Title	無線通信システムにおけるコンピュートアンドフォワード方式の応用
Author(s)	HASAN, MOHAMMAD NUR
Citation	
Issue Date	2020-09
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/16998
Rights	
Description	Supervisor:KURKOSKI Brian Michael, 先端科学技術研究科, 博士

Applications of Compute-and-Forward Methods in Wireless Communication Systems

Mohammad Nur Hasan

Japan Advanced Institute of Science and Technology

Doctoral Dissertation

Applications of Compute-and-Forward Methods in Wireless Communication Systems

Mohammad Nur Hasan

Supervisor : Professor Brian Kurkoski

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
Information Science

September 2020

Supervised by	:	Professor Brian Kurkoski
Reviewed by	:	Professor Tad Matsumoto
		Professor Razvan Beuran
		Professor Emanuele Viterbo
		Professor Khoirul Anwar

Abstract

Interference is a fundamental issue in wireless communication systems, in which multiple transmissions occur at the same time over a common medium. Due to the tremendous increase of number of wireless communication devices in the recent years, this issue has now become more crucial than ever. Conventionally, in the design of a wireless communication system, interference is usually avoided because it may substantially limit the reliability and the throughput of the system. However, with massive number of wireless communication devices and limited wireless communication resources, avoiding interference becomes almost impossible. Compute-and-forward is a new technique that deals with this issue elegantly. Rather than avoiding interference or treating it as noise, compute-and-forward embraces and exploits it by computing linear functions of the transmitted messages directly from interfering signal. Thus, it allows simultaneous transmissions, which consequently results in an increase of network throughput.

Owing to its promising advantages, compute-and-forward has found many applications in various wireless communication scenarios such as multi-source multi-relay channels, two-way relay channels, multiple-access relay channels, multiple-input multiple-output (MIMO) systems, etc. In this dissertation, we study applications of compute-and-forward methods in wireless communication systems, investigate the issues that arise in the existing work, and then propose some solutions to solve them such that the system performance, e.g., network throughput and outage probability, is improved. In particular, this dissertation focuses on two different wireless communication scenarios: multiple-access relay channels and precoded MIMO systems.

In the multiple-access relay channels with compute-and-forward, the destination and the relay compute linear functions of the transmitted messages which are then used by the destination to recover the transmitted messages. The main issue in this system is that the final linear functions obtained by the destination must be linearly independent, and hence, cooperation between the destination and the relay must be carefully designed. To cope with this issue, we propose two cooperation strategies. In the first strategy, the relay helps the destination by forwarding its "local best" linear functions without taking into account whether it is linearly independent of that of the destination. While in the second strategy, the relay forwards its optimal linear function ensuring the linear functions obtained by the destination are linearly independent. We show that both of the strategies outper-

form existing strategies available in the literature. It is also shown that the second strategy achieves better error performance compared to the first one with the cost of additional overhead for feedback.

For precoded MIMO systems, we are particularly interested in the unitary precoded integer-forcing (UPIF) MIMO where a unitary matrix is used as the precoder matrix for MIMO systems employing integer-forcing linear receivers. Integer-forcing is essentially another form of compute-and-forward. It has been shown that UPIF achieves full diversity gain while allowing full transmission rate. However, it is not easy to find the optimal unitary precoder matrix. No efficient algorithm is available for this problem. Instead of unitary precoder matrices, we propose orthogonal precoder matrices for the same systems. We show that not only has lower complexity, the proposed orthogonal precoder has performance advantage in terms of achievable rate and error rate. Further, we propose an efficient algorithm based on the steepest gradient algorithm that exploits the geometrical properties of orthogonal matrices as a Lie group. The proposed algorithm has low complexity and can be easily applied to an arbitrary MIMO configuration. We also show that the proposed orthogonal precoder outperforms the X-precoder, a precoder designed specifically for quadrature amplitude modulation (QAM), in high-order QAM schemes, e.g., 64- and 256-QAM.

Keywords: Lattice network coding, multiple-access relay channel, compute-and-forward, integer-forcing, MIMO

Acknowledgments

I want to take this opportunity to express my appreciation for numerous individuals who have made my journey in pursuing doctoral degree a rewarding experience.

I am deeply grateful towards my supervisor, Prof. Brian Kurkoski for giving me the opportunity and freedom to pursue my research under his supervision. His support and guidance are essential during my study in JAIST. Without him, I would never have completed this dissertation. I am also grateful towards Prof. Tad Matsumoto. His critical comments and suggestions certainly enhanced the quality of my research and this dissertation.

I would like to also to express my gratitude towards Porf. Emanuele Viterbo and Prof. Amin Sakzad for their hospitality during my visit in Melbourne University. Even though the visit was only three months, it was very fruitful; owing to their help and expertise.

I am deeply thankful to my dearest family; my wife Dian Novitasari, my boy Rizqi, my little princess Hikari, and four of my parents. Love and support from them have played a big part in this journey. Their presence are absolutely a power that has helped me achieve what I have achieved.

Table of Contents

List of Figures

Abbreviations

AWGN	Additive White Gaussian Noise
CF	Compute-and-Forward
CSI	Channel State Information
CSIT	Channel State Information at Transmitters
DMT	Diversity Multiplexing Tradeoff
EM	Electromagnetic
FB	Feedback
IF	Integer-Forcing
IF-MIMO	Multiple-Input Multiple-Output with Integer-Forcing
HKZ	Hermite-Korkine-Zolotareff
LLL	Lenstra–Lenstra–Lovász
LNC	Lattice Network Coding
MARC	Multiple-Access Relay Channel
MARC-CF	Multiple-Access Relay Channel with Compute-and-Forward
MIMO	Multiple-Input Multiple-Output
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
PLNC	Physical-Layer Network Coding
QAM	Quadrature Amplitude Modulation
SD	Sphere Decoding

SE Schnorr-Euchner

SG Steepest Gradient

SIC Successive Interference Cancellation

SISO Single-Input Single-Output

SNR Signal-to-Noise Power Ratio

SVD Singular Value Decomposition

TWRC Two-Way Relay Channel

UPIF Unitary Precoded Integer-Forcing

WER Word-Error-Rate

XOR Exclusive Or

ZF Zero-Forcing

Notations

$\mathbf{0}$	Zero vector
$\mathbf{a}$	(Gaussian) Integer coefficient vector
$\mathbf{A}$	(Gaussian) Integer coefficient matrix
$\mathbf{B}$	Equalizing matrix
$\mathbb{C}$	Complex number
$\mathbf{D}$	Diagonal matrix from SVD decomposition of $\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{V}^H$
$\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s})$	Lattice codebook constructed from Λ_c and Λ_s, also denoted as only $\mathcal{C}$
$\mathbb{F}_p$	Finite field of size p
$\mathbf{H}$	Channel Matrix
$\mathbf{I}$	Identity matrix
M	Number of sources or transmit antennas
n	Code length
$O(M)$	Orthogonal group of dimension M
$\mathbf{P}$	Orthogonal precoder matrix
$\check{\mathbf{P}}$	Unitary precoder matrix
$\mathbf{Q}$	Coefficient matrix in a finite field domain
R	Code rate
R_cp	Computation rate
R_IF	Achievable rate of integer-forcing MIMO
R_t	Target rate

$\mathbb{R}$	Real numbers
$U(M)$	Unitary group of dimension M
$\mathbf{V}$	Unitary matrix from SVD decomposition of $\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{V}^H$
$\mathcal{V}_\Lambda$	Fundamental Voronoi region of lattice Λ
$\mathbf{w}$	Information message vector randomly generated from $\mathbb{F}$
$\mathbf{W}$	Unitary matrix from SVD decomposition of $\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{V}^H$
$\mathbf{x}$	Transmitted lattice codeword (vector)
$\mathbf{X}$	Transmitted lattice codewords from all transmitters or antennas (matrix)
$\mathbf{y}$	Received signal (vector)
$\mathbf{Y}$	Received vector at all antennas (matrix)
$\mathbf{y}_{\text{eff}}$	Effective received signal (vector)
$\mathbf{Y}_{\text{eff}}$	Effective received vector at all antennas (matrix)
$\mathbf{z}$	Noise vector
$\mathbf{Z}$	Noise matrix
$\mathbf{z}_{\text{eff}}$	Effective noise vector
$\mathbb{Z}$	Integer numbers
$\mathbb{Z}[i]$	Integer Gaussian numbers
δ_{ij}	Distance from terminal i to terminal j
γ	Signal-to-noise power ratio (SNR)
$\lambda_l(\mathbf{G})$	l-th successive minimum of a lattice with a generator matrix $\mathbf{G}$
Λ_{c}	Coding lattices
Λ_{s}	Shaping lattices
σ^2_{eff}	Variance of effective noise
$\mathtt{SNR}_{\text{eff}}$	The minimum effective SNR of channel streams

Chapter 1

Introduction

1.1 Background and Motivation

Wireless communication systems have become indispensable parts in our life. This can be seen from the fact that the number of connected wireless devices around us is dramatically increasing. A recent study shows that over the next few years, wireless data traffic will increase by multiple orders of magnitude [4]. This tremendous growing traffic is a result of the proliferation of mobile devices such as tablets, smart-phones and the growing demands for high-quality data services such as video on demand, online gaming, social networking. Meanwhile, the wireless communication resources, i.e., wireless radio spectrum and power, are limited. Thus, it is a major challenge for wireless industry to cope with this issue.

With a huge number of wirelessly-connected devices and limited wireless radio spectrum, it is quite likely that multiple devices transmit their information simultaneously via a common wireless medium. This situation results in interference that degrades the reliability of the information transmission. In fact, interference has long been one of fundamental wireless communication problems. In the conventional design of a wireless communication system, interference is usually avoided because it may substantially limit the reliability and the throughput of the system. However, in this current situation, avoiding interference becomes harder than ever.

A few years back, the way of thinking about interference changed with the introduction of network coding [5] and physical-layer network coding (PLNC) [6–9]. Rather than treating interference as an unwanted enemy, network coding and in particular PLNC embrace and make use of it to improve wireless resource utilization. Network coding was initially proposed for wired networks to boost network throughput by allowing intermediate nodes to forward functions of their received packets rather than just route them. This idea was then extended to the physical layer, and hence the name, where the network coding operation occurs naturally in superimposed electromagnetic (EM) waves. It is widely understood that when multiple EM waves come together in the same medium, they add. This addition operation can be considered as an aspect of network coding that is performed by nature. Because PLNC allows multiple terminals to make transmission simultaneously, it is easy to see that it can significantly improve network throughput. Due to this promising advantage, a large number of strategies have been proposed to realize PLNC. For examples, Katti *et al.* proposed analog network coding in [10] and XORs in the air in [11]. Other PLNC schemes can be found in [6–9, 12, 13].

Compute-and-forward [14] is one of PLNC schemes that has drawn a lot of attention. It was proposed by Nazer and Gastpar based on nested lattice codes [15, 16] which have been proven to achieve additive white Gaussian noise (AWGN) channel capacity [17]. Compute-and-forward exploits the property of lattice codes where a superposition or a linear combination of codewords is another codeword. This means that a linear combination can be decoded directly without decoding each original codeword. Compute-and-forward does not only simplifies the way of computing a function of the transmitted messages, but also tackles a fundamental problem faced by other PLNC schemes, the fading problem, which is another impairment of the wireless medium. Unlike other strategies, compute-and-forward infers a linear combinations of the transmitted messages by introducing integer coefficients that are selected based on fading coefficients. It turns out that this feature allows compute-and-forward to achieve higher spectral efficiency and to handle a variety of wireless communication scenarios.

Since its introduction, compute-and-forward has found various applications such as the two-way relay channels (TWRC) [18, 19], multi-source multi-relay channels [14], multiple-access relay channels (MARC) [20–22], and multiple-access channels (MAC) [23]. In some wireless scenarios, compute-and-forward has been shown to be preferable compared to other network coding strategies. As an example, we may take a look at the application of compute-and-forward to TWRC [18, 19], which is a classic network scenario where network coding enhances network throughput significantly [8, 24, 25]. The achievable-rate performance of compute-

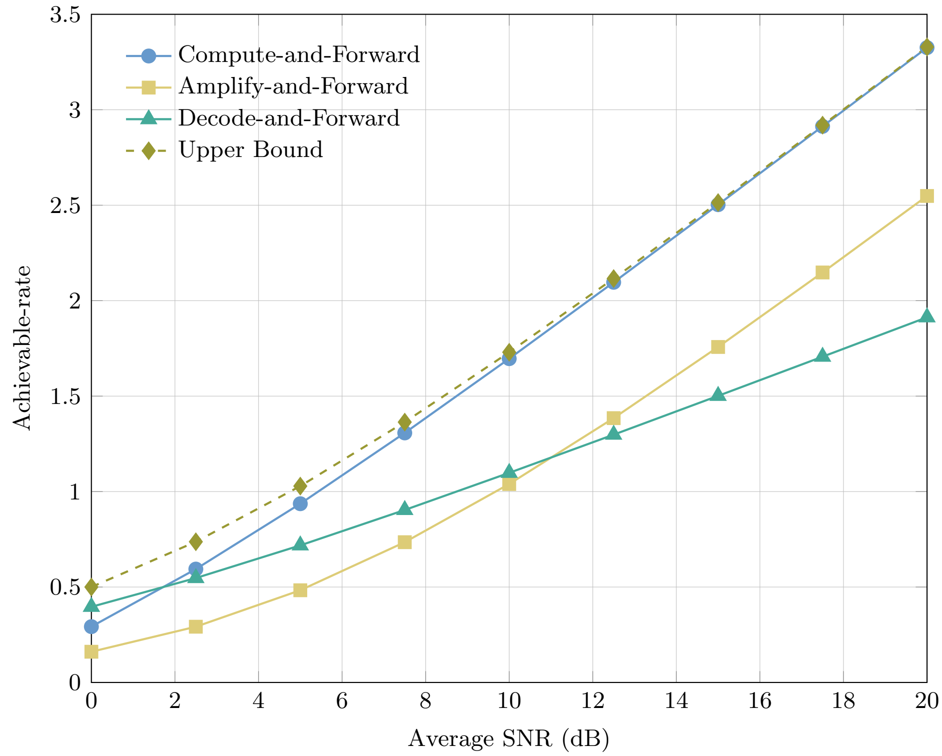


Fig. 1.1. Achievable-rates of compute-and-forward, decode-and-forward, and amplify-and-forward for the two-way relay channels (TWRC) [1].

and-forward, decode-and-forward, and amplify-and-forward strategies for TWRC is presented in Fig. 1.1, see [1] for details. It can be clearly seen that compute-and-forward offers significantly higher achievable rate than the others. In high signal-to-noise power ratio (SNR), compute-and-forward achieves rates close to the upper-bound.

Besides its application to multi-terminal relaying networks, compute-and-forward has also been applied to multiple-input multiple-output (MIMO) systems [3, 26]. The idea of compute-and-forward is extended to create a new type of linear MIMO receivers called integer-forcing linear receivers [26]. Instead of attempting to recover the transmitted messages directly, integer-forcing receivers compute integer linear combinations of the transmitted codewords first and then decodes the transmitted messages by solving a simple linear equation system. It has been shown that integer-forcing linear receivers yields higher achievable rates compared to the widely known zero-forcing (ZF) and minimum mean square error (MMSE) receivers [27] with nearly the same decoding complexity as those for slow-fading channels. Furthermore, integer-forcing receivers have been proven to achieve the optimal diversity-multiplexing tradefoff [28].

Despite its promising advantages, there are some issues that need to be tackled for applying compute-and-forward to a wireless network system. For instance, when applying compute-and-forward to a multi-source multi-relay network, linear combinations obtained by the final destination must be linearly independent. Therefore, a careful design is needed for applying compute-and-forward to a certain network scenario. This dissertation focuses on applications of compute-and-forward to some wireless scenarios, investigate the potential problems, and propose solutions for them. The final objective is to improve the network performance, e.g., error rate and throughput, of the considered networks.

1.2 Contributions

This dissertation has two major contributions: (i) two new cooperation strategies for employing compute-and-forward to multiple-access relay channels (MARC) and (ii) a new orthogonal precoding scheme for MIMO with integer-forcing. Short summaries are given below, but the details are provided in Chapters 3 and 4.

1.2.1 Cooperation Strategies for MARC with Compute-and-Forward

In MARC, multiple sources want to deliver their information to one common destination with the help of one relay terminal. When compute-and-forward is employed, the sources are allowed to transmit their information simultaneously to both the destination and the relay, which results in an increase of the network throughput. The relay computes an integer linear combination of the transmitted messages and forwards it to the destination. The destination then tries to recover the original messages from its own linear combinations and the one forwarded by the relay. The main challenge of this network scenario is that to correctly decode the transmitted messages, the linear combinations obtained by the destination needs to be linearly independent. Therefore the relay somehow has to take into account the linear combinations possessed by the destination when selecting its linear combination. Some effort has been made to cope with this challenge. For example, Soussi *et al.* [20] proposed a global optimization for finding global optimal linear combinations. Their proposed approach successfully improves the achievable rate. However, this improvement comes with the cost of a large communication overhead since it assumes that channel state information (CSI) are known to all nodes. Insausti *et al.* [22] propose a further improvement by treating a linear combination forwarded by the relay as a helper. Both [20] and [22] focused on improving achievable rate performance and ignore the outage probability performance which is one of the main objectives of cooperative or relaying networks. We found that both of them fail to achieve full diversity gain. In this dissertation, we propose two cooperation strategies for MARC with CF. We show that the proposed strategies outperform the existing strategies in terms of outage probability. We also show that one of our proposed strategies achieves full diversity gain. See Chapter 3 for more details.

1.2.2 Orthogonal Precoder for Integer-Forcing MIMO

The compute-and-forward idea was extended to MIMO systems producing a new MIMO linear receiver named as integer-forcing [26]. It was shown that integer-forcing achieves diversity-multiplexing tradefoff [28, 29]. In a recent work, Sakzad and Viterbo proposed a precoding scheme for MIMO with integer-forcing where precoder matrices are selected from unitary groups.[1] It was shown that their precoding scheme achieves full diversity gain while allowing full-rate transmission. The main issue in this scheme is that it is hard to find the optimal unitary precoder matrices. To the best of our knowledge, there is no efficient algorithm available to solve this problem in the literature. In this dissertation, instead of unitary groups, we propose a similar precoding scheme based on orthogonal groups. It is quite intuitive that by selecting precoder matrices from orthogonal groups, the complexity of the system can be greatly reduced. One may think that the complexity advantage comes with the cost of performance. However, we show an interesting result that the performance, e.g., achievable rate and error-rate, of the orthogonal precoding scheme is better than the unitary one. Further, we propose an efficient algorithm for finding "good" orthogonal precoder matrices based on the steepest gradient algorithm with Lie group approach. In addition, we numerically show that for higher quadrature amplitude modulations (QAM), the proposed precoding scheme outperforms the X-precoder [30] even though it is designed specifically for QAM. For the details, see Chapter 4.

[1] Unitary groups are groups of unitary matrices.

1.3 Dissertation Outline

This dissertation is organized as follows.

In Chapter 1 (this chapter), we describe the background and motivation of this research. We then summarize the contributions of this research and describe the organization of this dissertation. The general notations adopted in this dissertation are presented at the end of this chapter.

Chapter 2 provides a brief introduction of background knowledge required in this research. It includes lattices (and lattice codes), and compute-and-forward which is the main topic of this research.

We discuss one application of compute-and-forward in wireless cooperative networks in Chapter 3. In particular, we investigate how to efficiently employ compute-and-forward in MARC where multiple sources want to deliver information to a common destination with the help of a relay terminal. Two new cooperation protocols for MARC employing compute-and-forward are proposed. Semi-theoretical and numerical analyses are given along with discussion of the advantages and disadvantages of the proposed protocols. The comparison between the proposed protocols and existing protocols are also presented.

Subsequently, we investigate another application of compute-and-forward to a precoded MIMO system in Chapter 4. We are particularly interested in the work of Sakzad and Viterbo [3] where a unitary precoder is employed in MIMO systems with integer-forcing.[2] They showed that their proposed scheme achieves full diversity while allowing full transmission rate. In this chapter, instead of a unitary precoder, we propose an orthogonal precoder for the same systems. In addition to its computational complexity advantage, the orthogonal precoder is shown to achieve better performance in terms of achievable rates, error-rate, and outage probability. We also propose an efficient algorithm based on steepest gradient descent with Lie groups approach for finding a "good" orthogonal precoder. Complexity analysis and numerical results are presented.

Finally, in Chapter 5, we summarize the general conclusions and direction of future work.

[2] Integer-forcing is essentially another form of compute-and-forward scheme.

1.4 General Notations

In this dissertation we use the following general notations.

$\mathbb{R}, \mathbb{C}, \mathbb{Z}$ denote real, complex, and integer numbers, respectively. $\mathbb{Z}[i]$ denotes the Gaussian integers. $\mathbb{F}_p$ represents the finite field of size p, where p is always assumed to be a prime number. In the finite field domain, $\oplus$ denotes addition and $\bigoplus$ denotes summation.[3] For any complex number, $\Im(\cdot)$ and $\Re(\cdot)$ denote its real and imaginary components, respectively. Boldface lowercase letters denote column vectors, e.g., $\mathbf{a} \in \mathbb{Z}^n$, while boldface uppercase letters denote matrices, e.g., $\mathbf{A} \in \mathbb{Z}^{n\times n}$. The Hermitian and the regular transpose operations are expressed by $(\cdot)^H$ and $(\cdot)^T$, e.g., $\mathbf{A}^H$ and $\mathbf{A}^T$, respectively. The inversion of the regular transpose is denoted by $(\cdot)^{-T}$, e.g., $\mathbf{A}^{-T} \triangleq (\mathbf{A}^T)^{-1}$. The determinant of a matrix is denoted as $\det(\cdot)$, e.g., $\det(\mathbf{A})$. $\mathrm{rank}(\cdot)$ denotes the rank of a matrix, e.g., $\mathrm{rank}(\mathbf{A})$. $\mathbf{I}$ denotes the identity matrix, where the dimension is clear from the context.

$O(n)$ and $U(n)$ respectively denote the orthogonal and unitary groups of dimension n.[4] The matrix exponential is defined as $\exp(\mathbf{A}) \triangleq \sum_{m=0}^{\infty} \frac{\mathbf{A}^m}{m!}$. We define $\log^+(x) \triangleq \max(\log(x), 0)$ and the general logarithm operation is with base 2, unless otherwise stated. $\mathbb{E}\{\cdot\}$ denotes the expectation of a random variable.

[3] Any multiplication operation within a formula containing $\oplus$ or $\bigoplus$ is also over finite field.
[4] Orthogonal and unitary groups are groups of orthogonal and unitary matrices, respectively.

Chapter 2

Preliminaries

The main role of this chapter is to provide basic knowledge required in understanding this dissertation. First, an overview of lattices and lattice codes is briefly given. For more details, the reader is referred to [15, 16, 31]. Subsequently, the main concept of the compute-and-forward scheme [14] which is the main topic of this dissertation is described.

2.1 Lattices and Lattice Codes

2.1.1 Lattices

A lattice is a discrete subgroup of the Euclidean space with the ordinary vector addition operation. The origin, i.e., the zero vector $\mathbf{0}$, is the identity element of any lattice. Let Λ be a lattice. If $\mathbf{t}$ is an element of Λ, then so is $-\mathbf{t}$. For any $\mathbf{t}_1, \mathbf{t}_2 \in \Lambda$, an integer linear combination of those elements (points) will result in another element of Λ, i.e., $a_1\mathbf{t}_1 + a_2\mathbf{t}_2 \in \Lambda$, with $a_1, a_2 \in \mathbb{Z}$. This is one of lattice properties that is exploited by the compute-and-forward scheme as we will see later. It is easy to see that lattices are countably infinite sets. Using a generator matrix, lattices can be formally defined as follows.

Definition 2.1.1 (Lattice)**:** Let column vectors $\mathbf{g}_1, \dots \mathbf{g}_n \in \mathbb{R}^n$ be a basis forming a full-rank generator matrix $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_n] \in \mathbb{R}^{n\times n}$.[1] The lattice formed by $\mathbf{G}$

[1] A generator of a lattice need not to be full-rank. However, in this dissertation we limit our discussion to only lattices with full-rank generator matrices.

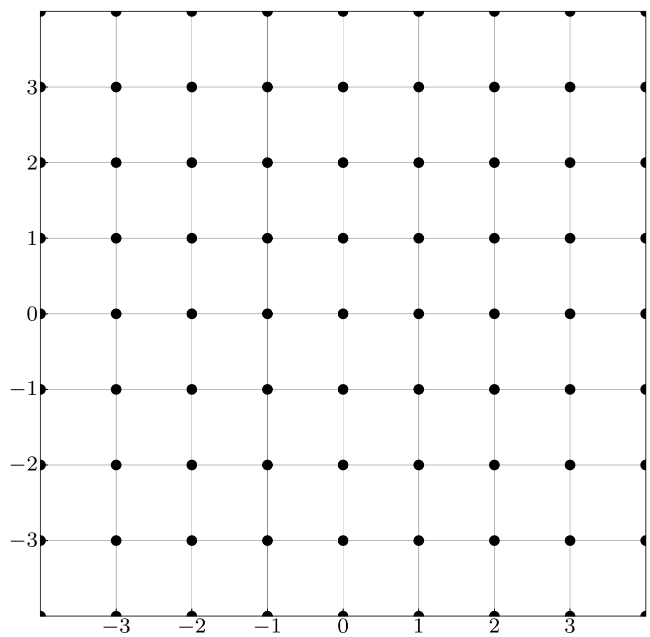


Fig. 2.1. The two-dimensional integer lattice $\mathbb{Z}^2$.

is defined as all integral combinations of the column vectors of $\mathbf{G}$, i.e.,[2]

$$\Lambda(\mathbf{G}) = \{\mathbf{G}\mathbf{a} : \mathbf{a} \in \mathbb{Z}^n\}. \tag{2.1}$$

The simplest lattice is the integer lattice, $\mathbb{Z}^n$. In one dimension, this is just the integers $\ldots, -2, -1, 0, 1, 2, \ldots$. For two dimension, the integer lattice is visualized in Fig 2.1. $\mathbb{Z}^n$ can be generated using the identity matrix of dimension n. In other words,

$$\mathbb{Z}^n = \{\mathbf{I}\mathbf{a} : \mathbf{a} \in \mathbb{Z}^n\}. \tag{2.2}$$

Given a lattice, the generator matrix is not unique. A lattice is invariant to a unimodular transformation of its basis.

Example 2.1.1 (Hexagonal lattice)**:** The two-dimensional hexagonal lattice, shown in Fig 2.2, can be generated by the basis $\mathbf{g}_1 = [0, 2]^T$ and $\mathbf{g}_2 = [\sqrt{3}, 1]^T$. It can also be generated by another basis, e.g., $\tilde{\mathbf{g}}_1 = [-\sqrt{3}, 1]^T$ and $\tilde{\mathbf{g}}_2 = [-\sqrt{3}, -1]^T$. Let $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2]$ and $\tilde{\mathbf{G}} = [\tilde{\mathbf{g}}_1, \tilde{\mathbf{g}}_2]$. It can be shown that $\mathbf{G}$ and $\tilde{\mathbf{G}}$ are related by

[2]We use Λ to denote a lattice when its generator matrix is undefined.

