1}\right) < \mathrm{P}_{\mathrm{Re}}\left(\mathbf{xVD}^{-1}\right) \le \mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}}\right) \tag{26}$$

$$\mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}}\right) \le \mathrm{P}_{\omega}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}} + \omega\mathbf{1}\right) \tag{27}$$

And in the case where the closest integer to the point $u+v\omega$ is given by $\lceil u \rceil + \lceil v \rceil \omega$, we obtain the bounds,

$$\mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}} - \mathbf{1}\right) < \mathrm{P}_{\mathrm{Re}}\left(\mathbf{xVD}^{-1}\right) \le \mathrm{P}_{\mathrm{Re}}\left(\mathbf{a}^{\mathrm{opt}}\right) \tag{28}$$

$$\mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}} - \omega\mathbf{1}\right) \le \mathrm{P}_{\omega}\left(\mathbf{xVD}^{-1}\right) < \mathrm{P}_{\omega}\left(\mathbf{a}^{\mathrm{opt}}\right) \tag{29}$$

Therefore, we can see that the four neighbouring Eisenstein integers to the optimal complex value solution all satisfy the form given in Theorem 2. Since we know the closest Eisenstein integer, or the optimal Eisenstein integer solution to the SVP, has to be one of the four

Eisenstein integer points, we can therefore conclude that the optimal Eisenstein integer solution abides by the form indicated in Theorem 2.