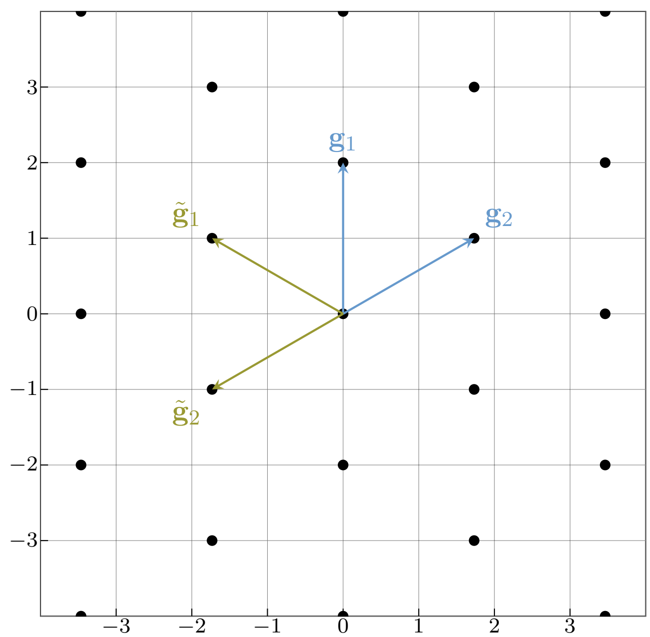


Fig. 2.2. The hexagonal lattice can be generated using basis vectors $\mathbf{g}_1 = [\sqrt{3}, 1]^T$ and $\mathbf{g}_2 = [0, 2]^T$. It can also be generated by another basis, e.g., $\tilde{\mathbf{g}}_1 = [-\sqrt{3}, 1]^T$ and $\tilde{\mathbf{g}}_2 = [-\sqrt{3}, -1]^T$.

a unimodular matrix $\mathbf{T}$.[3] To be precise,

$$\mathbf{G} = \tilde{\mathbf{G}}\mathbf{T}, \quad \mathbf{T} = \begin{bmatrix} 1 & 0 \\ -1 & -1 \end{bmatrix}. \tag{2.3}$$

A subgroup of a lattice is also a lattice, and this lattice is said to be *nested* in the other lattice.

Definition 2.1.2 (Nested Lattices)**:** Given lattices Λ and Λ_1, it is said that the lattic Λ is nested in Λ_1, if $\Lambda \subseteq \Lambda_1$. More generally, a sequence of lattices $\Lambda, \Lambda_1, \ldots, \Lambda_L$ is nested if $\Lambda \subseteq \Lambda_1 \subseteq \cdots \subseteq \Lambda_L$.

There are some important notions associated with lattices. A lattice quantizer maps any point in $\mathbb{R}^n$ to the nearest lattice point. In particular, this quantizer will be used in the decoding of a lattice.

Definition 2.1.3 (Lattice Quantization)**:** A lattice quantizer $Q_\Lambda : \mathbb{R}^n \to \Lambda$ maps every point $\mathbf{x} \in \mathbb{R}^n$ to the nearest $\mathbf{t} \in \Lambda$ in Euclidean distance,

$$Q_\Lambda(\mathbf{x}) = \arg\min_{\mathbf{t} \in \Lambda} \|\mathbf{x} - \mathbf{t}\|. \tag{2.4}$$

[3] A unimodular matrix $\mathbf{T}$ is an integer matrix with a unit absolute determinant, $\det(\mathbf{T}) = \pm 1$

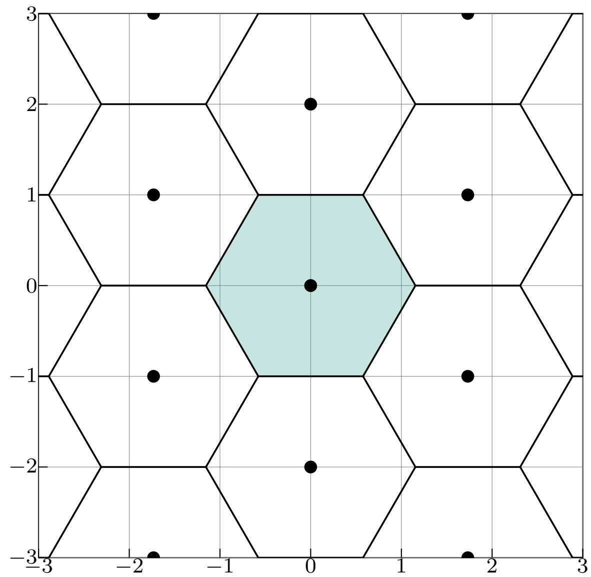


Fig. 2.3. The Voronoi regions of the hexagonal lattice. The middle one is the fundamental Voronoi region, i.e., the Voronoi region with respect to the origin.

A lattice induces a partition of the Euclidean space into congruent regions. With respect to a given lattice Λ, there are many ways to divide the space into congruent regions. The most important one is the Voronoi region.

Definition 2.1.4 (Voronoi Region)**:** The Voronoi region of an n-dimensional real-valued lattice Λ associated with a lattice point $\mathbf{t}$, denoted by $\mathcal{V}_{\mathbf{t}}$, consists of all points of the underlying space that are closer to $\mathbf{t}$ than to any other lattice point,

$$\mathcal{V}_{\mathbf{t}} = \left\{ \mathbf{x} \in \mathbb{R}^n : |\mathbf{x} - \mathbf{t}| \leq \left|\mathbf{x} - \tilde{\mathbf{t}}\right| \text{ for all } \tilde{\mathbf{t}} \in \Lambda \backslash \mathbf{t} \right\}. \tag{2.5}$$

Using the lattice quantization definition, it can also be written as $\mathcal{V}_{\mathbf{t}} = \{\mathbf{x} \in \mathbb{R}^n : Q_\Lambda(\mathbf{x}) = \mathbf{t}\}$. In this work, we use the term *fundamental Voronoi region* to represent the Voronoi region associated with the origin $\mathbf{0}$;

$$\mathcal{V}_\Lambda = \{\mathbf{x} \in \mathbb{R}^n : Q_\Lambda(\mathbf{x}) = \mathbf{0}\}. \tag{2.6}$$

Note that there are some points in the space that have equal distance to two or more lattice points. However, each point can belong to only one Voronoi region. The tie-breaking is performed in a systematic manner such that the resulting Voronoi regions are congruent. The volume of the Voronoi region of a lattice Λ

with a generator matrix $\mathbf{G}$ is given by

$$\mathrm{Vol}(\mathcal{V}_\Lambda) = |\det(\mathbf{G})| \triangleq \mathrm{Vol}(\Lambda) \tag{2.7}$$

Definition 2.1.5 (Moments)**:** The second moment of a lattice Λ is defined as the second moment per dimension of a random variable X which is uniformly distributed over the fundamental Voronoi region $\mathcal{V}_\Lambda$

$$\sigma_\Lambda^2 = \frac{1}{n}\mathbb{E}\{\|X\|^2\} = \frac{1}{n\mathrm{Vol}(\mathcal{V}_\Lambda)} \int_{\mathcal{V}_\Lambda} \|\mathbf{x}\|^2 \, d\mathbf{x}. \tag{2.8}$$

The normalized second moment of a lattice is given by

$$G(\Lambda) = \frac{\sigma_\Lambda^2}{(\mathrm{Vol}(\mathcal{V}_\Lambda))^{2/n}}. \tag{2.9}$$

Any point $\mathbf{x} \in \mathbb{R}^n$ can be uniquely expressed as the sum of a lattice point and a point in the fundamental Voronoi region, i.e, $\mathbf{x} = Q_\Lambda(\mathbf{x}) + (\mathbf{x} - Q_\Lambda(\mathbf{x}))$, where $\mathbf{x} - Q_\Lambda(\mathbf{x})$ is the quantization error from which we can define the lattice modulo operation.

Definition 2.1.6 (Modulus)**:** Modulo operation with respect to the lattice Λ on a point $\mathbf{x} \in \mathbb{R}^n$ is defined as the residual of the lattice quantization,

$$[\mathbf{x}] \bmod \Lambda = \mathbf{x} - Q_\Lambda(\mathbf{x}). \tag{2.10}$$

For all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $a \in \mathbb{Z}$, $\beta \in \mathbb{R}$, and $\Lambda \subseteq \Lambda_1$, the mod Λ operation satisfies:

$$[\mathbf{x} + \mathbf{y}] \bmod \Lambda = [[\mathbf{x}] \bmod \Lambda + \mathbf{y}] \bmod \Lambda, \tag{2.11}$$

$$[Q_{\Lambda_1}(\mathbf{x})] \bmod \Lambda = [Q_{\Lambda_1}([\mathbf{x}] \bmod \Lambda)] \bmod \Lambda, \tag{2.12}$$

$$[a\mathbf{x}] \bmod \Lambda = [a[\mathbf{x}] \bmod \Lambda] \bmod \Lambda, \tag{2.13}$$

$$\beta[\mathbf{x}] \bmod \Lambda = [\beta\mathbf{x}] \bmod \beta\Lambda. \tag{2.14}$$

Similar to any vector space, a lattice also has a dual space. Lattices Λ and Λ^* are dual or reciprocal if the inner products of their points are integers, i.e., for all $\mathbf{t} \in \Lambda$ and $\mathbf{t}^* \in \Lambda^*$, $\langle \mathbf{t}, \mathbf{t}^* \rangle \in \mathbb{Z}$. Using the generator matrix, it can also be defined as follows.

Definition 2.1.7 (Dual Lattice)**:** For a lattice $\Lambda(\mathbf{G})$ with a full-rank generator matrix $\mathbf{G} \in \mathbb{R}^{n\times n}$, the dual lattice is

$$\Lambda^*(\mathbf{G}) \triangleq \Lambda(\mathbf{G}^{-T}) = \{\mathbf{G}^{-T}\mathbf{a} : \mathbf{a} \in \mathbb{Z}^n\}. \tag{2.15}$$

Another important notion related to a lattice is the successive-minima which will be exploited in the Chapters 3 and 4.

Definition 2.1.8 (Successive Minima)**:** For an n-dimensional lattice $\Lambda(\mathbf{G})$, the l-th successive minimum, $1 \leq l \leq n$, is defined as

$$\lambda_l(\mathbf{G}) \triangleq \min_{\mathbf{v}_1,\ldots,\mathbf{v}_l \in \Lambda(\mathbf{G})} \max\{\|\mathbf{v}_1\|, \ldots, \|\mathbf{v}_l\|\}, \tag{2.16}$$

where the minimum is taken over all sets of l linearly independent vectors in $\Lambda(\mathbf{G})$. In other words, $\lambda_l(\mathbf{G})$ is the smallest real number r such that there exist l linearly independent vectors $\mathbf{v}_1, \ldots, \mathbf{v}_l \in \Lambda(\mathbf{G})$ with $\|\mathbf{v}_1\|, \ldots, \|\mathbf{v}_l\| \leq r$.

Note that the first successive minimum of $\Lambda(\mathbf{G})$, i.e., $\lambda_1(\mathbf{G})$, is its minimum distance. The successive minima are non-decreasing,

$$\lambda_1(\mathbf{G}) \leq \lambda_2(\mathbf{G}) \leq \cdots \leq \lambda_n(\mathbf{G}). \tag{2.17}$$

All the definitions given above are specifically for real-valued lattices. However, it can be extended to complex-valued lattices.

Definition 2.1.9 (Complex-Valued Lattice)**:** Given a full-rank generator matrix $\breve{\mathbf{G}} \in \mathbb{C}^{n\times n}$, the complex-valued lattice $\Lambda(\breve{\mathbf{G}})$ is defined similarly to real-valued lattices as

$$\Lambda(\breve{\mathbf{G}}) = \{\breve{\mathbf{G}}\breve{\mathbf{a}} : \breve{\mathbf{a}} \in \mathbb{Z}[i]^n\}. \tag{2.18}$$

Note that $\mathbb{Z}[i] = \{a + bi : a, b \in \mathbb{Z}\}$.

2.1.2 Lattice Codes

A lattice is an infinite structure that has no power constraint. In order to be useful for communications, it is necessary to select finite subset of the lattice points such that a certain power constraint is satisfied. A lattice code $\mathcal{C}$ is finite codebook whose codewords are selected from a lattice.

Given a lattice Λ_s and $\mathbf{x} \in \mathbb{R}^n$, the set $\mathbf{x} + \Lambda_\mathrm{s}$ is a *coset* of Λ_s in $\mathbb{R}$. Consider nested lattices $\Lambda_\mathrm{s} \subseteq \Lambda_\mathrm{c}$.[4] For each $\mathbf{t} \in \Lambda_\mathrm{c}$, $\mathbf{t} + \Lambda_\mathrm{s}$ is called a coset of Λ_s in Λ_c. Further, the point of $\mathbf{t} \bmod \Lambda_\mathrm{s}$ is called the *coset leader* of $\mathbf{t} + \Lambda_\mathrm{s}$. We then define a quotient $\Lambda_\mathrm{c}/\Lambda_\mathrm{s}$ as the set of all distinct cosets of Λ_s in Λ_c.

Definition 2.1.10 (Nested Lattice Code [15–17])**:** Given two lattices Λ_c and Λ_s where $\Lambda_\mathrm{s} \subseteq \Lambda_\mathrm{c}$, the nested lattice code $\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s})$ is defined as the coset leaders of the quotient group $\Lambda_\mathrm{c}/\Lambda_\mathrm{s}$;

$$\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s}) = \Lambda_\mathrm{c} \bmod \Lambda_\mathrm{s} = \{\mathbf{t} \bmod \Lambda_\mathrm{s} : \mathbf{t} \in \Lambda_\mathrm{c}\}. \tag{2.19}$$

Geometrically, $\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s})$ contains all points of Λ_c that are within the fundamental Voronoi region of Λ_s, i.e.,

$$\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s}) = \Lambda_\mathrm{c} \cap \mathcal{V}_{\Lambda_\mathrm{s}}. \tag{2.20}$$

The rate of a nested lattice code is

$$R = \frac{1}{n} \log |\mathcal{C}(\Lambda_\mathrm{c}/\Lambda_\mathrm{s})| = \frac{1}{n} \log \frac{\mathrm{Vol}\,(\mathcal{V}_{\Lambda_\mathrm{s}})}{\mathrm{Vol}\,(\mathcal{V}_{\Lambda_\mathrm{c}})}. \tag{2.21}$$

Example 2.1.2 (Nested Hexagonal Lattice)**:** Let A_2 denote the hexagonal lattice in two dimensions. Let $\Lambda_\mathrm{c} = A_2$ and $\Lambda_\mathrm{s} = 3A_2$. The nested hexagonal lattice can be constructed by taking all the coset leaders of the quotient $A_2/3A_2$. This code is visualized in Fig 2.4. The rate of this code is

$$R = \frac{1}{2} \log \frac{\mathrm{Vol}\,(3A_2)}{\mathrm{Vol}\,(A_2)} = \log 3. \tag{2.22}$$

This type of nested lattice code where the shaping and coding lattices are self-similar is called self-similar nested lattice codes [31].

2.2 Compute-and-Forward

Compute-and-forward has found many applications including Gaussian networks, multiple-access relay channels, random access, MIMO, etc. In this section, we

[4] In this dissertation, we use Λ_c to denote a fine lattice that is used for coding and Λ_s to denote a coarse lattice that is usually used for shaping.

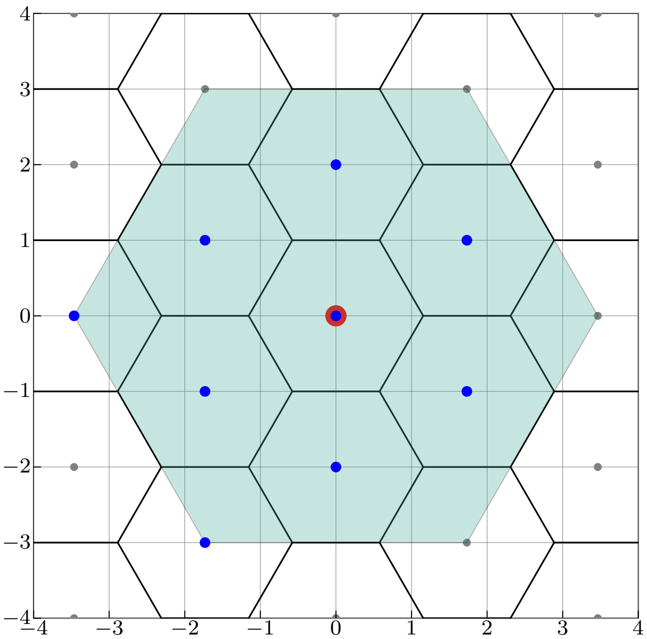


Fig. 2.4. The nested lattice code constructed from the hexagonal lattice A_2. In this figure, $\Lambda_{\mathrm{c}} = A_2$ and $\Lambda_{\mathrm{s}} = 3A_2$. The red point is a point in Λ_{s}, the gray points are in Λ_{c}, and the blue points are codewords of $\mathcal{C}(\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}})$.

provide a brief introduction of compute-and-forward in a very simple network configuration where there are multiple transmitters expecting to transmit information to one common receiver. The receiver is interested in *computing* a linear combination of all the transmitted messages, rather than each individual message. The receiver may *forward* the decoded linear combination to another node if needed — this is a common transmission pattern in wireless cooperative networks. Conventionally, the problem of computing a linear combination of transmitted messages is solved using a multiple access technique such as successive interference cancellation (SIC); each message is successively decoded, re-encoded, and subtracted from the received signal, and finally a linear combination is computed from the decoded messages. In contrast, compute-and-forward only employs one time single-input single-output (SISO) decoding to retrieve a linear combination, which is very efficient. Moreover, the achievable rate of compute-and-forward is often higher than that achieved by the conventional techniques.

Let $\mathcal{C} \triangleq \mathcal{C}(\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}})$ be a nested lattice code of n-dimension with rate R. The second moment of the shaping lattice Λ_{s} is assumed to be P. Consider a network with M transmitters and one receiver. For simplicity, we assume a symmetric

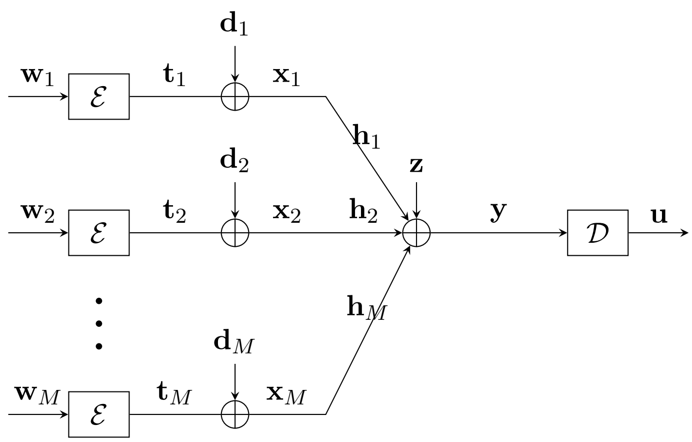


Fig. 2.5. M transmitters transmit messages to one common receiver. Instead of each individual message of the transmitters, the receiver is interested in a linear combination of the transmitted messages.

transmission scenario where all transmitters employ the same code $\mathcal{C}$ (with the same rate R). The extension to an asymmetric case can be found in [14,32,33]. The compute-and-forward scheme is illustrated in Fig. 2.5. Each transmitter, indexed by $m = 1, \ldots, M$, randomly generates a message $\mathbf{w}_m \in \mathbb{F}_p^{nR}$. The receiver desires to retrieve a linear combination of $\mathbf{w}_m$. Let $\mathbf{u}$ be the desired linear combination in the form of

$$\mathbf{u} = \bigoplus_{m=1}^{M} q_m \mathbf{w}_m, \tag{2.23}$$

where q_m is a coefficient taking value in $\mathbb{F}_p$. The summation denoted by $\bigoplus$ and the multiplication operations are performed over $\mathbb{F}_p$. Note that the receiver is not limited to decode only one linear combination, it is possible for it to compute multiple linear combinations.

Let $\mathcal{E}$ be a bijective mapping from a finite field $\mathbb{F}_p^{nR}$ to the codebook $\mathcal{C}$, i.e., $\mathcal{E} : \mathbb{F}_p^{nR} \to \mathcal{C}$. Using $\mathcal{E}$, $\mathbf{w}_m$ is encoded to a codeword $\mathbf{t}_m \in \mathcal{C}$ as

$$\mathbf{t}_m = \mathcal{E}(\mathbf{w}_m). \tag{2.24}$$

Prior to transmission, a random dither vector $\mathbf{d}_m$ distributed uniformly over $\mathcal{V}_{\Lambda_\mathrm{s}}$ is added to form a symbol $\mathbf{x}_m$:

$$\mathbf{x}_m = [\mathbf{t}_m - \mathbf{d}_m] \bmod \Lambda_\mathrm{s}. \tag{2.25}$$

The purpose of adding a dither vector is to make the transmitted vector $\mathbf{x}_m$ independent of the underlying lattice point $\mathbf{t}_m$ and uniformly distributed over $\mathcal{V}_{\Lambda_\mathrm{s}}$ [17]. It also ensures that $\mathbb{E}\{\|\mathbf{x}_m\|^2\} = nP$. Dither vectors are made available at the receiver.[5]

The transmitters simultaneously transmit $\mathbf{x}_m$, and accordingly the receiver receives a noisy superposition signal

$$\mathbf{y} = \sum_{m=1}^{M} h_m \mathbf{x}_m + \mathbf{z}, \tag{2.26}$$

where h_m denotes a channel coefficient from transmitter m and the receiver, and $\mathbf{z}$ denotes a noise vector at the receiver. All channel coefficient h_m and all elements of $\mathbf{z}$ are randomly distributed over Gaussian distribution $\mathcal{N}(0,1)$.

With our bijective mapping $\mathcal{E}$, we can retrieve the desired linear combination $\mathbf{u}$ if we have an equivalent linear combination (equation) of lattice codewords in $\mathcal{C}$. Let $\mathbf{v}$ be a lattice equation defined as follows.

Definition 2.2.1 (Lattice Equation)**:** A lattice equation $\mathbf{v}$ is an integral combination of lattice codewords $\mathbf{t}_m \in \mathcal{C}$ modulo the shaping lattice Λ_s:

$$\mathbf{v} = \sum_{m=1}^{M} a_m \mathbf{t}_m \bmod \Lambda_\mathrm{s}, \tag{2.27}$$

where $a_m \in \mathbb{Z}$. We refer to $\mathbf{a} = [a_1, \ldots, a_M]^T$ as a *coefficient vector.*

In order to retrieve $\mathbf{u}$, we first need to obtain $\mathbf{v}$. However, we have to overcome two sources of noise before we decode $\mathbf{v}$. The first one is the channel noise $\mathbf{z}$ and the second is the one caused by the fact that channel coefficients h_m are not equal to the desired equation coefficients. To overcome these, we scale the received signal appropriately such that the scaled channel coefficients are close to the integer coefficients and the resulting effective noise is as small as possible. Subsequently, we remove the dither vectors and quantize the resulting signal to the nearest lattice point. To be precise, using $\alpha \in \mathbb{R}$, the received signal $\mathbf{y}$ is scaled and the dither

[5] It has been shown that random dithers can be replaced by deterministic ones which means that no common randomness is required [14].

vectors are removed as

$$
\begin{aligned}
\tilde{\mathbf{y}} &= \left[\alpha\mathbf{y} + \sum_{m=1}^{M} a_m\mathbf{d}_m\right] \bmod \Lambda_{\mathrm{s}} && (2.28)\\
&= \left[\sum_{m=1}^{M} \alpha h_m\mathbf{x}_m + \alpha\mathbf{z} + \sum_{m=1}^{M} a_m\mathbf{d}_m\right] \bmod \Lambda_{\mathrm{s}} && (2.29)\\
&= \left[\sum_{m=1}^{M} a_m\mathbf{x}_m + \sum_{m=1}^{M} a_m\mathbf{d}_m + \sum_{m=1}^{M}(\alpha h_m - a_m)\mathbf{x}_m + \alpha\mathbf{z}\right] \bmod \Lambda_{\mathrm{s}} && (2.30)\\
&= \left[\sum_{m=1}^{M} a_m\left[\mathbf{t}_m - \mathbf{d}_m\right] \bmod \Lambda_{\mathrm{s}} + \sum_{m=1}^{M} a_m\mathbf{d}_m + \sum_{m=1}^{M}(\alpha h_m - a_m)\mathbf{x}_m + \alpha\mathbf{z}\right] \bmod \Lambda_{\mathrm{s}} \\
&= \left[\sum_{m=1}^{M} a_m\mathbf{t}_m + \sum_{m=1}^{M}(\alpha h_m - a_m)\mathbf{x}_m + \alpha\mathbf{z}\right] \bmod \Lambda_{\mathrm{s}} && (2.31)\\
&= \left[\mathbf{v} + \mathbf{z}_{\mathrm{eff}}\right] \bmod \Lambda_{\mathrm{s}}, && (2.32)
\end{aligned}
$$

where $\mathbf{z}_{\mathrm{eff}} = \sum_{m=1}^{M}(\alpha h_m - a_m)\mathbf{x}_m + \alpha\mathbf{z}$ is the resulting effective noise. Owing to the dither vectors, $\mathbf{z}_{\mathrm{eff}}$ is independent of $\mathbf{v}$ and its variance can be defined as [14]

$$
\begin{aligned}
\sigma^2_{\mathrm{eff}} &\triangleq \frac{1}{n}\mathbb{E}\{\|\mathbf{z}_{\mathrm{eff}}\|^2\} && (2.33)\\
&= \|\alpha\mathbf{h} - \mathbf{a}\|^2 P + \alpha^2, && (2.34)
\end{aligned}
$$

where $\mathbf{h} = [h_1, \ldots, h_M]^T$ is the channel coefficient vector.

The receiver then produces an estimate for $\mathbf{v}$ by applying the lattice quantizer associated with Λ_{c} to $\tilde{\mathbf{y}}$:

$$\hat{\mathbf{v}} = \left[Q_{\Lambda_{\mathrm{c}}}(\tilde{\mathbf{y}})\right] \bmod \Lambda_{\mathrm{s}}. \tag{2.35}$$

Let $\mathcal{V}_{\Lambda_{\mathrm{c}}}$ be the fundamental Voronoi region of Λ_{c}. The probability that the receiver estimates $\mathbf{v}$ incorrectly is upper bounded by the probability that the effective noise $\mathbf{z}_{\mathrm{eff}}$ lies outside $\mathcal{V}_{\Lambda_{\mathrm{c}}}$:

$$\Pr(\hat{\mathbf{v}} \neq \mathbf{v}) \leq \Pr\left(\mathbf{z}_{\mathrm{eff}} \notin \mathcal{V}_{\Lambda_{\mathrm{c}}}\right). \tag{2.36}$$

Given an integer coefficient vector $\mathbf{a}$, the scaling factor α has to be determined such that σ^2_{eff} is minimized. This ensures that the error probability is minimized.

From [14], it is found that the optimal value of α is the minimum mean square error (MMSE) coefficient

$$\alpha_{\text{opt}} = \frac{P\mathbf{h}^T\mathbf{a}}{1 + P\left\|\mathbf{h}\right\|^2}. \tag{2.37}$$

Once the receiver obtains $\hat{\mathbf{v}}$, an estimate of the desired linear equation $\mathbf{u}$ can be recovered by applying to $\hat{\mathbf{v}}$ the inverse of $\mathcal{E}$,

$$\hat{\mathbf{u}} = \mathcal{E}^{-1}(\hat{\mathbf{v}}). \tag{2.38}$$

Note that because $\mathcal{E}$ is a bijective mapping, $\Pr(\hat{\mathbf{u}} \neq \mathbf{u}) = \Pr(\hat{\mathbf{v}} \neq \mathbf{v})$.

The achievable rate of the compute-and-forward scheme is given in the following theorem.

Theorem 2.2.1 (Computation Rate for Real-Valued Channels [14])**:** Consider a real-valued Gaussian network with M transmitters that simultaneously transmit their messages with average power constraint P to a receiver. Let $\mathbf{h} = [h_1, \ldots, h_M]^T \in \mathbb{R}^M$ be the channel coefficients. The receiver can decode an integer linear combination of the transmitted messages with low error probability so long as the message rate is less than the *computation rate* given by

$$R_{\text{cp}}(\mathbf{a}, \mathbf{h}) = \frac{1}{2}\log^+\left(\left(\left\|\mathbf{a}\right\|^2 - \frac{P(\mathbf{h}^T\mathbf{a})^2}{1 + P\left\|\mathbf{h}\right\|^2}\right)^{-1}\right) \tag{2.39}$$

where $\mathbf{a} = [a_1, \ldots, a_M]^T \in \mathbb{Z}^M$ denotes the integer coefficient vector.

In the above discussion we only assume real-valued channels. However, the compute-and-forward scheme also works similarly in complex-valued channels. One way to deal with complex-valued channels will be demonstrated in Chapter 3.

Chapter 3

Compute-and-Forward in Multiple Access Relay Channels

In this chapter, we study the application of compute-and-forward to multiple access relay channels (MARC). How to apply compute-and-forward to the MARC is not straightforward. Cooperation between the destination and the relay has to be designed carefully such that the destination can recover transmitted messages correctly with higher probability. Our work here focuses on the design of cooperation strategies between the destination and the relay in the MARC employing compute-and-forward.

3.1 Introduction

3.1.1 Background and Related Work

Network coding [5, 9, 19, 34] has become an important networking strategy to improve the spectral efficiency of wireless communication networks. In contrast to simple forwarding, network coding allows intermediate nodes to "combine" the received messages before forwarding them to next nodes, to reduce the required number of transmissions. On the other hand, cooperative communication is an effective method to enlarge network coverage, increase transmission robustness, and improve power efficiency by exploiting spatial diversity without additional antennas [35–37]. However, the gains achieved by cooperative communications in practice come with a loss of spectral efficiency due to half-duplex operation [36].

Thus, it is beneficial to apply network coding to cooperative networks to achieve reliable communications with high spectral efficiency.

In this chapter, we design a network coding scheme for multiple access relay channel (MARC), which is an important class of wireless cooperative networks. In the MARC, multiple sources want to deliver messages to one common destination with the present of one relay terminal [38–42]. The applications of such networks include sensor and ad-hoc networks and uplink for cellular networks with an intermediate node as a relay. The MARC also has found a use case in the *LTE Advanced* mobile communication standard [22, 43]. It has been shown that network coding significantly improves the spectral efficiency of the MARC. For example, in the conventional two-source MARC, four orthogonal transmissions are required, where the sources transmit their messages in turn and the relay forwards the messages one by one to the destination. With network coding, number of transmissions is reduced to only three; the first and the second transmissions are used by the sources to transmit messages in turn, while the third transmission is used by the relay to forward the *network coded* version of the transmitted messages to the destination [39].

Recently, Nazer and Gastpar proposed a new network coding and relaying scheme, known as compute-and-forward [14]. It views interference as an advantage rather than a severe problem to avoid and allows sources in a relay network to simultaneously transmit their messages via a non-orthogonal channel. Each relay directly *computes* an integer linear combination of the transmitted messages from the received superimposed signals without decoding each transmitted message separately, and then forwards it to the destination. Given a sufficient number of linear combinations, the destination can recover the transmitted messages so long as the coefficient matrix, that is the matrix composed of the coefficients of the linear combinations, is full rank.

Allowing sources to transmit their messages via one non-orthogonal channel is an appealing advantage of compute-and-forward. It is easy to see its potential for improving the spectral efficiency of the MARC. When compute-and-forward is applied to the MARC, the required number of transmissions can be reduced to only two; the first is used by the sources to broadcast their messages to the relay and destination at once and the second is used by the relay to forward its computed linear combination to the destination. Note that this advantage is applicable to any

MARC with any number of sources, not limited to the MARC with two sources. Despite its promising advantage, however, it is not straightforward to efficiently apply compute-and-forward to MARC as the destination requires the resulting coefficient matrix to be full rank.

It is possible to naively apply the original compute-and-forward [14] to the MARC by allowing the destination and the relay to compute their local optimal integer linear combinations independently. Nevertheless, it may result in a rank deficient coefficient matrix which causes a decoding failure. Therefore, cooperation between the destination and the relay in computing their integer linear combinations is necessary. In [20], Soussi *et al.* made an attempt to solve this issue. They proposed a global optimization technique such that the destination and the relay select the global optimal linearly independent combinations with respect to achievable rate. They showed that with this strategy, compute-and-forward achieves higher achievable symmetric-rate compared to other relaying strategies such as decode-and-forward and amplify-and-forward. The achievable rate improvement in their work, however, relies on the assumption that all channel state information (CSI) are known to all nodes. Insausti *et al.* proposed another strategy for applying compute-and-forward to the MARC. Different with [20], the relay is allowed to select its local best linear combination; based on this the destination adjusts its linear combination ensuring a full rank coefficient matrix. This strategy is more efficient than the one proposed in [20]. Moreover, it was shown that it also achieves higher achievable rate. Both of the aforementioned works focused on the achievable rate performance without investigating outage performance. Because one of the main objectives of wireless cooperative networks is to increase transmission reliability, it is important to make sure that the outage performance is also increased when compute-and-forward is employed. In this work we design efficient cooperation strategies for the MARC employing compute-and-forward that improve outage performance as well.

3.1.2 Summary of Contributions

In this chapter we propose two efficient cooperation strategies for applying compute-and-forward to the MARC. Both of the proposed strategies utilize as few transmissions as possible while improving the outage probability performance. We provide semi-theoretical and numerical analyses to evaluate the outage performance of the proposed strategies.

The first proposed strategy helps the destination by forwarding its best linear combinations without taking into account whether the resulting coefficient matrix is full rank. We show that even though it does not achieve full diversity gain,[1] it outperforms the existing strategies [20, 22] in terms of outage probability and throughput. The second strategy is similar to the first except that the relay helps the destination by a sending "good" linear combination that ensures a full rank coefficient matrix. We show that the second strategy offers the best performance in terms of outage probability and network throughput. Moreover, we show that it achieves full diversity gain of the MARC.

3.2 Multiple Access Relay Channel Model

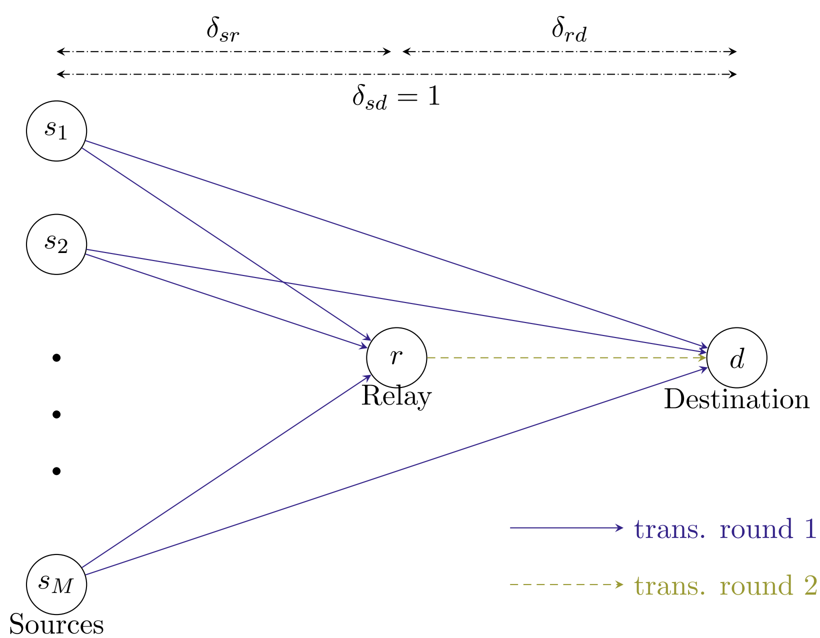


Fig. 3.1. M-users MARC model.

We begin by describing the system model of the MARC considered in this chapter. As illustrated in Fig. 3.1, our model consists of M sources denoted as s_m, $m = 1, \ldots, M$, one destination d and one relay terminal r. The sources want to transmit information messages to the destination. All wireless links are assumed

[1]In this work, we consider the full-diversity gain of the MARC is the order of two. This is because all the sources transmit independent messages.

to be Rayleigh block fading channels where the fading coefficients remain constant within a block of symbols, but change independently from one block to the other according to a circularly symmetric complex Gaussian distribution with zero mean and unit variance.

As shown in Fig. 3.1, there are three types of directed transmission links: sources-destination, sources-relay, and relay-destination links. All transmitters (sources and relay) have the same average transmit power P. For $i \in \{s_1, \dots, s_M, r\}$, $j \in \{r, d\}$, and $i \neq j$, let γ_{ij}, δ_{ij}, g_{ij}, and h_{ij} be the average signal-to-noise power ratio (SNR), distance, geometric gain, and channel gain of wireless link from terminal i to terminal j, respectively. The geometric gain captures the effect of path loss which is a function of distance, i.e., $g_{ij} = \delta_{ij}^{-\kappa}$, where κ is the path loss exponent [44,45]. The channel gain h_{ij} and noise at every node are randomly distributed over $\mathcal{CN}(0,1)$. Thus, the average SNR can be defined as $\gamma_{ij} \triangleq P g_{ij}$. For simplicity, we assume that all sources have the same distance to the destination and also have the same distance to the relay, To be more precise, let δ_{sd} and δ_{sr} be positive real scalars. We assume that for all $m \in \{1, \dots, M\}$, $\delta_{s_m d} = \delta_{sd}$ and $\delta_{s_m r} = \delta_{sr}$. This assumption implies that the sources have one common geometric gain to the destination and another common geometric gain to the relay, i.e., $\forall m$, $g_{s_m d} = g_{sd}$ and $g_{s_m r} = g_{sr}$. Accordingly, the average SNR of each source to the destination can be defined as $\gamma_{s_m d} = \gamma_{sd} = P g_{sd}$, and to the relay as $\gamma_{s_m r} = \gamma_{sr} = P g_{sr}$.

3.3 MARC with Compute-and-Forward (MARC-CF)

3.3.1 Encoding Scheme

For simplicity, we consider symmetric MARC where all sources transmit with the same rate R. For a positive integer n, consider nested lattices $\Lambda_{\mathrm{s}} \subseteq \Lambda_{\mathrm{c}} \subset \mathbb{R}^n$ and let $\mathcal{C} \triangleq \mathcal{C}(\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}})$ be a nested lattice code with rate $R/2$. Similar to the previous chapter, Λ_{c} represents a fine lattice used for coding and Λ_{s} represents a coarse lattice used for shaping to ensure that the average power constraint is satisfied. The second moment of Λ_{s} is assumed to be $P/2$. For a prime number p, let $\mathcal{E}$ be a bijective mapping from $\mathbb{F}_p^{nR/2}$ to $\mathcal{C}$, i.e.,

$$\mathcal{E} : \mathbb{F}_p^{nR/2} \to \mathcal{C}. \tag{3.1}$$

The bijective mapping $\mathcal{E}$ will be employed by the sources and the relay for encoding their messages to lattice codewords.

The encoding scheme is performed as follows. Each source s_m randomly generates two information vectors $\mathbf{w}_m^{\mathrm{Re}}, \mathbf{w}_m^{\mathrm{Im}} \in \mathbb{F}_p^{nR/2}$. Together, these information vectors form $\mathbf{w}_m = (\mathbf{w}_m^{\mathrm{Re}}, \mathbf{w}_m^{\mathrm{Im}}) \in \mathbb{F}_p^{nR}$, which is then encoded to a complex-valued vector in the following way. $\mathbf{w}_m^{\mathrm{Re}}$ and $\mathbf{w}_m^{\mathrm{Im}}$ are respectively mapped to $\mathbf{x}_m^{\mathrm{Re}} \in \mathcal{C}$ and $\mathbf{x}_m^{\mathrm{Im}} \in \mathcal{C}$ using $\mathcal{E}$, i.e.,

$$\mathbf{x}_m^{\mathrm{Re}} = \mathcal{E}(\mathbf{w}_m^{\mathrm{Re}}), \tag{3.2}$$

$$\mathbf{x}_m^{\mathrm{Im}} = \mathcal{E}(\mathbf{w}_m^{\mathrm{Im}}). \tag{3.3}$$

Subsequently, these vectors form a complex-valued vectors $\mathbf{x}_m = \mathbf{x}_m^{\mathrm{Re}} + i\mathbf{x}_m^{\mathrm{Im}} \in \mathbb{C}^n$ which is to be broadcast to the destination and the relay. We assume that a dithering technique is employed. However, we omit the notations for the sake of ease exposition. Dithering is important for ensuring the resulting effective noise is independent of the underlying lattice codewords. Furthermore, it ensures that each $\mathbf{x}_m$ satisfies the average power constraint $\mathbb{E}\{\|\mathbf{x}_m\|^2\} \leq nP$.

3.3.2 Transmission Rounds

The end-to-end information transmission is divided into two rounds. In the first round, the sources simultaneously broadcast $\mathbf{x}_m$ to the destination and the relay. For $j \in \{r, d\}$, let $\mathbf{z}_j$ be a noise vector at terminal j, and recall that h_{ij} denotes the channel coefficient from terminal i to j, $i \in \{s_1, \ldots, s_M\}$. The destination and the relay respectively receive noisy superposition signals

$$\mathbf{y}_d^{(1)} = \sum_{m=1}^{M} \sqrt{g_{sd}} h_{s_m d} \mathbf{x}_m + \mathbf{z}_d^{(1)}, \tag{3.4}$$

$$\mathbf{y}_r^{(1)} = \sum_{m=1}^{M} \sqrt{g_{sr}} h_{s_m r} \mathbf{x}_m + \mathbf{z}_r^{(1)}. \tag{3.5}$$

At the end of the first round, the relay does not attempt to decode $\mathbf{w}_1, \ldots, \mathbf{w}_M$ separately as usually done in conventional MARC schemes. Rather, it adopts the compute-and-forward technique to directly decode a linear combination of $\mathbf{w}_1, \ldots, \mathbf{w}_M$. Let $\mathbf{u}_r \triangleq f_r(\mathbf{w}_1, \ldots, \mathbf{w}_M)$ be the desired linear combination. How to select a "good" $\mathbf{u}_r$ is one of the problems that we will tackle in the next section. In the second round, the relay encodes $\mathbf{u}_r$ to a complex-valued vector $\mathbf{x}_r \in \mathbb{C}^n$ using the same encoding scheme described in Subsection 3.3.1 and forwards it to the destination. We shall note that to increase the transmission efficiency, the second round is only used when some conditions is met. It will be discussed further in Section 3.4. The received signal at the destination is given by

$$\mathbf{y}_d^{(2)} = \sqrt{g_{rd}} h_{rd} \mathbf{x}_r + \mathbf{z}_d^{(2)}. \tag{3.6}$$

3.3.3 Computing Linear Combinations

As mentioned above, by the end of the first transmission round, the relay decodes a combination of the transmitted information messages. In fact, it is not only the relay. Because at least M linear combinations are required to recover all the transmitted information messages, the destination also needs to decode at least $M-1$ linear combinations. For simplicity, let us focus on how a receiver, which may represent either the destination or the relay, decodes some linear combinations.

Before going further, let us rewrite the received signals (3.4) or (3.5) in a simpler form, omitting the notations for the relay or the destination. Let $\mathbf{h} = [h_1, \ldots, h_M] \in \mathbb{C}^M$ be the channel coefficient vector and g be the geometric gain from the sources to the receiver. The received signal is rewritten as

$$\mathbf{y} = \sum_{m=1}^{M} \sqrt{g} h_m \mathbf{x}_m + \mathbf{z}. \tag{3.7}$$

Assume that the receiver expects to decode $L \leq M$ linear combinations $\mathbf{u}_1, \ldots, \mathbf{u}_L \in \mathbb{F}_p^{nR}$. For $l \in \{1, \ldots, L\}$, the receiver selects coefficients $q_{lm}^{\mathrm{Re}}, q_{lm}^{\mathrm{Im}} \in \mathbb{F}_p$ and attempts to decode two equations

$$\mathbf{u}_l^{\mathrm{Re}} = \bigoplus_{m=1}^{M} q_{lm}^{\mathrm{Re}} \mathbf{w}_m^{\mathrm{Re}} \oplus (-q_{lm}^{\mathrm{Im}}) \mathbf{w}_m^{\mathrm{Im}}, \tag{3.8}$$

$$\mathbf{u}_l^{\mathrm{Im}} = \bigoplus_{m=1}^{M} q_{lm}^{\mathrm{Im}} \mathbf{w}_m^{\mathrm{Re}} \oplus (q_{lm}^{\mathrm{Re}}) \mathbf{w}_m^{\mathrm{Im}}, \tag{3.9}$$

where $(-q_m)$ denotes the additive inverse of q_m. The linear combinations $\mathbf{u}_l$ is then obtained as $\mathbf{u}_l \triangleq (\mathbf{u}_l^{\mathrm{Re}}, \mathbf{u}_l^{\mathrm{Im}})$.

Although the desired linear combinations are evaluated over the finite field

$\mathbb{F}_p$, the channels operates over the complex field $\mathbb{C}$. This issue can be overcome by exploiting the real-valued decomposition of a complex-valued number. To be precise, the receiver selects an integer linear coefficient $\mathbf{a}_l = [a_{l1}, \ldots, a_{lM}] \in \mathbb{Z}[i]^M$ and decodes the corresponding lattice equation

$$\mathbf{v}_l = \sum_{m=1}^{M} a_{lm}\mathbf{x}_m \bmod \Lambda_{\mathrm{s}}. \tag{3.10}$$

Now, $\mathbf{v}_l$ can be written as $\mathbf{v}_l = \mathbf{v}_l^{\mathrm{Re}} + i\mathbf{v}_l^{\mathrm{Im}}$, where

$$\mathbf{v}_l^{\mathrm{Re}} \triangleq \Re(\mathbf{v}_l) = \left[\sum_{m=1}^{M} \Re(a_{lm})\mathbf{x}_m^{\mathrm{Re}} - \Im(a_{lm})\mathbf{x}_m^{\mathrm{Im}}\right] \bmod \Lambda_{\mathrm{s}} \tag{3.11}$$

$$\mathbf{v}_l^{\mathrm{Im}} \triangleq \Im(\mathbf{v}_l) = \left[\sum_{m=1}^{M} \Im(a_{lm})\mathbf{x}_m^{\mathrm{Re}} + \Re(a_{lm})\mathbf{x}_m^{\mathrm{Im}}\right] \bmod \Lambda_{\mathrm{s}}. \tag{3.12}$$

Once the destination obtains $\mathbf{v}_l$, it can recover $\mathbf{u}_l^{\mathrm{Re}}$ and $\mathbf{u}_l^{\mathrm{Im}}$ using the inverse of $\mathcal{E}$ as follows

$$\mathbf{u}_l^{\mathrm{Re}} \triangleq \mathcal{E}^{-1}(\mathbf{v}_l^{\mathrm{Re}}) = \bigoplus_{m=1}^{M} q_{lm}^{\mathrm{Re}}\mathbf{w}_m^{\mathrm{Re}} \oplus (-q_{lm}^{\mathrm{Im}})\mathbf{w}_m^{\mathrm{Im}}, \tag{3.13}$$

$$\mathbf{u}_l^{\mathrm{Im}} \triangleq \mathcal{E}^{-1}(\mathbf{v}_l^{\mathrm{Im}}) = \bigoplus_{m=1}^{M} q_{lm}^{\mathrm{Im}}\mathbf{w}_m^{\mathrm{Re}} \oplus (q_{lm}^{\mathrm{Re}})\mathbf{w}_m^{\mathrm{Im}}. \tag{3.14}$$

Given the choice of $\mathbf{a}_l$, q_{lm}^{Re} and q_{lm}^{Im} are equivalent to $q_{lm}^{\mathrm{Re}} = \Re(a_{lm}) \bmod p$ and $q_{lm}^{\mathrm{Im}} = \Im(a_{lm}) \bmod p$. This implies that the selection of integer coefficient in the Gaussian integer domain corresponds to the selection of coefficients in the finite field domain. How to select "good" integer coefficients will be addressed later.

The next problem is then how to obtain the lattice equation $\mathbf{v}_l$. Similar to the compute-and-forward with real-valued channels described in Chapter 2, the receiver scales $\mathbf{y}$ with a scaling factor $\alpha_l \in \mathbb{C}$ and computes

$$\tilde{\mathbf{y}}_l = [\alpha_l \mathbf{y}] \bmod \Lambda_{\mathrm{s}} \tag{3.15}$$

$$= \left[\sum_{m=1}^{M} \alpha_l \sqrt{g} h_m \mathbf{x}_m + \alpha_l \mathbf{z}\right] \bmod \Lambda_{\mathrm{s}} \tag{3.16}$$

$$= \left[\sum_{m=1}^{M} a_{lm}\mathbf{x}_m + \sum_{m=1}^{M} (\alpha_l \sqrt{g} h_m \mathbf{x}_m - a_{lm}\mathbf{x}_m) + \alpha_l \mathbf{z} \right] \bmod \Lambda_{\mathrm{s}} \tag{3.17}$$

$$= [\mathbf{v}_l + \mathbf{z}_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g)] \bmod \Lambda_{\mathrm{s}}, \tag{3.18}$$

where

$$\mathbf{z}_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g) = \sum_{m=1}^{M} (\alpha_l \sqrt{g} h_m \mathbf{x}_m - a_{lm}\mathbf{x}_m) + \alpha_l \mathbf{z} \tag{3.19}$$

is the effective noise. Subsequently, it produces estimates of $\mathbf{v}_l^{\mathrm{Re}}$ and $\mathbf{v}_l^{\mathrm{Im}}$ by quantizing the real and imaginary components of $\tilde{\mathbf{y}}_l$ with respect to Λ_{c} and performs modulo operation on them with respect to Λ_{s}, i.e.,

$$\hat{\mathbf{v}}_l^{\mathrm{Re}} = Q_{\Lambda_{\mathrm{c}}} \left(\Re(\tilde{\mathbf{y}}_l) \right) \bmod \Lambda_{\mathrm{s}} \tag{3.20}$$

$$\hat{\mathbf{v}}_l^{\mathrm{Im}} = Q_{\Lambda_{\mathrm{c}}} \left(\Im(\tilde{\mathbf{y}}_l) \right) \bmod \Lambda_{\mathrm{s}}. \tag{3.21}$$

Finally, the estimates of $\mathbf{u}_l^{\mathrm{Re}}$ and $\mathbf{u}_l^{\mathrm{Im}}$ are obtained by using the inverse of $\mathcal{E}$,

$$\hat{\mathbf{u}}_l^{\mathrm{Re}} = \mathcal{E}^{-1} \left(\hat{\mathbf{v}}_l^{\mathrm{Re}} \right), \tag{3.22}$$

$$\hat{\mathbf{u}}_l^{\mathrm{Im}} = \mathcal{E}^{-1} \left(\hat{\mathbf{v}}_l^{\mathrm{Im}} \right). \tag{3.23}$$

The estimate of $\mathbf{u}_l$ is then recovered as $\hat{\mathbf{u}}_l = \left(\hat{\mathbf{u}}_l^{\mathrm{Re}}, \hat{\mathbf{u}}_l^{\mathrm{Im}} \right)$.

In order for the receiver to be able to decode $\mathbf{u}_l$ with low error probability, the scaling factor α_l has to be chosen such that the variance of the effective noise $\mathbf{z}_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g)$ is minimized. Let $\sigma^2_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g)$ be the variance of $\mathbf{z}_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g)$ defined as

$$\sigma^2_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g) \triangleq \frac{1}{n} \mathbb{E} \left\{ \left\| \mathbf{z}_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g) \right\|^2 \right\} \tag{3.24}$$

$$= \left\| \alpha_l \sqrt{g} \mathbf{h} - \mathbf{a}_l \right\|^2 P + |\alpha_l|^2. \tag{3.25}$$

One can easily show that the optimal value for α_l is given by

$$\alpha_l^{\mathrm{opt}} \triangleq \arg\min_{\alpha_l} \sigma^2_{\mathrm{eff}}(\alpha_l, \mathbf{a}_l, \mathbf{h}, g) \tag{3.26}$$

$$= \frac{P\sqrt{g}\mathbf{h}^H \mathbf{a}_l}{1 + Pg \left\| \mathbf{h} \right\|^2}. \tag{3.27}$$

As a summary, the receiver decodes linear combination $\mathbf{u}_l$ in the following way. It first selects an integer coefficient vector $\mathbf{a}_l \in \mathbb{Z}[i]^M$, then computes α_l^{opt} and uses it as the scaling factor α_l. Next, it scales the received signal and performs modulo operation with respect to Λ_{s}. Finally, the desired linear combination is obtained by performing operations described in (3.20), (3.21), (3.22), and (3.23) sequentially.

The next question is then how to choose the integer coefficient vector $\mathbf{a}_l$. In principle, it is possible for the receiver to choose any integer coefficient vector. However, the selection of the coefficient vector has a significant impact on the achievable computation rate and consequently on the error performance. Therefore, $\mathbf{a}_l$ has to be chosen wisely.

The achievable computation rate of compute-and-forward in complex-valued channels is given in the following theorem.

Theorem 3.3.1 (Computation Rate for Complex-Valued Channels [14])**:** Consider a complex-valued Gaussian network with M transmitters that simultaneously transmit their messages with average power constraint P to a receiver. Let $\mathbf{h} = [h_1, \ldots, h_M]^T \in \mathbb{C}^M$ be the channel coefficients and g be the geometric gain from the transmitters to the receiver. Given a coefficient vector $\mathbf{a} = [a_1, \ldots, a_M] \in \mathbb{Z}[i]$, the receiver can decode the corresponding linear combination of transmitted messages with low error probability so long as the message rate is less than the *computation rate* given by

$$R_{\text{cp}}(\mathbf{a}_l, \mathbf{h}, g) = \log^+\left(\left(\|\mathbf{a}\|^2 - \frac{Pg\left|\mathbf{h}^H\mathbf{a}\right|^2}{1 + Pg\|\mathbf{h}\|^2}\right)^{-1}\right). \tag{3.28}$$

From the above theorem, it is clear that integer coefficient vectors should be chosen such that the computation rate is maximized. For $l \in \{1, \ldots, L\}$, let $\mathbf{a}_l$ be the integer coefficient vectors of the desired linear combination $\mathbf{u}_l$, respectively. Let $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_L]^T$. Note that $\mathbf{a}_1, \ldots, \mathbf{a}_L$ are linearly independent, and thus, $\text{rank}(\mathbf{A}) = L$. The receiver should choose $\mathbf{A}$ such that

$$\mathbf{A} = \underset{\substack{\tilde{\mathbf{A}} = [\tilde{\mathbf{a}}_1, \ldots, \tilde{\mathbf{a}}_L] \in \mathbb{Z}[i]^{L \times M}, \\ \text{rank}(\tilde{\mathbf{A}}) = L}}{\arg\max} \; \min_{l=1,\ldots,L} R_{\text{cp}}(\tilde{\mathbf{a}}_l, \mathbf{h}, g). \tag{3.29}$$

It can be shown that the computation rate $R_{\text{cp}}(\mathbf{a}_l, \mathbf{h}, g)$ can be written as [13]

$$R_{\text{cp}}(\mathbf{a}_l, \mathbf{h}, g) = \log^+ \left(\frac{1}{\mathbf{a}_l^H \mathbf{M} \mathbf{a}_l} \right), \tag{3.30}$$

where

$$\mathbf{M} = \mathbf{I} - \frac{Pg}{1 + Pg \left\| \mathbf{h} \right\|^2} \mathbf{h}\mathbf{h}^H. \tag{3.31}$$

One can observe that $\mathbf{M}$ is a positive definite matrix, and thus, using Cholesky factorization, it can further be decomposed into

$$\mathbf{M} = \mathbf{B}^H \mathbf{B}, \tag{3.32}$$

where $\mathbf{B}$ is an upper triangular matrix.

The problem defined in (3.29) now can be written as

$$\mathbf{A} = \underset{\substack{\tilde{\mathbf{A}} = [\tilde{\mathbf{a}_1}, \ldots, \tilde{\mathbf{a}_L}] \in \mathbb{Z}[i]^{L \times M}, \\ \text{rank}\left(\tilde{\mathbf{A}}\right) = L}}{\arg\max} \; \min_{l=1,\ldots,L} \log^+ \left(\frac{1}{\tilde{\mathbf{a}}_l{}^H \mathbf{M} \tilde{\mathbf{a}}_l} \right) \tag{3.33}$$

$$= \underset{\substack{\tilde{\mathbf{A}} = [\tilde{\mathbf{a}_1}, \ldots, \tilde{\mathbf{a}_L}] \in \mathbb{Z}[i]^{L \times M}, \\ \text{rank}\left(\tilde{\mathbf{A}}\right) = L}}{\arg\min} \; \max_{l=1,\ldots,L} \tilde{\mathbf{a}}_l{}^H \mathbf{M} \tilde{\mathbf{a}}_l \tag{3.34}$$

$$= \underset{\substack{\tilde{\mathbf{A}} = [\tilde{\mathbf{a}_1}, \ldots, \tilde{\mathbf{a}_L}] in \mathbb{Z}[i]^{L \times M}, \\ \text{rank}\left(\tilde{\mathbf{A}}\right) = L}}{\arg\min} \; \max_{l=1,\ldots,L} \left\| \mathbf{B} \tilde{\mathbf{a}}_l \right\|^2. \tag{3.35}$$

From (3.35), it can be said that finding L “best” integer coefficient vectors $\mathbf{a}_1, \ldots, \mathbf{a}_L$ with respect to the computation rate is equivalent to finding integer vectors providing L successive minima of the lattice with generator matrix $\mathbf{B}$. See the definition of the successive minima of a lattice in Definition 2.1.8. Thus, to find $\mathbf{A}$, we can employ existing algorithms for finding the successive minima of a lattice such that the Fincke-Pohst algorithm [46], the Schnorr-Euchner algorithm [47], the LLL algorithm [48], and their variations [49–54].

3.3.4 Recovering Information Messages

We have discussed that upon receiving noisy superposition signal in (3.4) and (3.5), the destination and the relay compute linear combinations of the transmitted messages. However, the ultimate goal of the destination is to recover the transmitted information messages $\mathbf{w}_1, \ldots, \mathbf{w}_M$. To this end, the destination requires at least M linear combinations. Let us assume that the destination possesses linear combinations $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_M$. These linear combinations may be obtained either by itself, or with the help of the relay. How these linear combinations are obtained will be addressed in the next section.

Let $\mathbf{a}_1, \ldots, \mathbf{a}_M \in \mathbb{Z}[i]^M$ be the integer coefficient vectors corresponding to $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_M$ and let $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_M]^T$; we refer to $\mathbf{A}$ as the *integer coefficient matrix* or just *coefficient matrix.* The corresponding coefficient matrix in $\mathbb{F}_p$ can be written as

$$\mathbf{Q} = \begin{bmatrix} \Re(\mathbf{A}) & -\Im(\mathbf{A}) \\ \Im(\mathbf{A}) & \Re(\mathbf{A}) \end{bmatrix} \bmod p, \tag{3.36}$$

where the modulo operation is element-wise. For all $m \in \{1, \ldots, M\}$, let $\hat{\mathbf{w}}_m = (\hat{\mathbf{w}}_m^{\text{Re}}, \hat{\mathbf{w}}_m^{\text{Im}})$ be the estimate of $\mathbf{w}_m$. The destination decodes the transmitted information messages by solving the following linear equation

$$\begin{bmatrix} \hat{\mathbf{u}}_1^{\text{Re}} \\ \vdots \\ \hat{\mathbf{u}}_M^{\text{Re}} \\ \hat{\mathbf{u}}_1^{\text{Im}} \\ \vdots \\ \hat{\mathbf{u}}_M^{\text{Im}} \end{bmatrix} = \mathbf{Q} \begin{bmatrix} \hat{\mathbf{w}}_1^{\text{Re}} \\ \vdots \\ \hat{\mathbf{w}}_M^{\text{Re}} \\ \hat{\mathbf{w}}_1^{\text{Im}} \\ \vdots \\ \hat{\mathbf{w}}_M^{\text{Im}} \end{bmatrix} \tag{3.37}$$

where all operations are performed over finite field $\mathbb{F}_p$. It should be noted that the destination can solve the above linear equation system if and only if the matrix $\mathbf{Q}$ is full-rank. Therefore, we should take into account the probability that the coefficient matrix is not full rank when designing MARC with compute-and-forward. Nevertheless, it has been shown in [14, Sec. VI] by taking the blocklength n and field size p to be large enough, rather than checking rank of $\mathbf{Q}$ over $\mathbb{F}_q$, it is suffice to check whether $\mathbf{A}$ is full-rank over $\mathbb{C}$, which is certainly easier.

3.4 Proposed Cooperation Strategies

In this section we propose two cooperation strategies between the destination and the relay. In the first strategy, the relay helps the destination by providing its local "best" linear combination without taking into account whether the resulting coefficient matrix is full-rank or not. In the second strategy, the relay assists the destination by forwarding the "best" linear combination that ensures the resulting coefficient matrix is full-rank. Since the relay needs to know the linear combinations that the destination possesses, a sufficient amount of feedback is needed in the second strategy.

Before performing cooperation, the destination and the relay prepare M linearly independent coefficient vectors. Note that at this stage, the corresponding linear combinations are not yet decoded. Let $\mathcal{A}_d = \{\mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_M}\}$ and $\mathcal{A}_r = \{\mathbf{a}_{r_1}, \ldots, \mathbf{a}_{r_M}\}$ be the sets of coefficient vectors prepared by the destination and the relay, respectively. The elements of $\mathcal{A}_d$ and $\mathcal{A}_r$ are sorted based on the resulting computation rates. Specifically, let $\mathbf{h}_{sd} = [h_{s_1d}, \ldots, h_{s_Md}]$ and $\mathbf{h}_{sr} = [h_{s_1r}, \ldots, h_{s_Mr}]$, and define

$$R_{\text{cp},d}^{(m)} \triangleq R_{\text{cp}}(\mathbf{a}_{d_m}, \mathbf{h}_{sd}, g_{sd}), \tag{3.38}$$

$$R_{\text{cp},r}^{(m)} \triangleq R_{\text{cp}}(\mathbf{a}_{r_m}, \mathbf{h}_{sr}, g_{sr}). \tag{3.39}$$

The coefficient vectors $\mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_M}$ and $\mathbf{a}_{r_1}, \ldots, \mathbf{a}_{r_M}$ are respectively sorted such that $R_{\text{cp},d}^{(1)} \geq \cdots \geq R_{\text{cp},d}^{(M)}$ and $R_{\text{cp},r}^{(1)} \geq \cdots \geq R_{\text{cp},r}^{(M)}$. Moreover, the elements of $\mathcal{A}_d$ and $\mathcal{A}_r$ are integer vectors that provide M successive minima of their corresponding lattices generated by a matrix $\mathbf{B}$ described in (3.32). This implies that $\mathbf{a}_{d_1}$ and $\mathbf{a}_{r_1}$ are the local optimal coefficient vectors at the destination and the relay, respectively.

Based on $\mathcal{A}_d$ and $\mathcal{A}_r$, we propose two cooperation strategies as follows.

3.4.1 Limited Feedback Strategy

The first strategy is very simple, yet it outperforms the existing cooperation strategies in the literature. In this strategy, there are two steps for decoding the transmitted messages. The first is, upon receiving $\mathbf{y}_d^{(1)}$, the destination directly attempts to decode the transmitted information messages without the help of the relay. We refer to this step of decoding as the *direct decoding*. Specifically, based on the integer coefficient vector in $\mathcal{A}_d$, the destination decodes the corresponding M linear combinations. Let $\hat{\mathbf{u}}_{d_1}, \ldots, \hat{\mathbf{u}}_{d_M}$ be the decoded linear combinations and $\mathbf{A}_d = [\mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_M}]^T$. Based on $\hat{\mathbf{u}}_{d_1}, \ldots, \hat{\mathbf{u}}_{d_M}$ and $\mathbf{A}_d$, the transmitted messages are recovered by solving the resulting equation system similar to (3.37).

If the destination successfully decodes the transmitted messages, the sources can transmit the next messages. Otherwise, the destination sends feedback to the sources and the relay, asking the sources to wait and the relay to help the decoding. The feedback size is *limited* to only 1-bit and is assumed to always be received correctly.[2] Note that by default, the second round of transmission is not required; the destination will not send any feedback when the direct decoding succeeds. This scheme ensures high transmission efficiency.

The second step of decoding, to which we refer as the *cooperative decoding*, is carried out when the relay receives feedback from the destination. The relay chooses its local best coefficient vector $\mathbf{a}_{r_1}$, decodes the corresponding linear combinations and forwards it to the destination. Let $\hat{\mathbf{u}}_{r_1}$ be the linear combination forwarded by the relay. The destination now has an additional linear combination $\hat{\mathbf{u}}_{r_1}$ with coefficient vector $\mathbf{a}_{r_1}$. Let $\mathbf{A}_{\text{cop}} = [\mathbf{a}_{r_1}, \mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_{M-1}}]^T$. Based on $\hat{\mathbf{u}}_{r_1}, \hat{\mathbf{u}}_{d_1}, \ldots, \hat{\mathbf{u}}_{d_{M-1}}$ and $\mathbf{A}_{\text{cop}}$, the destination then again decodes the transmitted messages. We shall note that $\mathbf{A}_{\text{cop}}$ may not be full-rank which clearly will prevent the destination to decode the information messages correctly.

3.4.2 Sufficient Feedback Strategy

The second strategy is quite similar to the first in the sense that it also uses two decoding steps. The first step is exactly the same as the limited feedback strategy. The destination attempts to *directly* decode the information messages by using its own linear combinations. Using $\hat{\mathbf{u}}_{d_1}, \ldots, \hat{\mathbf{u}}_{d_M}$ and $\mathbf{A}_d$ the destination recovers $\mathbf{w}_1, \ldots, \mathbf{w}_M$ by solving an equation system similar to (3.37) . If the direct decoding succeeds, the sources can transmit their next information messages. Otherwise, the destination sends feedback to the sources and the relay. The feedback must contain information about the $M-1$ best integer coefficient vectors of the destination, i.e., $\mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_{M-1}}$. Compared to the first strategy, the feedback size is *sufficiently*

[2]The destination can only ask for help one time. The case for multiple requests in the form of an automatic repeat request (ARQ) protocol is left for future work, cf. Chapter 5

bigger. However, it is far smaller compared to the overhead used in [20] where channel gains are known to all terminals. Besides perfectly received, it is also assumed that the feedback size is negligible compared to the amount of information that can be transmitted within one coherence time. In practice, the feedback required will require at least $M(M-1)\log p$ bits.

Let $\mathbf{a}_{r_*}$ be the integer coefficient vector selected by the relay. Let $\mathbf{A}_{rl} = [\mathbf{a}_{r_l}, \mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_{M-1}}]^T$. The relay must select $\mathbf{a}_{r_*}$ such that

$$\mathbf{a}_{r_*} = \underset{\substack{\mathbf{a}_{r_l} \in \mathbf{a}_{r_1}, \ldots, \mathbf{a}_{r_M}, \\ \text{rank}(\mathbf{A}_{rl}) = M}}{\arg\max} \; R^{(l)}_{\text{cp},r}. \tag{3.40}$$

This way of selecting coefficient vector ensures that the resulting coefficient matrix is full rank while keeping the achievable rate as high as possible. Subsequently, the relay decodes the linear combination of the messages that corresponds to the selected coefficient vector $\mathbf{a}_{r_*}$ and the forwards it to the destination. Now, because the destination has enough linear combinations, it can re-decode the information messages by solving the resulting linear equation system according to (3.37).

3.5 Performance Analysis

This section provides performance analysis of the proposed cooperation strategies in terms of outage probability. In wireless communication systems, an outage event is the situation where the channel condition is poor such that transmitters and receivers cannot communicate reliably at a certain fixed data rate. The probability of outage events is widely used as a metric to evaluate the performance of wireless communication systems. It is a very useful metric that can be used to analyze the relationship among the communication efficiency, reliability, SNR, and channel fading [55].

3.5.1 Limited Feedback Strategy

We start from the performance analysis of the limited feedback (lim-FB) strategy. In this strategy, the destination has two possible ways of decoding the transmitted messages, the direct decoding and cooperative decoding. In the direct decoding, the destination attempts to decode the transmitted messages by itself. Specifically, it computes linear combinations with coefficients $\mathbf{a}_{d_1}, \ldots, \mathbf{a}_{d_M}$ and solves the corresponding linear equation system. Let e_1 be the outage event for the direct decoding. e_1 is defined as

$$e_1 \triangleq \left\{ \min_{m \in \{1,\dots,M\}} R_{\mathrm{cp},d}^{(m)} < R \right\} \tag{3.41}$$

$$= \left\{ R_{\mathrm{cp},d}^{(M)} < R \right\}, \tag{3.42}$$

where R is the coding rate employed by the sources and the relay. Note that (3.42) is due to the fact $R_{\mathrm{cp},d}(1) \geq \cdots \geq R_{\mathrm{cp},d}^{(M)}$, see Section 3.4. Intuitively, we can think that the outage event for direct decoding is determined by the worst coefficient vector $\mathbf{a}_{d_M}$.

In the cooperative decoding, the relay forwards its local best linear combination and the destination uses its $M-1$ best linear combinations and solves the resulting linear equation system to decode the transmitted messages. In order for the cooperative decoding to succeed, all the linear combinations have to be correctly decoded and the resulting coefficient matrix has to be full rank. Let e_2 be the outage event for the cooperative decoding and $\mathbf{A}$ be the resulting coefficient matrix. The outage event in the cooperative decoding is defined as

$$\begin{aligned} e_2 &\triangleq \left\{ \min_{m \in \{1,\dots,M-1\}} R_{\mathrm{cp},d}^{(m)} < R \right\} \cup \left\{ R_{\mathrm{cp},r}^{(1)} < R \right\} \cup \left\{ R_{\mathrm{rd}} < R \right\} \cup \left\{ \mathrm{rank}(\mathbf{A}) < M \right\} \\ &= \left\{ R_{\mathrm{cp},d}^{(M-1)} < R \right\} \cup \left\{ R_{\mathrm{cp},r}^{(1)} < R \right\} \cup \left\{ R_{\mathrm{rd}} < R \right\} \cup \left\{ \mathrm{rank}(\mathbf{A}) < M \right\} \end{aligned} \tag{3.43}$$

where $R_{\mathrm{rd}} = \log(1 + |h_{rd}|^2 \gamma_{rd})$ is the achievable rate of the point-to-point relay-destination link.

Let $P_{\mathrm{def}} \triangleq \Pr(\mathrm{rank}(\mathbf{A}) < M)$. At the end, the destination fails to decode the transmitted messages if and only if both the direct and the cooperative decodings fail. Therefore, the outage probability is given by

$$\begin{aligned} P_{\mathrm{out}} &\triangleq \Pr(e_1 \cap e_2) \qquad (3.44) \\ &= \Pr\Big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \big(\{R_{\mathrm{cp},d}^{(M-1)} < R\} \cup \{R_{\mathrm{cp},r}^{(1)} < R\} \cup \{R_{\mathrm{rd}} < R\} \\ &\quad \cup \{\mathrm{rank}(\mathbf{A}) < M\} \big) \Big) \\ &= \Pr\Big(\{R_{\mathrm{cp},d}^{(M-1)} < R\} \cup \big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \{R_{\mathrm{cp},r}^{(1)} < R\} \big) \\ &\quad \cup \big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \{R_{\mathrm{rd}} < R\} \big) \cup \big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \{\mathrm{rank}(\mathbf{A}) < M\} \big) \Big) \end{aligned}$$

$$
\begin{aligned}
&\overset{(a)}{\leq} \Pr\left(\{R_{\text{cp},d}^{(M-1)} < R\}\right) + \Pr\left(\{R_{\text{cp},d}^{(M)} < R\} \cap \{R_{\text{cp},r}^{(1)} < R\}\right) \\
&\quad + \Pr\left(\{R_{\text{cp},d}^{(M)} < R\} \cap \{R_{\text{rd}} < R\}\right) + \Pr\left(\{R_{\text{cp},d}^{(M)} < R\} \cap \{\text{rank}(\mathbf{A}) < M\}\right)\Big) \\
&\overset{(b)}{\leq} \Pr\left(R_{\text{cp},d}^{(M-1)} < R\right) + \Pr\left(R_{\text{cp},d}^{(M)} < R\right) \Pr\left(R_{\text{cp},r}^{(1)} < R\right) \\
&\quad + \Pr\left(R_{\text{cp},d}^{(M)} < R\right) \Pr\left(R_{\text{rd}} < R\right) + \min\left\{ \Pr\left(R_{\text{cp},d}^{(M)} < R\right), P_{\text{def}}\right\}
\end{aligned} \tag{3.45}
$$

where (a) is due to the union bound and (b) is because the channels of sources-destination, sources-relay, and relay-destination are independent. The last part of (b) is due to the Fréchet bound.

Let us recall the definition of diversity order [29] achieved by a system.

Definition 3.5.1 (Diversity order)**:** For a system with outage probability P_{out}, the diversity order of the system is defined as

$$
d \triangleq -\lim_{\gamma\to\infty} \frac{\log P_{\text{out}}}{\log \gamma}, \tag{3.46}
$$

where γ is the average SNR of the channel.

From the above definition, we can estimate the diversity order of a system from its outage probability curve. We will use this for observing our numerical results from computer simulations. For simplicity of presentation, we will use the following notation.

Definition 3.5.2 (Exponential Equality)**:** A function $f(\gamma)$ is said to be exponentially equal to b, denoted by $f(\gamma) \doteq \gamma^b$, if

$$
\lim_{\gamma\to\infty} \frac{\log f(\gamma)}{\log b} = b. \tag{3.47}
$$

The relation $\dot{\leq}$ is defined similarly.

From (3.45), we can see that the diversity order achieved by the lim-FB strategy depends on the outage probability of the selected linear combinations at the destination and the relay, the outage probability of the relay-destination link, and the probability of rank deficient coefficient matrix during the cooperative decoding. For the relay-destination link, because it is just a *normal* point-to-point wireless communication, the achieved diversity order is one [27]. For the compute-and-forward at the destination and the relay, it is possible to prove that they achieve

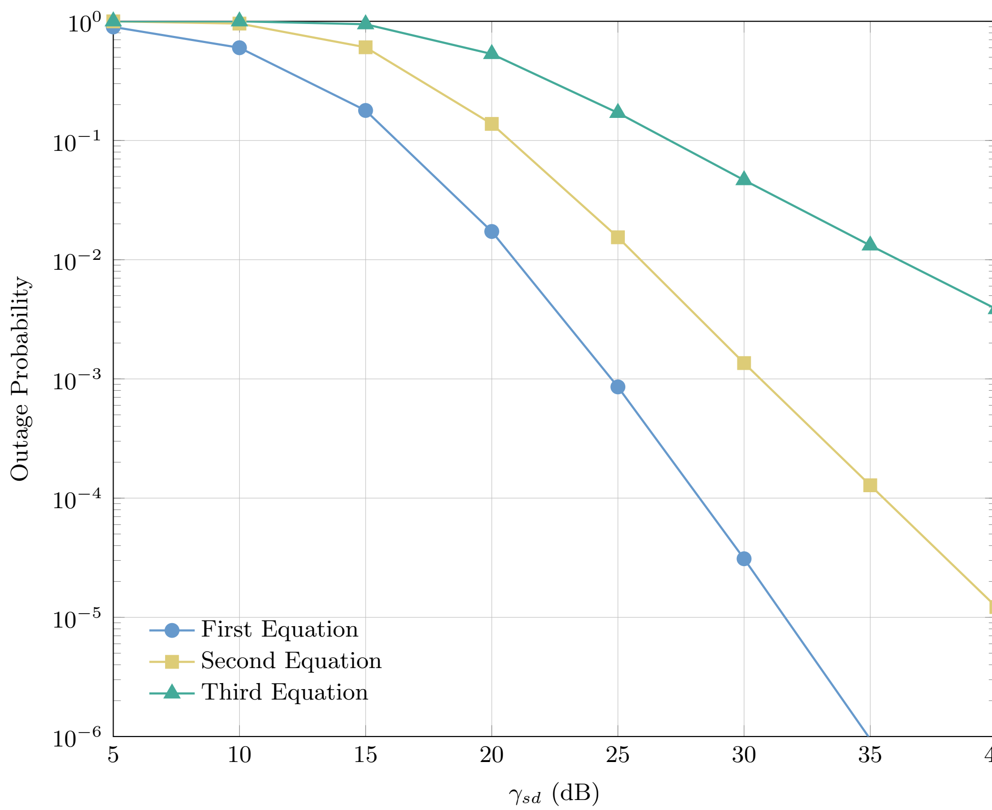


Fig. 3.2. The outage probability of the $M = 3$ best linear equations of a compute-and-forward system with three transmitters. The first best equation achieves third-order diversity, while the second achieves second-order diversity. The last equation achieves first-order diversity.

full diversity gain when they use their best coefficient vector. However, we found that it is very hard to theoretically show the achieved diversity order when a non-best linear coefficient vector is selected. For this reason, we numerically evaluate the outage probabilities of the compute-and-forward with M best linear coefficient vectors. By "best" here, we mean the coefficient vectors that provide successive minima of the resulting lattice $\mathbf{B}$ in (3.32). In Fig. 3.2, we show numerical results of outage probabilities for each equation of a compute-and-forward system with three transmitters. Based on the slopes of the outage probability curves, we can see that the first best linear equation achieves diversity gain of order three, while the second best achieves diversity gain of order two. Lastly, the worst linear equation achieves first-order diversity gain. Similar results for the MARC with two sources are also shown if Fig. 3.3 where the best linear equation achieves full diversity order and the worst linear equation only achieves the first-order. Therefore, we can

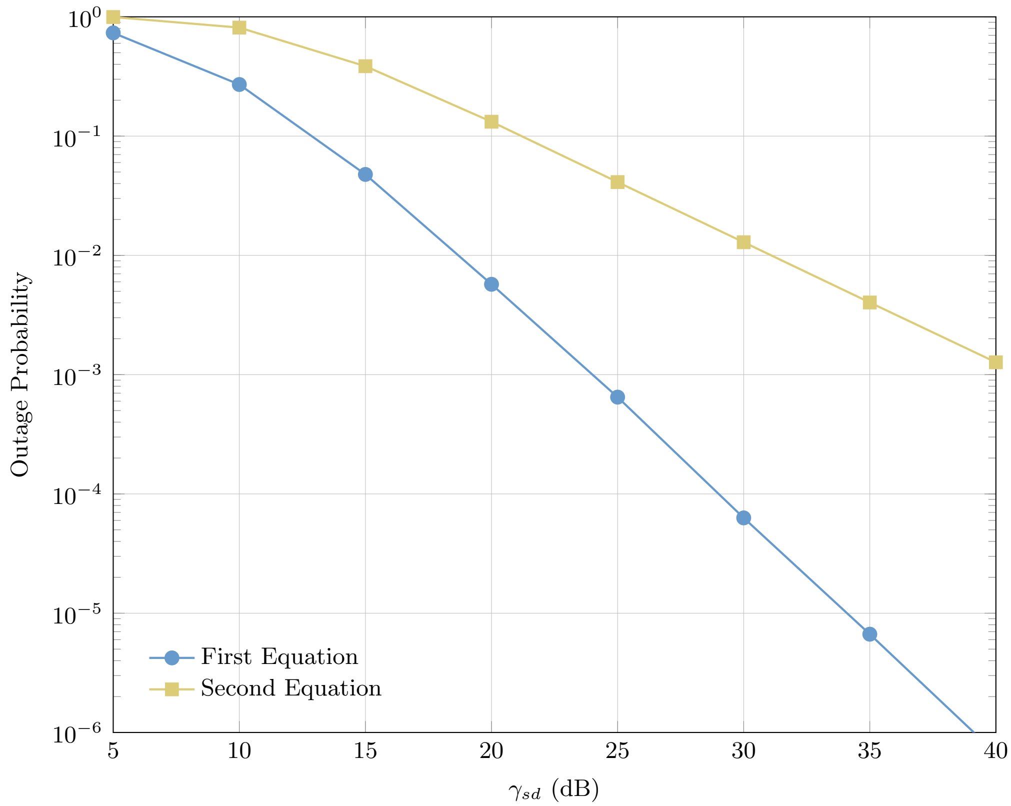


Fig. 3.3. The outage probability of the $M = 2$ best linear equations of a compute-and-forward system with two transmitters. The first best equation achieves second-order diversity and the last equation achieves first-order diversity.

conjecture that the m-th best linear equation of a compute-and-forward system achieves diversity gain of order $M - m + 1$.

Now we are left with the probability of rank deficient coefficient matrix. Unfortunately, it is also hard to obtain a closed-form expression for P_{def}. Similarly, we use a numerical approach to show the behavior of P_{def}. In Fig. 3.4, we show the rank deficient probability of the coefficient matrices constructed during the cooperative decoding of the lim-FB strategy. The evaluation is performed by adjusting the position of the relay relative to the sources δ_{sr} which impacts the average SNR. We assume that the relay-destination link is perfect. It can be observed that the rank deficient probability gets lower as the position of the relay is closer to the sources. Moreover, we can also observe from the slopes of the curves that the rank deficient probability has an equivalent diversity gain of order less than one.

Let k be the diversity order related to the rank deficient probability of the

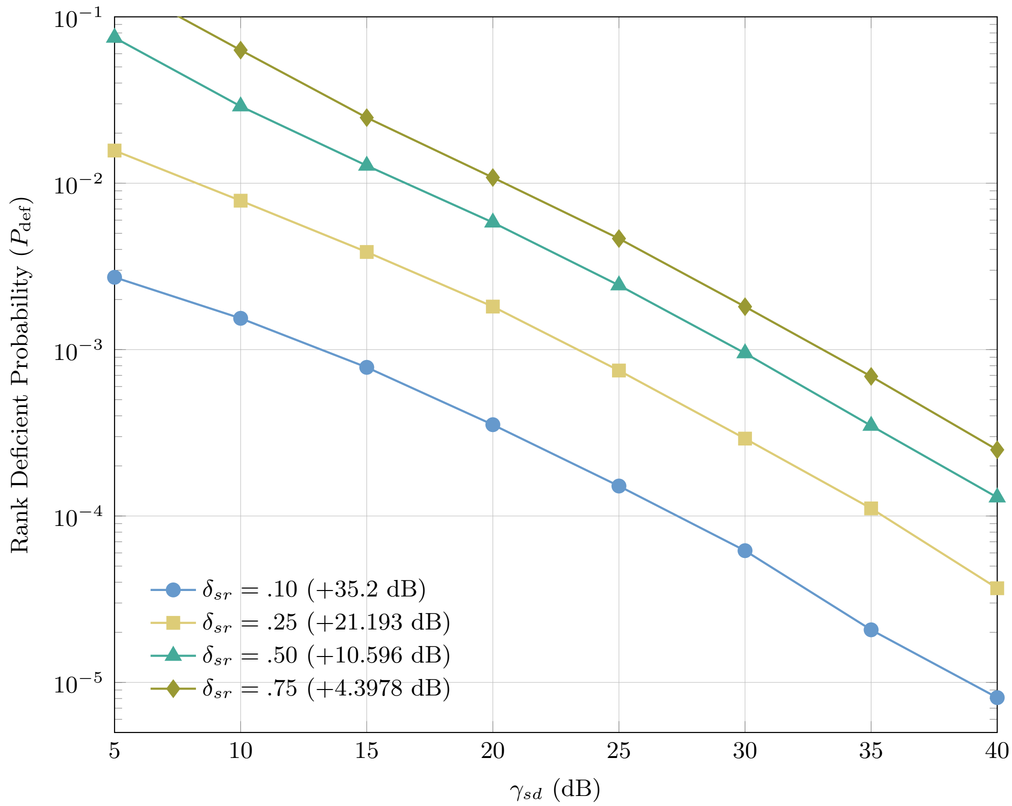


Fig. 3.4. Probability of rank deficient coefficient matrix (P_{def}) of MARC with two sources.

coefficient matrix. From the above numerical results, we conjecture that $k < 1$. The outage probability of the lim-FB strategy now can be written as

$$\begin{aligned} P_{\text{out}} &\leq \Pr\left(R_{\text{cp},d}^{(M-1)} < R\right) + \Pr\left(R_{\text{cp},d}^{(M)} < R\right) \Pr\left(R_{\text{cp},r}^{(1)} < R\right) \\ &\quad + \Pr\left(R_{\text{cp},d}^{(M)} < R\right) \Pr\left(R_{\text{rd}} < R\right) + \min\left\{ \Pr\left(R_{\text{cp},d}^{(M)} < R\right), P_{\text{def}}\right\} \\ &\overset{(a)}{\dot{\leq}} \frac{\xi_1}{\gamma_{sd}^2} + \frac{\xi_2}{\gamma_{sd}}\frac{\xi_3}{\gamma_{sd}^M} + \frac{\xi_2}{\gamma_{sd}}\frac{\xi_4}{\gamma_{sd}} + \frac{\xi_2}{\gamma_{sd}} \qquad (3.48) \\ &\dot{\leq} \frac{\xi}{\gamma_{sd}}, \qquad (3.49) \end{aligned}$$

where ξ, ξ_1, ξ_2, ξ_3, and ξ_4 are positive constants. In (a), because $k < 1$, in the high SNR regime, $\min\{\Pr(R_{\text{cp},d}^{(M)} < R), P_{\text{def}}\} = \Pr(R_{\text{cp},d}^{(M)} < R)$. From the above result, we can see that the lim-FB cannot achieve the full-diversity gain of the MARC. However, the bound in (3.49) is loose because of the Fréchet bound. As we will see in the next section, the lim-FB nearly achieves diversity gain of order two and its outage performance is significantly better compared to the existing strategies.

3.5.2 Sufficient Feedback Strategy

The outage performance of the sufficient feedback (suf-FB) strategy is quite similar to the lim-FB strategy. In the suf-FB strategy, there are also two possible ways for the destination to decode the transmitted messages. The first one is the direct decoding, which is exactly the same as that of the lim-FB. Therefore, the outage event of the direct decoding in the suf-FB is also given by

$$e_1 = \left\{ R_{\mathrm{cp},d}^{(M)} < R \right\}. \tag{3.50}$$

The second one is the cooperative decoding, where the relay select its best linear combination $\mathbf{a}_{r_*}$ that is linearly independent of the first $M-1$ linear combinations of the destination. Therefore, the resulting coefficient matrix is always full rank. As a result, the outage event of the cooperative decoding depends only on the sources-destination, the sources-relay, and the relay-destination links. Let $R_{\mathrm{cp},r}^{(*)} \triangleq R_{\mathrm{cp}}(\mathbf{a}_{r_*}, \mathbf{h}_{sr}, g_{sr})$. The outage probability of the suf-FB strategy is given by

$$\begin{aligned} e_2 &\triangleq \Big\{ \min_{m \in \{1,\ldots,M-1\}} R_{\mathrm{cp},d}^{(m)} < R \Big\} \cup \Big\{ R_{\mathrm{cp},r}^{(*)} < R \Big\} \cup \Big\{ R_{\mathrm{rd}} < R \Big\} \\ &= \Big\{ R_{\mathrm{cp},d}^{(M-1)} < R \Big\} \cup \Big\{ R_{\mathrm{cp},r}^{(*)} < R \Big\} \cup \Big\{ R_{\mathrm{rd}} < R \Big\} \end{aligned} \tag{3.51}$$

Similar to the lim-FB strategy, the overall outage for the suf-FB strategy occurs if and only if the direct and the cooperative decodings fail. Therefore, the outage probability of the suf-FB is given by

$$\begin{aligned} P_{\mathrm{out}} &\triangleq \Pr(e_1 \cap e_2) \qquad (3.52) \\ &= \Pr\Big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \big(\{R_{\mathrm{cp},d}^{(M-1)} < R\} \cup \{R_{\mathrm{cp},r}^{(*)} < R\} \cup \{R_{\mathrm{rd}} < R\} \Big) \\ &= \Pr\Big(\{R_{\mathrm{cp},d}^{(M-1)} < R\} \cup \big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \{R_{\mathrm{cp},r}^{(*)} < R\} \big) \\ &\quad \cup \big(\{R_{\mathrm{cp},d}^{(M)} < R\} \cap \{R_{\mathrm{rd}} < R\} \big) \\ &\overset{(a)}{\leq} \Pr\big(R_{\mathrm{cp},d}^{(M-1)} < R \big) + \Pr\big(R_{\mathrm{cp},d}^{(M)} < R \big) \Pr\big(R_{\mathrm{cp},r}^{(*)} < R \big) \\ &\quad + \Pr\big(R_{\mathrm{cp},d}^{(M)} < R \big) \Pr\big(R_{\mathrm{rd}} < R \big) \\ &\overset{(b)}{\leq} \Pr\big(R_{\mathrm{cp},d}^{(M-1)} < R \big) + \Pr\big(R_{\mathrm{cp},d}^{(M)} < R \big) \Pr\big(R_{\mathrm{cp},r}^{(M)} < R \big) \end{aligned}$$

$$+ \Pr\left(R_{\mathrm{cp},d}^{(M)} < R\right) \Pr\left(R_{\mathrm{rd}} < R\right) \tag{3.53}$$

where (a) is due to union bound and in (b) we bound P_{out} by selecting the worst linear combinations at the relay.

Using the results shown in the previous subsection, we can see that the suf-FB strategy can achieve second-order diversity. Specifically, P_{out} can be written as

$$\begin{aligned} P_{\mathrm{out}} \leq \Pr\left(R_{\mathrm{cp},d}^{(M-1)} < R\right) &+ \Pr\left(R_{\mathrm{cp},d}^{(M)} < R\right) \Pr\left(R_{\mathrm{cp},r}^{(M)} < R\right) \\ &+ \Pr\left(R_{\mathrm{cp},d}^{(M)} < R\right) \Pr\left(R_{\mathrm{rd}} < R\right) && (3.54) \\ &\dot{\leq} \frac{\xi_1}{\gamma_{sd}^2} + \frac{\xi_2}{\gamma_{sd}} \frac{\xi_3}{\gamma_{sd}} + \frac{\xi_2}{\gamma_{sd}} \frac{\xi_4}{\gamma_{sd}} && (3.55) \\ &\dot{\leq} \frac{\xi}{\gamma_{sd}^2}, && (3.56) \end{aligned}$$

with other positive constants ξ, ξ_1, ξ_2, ξ_3, and ξ_4.

3.6 Numerical Evaluation

In this section, we provide results of computer simulations performed to evaluate the performance of the proposed cooperation strategies compared to the approaches found in the literature. Since we focus on the design of cooperation strategies for applying compute-and-forward to the MARC, we mainly compare our proposed strategies to the approaches proposed by Soussi *et al.* [20] and Insausti *et al.* [22]. To the best of our knowledge, [20] and [22] are the only works available in the literature that addressed the problem of applying compute-and-forward to MARC.

Before going into the details, let us first briefly describe the approaches proposed in [20] and [22]. In [20], Soussi *et al.* proposed a global optimization for choosing linear combinations at the relay and the destination. Given channel state information (CSI) is known to all terminals, the relay and the destination select their optimal linear combinations maximizing the achievable rate. The relay then forwards its linear combination to the destination. Finally, having sufficient linear combinations, the destination attempts to recover the transmitted messages. We shall note that this approaches was proposed specifically for MARC with two sources. Extending it to MARC with any number of sources would require a significant effort. It has been shown that this approach achieves better achievable rate compared to other relaying strategies such as amplify-and-forward and decode-and-forward. However, it has a drawback that it requires huge communication overhead due to the fact that all CSI is known to all terminals. Moreover, it is easy to prove that this approach does not achieve full-diversity gain as it has a bottleneck in the link between the relay and the destination.

In [22], Insausti *et al.* proposed a different approach where the relay is allowed to choose its best linear combination yielding optimal computation rate and to forward it to the destination. The destination then chooses linear combinations that are linearly independent of the one from the relay and decodes the transmitted messages. If the decoding fails, the destination computes one more linear combination from its received signal and again decodes the transmitted messages. This approach is quite similar to our proposed strategy with limited feedback. Indeed, they achieve the same outage probability performance as we will see later. They differ in the way they utilize the transmission rounds. In the Insausti approach, two transmission rounds are always used. While in our proposed strategy, we try to use only one transmission round as much as possible to maintain the transmission efficiency.

Recall that we assume all the sources have the same distance to the destination and also to the relay. The distance from the sources to the destination is denoted by δ_{sd}, and to the relay is denoted by δ_{sr}. The relay has distance δ_{rd} to the destination. For simplicity, we normalize $\delta_{sd} = 1$, and assume $\delta_{sr} + \delta_{rd} = \delta_{sd}$. See the illustration in Fig. 3.1. The corresponding average SNRs are calculated with the assumption of path-loss exponent is equal to 3.52 [44,45]. In the simulations, we consider MARC with two sources and transmission rate $R = 2$. The performance is evaluated in three scenarios as follows.

1. *First scenario:* In this scenario, we evaluate the condition where the relay is closer to the sources than to the destination. Specifically, we set the distance from the sources to the relay $\delta_{sr} = 0.25$, while from the relay to the destination is $\delta_{rd} = 0.75$. This scenario is equivalent to the setting of average SNRs $\gamma_{sr} = \gamma_{sd} + 21.19$ dB and $\gamma_{rd} = \gamma_{sd} + 4.39$ dB.

2. *Second scenario:* In this scenario, we assume the distance from the sources to the relay is equal to the distance from the relay to the destination, i.e., $\delta_{sr} = \delta_{rd} = 0.5$. In other words, the relay is half-way between the sources

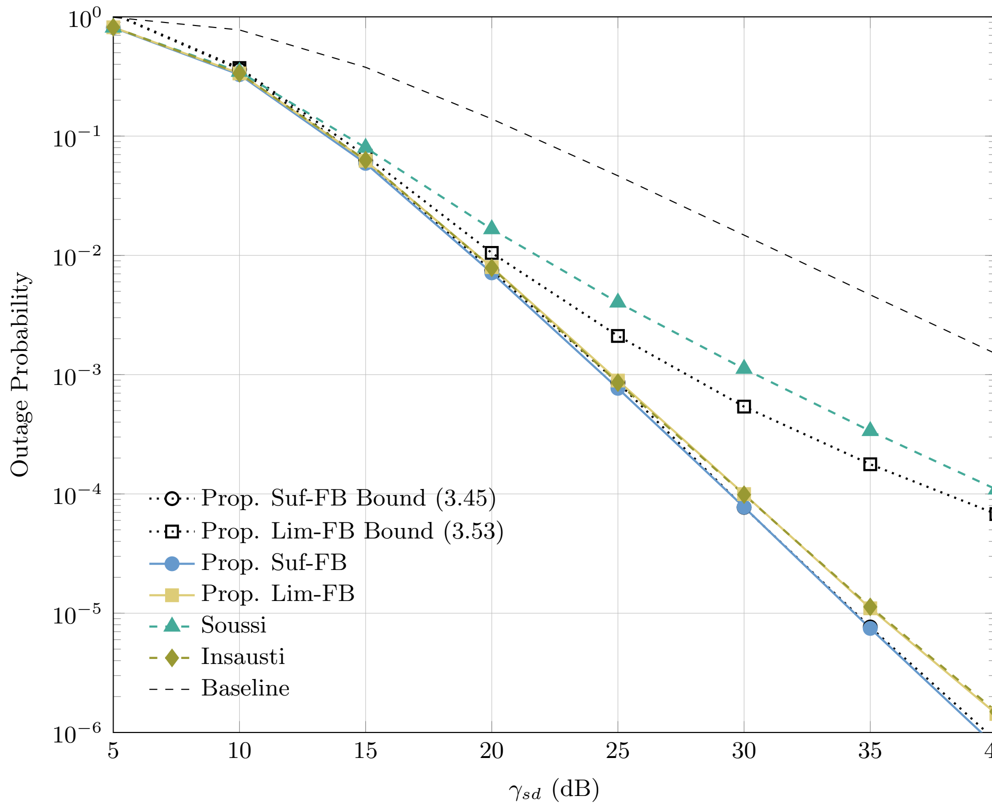


Fig. 3.5. Outage probabilities of the two-source MARC in the first scenario.

and the destination. With the same path-loss exponent, the resulting average SNRs are $\gamma_{sr} = \gamma_{rd} = \gamma_{sr} + 10.59$ dB.

3. *Third scenario:* This scenario is the opposite of the first one. It is assumed that the relay is positioned closer to the destination than to the relay. In particular, we assume $\delta_{sr} = 0.25$ and $\delta_{rd} = 0.75$. As a result, the average $\gamma_{sr} = \gamma_{sd} + 4.39$ dB and $\gamma_{rd} = \gamma_{sd} + 21.19$ dB.

The outage probability results for the first, second, and third scenarios are presented in Figs. 3.5, 3.6, and 3.7, respectively. Additionally, we present the baseline outage probability for the case when the sources send their information to the destination without a relay so that we can see how much improvement is gained when the relay is employed. Furthermore, outage probability bounds (3.45) and (3.53) are also presented. It can be observed that the outage probability bound of the lim-FB is quite loose which is a result of using the Fréchet bound in the derivation. From here on we refer to the strategy proposed by Soussi *et al.* [20] as

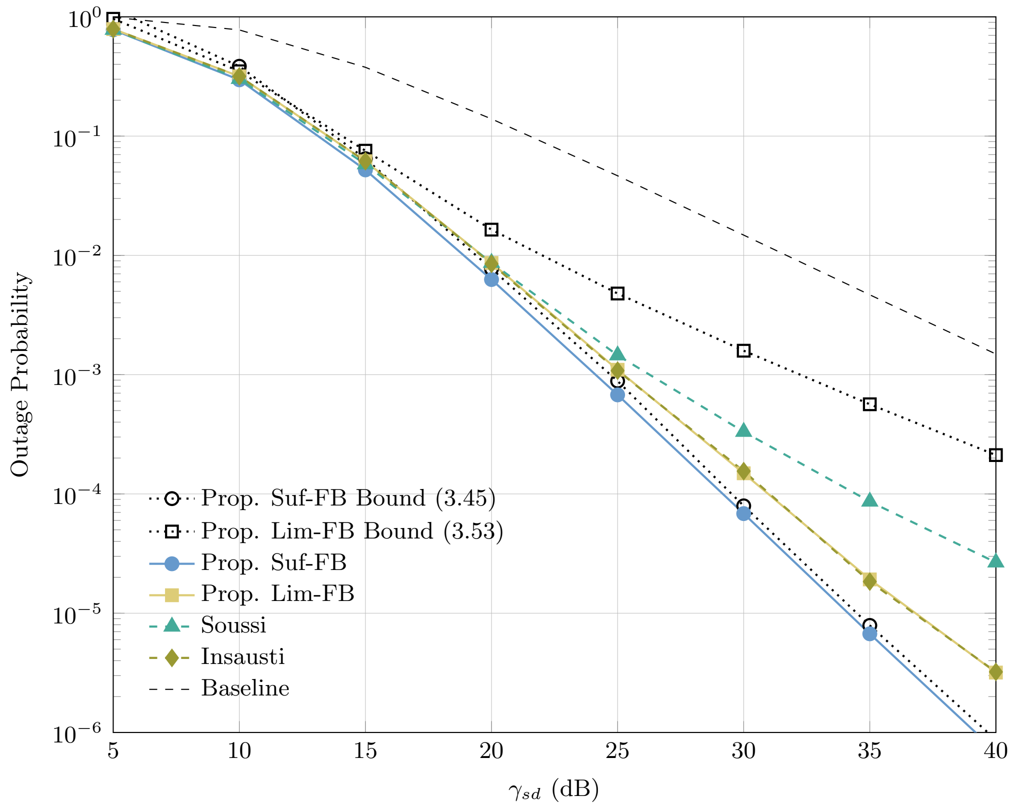


Fig. 3.6. Outage probabilities of the two-source MARC in the second scenario.

the *Soussi strategy*, and the strategy proposed by Insausti *et al.* [22] as the *Insausti strategy*.

From Figs. 3.5 and 3.6, we can clearly see that the Soussi strategy exhibits the highest outage performance. This is because even though the linear combinations selected by the relay and the destination are globally optimized, the destination can correctly recover the transmitted message if and only if it correctly decodes its own linear combination and the one from the relay. Therefore, if there is an outage in either sources-relay link, sources-destination link, or relay-destination link, the final decoding at the destination will fail. This means that the relay does not act as a helper, rather, its presence is mandatory. Even though compute-and-forward at the destination and the relay may achieve good outage performance, the point-to-point communication from the relay to the destination can only achieve the first-order diversity gain. And hence, the Soussi strategy suffers from a bottleneck performance at the relay-destination link. This fact can be seen from the three scenarios where the outage performance of the Soussi strategy gets better as the

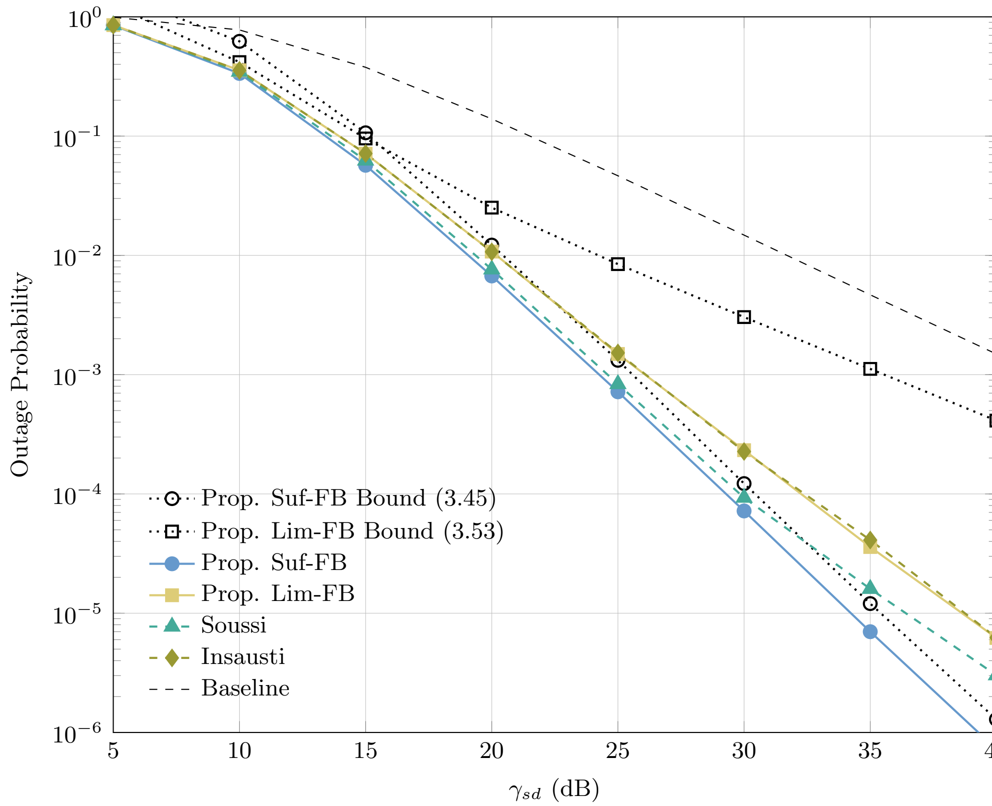


Fig. 3.7. Outage probabilities of the two-source MARC in the third scenario.

distance of the relay to the destination gets closer, i.e., γ_{rd} gets larger. Moreover, it can be seen from the slopes of the outage probability curves that this strategy can only achieve the first-order diversity.

In Figs. 3.5 and 3.6, it is shown that a significant outage performance improvement over the Soussi strategy is achieved by the Insausti strategy in the first and the second scenarios. For the third scenario, even though at low SNR regime the Soussi strategy has lower outage probability, it can be predicted that eventually the Insausti strategy is better in high SNR regime as the slope of its outage probability curve is steeper. This improvement is a result of giving the destination two possible ways of decoding the transmitting messages. The first is with the help of the relay, and the second is by using linear combinations decoded by itself. Thus, it can be thought that the relay acts as a helper where its existence is not mandatory, i.e., it is possible for the destination to decode the transmitted messages without the relay. It can also be observed that our proposed lim-FB strategy achieves the same outage performance as the Insausti strategy. This is because they are quite similar in the sense that the destination has two possible ways for decoding the transmitted messages and treat the relay as a useful helper. If we carefully observe the slopes of the outage performance of the lim-FB and the Insausti strategies, they do not achieve second-order diversity gain. The main reason behind this is that the local best linear combination selected by the relay may not be linearly independent of the $M-1$ best linearly combinations of the destination. We also observe that the performance of the Insausti and the lim-FB strategies degrades as the relay gets closer to the destination or as the average SNR from the sources to the relay gets smaller. This is related to the probability of the rank deficient coefficient matrix. As we have seen in Fig. 3.4, the smaller the difference between γ_{sd} and γ_{sr}, the higher the probability of rank deficient coefficient matrix. Hence, for the lim-FB and the Insausti strategies, it is better to place the relay closer to the sources.

The best outage performance is achieved by the suf-FB strategy. Based on the slopes of the curves shown in Figs. 3.5, 3.6, and 3.7, one can see that the suf-FB strategy achieves the second-order diversity gain. This agrees with our analysis in Subsection 3.5.2. The main reason for this is that the destination has two possible ways in decoding the transmitted messages, the direct and the cooperative decoding. Moreover, unlike in the lim-FB, the resulting coefficient matrices in the suf-FB are guaranteed to always be full-rank.

Next, we evaluate network throughput performance which is defined as the ratio of the correctly received messages to number of transmission rounds utilized. For the proposed strategies, because the second round of transmission is utilized only when the direct decoding fails, the network throughput is defined as

$$T_{\text{prop}} = \frac{M(1-P_{\text{out}})}{1+P_{\text{out}}^{\text{dir}}}, \tag{3.57}$$

where $P_{\text{out}}^{\text{dir}} \triangleq \Pr\{R_{\text{cp},d}^{(M)} < R\}$ is the outage probability of the direct decoding. On the other hand, because the Soussi and the Insausti strategies always use two transmission round, their network throughput is given by

$$T_{\text{exist}} = \frac{M(1-P_{\text{out}})}{2}. \tag{3.58}$$

We found that in terms of network throughput, the performance of each strat-

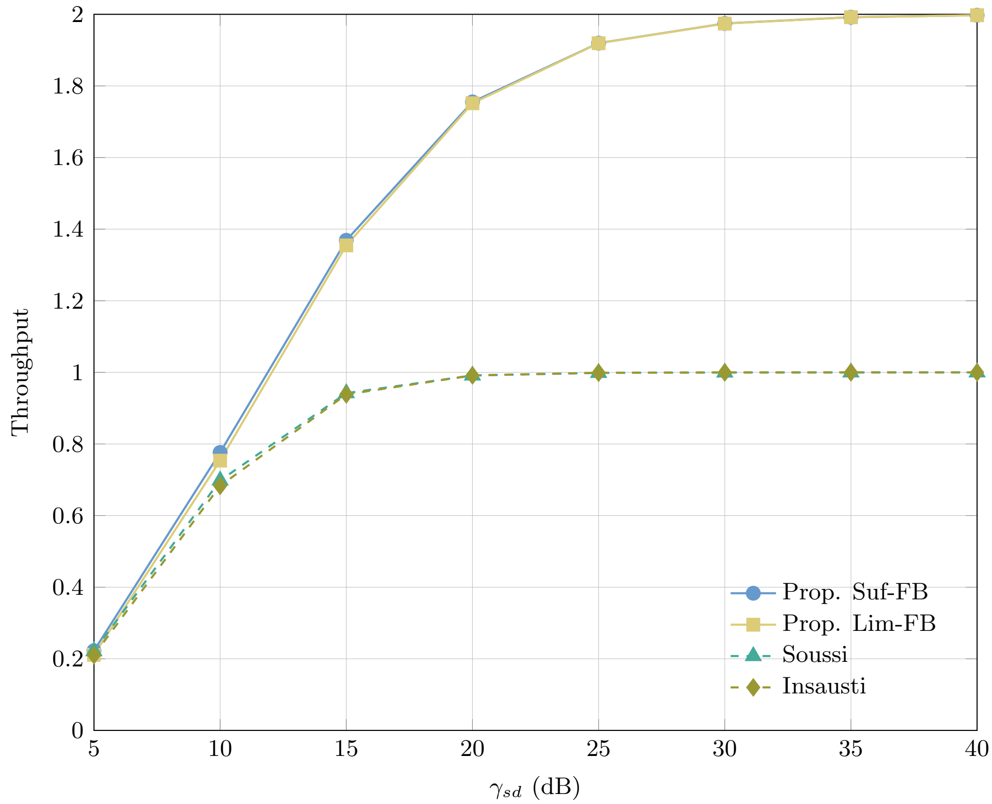


Fig. 3.8. Network throughput of the two-source MARC in the second scenario.

egy in all scenarios is similar. Therefore, it is sufficient to only present the network throughput performance of one of the three scenarios; Fig. 3.8 presents the network throughput of the second scenario. It can be clearly seen that both the proposed strategies nearly achieve the maximum throughput of two messages per transmission, while the existing strategies can only approach maximum of one message per transmission. This is because the proposed strategies requires less than two transmissions on average to deliver two messages. In fact, in the high SNR regime, close to one transmission is required on average. On the other hand, the existing strategies always utilize two transmission rounds to deliver two messages. Therefore, the maximum network throughput they can achieve is one message per transmission rounds. Thus, it can be concluded that the proposed strategies have higher transmission efficiency compared to the existing strategies.

We shall note that the proposed strategies, in particular the suf-FB strategy, indeed need additional communication overhead for achieving the aforementioned advantages. However, the additional overhead size is not as big as the Soussi strategy. Moreover, the feedback is only sent when the destination fails to decode the transmitted messages by itself. Hence, in the higher SNR regime, only a small amount of feedback is required.

3.7 Summary

In this chapter, we have studied the application of compute-and-forward to multiple-access relay channels (MARC). We proposed two cooperation strategies between the relay and the destination. The proposed strategies are opportunistic in the sense that they use transmission rounds as few as possible to increase the transmission efficiency while improving outage probability performance of MARC. We have shown that both of the proposed strategies improves network throughput remarkably, twice of that of the existing strategies [20, 22]. It is shown that the first strategy called the lim-FB strategy achieves diversity gain close to the second-order, which is a significant improvement over [20]. A better outage probability enhancement is achieved the second strategy, namely the suf-FB strategy, where the full-diversity gain of the MARC is achieved.

Chapter 4

Compute-and-Forward in Precoded MIMO Systems

Different from Chapter 3, in this chapter we study another type of wireless communication scenarios that adopts compute-and-forward methods. In particular, we study precoded multiple-input multiple-output (MIMO) systems that employ integer-forcing linear receivers [26]. Note that integer-forcing in principle is another form compute-and-forward methods.

4.1 Introduction

4.1.1 Background and Related Work

MIMO systems where multiple antennas are used at both transmitter and receiver in a wireless communication system has been regarded as one key technology to cope with major wireless network challenges. As we are aware that wireless networks are now facing unprecedented challenges as the number of wirelessly-connected devices such as smartphones, tablets, computers, and sensors is dramatically increasing. Furthermore, the emergence of abundant software applications demanding high quality media, e.g., images and videos, results in the tremendous increase of the global network traffic. This situation leads to the demands of massive wireless network access and high data transmission rate. The scarcity of the available spectrum frequency makes these challenges more difficult to overcome. In this situation MIMO strategy comes in handy because by exploiting multi-path scattering, MIMO offers significant improvement in terms of transmission reliability (diversity gain) and data transmission rate.

To realize the advantages of MIMO, it is important to design an optimal or near-optimal receiver. A maximum likelihood (ML) receiver has optimal rates and probability of error [49]. However, its complexity increases exponentially with respect to the number of antennas. As alternatives, zero-forcing (ZF) or minimum mean square error (MMSE) receivers are often employed [27]. These receivers apply a linear transformation such that the MIMO channel can be seen as a sequence of single-input single-output (SISO) channels, and hence, the decoding complexity is greatly reduced. However, this advantage comes with the cost of a performance loss which can be significant especially in the low signal-to-noise power ratio (SNR) regime. Inspired by compute-and-forward, Zhan *et al.* proposed a MIMO linear receiver called integer-forcing (IF) receiver [26] which achieves significantly better error performance than ZF and MMSE receivers with nearly the same decoding complexity for slow-fading channels. Similar to the compute-and-forward case, the transmitter in the IF receiver framework employs nested lattice codes and the receiver approximates the channels with a "good" full rank *integer* matrix $\mathbf{A}$. Since an integer linear combination of lattice codewords is again a codeword, the receiver can use SISO decoding to decode each linear combination, and subsequently recover the transmitted messages by solving a simple linear equation system. It has been shown that IF receivers achieve the optimal diversity-multiplexing tradeoff (DMT) [28, 29] and yield numerical error performance that is quite close to that of the optimal ML receiver [26, 54].

While the advantages of MIMO can be achieved when the channel state information (CSI) is only available at the receiver, these can be further enhanced when the transmitter has some level of knowledge of CSI. The transmitter exploits CSI for encoding information symbols prior to transmissions to increase the reliability against the channel fluctuations; this technique is known as *precoding* [56]. Many precoding schemes are designed for MIMO with quadrature amplitude modulations (QAM) and ML receivers. For instance, Vrigneau *et al.* [57] proposed a specific precoding scheme for 4-QAM MIMO systems with ML receivers. This precoding is optimal and has been shown to outperform all MMSE receiver-based precodings. However, despite its optimality, it is hard to further extend the idea to higher-order QAM because of its high complexity. In [30], Mohammed *et al.*

proposed precoding schemes for more general QAM with ML receivers, namely X- and Y-precoders. These precoding schemes can achieve error performance close to that of [57] and can be easily employed for an arbitrary MIMO configuration. However, when full transmission rate is used, X- and Y-precoders cannot achieve full diversity gain. Moreover, since they are designed based on the minimum distance of the received QAM constellations, the error performance degrades as the constellation size increases.

In this chapter we focus on precoding schemes for MIMO with integer-forcing receivers (IF-MIMO). The performance of this kind of precoding is not dictated by the minimum distance of received constellations, and hence, it can excel in high-order modulation schemes. In [3], Sakzad and Viterbo proposed *unitary precoded integer-forcing* (UPIF), a precoding scheme designed for IF-MIMO where the precoder matrices are from groups of unitary matrices. They showed that UPIF achieves full diversity gain while allowing full rate transmission. Two types of UPIF were introduced. The first type of precoder (UPIF I) is designed for each channel realization based on the *minimum distance* of a lattice generated by the precoder matrix. The second type of precoder (UPIF II) is designed for all channel realizations based on the *minimum product distance* [58] of the generated lattice. We are particularly interested in UPIF I where the precoder matrix adapts to each channel realization. Finding the optimal precoder matrix of UPIF I is a hard problem due to the involvement of the unitary constraint [59] and the lattice minimum distance problem [46, 47, 60]. For 2×2 MIMO systems, a simple parameterization technique finds the optimal UPIF I precoder matrix [3]. But for higher-order MIMO, this technique is computationally expensive because an exhaustive search over multiple parameters is required. Our work here addresses this problem and proposes an efficient algorithm for finding good orthogonal precoder matrices that are applicable to any MIMO dimension.

4.1.2 Summary of Contributions

The summary and contributions of our work in this chapter are as follows.

1. In [3] it is shown that the search space for optimal UPIF I precoder matrices is groups of unitary matrices. However, we argue that it is sufficient and even superior to only search over groups of orthogonal matrices.[1] Uni-

[1] Groups of orthogonal matrices are sub-groups of groups of unitary matrices.

tary precoder matrices do not guarantee better achievable rate and outage probability than orthogonal precoder matrices; this is shown using Propositions 4.3.1 and 4.4.1. Via numerical evaluations we confirm that indeed the orthogonal precoder outperforms its unitary counterpart in terms of achievable rate, outage probability, and error rate. Besides the performance advantage, the orthogonal precoder also has lower complexity as the dimension of orthogonal matrices is half that of unitary matrices in real-valued domain. In other words, we show that the orthogonal precoder is more favorable in terms of both performance and complexity compared to unitary precoder for UPIF I.

2. We propose an efficient algorithm for finding good orthogonal precoder matrices. This algorithm is based on the steepest gradient algorithm and exploits the geometrical properties of orthogonal matrices as a Lie group [2, 59, 61]. The main difficulty of the optimization problem comes from the simultaneous inclusions of (i) an orthogonality constraint and (ii) the lattice minimum distance problem. Without the minimum distance problem, we could immediately use existing steepest gradient algorithms. However, the inclusion of (ii) makes the optimization problem non-differentiable and much harder. Our approach is to divide the problem into two sub-problems, and develop algorithms based on steepest gradient and random search algorithms to solve them. Discussion of the proposed algorithm is presented in Section 4.5. Compared to the parameterization technique [3, 62], the proposed algorithm has lower complexity — the proposed algorithm has polynomial complexity of $\mathcal{O}(M^4 \log M)$, while the parameterization technique has exponential $\mathcal{O}(\nu^{M(M-1)/2)} M^4 \log M)$, where M is the number of antennas and ν is a constant, cf. Section 4.6.

3. We present and analyze the results of computer simulations comparing the proposed schemes with existing schemes. The numerical results show that:

   - Orthogonal precoder matrices are superior to unitary precoder matrices for integer-forcing MIMO.
   - Despite its lower complexity, the proposed steepest gradient-based algorithm achieves performance identical to the parameterization technique.

- Even though X-precoders are designed specifically for QAM, our proposed schemes are remarkably better (in terms probability of error) in high-order QAM schemes, e.g., 64- and 256-QAM.
- The proposed schemes outperform UPIF II in some scenarios, e.g., 4×4 MIMO.

4.2 Integer-Forcing MIMO with Orthogonal Precoder

Without loss of generality, we consider a point-to-point MIMO system where each transmission end is equipped with M antennas, i.e., an $M \times M$ MIMO system. The channels are assumed to be quasi-static flat-fading, remaining constant over one coherence interval. CSI is known to both transmitter and receiver. Denoted by $\mathbf{H} \in \mathbb{C}^{M \times M}$, the channel matrix is decomposed to $\mathbf{H} = \mathbf{W}\mathbf{D}\mathbf{V}^H$ using the singular value decomposition (SVD). $\mathbf{W}, \mathbf{V} \in \mathbb{C}^{M \times M}$ are unitary matrices, i.e., $\mathbf{W}\mathbf{W}^H = \mathbf{V}\mathbf{V}^H = \mathbf{I}$, and $\mathbf{D} \triangleq \mathrm{diag}(d_1, d_2, ..., d_M) \in \mathbb{R}^{M \times M}$ is a diagonal matrix with $d_1 \geq d_2 \geq \cdots \geq d_M$.

Let $\mathcal{C}$ be a codebook of a nested lattice $\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}} \subset \mathbb{C}^n$ with coding rate R. Let $\mathbf{w}_m$, $m = 1, ..., M$, be information messages to be transmitted across MIMO channels. These messages are encoded to lattice codewords $\mathbf{x}_m \in \mathcal{C}$ using a bijective mapping $\mathcal{E}$, i.e., $\mathcal{E}(\mathbf{w}_m) = \mathbf{x}_m$. Each $\mathbf{x}_m$ satisfies $\frac{1}{n}\mathbb{E}||\mathbf{x}_m||^2 = \gamma$. Let $\mathbf{X} = [\mathbf{x}_1 \ \cdots \ \mathbf{x}_M]^T \in \mathbb{C}^{M \times n}$. Prior to transmissions, $\mathbf{X}$ is precoded such that $\mathbf{X}_{\mathrm{prec}} = \mathbf{V}\mathbf{P}\mathbf{X}$, where $\mathbf{P} \in \mathbb{R}^{M \times M}$ is an orthogonal matrix. We refer to the matrix $\mathbf{P}$ as the *precoder matrix*, which is subject to the optimization problem in this work. The received signal at the receiver is

$$\mathbf{Y} = \mathbf{H}\mathbf{X}_{\mathrm{prec}} + \mathbf{Z}. \tag{4.1}$$

The entries of $\mathbf{H}$ and $\mathbf{Z} \in \mathbb{C}^{N \times n}$ are i.i.d. complex Gaussian random variables $\sim \mathcal{CN}(0, 1)$. We assume that random dithering is employed to ensure that the $\mathbf{x}_m$ is uniformly distributed over the fundamental Voronoi region of Λ_{s}. However, for simplicity, we omit the dithering notations from the exposition. Upon receiving $\mathbf{Y}$, the receiver multiplies it by $\mathbf{W}^H$, and thus,

$$\tilde{\mathbf{Y}} = \mathbf{W}^H\mathbf{Y} = \mathbf{W}^H\mathbf{H}\mathbf{X}_{\mathrm{prec}} + \mathbf{W}^H\mathbf{Z} \tag{4.2}$$

$$= \mathbf{W}^H\mathbf{W}\mathbf{D}\mathbf{V}^H\mathbf{V}\mathbf{P}\mathbf{X} + \mathbf{W}^H\mathbf{Z} \tag{4.3}$$

$$= \mathbf{DPX} + \tilde{\mathbf{Z}}, \tag{4.4}$$

with $\tilde{\mathbf{Z}} = \mathbf{W}^H\mathbf{Z}$ whose entries still follow $\mathcal{CN}(0,1)$ because $\mathbf{W}$ is unitary.

The receiver employs an IF receiver [26] which transforms the resulting channel in (4.4) into M effective point-to-point sub-channels. Hence the receiver can decode the transmitted messages using a SISO decoding rather than joint decoding across all receive antennas. In principle, the IF receiver approximates the resulting MIMO channel $\mathbf{DP}$ with an invertible integer matrix[2] $\mathbf{A} \in \mathbb{Z}^{M\times M}$ by selecting an equalizing matrix $\mathbf{B} \in \mathbb{R}^{M\times M}$ and computes[3]

$$\mathbf{Y}_{\text{eff}} = [\mathbf{B}\tilde{\mathbf{Y}}] \bmod \Lambda_{\text{s}} \tag{4.5}$$

$$= [\mathbf{BDPX} + \mathbf{B}\tilde{\mathbf{Z}}] \bmod \Lambda_{\text{s}} \tag{4.6}$$

$$= [\mathbf{AX} + (\mathbf{BDP} - \mathbf{A})\mathbf{X} + \mathbf{B}\tilde{\mathbf{Z}}] \bmod \Lambda_{\text{s}}. \tag{4.7}$$

Let $\mathbf{y}_{\text{eff},m}^T$, $\mathbf{a}_m^T$, and $\mathbf{b}_m^T$ be the m-th rows of $\mathbf{Y}_{\text{eff}}$, $\mathbf{A}$, and $\mathbf{B}$, respectively. The effective received signal at sub-channel m can be written as

$$\mathbf{y}_{\text{eff},m}^T = [\mathbf{a}_m^T\mathbf{X} + (\mathbf{b}_m^T\mathbf{DP} - \mathbf{a}_m^T)\mathbf{X} + \mathbf{b}_m^T\tilde{\mathbf{Z}}] \bmod \Lambda_{\text{s}} \tag{4.8}$$

$$= [\mathbf{c}_m^T + \mathbf{z}_{\text{eff},m}^T] \bmod \Lambda_{\text{s}}, \tag{4.9}$$

where $\mathbf{c}_m^T = \mathbf{a}_m^T\mathbf{X} \bmod \Lambda_{\text{s}}$ is the desired linear combination, and

$$\mathbf{z}_{\text{eff},m}^T = [(\mathbf{b}_m^T\mathbf{DP} - \mathbf{a}_m^T)\mathbf{X} + \mathbf{b}_m^T\tilde{\mathbf{Z}}] \bmod \Lambda_{\text{s}} \tag{4.10}$$

is the effective noise at sub-channel m.

Owing to the linearity property of $\mathcal{C}$, the linear combination $\mathbf{c}_m$ happens to be a codeword, and thus, the next step of the IF receiver is to decode $\mathbf{c}_m$ from the effective point-to-point sub-channel in (4.9). Let $\hat{\mathbf{c}}_m$ be the estimate of $\mathbf{c}_m$. $\hat{\mathbf{c}}_m$ is obtained using $\hat{\mathbf{c}}_m = Q_{\Lambda_{\text{c}}}(\mathbf{y}_{\text{eff},m})$, where $Q_{\Lambda_{\text{c}}}(\cdot)$ is the decoding or quantization function with respect to Λ_{c}. Let $\hat{\mathbf{C}} = [\hat{\mathbf{c}}_1, ..., \hat{\mathbf{c}}_M]^T$, and $\hat{\mathbf{X}}$ and $\hat{\mathbf{w}}_m$ be the estimates of $\mathbf{X}$ and $\mathbf{w}_m$, respectively. The transmitted symbols are obtained by solving $\hat{\mathbf{X}} = \mathbf{A}^{-1}\hat{\mathbf{C}}$, and finally the information messages are recovered using $\hat{\mathbf{w}}_m = \mathcal{E}^{-1}(\hat{\mathbf{x}}_m)$.

[2]Note that if $\mathbf{P}$ is a unitary matrix (complex-valued), then $\mathbf{A} \in \mathbb{Z}[i]^{M\times M}$ and $\mathbf{B} \in \mathbb{C}^{M\times M}$.

[3]mod Λ_{s} is modulo operation on each row of the corresponding matrix with respect to the shaping lattice Λ_{s}.

4.3 Performance Metrics

Consider the performance of this MIMO system. First, define the variance of $\mathbf{z}_{\text{eff},m}$ as

$$\begin{aligned}\sigma^2_{\text{eff},m} &\triangleq \frac{1}{n}\mathbb{E}\left\|(\mathbf{b}_m^T\mathbf{D}\mathbf{P}-\mathbf{a}_m^T)\mathbf{X}+\mathbf{b}_m^T\tilde{\mathbf{Z}}\right\|^2 \\ &= \gamma\left\|\mathbf{b}_m^T\mathbf{D}\mathbf{P}-\mathbf{a}_m^T\right\|^2+\left\|\mathbf{b}_m^T\right\|^2. \end{aligned} \tag{4.11}$$

To achieve a reliable communication system, $\mathbf{b}_m$ should be chosen such that the effective noise variance $\sigma^2_{\text{eff},m}$ is minimized. The optimal $\mathbf{b}_m$ is [3]

$$\mathbf{b}^T_{\text{opt},m} = \gamma\mathbf{a}_m^T(\mathbf{D}\mathbf{P})^T(\mathbf{I}+\gamma\mathbf{D}\mathbf{P}(\mathbf{D}\mathbf{P})^T)^{-1}. \tag{4.12}$$

Substituting $\mathbf{b}^T_{\text{opt},m}$ into (4.11) results in

$$\sigma^2_{\text{eff},m} = \gamma\mathbf{a}_m^T(\mathbf{I}+\gamma(\mathbf{D}\mathbf{P})^T\mathbf{D}\mathbf{P})^{-1}\mathbf{a}_m \tag{4.13}$$

$$= \gamma\mathbf{a}_m^T\mathbf{P}^T(\mathbf{I}+\gamma\mathbf{D}^T\mathbf{D})^{-1}\mathbf{P}\mathbf{a}_m. \tag{4.14}$$

Because $(\mathbf{I}+\gamma\mathbf{D}^T\mathbf{D})^{-1}$ is a positive definite matrix, it admits Cholesky decomposition

$$(\mathbf{I}+\gamma\mathbf{D}^T\mathbf{D})^{-1} = \mathbf{L}\mathbf{L}^T. \tag{4.15}$$

Now let

$$\mathbf{L}_{\text{P}} \triangleq \mathbf{P}^T\mathbf{L}. \tag{4.16}$$

Hence, $\sigma^2_{\text{eff},m}$ can be expressed as

$$\sigma^2_{\text{eff},m} = \gamma\mathbf{a}_m^T\mathbf{P}^T\mathbf{L}\mathbf{L}^T\mathbf{P}\mathbf{a}_m \tag{4.17}$$

$$= \gamma\left\|\mathbf{L}_{\text{P}}^T\mathbf{a}_m\right\|^2. \tag{4.18}$$

Define the effective SNR of the worst sub-channel, i.e., the channel with the highest effective noise variance, as

$$\texttt{SNR}_{\text{eff}} \triangleq \min_{m=1,\ldots,M}\frac{\frac{1}{n}\mathbb{E}\left\|\mathbf{c}_m\right\|^2}{\sigma^2_{\text{eff},m}} \tag{4.19}$$

$$= \min_{m=1,\ldots,M} \frac{1}{\left\|\mathbf{L}_{\mathrm{P}}^{T}\mathbf{a}_m\right\|^2}. \tag{4.20}$$

Note that because $\mathbf{c}_m$ is a codeword, $\frac{1}{n}\mathbb{E}\left\|\mathbf{c}_m\right\|^2 = \gamma$. Clearly, to recover the information messages, all $\mathbf{c}_m$'s must be decoded correctly. Therefore, the matrix $\mathbf{A}$ has to be chosen such that $\mathtt{SNR}_{\text{eff}}$ is maximized. Define the optimal matrix $\mathbf{A}$ as

$$\mathbf{A}_{\text{opt}} = \arg\max_{\substack{\mathbf{A}\in\mathbb{Z}^{M\times M}\\ \det(\mathbf{A})\neq 0}} \min_{m=1,\ldots,M} \frac{1}{\left\|\mathbf{L}_{\mathrm{P}}^{T}\mathbf{a}_m\right\|^2} \tag{4.21}$$

$$= \arg\min_{\substack{\mathbf{A}\in\mathbb{Z}^{M\times M}\\ \det(\mathbf{A})\neq 0}} \max_{m=1,\ldots,M} \left\|\mathbf{L}_{\mathrm{P}}^{T}\mathbf{a}_m\right\|^2. \tag{4.22}$$

If $\mathbf{A}_{\text{opt}}$ is employed, then we have the optimal $\mathtt{SNR}_{\text{eff}}$ as

$$\mathtt{SNR}_{\text{eff,opt}} = \frac{1}{\lambda_M^2(\mathbf{L}_{\mathrm{P}}^{T})}, \tag{4.23}$$

where $\lambda_M(\mathbf{L}_{\mathrm{P}}^{T})$ is the largest successive minimum of the lattice $\Lambda(\mathbf{L}_{\mathrm{P}}^{T})$, see the definition of successive minima given in (2.16). Finding $\mathbf{A}_{\text{opt}}$ is one of crucial problems in the IF framework. Because this problem is equivalent to finding successive minima of a lattice, we can conveniently employ the sphere decoding algorithms [46,49] or the LLL algorithms [48,63]. We can also use the recently proposed algorithms specifically for IF-MIMO [50,51,53,54,64,65].

Assume that a "good" nested lattice code $\mathcal{C}$ [14,16,17,26] is employed at the transmitter. In the IF receiver framework, the worst sub-channel constitutes a performance bottleneck. Therefore, if the rate of $\mathcal{C}$ satisfies

$$R < \log(\mathtt{SNR}_{\text{eff,opt}}), \tag{4.24}$$

then all sub-channels $m = 1, ..., M$ can decode their linear combination $\mathbf{c}_m$ with a low error probability. This implies that the achievable rate of this MIMO system is

$$R_{\text{IF}} = M\log(\mathtt{SNR}_{\text{eff,opt}}). \tag{4.25}$$

Let R_{t} be the target rate of the system. The outage probability of the system is defined as

$$P_{\text{out}} \triangleq \Pr(R_{\text{IF}} < R_{\text{t}}) = \Pr(M \log(\texttt{SNR}_{\text{eff,opt}}) < R_{\text{t}}) \tag{4.26}$$

$$= \Pr\left(\texttt{SNR}_{\text{eff,opt}} < 2^{R_{\text{t}}/M}\right). \tag{4.27}$$

From (4.25) and (4.27), we know that to improve the performance in terms of achievable rate and outage probability, $\texttt{SNR}_{\text{eff,opt}}$ should be maximized. This maximization is rather difficult because $\texttt{SNR}_{\text{eff,opt}}$ is a function of the largest successive minimum of a lattice. However, we can bound $\texttt{SNR}_{\text{eff,opt}}$ with the minimum distance of its dual lattice, which makes the optimization easier. For this purpose, we use the following proposition.

Proposition 4.3.1: Consider the aforementioned IF-MIMO system with an orthogonal precoder matrix $\mathbf{P}$. The effective SNR of the worst sub-channel is lower bounded by

$$\texttt{SNR}_{\text{eff,opt}} \geq \frac{\lambda_1^2(\mathbf{L}_{\text{P}}^{-1})}{M^2}, \tag{4.28}$$

where $\mathbf{L}_{\text{P}}$ is defined in (4.16) and $\lambda_1(\mathbf{L}_{\text{P}}^{-1})$ is the minimum distance of lattice $\Lambda(\mathbf{L}_{\text{P}}^{-1})$, which is the dual lattice of $\Lambda(\mathbf{L}_{\text{P}}^{T})$.

Proof. The proof follows the one given in [28]. Let $\Lambda(\mathbf{G})$ be a real-valued lattice generated by a full rank matrix $\mathbf{G} \in \mathbb{R}^{M\times M}$ and let $\Lambda(\mathbf{G}^{-T})$ be its dual lattice. In [66] Banaszczyk proved that the successive minima of $\Lambda(\mathbf{G})$ and $\Lambda(\mathbf{G}^{-T})$ have the following relationship

$$\lambda_m(\mathbf{G})\lambda_{M-m+1}(\mathbf{G}^{-T}) \leq M, \tag{4.29}$$

for $1 \leq m \leq M$.

From (4.23), we have $\texttt{SNR}_{\text{eff,opt}} = 1/\lambda_M^2(\mathbf{L}_{\text{P}}^{T})$. The dual lattice of $\Lambda(\mathbf{L}_{\text{P}}^{T})$ is $\Lambda(\mathbf{L}_{\text{P}}^{-1})$, see Definition 2.1.7. And thus, by (4.29), it follows that

$$\texttt{SNR}_{\text{eff,opt}} \geq \frac{\lambda_1^2(\mathbf{L}_{\text{P}}^{-1})}{M^2}, \tag{4.30}$$

which is the desired result. ∎

Using Proposition 4.3.1, we now can bound the achievable rate of the system as

$$
\begin{aligned}
R_{\mathrm{IF}} &= M \log(\mathtt{SNR}_{\mathrm{eff,opt}}) && (4.31)\\
&\geq M \log\Big(\frac{\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})}{M^2}\Big) && (4.32)\\
&= 2M\Big(\log(\lambda_1(\mathbf{L}_{\mathrm{P}}^{-1})) - \log(M)\Big), && (4.33)
\end{aligned}
$$

and the outage probability as

$$
\begin{aligned}
P_{\mathrm{out}} &= \Pr\left(\mathtt{SNR}_{\mathrm{eff,opt}} < 2^{R_{\mathrm{t}}/M}\right) && (4.34)\\
&\leq \Pr\left(\frac{\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})}{M^2} < 2^{R_{\mathrm{t}}/M}\right) && (4.35)\\
&= \Pr\left(\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1}) < M^2 2^{R_{\mathrm{t}}/M}\right). && (4.36)
\end{aligned}
$$

Define the error probability of the system as

$$
P_{\mathrm{e}} = \Pr\big((\hat{\mathbf{w}}_1, ..., \hat{\mathbf{w}}_M) \neq (\mathbf{w}_1, ..., \mathbf{w}_M)\big). \tag{4.37}
$$

This error probability is dependent of the nested lattice code $\mathcal{C}$ employed by the system. From a practical point of view we may consider 2^{2q}-QAM constellations for a positive integer q, e.g., 4-QAM, 16-QAM, and 64-QAM. These constellations are equivalent to the nested lattice code $\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}}$ with $\Lambda_{\mathrm{c}} = \alpha\mathbb{Z}[i]$ and $\Lambda_{\mathrm{s}} = 2^q\Lambda_{\mathrm{c}}$, where α is a positive real number. Employing this code, the error probability of the system is given by the following proposition.

Proposition 4.3.2: If nested lattice code $\Lambda_{\mathrm{c}}/\Lambda_{\mathrm{s}}$, with $\Lambda_{\mathrm{c}} = \alpha\mathbb{Z}[i]$ and $\Lambda_{\mathrm{s}} = 2^q\Lambda_{\mathrm{c}}$, where $1 < q \in \mathbb{Z}$ and $\alpha = \sqrt{6\gamma}/2^{2q}$, is employed in an $M \times M$ IF-MIMO system, the error probability is bounded as

$$
P_{\mathrm{e}} \leq 4M \exp\Big(-\frac{3\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})}{2^{4q+1}M^2}\Big), \tag{4.38}
$$

where $\mathbf{L}_{\mathrm{P}}$ is defined in (4.16).

Proof. See Appendix A. ∎

4.4 Problem Statement

The performance metrics derived in (4.33), (4.36), and (4.38) suggest that to achieve a good performance in terms of achievable rate, outage probability, and error probability, we should choose precoder matrix $\mathbf{P}$ such that $\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})$ is maximized. Formally, we define the problem of finding the optimal $\mathbf{P}$ as

$$\mathbf{P}_{\mathrm{opt}} = \arg\max_{\mathbf{P}\in O(M)} \lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1}) \tag{4.39}$$

$$= \arg\max_{\mathbf{P}\in O(M)} \min_{\mathbf{v}\in\mathbb{Z}^M\setminus\mathbf{0}} \left\|\mathbf{L}^{-1}\mathbf{P}\mathbf{v}\right\|^2 . \tag{4.40}$$

In other words, we have to find an orthogonal matrix $\mathbf{P}$ such that the minimum distance of lattice $\Lambda(\mathbf{L}^{-1}\mathbf{P})$ is maximized.

Based on (4.40), one may argue that unitary precoder matrices can yield a larger $\lambda_1(\mathbf{L}_{\mathrm{P}}^{-1})$ than the orthogonal one. Indeed, that is the case. But, recall that we derive the bounds on performance metrics in (4.33), (4.36), and (4.38) in order to ease the optimization process. The performance of the system is more directly affected by $\mathtt{SNR}_{\mathrm{eff,opt}}$ or $\lambda_M(\mathbf{L}_{\mathrm{P}}^T)$ rather than by $\lambda_1(\mathbf{L}_{\mathrm{P}}^{-1})$. We introduce the following proposition for the case where a unitary matrix is employed as the precoder matrix.

Proposition 4.4.1: Consider a precoded IF-MIMO system similar to the aforementioned one except that the precoder matrix is unitary. Let $\check{\mathbf{P}} \in U(M)$ be the precoder matrix and $\mathbf{L}_{\check{\mathrm{P}}} = \check{\mathbf{P}}^H\mathbf{L}$ be the matrix corresponding to (4.16) in the orthogonal precoder case. The effective SNR of the worst sub-channel is bounded as

$$\mathtt{SNR}_{\mathrm{eff,opt}} \geq \frac{1}{4M^2}\lambda_1^2(\mathbf{L}_{\check{\mathrm{P}}}^{-1}). \tag{4.41}$$

Proof. Because $\check{\mathbf{P}}$ is a unitary matrix of dimension M, which is complex-valued, the resulting lattice $\Lambda(\mathbf{L}_{\check{\mathrm{P}}}^H)$ and its dual are also complex-valued with dimension M. In the real-valued domain, those lattices have dimension of $2M$. Hence, following the proof of Proposition 4.3.1, the desired result is obtained. ■

From Propositions 4.3.1 and 4.4.1, we can see that a larger $\lambda_1^2(\mathbf{L}_{\check{\mathrm{P}}}^{-1})$ of unitary precoder cannot guarantee that the corresponding $\mathtt{SNR}_{\mathrm{eff,opt}}$ is also higher than that of orthogonal precoder. In particular, consider the case where $\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1}) \leq$

$\lambda_1^2(\mathbf{L}_{\breve{\mathrm{P}}}^{-1}) < 4\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})$.[4] Even though $\lambda_1^2(\mathbf{L}_{\breve{\mathrm{P}}}^{-1}) \geq \lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})$, the corresponding lower bound on $\texttt{SNR}_{\text{eff,opt}}$ of unitary precoder is lower than that of orthogonal precoder. Hence, if we search for a precoder matrix over unitary groups, we may end up obtaining lower $\texttt{SNR}_{\text{eff,opt}}$ than in the case when we search over orthogonal groups even though the obtained optimal unitary precoder matrix may have larger $\lambda_1^2(\mathbf{L}_{\breve{\mathrm{P}}}^{-1})$. According to (4.25) and (4.27), a lower $\texttt{SNR}_{\text{eff,opt}}$ implies lower achievable rate and higher outage probability. To illustrate this phenomena more clearly, we provide a simple example in the following.

Example 4.4.1: Consider $\gamma = 30$ dB and a randomly generated channel matrix $\mathbf{H}$ with the resulting

$$\mathbf{L} = \begin{bmatrix} 0.0138 & 0 \\ 0 & 0.1595 \end{bmatrix}. \tag{4.42}$$

Performing optimization (4.40) over orthogonal matrices, we found a good orthogonal matrix

$$\mathbf{P} = \begin{bmatrix} 0.5475 & -0.8368 \\ 0.8368 & 0.5475 \end{bmatrix}. \tag{4.43}$$

This matrix yields $\lambda_1(\mathbf{L}_{\mathrm{P}}^{-1}) = 22.7196$ and $\lambda_M(\mathbf{L}_{\mathrm{P}}^{T}) = 0.0499$. The corresponding $\texttt{SNR}_{\text{eff,opt}}$ achieved by $\mathbf{P}$ is 26.04 dB.

On the other hand, searching over unitary matrices, we found a good unitary matrix

$$\breve{\mathbf{P}} = \begin{bmatrix} 0.0008 - 0.9387i & -0.3386 + 0.0654i \\ 0.3053 + 0.1602i & -0.2698 + 0.8991i \end{bmatrix}. \tag{4.44}$$

With this matrix we have $\lambda_1(\mathbf{L}_{\breve{\mathrm{P}}}^{-1}) = 25.0441$, $\lambda_M(\mathbf{L}_{\breve{\mathrm{P}}}^{H}) = 0.0550$, and $\texttt{SNR}_{\text{eff,opt}} = 25.19$ dB.

If we only consider (4.40), surely we will choose $\breve{\mathbf{P}}$ as our precoder matrix because $\lambda_1(\mathbf{L}_{\breve{\mathrm{P}}}^{-1}) > \lambda_1(\mathbf{L}_{\mathrm{P}}^{-1})$. However, if we take a look at the resulting $\texttt{SNR}_{\text{eff,opt}}$, then we must choose $\mathbf{P}$ as our precoder matrix because the resulting $\texttt{SNR}_{\text{eff,opt}}$ is higher which implies higher achievable rate and lower error-rate.

[4] Note that $\mathbf{L}_{\mathrm{P}}$ corresponds to the orthogonal precoder case, while $\mathbf{L}_{\breve{\mathrm{P}}}$ to the unitary case. From computer simulations, we found that most of the time (99.8%), orthogonal and unitary precoders result in this particular case.

The above example illustrates that the search for a precoder matrix over unitary matrices does not always result in better performance than the search over orthogonal matrices. We shall note that here we do not claim *theoretically* that orthogonal precoder is always better than its unitary counterpart. Rather, the fact that the search for precoder matrix over unitary groups may result in worse performance than the search over orthogonal groups in spite of its higher complexity, gives us sufficient reason to adopt and recommend the orthogonal precoder.

One numerical example may not be able to validate the superiority of orthogonal precoder in terms of performance. Therefore, we also carried out numerical simulations with more than 10^5 channel realizations for each evaluated SNR and present the results in Figs 4.3 and 4.4 in Section 4.7. These results confirm that on average, indeed even though $\lambda_1^2(\mathbf{L}_{\breve{\mathrm{P}}}^{-1})$ is higher than $\lambda_1^2(\mathbf{L}_{\mathrm{P}}^{-1})$, orthogonal precoder achieves higher average $\mathtt{SNR}_{\mathrm{eff,opt}}$ and achievable rate, and lower outage and error probabilities than unitary precoder. Thus, now we can claim that finding the optimal IF-MIMO precoder matrix over orthogonal groups instead of unitary groups is beneficial in terms of both complexity and performance.

4.5 Finding the Optimal Precoder Matrix

To find the optimal orthogonal precoder matrix, let us first define the objective function as follow

$$J(\mathbf{P}) = \min_{\mathbf{v} \in \mathbb{Z}^M \backslash \mathbf{0}} \left\| \mathbf{L}^{-1} \mathbf{P} \mathbf{v} \right\|^2 . \tag{4.45}$$

The optimization problem in (4.40) can now be written as

$$\mathbf{P}_{\mathrm{opt}} = \underset{\mathbf{P} \in O(M)}{\arg\max}\, J(\mathbf{P}). \tag{4.46}$$

The difficulties of solving the optimization problem above lie within the combination of two major obstacles: (i) orthogonal matrix constraint and (ii) finding the minimum distance of the lattice $\Lambda(\mathbf{L}^{-1}\mathbf{P})$.

For a 2×2 MIMO system, a convenient parameterization of 2-dimensional orthogonal group was proposed in [3]. The orthogonal matrix $\mathbf{P}$ is parameterized using one angle θ as

$$\mathbf{P}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}. \tag{4.47}$$

With this parameterization, $\mathbf{P}_{\mathrm{opt}}$ can be estimated easily by performing a simple exhaustive search over only one parameter $\theta \in [0, \pi/4]$. Indeed, this technique performs very well for 2-dimensional orthogonal group. However, beyond that, it becomes unwieldy and prohibitively complex because the exhaustive search has to be done over $M(M-1)/2$ parameters (angles) [62] and the minimum distance of the resulting lattice has to be checked at every search or iteration.

A simple approach to solving optimization problems with orthogonality constraint is to perform gradient-based search algorithm such as the steepest gradient (SG) algorithm. Interestingly, by exploiting the geometrical properties of orthogonal group as a Lie group [2, 59, 61], the orthogonality constraint is always naturally satisfied at every step of the SG algorithm. This means that an optimization problem with an orthogonality constraint is transformed into an unconstrained one, which makes the optimization process easier. For this reason we will use the SG algorithm on Lie groups [2, 59, 61] to solve our problem. As general reference for the Lie group theory, see [67]. Unfortunately, the SG algorithm on Lie groups is not directly applicable to our problem. This is because our objective function in (4.45) in not purely constrained with orthogonality and it is not even differentiable because it depends not only on $\mathbf{P}$, but also on a discrete integer vector $\mathbf{v}$. To overcome this, we break the problem down into two sub-problems.

4.5.1 Sub-Problem 1: Local Search

Observe that by fixing the integer vector $\mathbf{v}$, we can transform the objective function in (4.45) into a differentiable function on which the SG algorithm can work. Assume that we start the search for the solution from an initial $\mathbf{P}_{\mathrm{i}} \in O(M)$. A vector at the minimum distance of $\Lambda(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}})$ is given by an integer vector $\mathbf{v}_{\mathrm{i}}$, i.e., $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}}) = \|\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}}\mathbf{v}_{\mathrm{i}}\|$. Define

$$\tilde{J}(\mathbf{P}) = \left\|\mathbf{L}^{-1}\mathbf{P}\mathbf{v}_{\mathrm{i}}\right\|^2. \tag{4.48}$$

Our first sub-problem is thus, given an initial $\mathbf{P}_{\mathrm{i}}$ with the corresponding $\mathbf{v}_{\mathrm{i}}$, find a "good" $\tilde{\mathbf{P}}_{\mathrm{opt}}$ such that

$$\tilde{\mathbf{P}}_{\mathrm{opt}} = \operatorname*{arg\,max}_{\substack{\mathbf{P}\in O(M),\\ \lambda_1(\mathbf{L}^{-1}\mathbf{P})=\left\|\mathbf{L}^{-1}\mathbf{P}\mathbf{v}_{\mathrm{i}}\right\|}} \tilde{J}(\mathbf{P}). \tag{4.49}$$

This means that we must find $\tilde{\mathbf{P}}_{\mathrm{opt}}$ that maximizes (4.48) such that the minimum distance of $\Lambda(\mathbf{L}^{-1}\tilde{\mathbf{P}}_{\mathrm{opt}})$ is still given by $\mathbf{v}_{\mathrm{i}}$, i.e., $\lambda_1(\mathbf{L}^{-1}\tilde{\mathbf{P}}_{\mathrm{opt}}) = \left\|\mathbf{L}^{-1}\tilde{\mathbf{P}}_{\mathrm{opt}}\mathbf{v}_{\mathrm{i}}\right\|$.[5] Because $O(M)$ is a manifold, we can think geometrically that the search is done by moving over the surface of $O(M)$ starting from $\mathbf{P}_{\mathrm{i}}$ to a point that satisfies (4.49). We can also think of this search as rotating the whole lattice points $\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}}\mathbb{Z}$ until a certain degree such that its minimum distance is maximized while keeping the integer vector giving its minimum distance remains unchanged.

Like the conventional SG algorithm, the search for the solution is performed by iteratively moving from one point to another in the search space in the steepest direction. Particularly, at ℓ-th iteration, a move from the current point $\mathbf{P}_\ell$ to $\mathbf{P}_{\ell+1}$ over $O(M)$ is made. This move is equivalent to the move from $\mathbf{I}$ to some point $\mathbf{R}_\ell \in O(M)$ such that $\mathbf{P}_{\ell+1} = \mathbf{R}_\ell\mathbf{P}_\ell$. The question is then how to choose the movement matrix $\mathbf{R}_\ell$.

For defining a movement in the steepest direction, we will make use of the corresponding Lie algebra $\mathfrak{o}(M)$ instead of $O(M)$ which is closed only under matrix multiplication. The Lie algebra $\mathfrak{o}(M)$ is the vector space of the $M \times M$ skew-symmetric matrices with additional Lie bracket operation in the form of matrix exponential [68]. Because $\mathfrak{o}(M)$ is a vector space which is closed under addition and scalar multiplication, it is easier to define a movement over $\mathfrak{o}(M)$ rather than over $O(M)$. $O(M)$ and $\mathfrak{o}(M)$ are connected by matrix exponential and matrix logarithm operators [67, Chapter 2]. As illustrated in Fig. 4.1, any point $\mathbf{P} \in O(M)$ can be mapped to a point $\mathbf{S} \in \mathfrak{o}(M)$ using $\mathbf{S} = \log(\mathbf{P})$ and any point $\mathbf{S}' \in \mathfrak{o}(M)$ can be mapped to a point $\mathbf{P}' \in O(M)$ using $\mathbf{P}' = \exp(\mathbf{S}')$. Thus, any movement in $O(M)$ is equivalent to a movement in $\mathfrak{o}(M)$, and vice versa.

Consider our SG algorithm at ℓ-th iteration. To move from $\mathbf{I}$ to $\mathbf{R}_\ell$, first, we map $\mathbf{I}$ to a point in $\mathfrak{o}(M)$, which is $\mathbf{0}$ because $\log(\mathbf{I}) = \mathbf{0}$. Then, from $\mathbf{0}$ we make a move to a point $\mathbf{S}_\ell$ over $\mathfrak{o}(M)$. Once $\mathbf{S}_\ell$ is found, we can compute $\mathbf{R}_\ell = \exp(\mathbf{S}_\ell)$,

[5]For any real lattice, $\|\mathbf{L}^{-1}\mathbf{P}\mathbf{v}_{\mathrm{i}}\| = \|\mathbf{L}^{-1}\mathbf{P}(-\mathbf{v}_{\mathrm{i}})\|$, and thus, we must also include $-\mathbf{v}_{\mathrm{i}}$ in the constraint. But for brevity, we omit it.

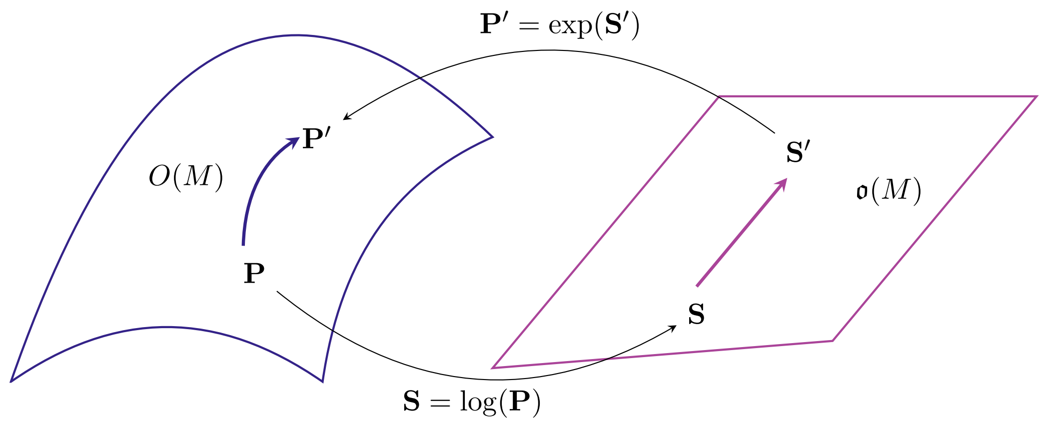


Fig. 4.1. $O(M)$ and $\mathfrak{o}(M)$ are connected by matrix exponential and logarithm operations [2]. A movement over $O(M)$ can be defined equivalently by a movement over $\mathfrak{o}(M)$.

and subsequently $\mathbf{P}_{\ell+1} = \mathbf{R}_\ell \mathbf{P}_\ell$. The movement matrix $\mathbf{S}_\ell$ has to be decided based on the steepest gradient of $\tilde{J}(\mathbf{P}_\ell)$ in the $\mathbf{S}$-space. Define $\Delta_\mathbf{P}\tilde{J}(\mathbf{P}_\ell)$ as the gradient of $\tilde{J}(\mathbf{P})$ in the $\mathbf{P}$-space at $\mathbf{P} = \mathbf{P}_\ell$. It is easy to derive that

$$\Delta_\mathbf{P}\tilde{J}(\mathbf{P}_\ell) = 2(\mathbf{L}^{-1})^2\mathbf{P}_\ell \mathbf{v}_\mathrm{i}\mathbf{v}_\mathrm{i}^T. \tag{4.50}$$

Using the result from [2], the steepest gradient of $\tilde{J}(\mathbf{P})$ in the $\mathbf{S}$-space at $\mathbf{P} = \mathbf{P}_\ell$ is given by

$$\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell) = \Delta_\mathbf{P}\tilde{J}(\mathbf{P}_\ell)\mathbf{P}_\ell^T - \mathbf{P}_\ell(\Delta_\mathbf{P}\tilde{J}(\mathbf{P}_\ell))^T. \tag{4.51}$$

For a constant μ, a move from $\mathbf{0}$ to $\mathbf{S}_\ell$ now can be defined as

$$\mathbf{S}_\ell = \mathbf{0} + \mu\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell) = \mu\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell). \tag{4.52}$$

We refer to μ as the *step size*. The move from $\mathbf{P}_\ell$ to $\mathbf{P}_{\ell+1}$ is thus can be written as

$$\mathbf{P}_{\ell+1} = \exp(\mu\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell))\mathbf{P}_\ell. \tag{4.53}$$

As in the general SG algorithm, choosing an appropriate step size is crucial for the convergence. A fixed step size can ensure a convergence close to a local optimum, but in general it requires many iterations. Therefore, it is desirable to select an appropriate step size at each iteration for a faster convergence. The appropriate step size is commonly determined based on the objective function. However, in our problem, the step size depends not only on the objective function, but also on the problem constraint; that is the integer vector providing the minimum distance of the corresponding lattice must not change. From here on we refer to this constraint as *integer vector constraint*. To select an appropriate step size at every iteration we propose the following two steps.

Step 1: In this step, the step size is determined based on the objective function. Consider a point in $O(M)$ emanating from $\mathbf{P}_\ell$ along the steepest direction $\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell)$ as a function of μ

$$\mathbf{P}(\mu) = \exp(\mu\Delta_\mathbf{S}\tilde{J}(\mathbf{P}_\ell))\mathbf{P}_\ell, \tag{4.54}$$

and define

$$\hat{J}(\mu) \triangleq \tilde{J}(\mathbf{P}(\mu)). \tag{4.55}$$

The step size at ℓ-th iteration is initially chosen such that

$$\mu_\ell = \arg\max_\mu \hat{J}(\mu). \tag{4.56}$$

The optimal μ_ℓ is difficult to find in general. Fortunately, our objective function $\hat{J}(\mu)$ in (4.56) has a desirable property that may be exploited to determine μ_ℓ. The matrix exponential of skew-symmetric matrices in (4.54) induces an *almost periodic* [69, 70] behavior of $\hat{J}(\mu)$ with respect to μ. As an example, given an orthogonal matrix $\mathbf{P} \in O(4)$ and a diagonal matrix $\mathbf{L} \in \mathbb{R}^{4\times 4}$, $\hat{J}(\mu)$ is drawn in Fig. 4.2, where it is shown that $\hat{J}(\mu)$ is periodic.[6] Therefore, to determine μ_ℓ, we can use existing techniques that are used for finding local maximums of almost periodic functions. In particular, we adopt the polynomial approximation technique proposed in [70]. With this technique we find the first local maximum of $\hat{J}(\mu)$ (point A in Fig. 4.2) and then choose the corresponding μ as the our initial μ_ℓ. If the integer vector constraint is satisfied with this initial μ_ℓ, a further adjustment is not needed.

Step 2: μ_ℓ obtained in the step 1 is chosen such that $\hat{J}(\mu)$ is maximized. This will not lead us to the solution of (4.49) if the integer vector constraint is not

[6] $\hat{J}(\mu)$ function in Fig. 4.2 is periodic, but in general, matrix exponential of of skew-symmetric matrices induces almost periodic function [70].

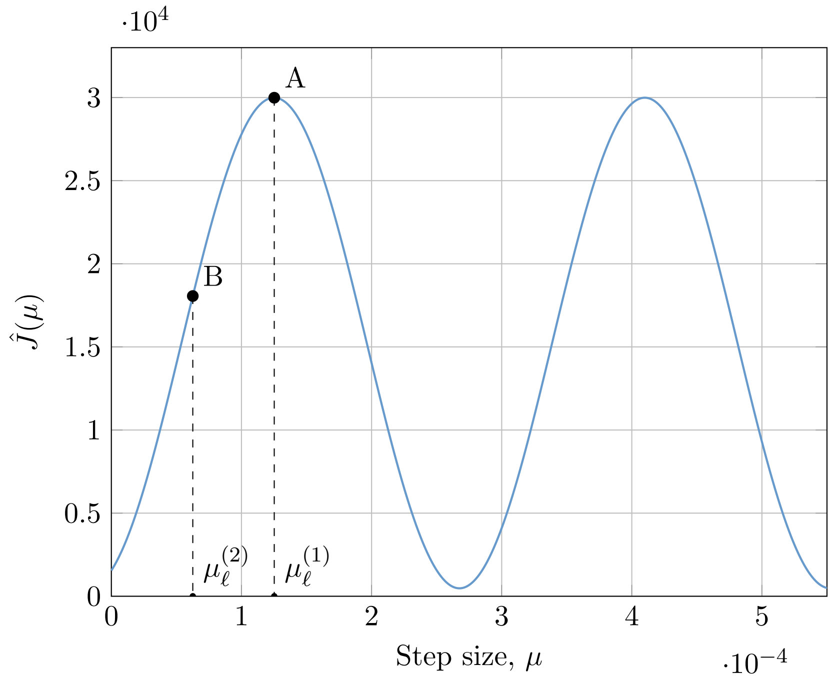


Fig. 4.2. An example of $\hat{J}(\mu)$ emanating from an orthogonal matrix $\mathbf{P}_\ell \in O(4)$ and a diagonal matrix $\mathbf{L} \in \mathbb{R}^{4\times 4}$ randomly generated at $\gamma = 30$ dB.

satisfied, i.e., the integer vector providing the minimum distance of $\Lambda(\mathbf{L}^{-1}\mathbf{P}(\mu_\ell))$ is different from that of $\Lambda(\mathbf{L}^{-1}\mathbf{P}_\mathrm{i})$. Therefore μ_ℓ obtained in the step 1 has to be further adjusted such that the constraint is always satisfied. Because the initial μ_ℓ from the step 1 provides the first local maximum of $\hat{J}(\mu)$, it is now easy to make readjustment. This adjustment is performed by iteratively halving μ_ℓ or dividing μ_ℓ by a constant $\zeta > 1$ until the constraint in satisfied.

We shall note that the process of finding μ_ℓ above always converges. From the step 1, μ_ℓ is initialized with a value giving the first local maximum of $\hat{J}(\mu)$ ($\mu_\ell^{(1)}$ in Fig. 4.2). If at this point, the integer vector constraint is satisfied, then μ_ℓ is found and step 2 is not needed. Otherwise, we have to reduce μ_ℓ, by dividing it with $\zeta > 1$. For example, with $\zeta = 2$, in Fig. 4.2 μ_ℓ will be reduced to $\mu_\ell^{(2)}$ that results in point B. If at this point the integer vector constraint is satisfied, then μ_ℓ is found. Otherwise, the process is repeated until the integer constraint is satisfied. Hence, it can be observed that this process always converges.

The summary of the algorithm for solving the sub-problem 1 is presented in Algorithm 1. This algorithm has two stopping conditions. The first one is the maximum number of iterations. This condition ensures that the complexity of

Algorithm 1 Local search: Finding a local optimal $\tilde{\mathbf{P}}_{\mathrm{opt}}$ from an initial orthogonal matrix $\mathbf{P}_{\mathrm{i}}$.

Input: $\mathbf{L}^{-1}$ and $\mathbf{P}_{\mathrm{i}}$.
Output: An estimate of local optimal precoder matrix $\tilde{\mathbf{P}}_{\mathrm{opt}}$.
1: Find $\mathbf{v}_{\mathrm{i}} \in \mathbb{Z}^M$ such that $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}}) = \left\|\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}}\mathbf{v}_{\mathrm{i}}\right\|$.
2: Initialize $\ell = 0$, $\mathbf{P}_\ell = \mathbf{P}_{\mathrm{i}}$.
3: Compute $\Delta_{\mathbf{S}}\tilde{J}(\mathbf{P}_\ell)$ as in (4.51).
4: Find μ_ℓ using the polynomial approximation [70].
5: Further adjust μ_ℓ:
while $\lambda_1(\mathbf{L}^{-1}\mathbf{P}(\mu_\ell)) \neq \left\|\mathbf{L}^{-1}\mathbf{P}(\mu_\ell)\mathbf{v}_{\mathrm{i}}\right\|$, set $\mu_\ell := \mu_\ell/2$.
6: Update $\mathbf{P}_{\ell+1} = \mathbf{P}(\mu_\ell)$ and $\ell := \ell + 1$. Iterate the steps 3 - 6 until convergence or until maximum iteration.
7: **return** $\tilde{\mathbf{P}}_{\mathrm{opt}} = \mathbf{P}_\ell$.

the algorithm does not exceed a certain level of complexity. The second condition is when the algorithm converges to a certain value. This means that if at some iteration, no further improvement on $\tilde{J}(\mathbf{P}_\ell)$ is achieved, then the algorithm stops. It has been shown that the SG algorithm with Lie group approach converges to an optimal point [2, 61]. If the integer vector constraint is ignored, our algorithm essentially tries to reach the same point. However, due to the constraint, our algorithm stops earlier at an edge point where the constraint is still being satisfied. Therefore, it is easy to see that the proposed algorithm also converges.

In Algorithm 1, the minimum distance of a lattice needs to be calculated. To this end, algorithms such as the Fincke-Pohst [46] algorithm or sphere decoding [49] algorithm and its variants [47, 52, 71], may be employed. One can also use the Lenstra-Lenstra-Lovász (LLL) algorithm [48] that exhibits much lower complexity. We found that the LLL algorithm proposed [64] also yields good performance when employed in our algorithm.

4.5.2 Sub-Problem 2: Global Search

The solution of the sub-problem 1 may not be the global optimal solution because given a starting point $\mathbf{P}_{\mathrm{i}}$, the search is performed over the surface limited to only around $\mathbf{P}_{\mathrm{i}}$. Therefore, to find the global optimal solution, it is crucial to select a good starting point $\mathbf{P}_{\mathrm{i}}$, which becomes our second sub-problem. We state our second sub-problem as follows: from $O(M)$, find a good matrix $\mathbf{P}_{\mathrm{i}}$ such that $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}})$ is as large as possible. This problem is indeed similar to our original

Algorithm 2 Global search: Finding a "good" initial orthogonal matrix $\mathbf{P}_{\mathrm{i}}$ for Algorithm 1.

Input: $\mathbf{L}^{-1}$.

Output: $\mathbf{P}_{\mathrm{i}} \in O(M)$ such that $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}})$ is large.

1: Initialize $\ell = 0$, $\mathbf{P}_{\mathrm{i}} := \mathbf{I}$.

2: Generate a random orthogonal matrix $\mathbf{P}_\ell$ with Haar measure distribution using [74].

3: If $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_\ell) > \lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}})$, $\mathbf{P}_{\mathrm{i}} := \mathbf{P}_\ell$.

4: $\ell := \ell + 1$ and repeat from step 2 for some iterations.

5: **return** $\mathbf{P}_{\mathrm{i}}$.

problem in (4.46), except that the solution of this sub-problem does not have to be optimal. A better or possibly optimal solution will be derived by refining the solution using Algorithm 1.

To solve this sub-problem, we adopt a random search technique. Random search has been widely used and is very suitable for ill-structured global optimization problem, where the objective function may be not differentiable, and possibly discontinuous over a continuous, discrete, or mixed continuous-discrete domain [72] just like exactly what we have in (4.46). Random search in general does not guarantee finding a global optimal solution. But it offers finding a good solution quickly. In literature, it has been shown that random search converges to the global optimal solution with some probability [72, 73].

The random search algorithm that we employ is quite straightforward and is summarized in Algorithm 2. The algorithm starts by initializing $\mathbf{P}_{\mathrm{i}} = \mathbf{I}$. Then, at every iteration ℓ an orthogonal matrix $\mathbf{P}_\ell$ is randomly generated with Haar measure distribution [74] and the minimum distance of the resulting lattice $\Lambda(\mathbf{L}^{-1}\mathbf{P}_\ell)$ is evaluated. If $\lambda_1(\mathbf{L}^{-1}\mathbf{P}_\ell) > \lambda_1(\mathbf{L}^{-1}\mathbf{P}_{\mathrm{i}})$, then $\mathbf{P}_\ell$ is kept as the temporary solution, i.e., $\mathbf{P}_{\mathrm{i}} := \mathbf{P}_\ell$. The more iterations we have, the higher probability that resulting $\mathbf{P}_{\mathrm{i}}$ is close to the global optimal solution $\mathbf{P}_{\mathrm{opt}}$. However, the complexity also increases as the number of iterations increases. Therefore, the stopping condition of Algorithm 2 depends on the desired level complexity, i.e., the maximum number of iterations allowed in the algorithm. In practice, we do not need many iterations because the result will be further refined using Algorithm 1. From computer simulations, we found that no significant gain is achieved after 30 iterations.

We shall emphasize that the proposed Algorithm 2 does not guarantee a convergence. This is because the algorithm is proposed based on a random search to cope

Algorithm 3 Finding $\mathbf{P}_{\text{opt}}$ for the original problem (4.40).

Input: $\mathbf{L}^{-1}$.
Output: $\mathbf{P}_{\text{opt}}$, a solution for (4.40).
1: Use Algorithm 2 to find $\mathbf{P}_{\text{i}}$.
2: With input $\mathbf{P}_{\text{i}}$, perform Algorithm 1 to obtain $\tilde{\mathbf{P}}_{\text{opt}}$.
3: Set $\mathbf{P}_{\text{opt}} := \tilde{\mathbf{P}}_{\text{opt}}$.
4: **return** $\mathbf{P}_{\text{opt}}$.

with our ill-structured optimization problem, which is non-differentiable and has a mixed continuous-discrete domain. It is well known that a random search does not guarantee a convergence, rather it may converge with some probability [72, 73].

4.5.3 Summary of the Proposed Algorithm

To find the solution for our original problem in (4.46), first, we perform a global search for a good candidate of $\mathbf{P}_{\text{i}}$ over $O(M)$ using Algorithm 2. The resulting $\mathbf{P}_{\text{i}}$ is then used as the starting point of the gradient-based local search following Algorithm 1, of which the result is expected to be an estimate of the global optimal solution. The overall algorithm is summarized in Algorithm 3.

We shall note that the proposed algorithm can also be applied to the unitary precoder case [3] with some modifications. First, all the regular matrix transpose operations are replaced with the Hermitian transpose. Then, the gradient in (4.50) is replaced with $\Delta_{\mathbf{P}}\tilde{J}(\mathbf{P}_\ell) = (\mathbf{L}^{-1})^H\mathbf{L}^{-1}\mathbf{P}_\ell\mathbf{v}_{\text{i}}\mathbf{v}_{\text{i}}^H$ and obviously we should generate a random unitary matrix instead of orthogonal one in the step 2 of Algorithm 2. The complexity of the unitary precoder case is clearly higher than the orthogonal precoder because most of the operations are done in complex-valued domain rather than real-valued domain.

4.6 Complexity Analysis

4.6.1 Complexity of Algorithm 3

This sub-section provides evaluation of computational complexity of Algorithm 3 and compares it to that of parameterization technique [3].

The parameterization technique introduced in [3] can be extended to higher dimensional MIMO [62]. In this case, the search for the optimal orthogonal precoder matrix is carried out over at least $M(M-1)/2$ parameters (angles). Denote these parameters as $\theta_1, ..., \theta_{M(M-1)/2}$. For simplicity, assume that $\theta_i, \forall i = 1, .., M(M-1)/2$, has a search space of $[0, 2\pi)$ which is discretized to ν samples. For each combination of samples of θ_i, an orthogonal matrix $\mathbf{P}(\theta_1, ..., \theta_{M(M-1)/2})$ is constructed and the minimum distance of the resulting lattice $\Lambda(\mathbf{L}^{-1}\mathbf{P}(\theta_1, ..., \theta_{M(M-1)/2}))$ is evaluated. Subsequently $\mathbf{P}(\theta_1, ..., \theta_{M(M-1)/2})$ that yields the largest minimum distance of $\Lambda(\mathbf{L}^{-1}\mathbf{P}(\theta_1, ..., \theta_{M(M-1)/2}))$ is chosen as the solution. To keep a low complexity, let us assume that the LLL algorithm with complexity of $\mathcal{O}(M^4 \log M)$ [64] is employed. The overall complexity of the parameterization technique is thus $\mathcal{O}(\nu^{M(M-1)/2} M^4 \log M)$.

Because the complexity of Algorithm 3 is dominated by the step 2 where Algorithm 1 is run, we only need to evaluate the complexity of Algorithm 1. The dominant operations in the Algorithm 1 are finding minimum distance of a lattice in the steps 1 and 5 and calculating matrix exponential in the steps 3, 4, and 5. To find the minimum distance of a lattice, we employ the same LLL algorithm with complexity of $\mathcal{O}(M^4 \log M)$ [64]. While for matrix exponential, there are many ways to calculate it. In literature, we found that the most efficient methods for calculating the matrix exponential exhibit computational complexity of $\mathcal{O}(M^3)$ [68]. Because the complexity of finding the minimum distance of a lattice is more dominant, we can ignore the complexity of computing a matrix exponential. Assume that we need ξ_i number of iterations to adjust the step size μ_ℓ in the step 5 and ξ_o number of iterations for Algorithm 1 to converge. Thus, the overall computational complexity of Algorithm 3 is $\mathcal{O}(\xi_\mathrm{o}\xi_\mathrm{i} M^4 \log M)$ or simply $\mathcal{O}(M^4 \log M)$. Now we can clearly see that the complexity of the proposed algorithm is much smaller than that of the parameterization technique.

4.6.2 Decoding Complexity

At the receiver side, the decoding complexity of the proposed scheme is nearly the same as ZF and MMSE receivers. This is because the IF receiver manipulates MIMO channels such that a SISO decoding can be employed, which is similar to ZF and MMSE receivers. An additional complexity comes from the step of finding a full-rank integer matrix $\mathbf{A}$. Consider slow-fading channels where the channel coefficients remain constant over a long period called *quasi-static channel interval.* Because $\mathbf{A}$ is essentially an approximation of the MIMO channels which remains constant during the interval, the search for $\mathbf{A}$ needs to be done only once in each static interval. This is in contrast to the general joint ML MIMO decoding in slow-fading channels. Assume that within the static interval, there are $T \in \mathbb{Z}$ number of codeword transmissions that can be made. In the joint ML decoding case, an optimal algorithm such as sphere decoding (SD) algorithm [46, 49] which has an exponential complexity has to be performed for each transmission; T times in one static interval. Assume that to find the optimal $\mathbf{A}$, the proposed scheme utilizes the same SD algorithm. In this case, the joint ML decoding would exhibit T times higher complexity than the proposed scheme.

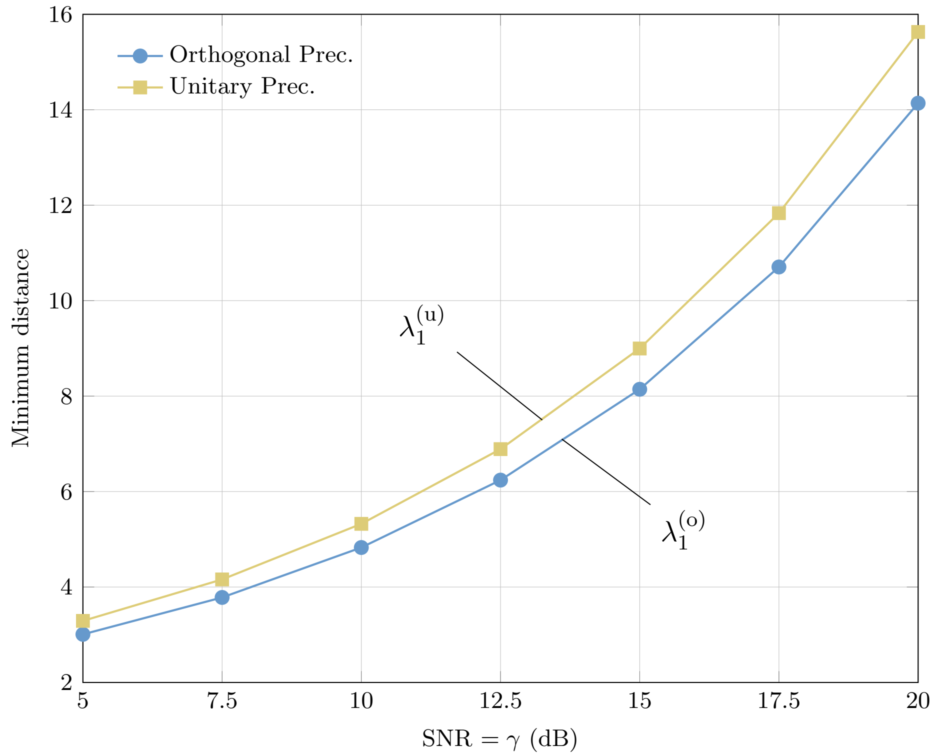
Even though a brute force method for finding the optimal integer matrix $\mathbf{A}$ has a high complexity of $\mathcal{O}(\gamma^M)$ [14], some effort has been made to develop more efficient algorithms. For instance, Ding *et al.* [50] developed an optimal algorithm based on the Schnorr-Euchner (SE) algorithm [47] to find the optimal $\mathbf{A}$ with computational complexity of $\mathcal{O}\left(M^4 \frac{(M\pi)^{M/2}}{\Gamma(M/2+1)}\right)$, where $\Gamma(\cdot)$ is the Gamma function. In a different approach, Wen *et al.* [51, 65] also exploited the SE algorithm to find the optimal $\mathbf{A}$ with significantly lower complexity. They showed that their algorithm is $\Omega(M)$ faster than [50], making it currently the most efficient existing optimal algorithm for finding the optimal integer matrix $\mathbf{A}$. To further reduce the complexity, Sakzad *et al.* [54] proposed an approximation algorithm based on the LLL algorithm with polynomial complexity of $\mathcal{O}(M^4 \log(2M))$. They also investigated other approximation algorithms based on Hermite-Korkine-Zolotareff (HKZ) and Minkowski lattice basis reduction algorithms, see [54] for more detail discussion. Other efficient algorithms can be found in [53, 64].

4.7 Numerical Evaluation

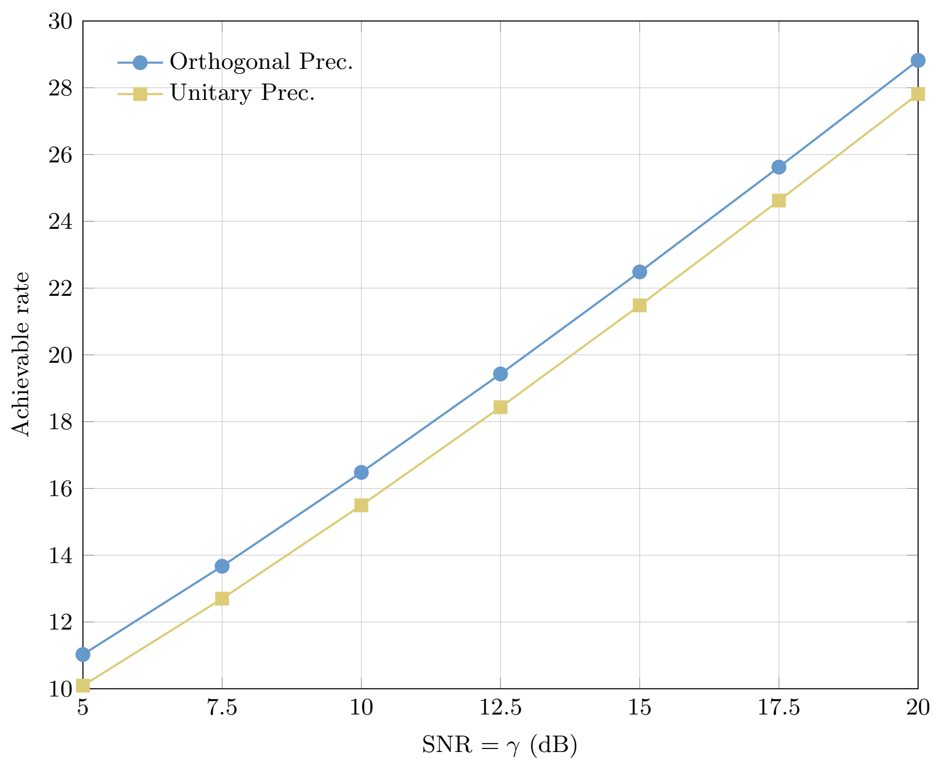
This section presents and analyzes the numerical results obtained from computer simulations conducted to compare the performance of the proposed schemes with existing schemes.

First, we compare the performance of orthogonal and unitary precoders.[7] For finding good orthogonal and unitary precoder matrices in the sense of (4.39), we use Algorithm 3 and its modified version described in Subsection 4.5.3, respec-

[7] Here, the orthogonal and unitary precoders refer to the precoders described in Section 4.2 where the precoder matrix is selected from groups of orthogonal and unitary matrices, respectively. The unitary precoder is exactly UPIF I.

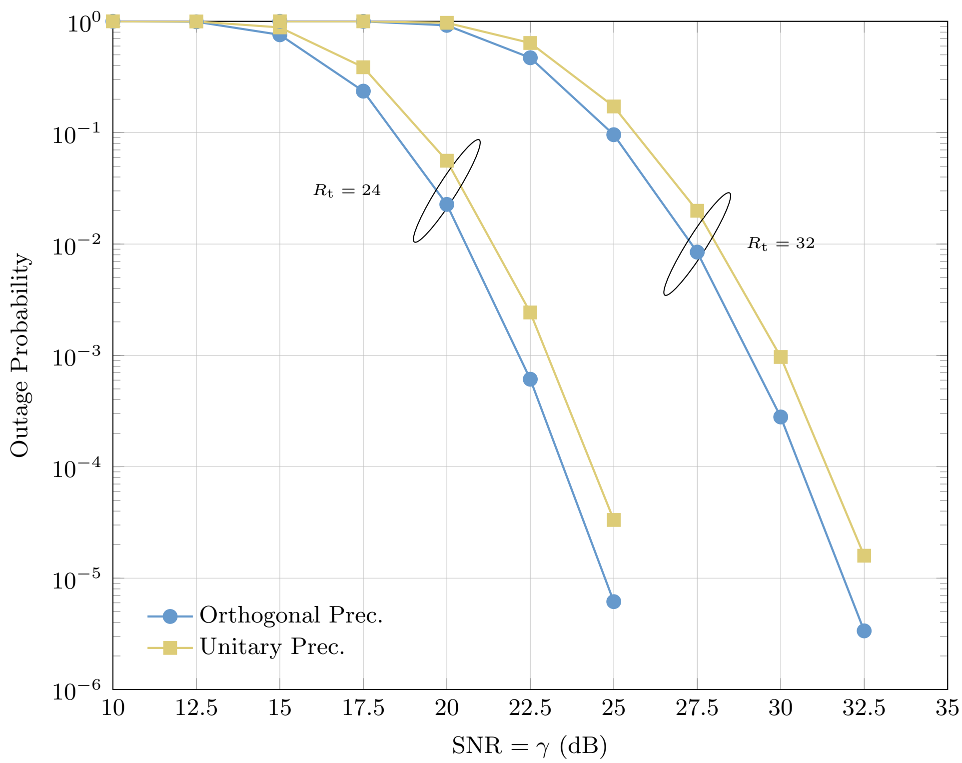


(a) Minimum distance of dual lattice

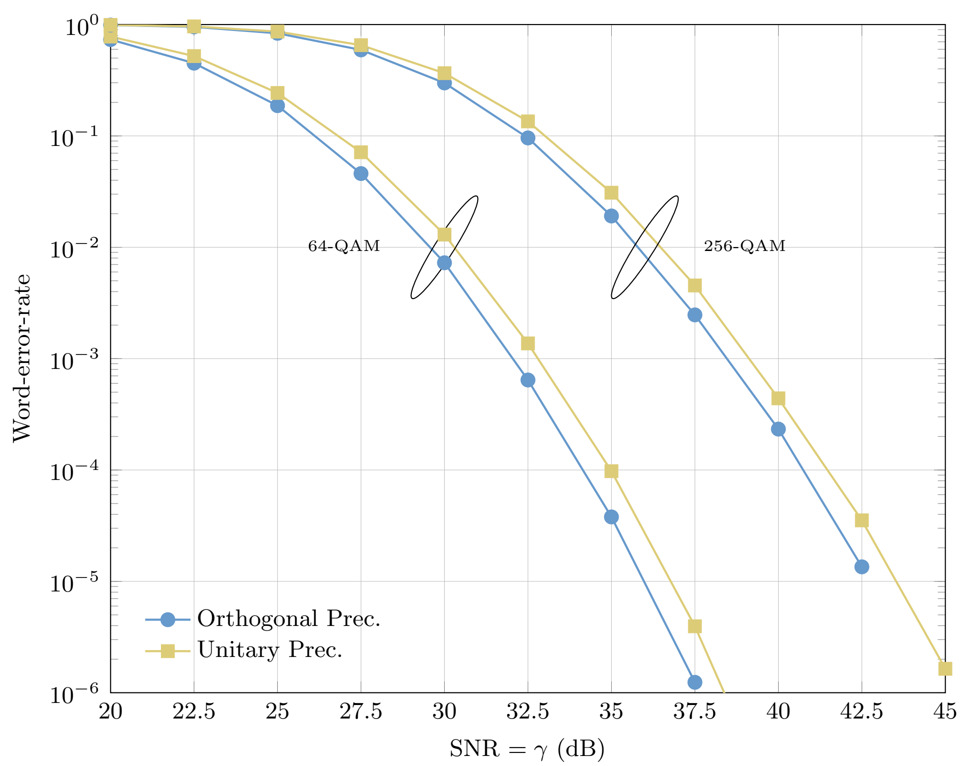


(b) Achievable rate

Fig. 4.3. Performance of orthogonal and unitary precoders in 4×4 MIMO : (a) average minimum distance of dual lattices $\Lambda(\mathbf{L}_{\mathrm{P}}^{-1})$ (orthogonal) and $\Lambda(\mathbf{L}_{\check{\mathrm{P}}}^{-1})$ (unitary), (b) average achievable rate, which is a function of $\mathtt{SNR}_{\mathrm{eff,opt}}$.



(a) Outage probability



(b) Word-error-rate

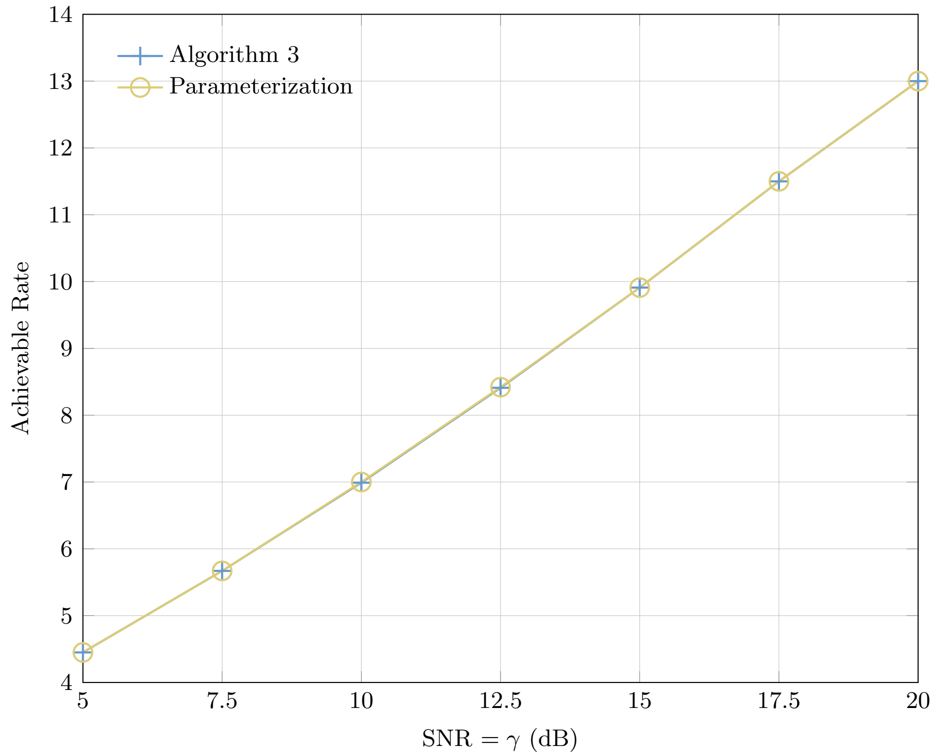
Fig. 4.4. Performance of the orthogonal and unitary precoders in 4×4 MIMO: (a) outage probability with target rate $R_t \in \{24, 32\}$, (b) word-error-rate with 64/256-QAM.

tively. Let $\lambda_1^{(\mathrm{o})} \triangleq \lambda_1(\mathbf{L}_{\mathrm{P}}{}^{-1})$ and $\lambda_1^{(\mathrm{u})} \triangleq \lambda_1(\mathbf{L}_{\check{\mathrm{P}}}{}^{-1})$ denote the minimum distance of the resulting dual lattices of orthogonal and unitary precoders, respectively (cf. Propositions 4.3.1 and 4.4.1). Fig. 4.3(a) shows the average of $\lambda_1^{(\mathrm{o})}$ and $\lambda_1^{(\mathrm{u})}$. Based on Fig. 4.3(a) and our main optimization problem (4.39), one may conclude that unitary precoder is better than orthogonal precoder because $\lambda_1^{(\mathrm{u})}$ is larger than $\lambda_1^{(\mathrm{o})}$. However, Fig. 4.3(b) shows the opposite, that orthogonal precoder has higher average achievable rates. A similar result is shown in Fig. 4.4 where orthogonal precoder has lower outage probability and word-error-rate (WER) than unitary precoder.[8] These results confirm our claim that for IF-MIMO precoding, in addition to the complexity advantage, searching for precoder matrices over orthogonal groups instead of unitary groups also offers performance advantage. This additional advantage is because the lower bound on $\mathtt{SNR}_{\mathrm{eff,opt}}$ of unitary precoder is smaller than that of orthogonal precoder as shown in Propositions 4.3.1 and 4.4.1. In fact, since the dimension of unitary matrices are twice that of orthogonal matrices in the real-valued domain, the largest successive minimum of the prime lattice $\Lambda(\mathbf{L}_{\check{\mathrm{P}}}^H)$ of unitary precoder is generally larger than that of the prime lattice $\Lambda(\mathbf{L}_{\mathrm{P}}^T)$ of orthogonal precoder, and hence its $\mathtt{SNR}_{\mathrm{eff,opt}}$ is smaller (see (4.23)), implying lower achievable rate and higher outage probability.

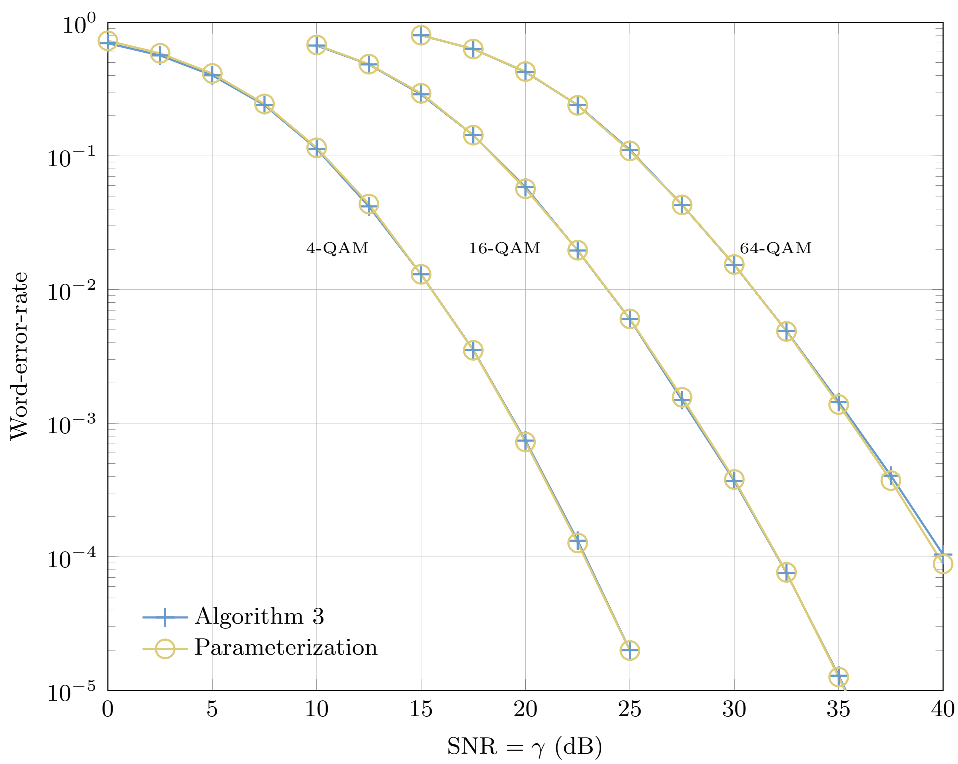
We then compare the performance of the parameterization technique [3] (proposed for UPIF I) and Algorithm 3. The parameterization was proposed in [3] for finding good orthogonal matrices for 2×2 IF-MIMO. Even though it is possible to extend this technique to higher dimension [62], it exhibits exponential complexity as described in Section 4.6.1. For this reason, we only compared them in the 2×2 IF-MIMO case. Fig. 4.5 depicts the results of achievable rate and WER performance of the parameterization algorithm of [3] compared to our proposed algorithm. It can be clearly seen that Algorithm 3 achieves nearly identical performance to the parameterization technique in various cases. Since Algorithm 3 has low complexity and yields good performance, we can easily employ it to realize orthogonal precoder for higher dimension IF-MIMO as we will see later.

Next, we compare the performance of the proposed orthogonal precoder with UPIF II. We employ Algorithm 3 for the proposed precoder. According to [3], the optimal precoder matrix for UPIF II should be chosen from *unitary* groups

[8] We define a word as $(\mathbf{w}_1, \ldots, \mathbf{w}_M)$. For calculating WER, we declare an error event when $(\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_M) \neq (\mathbf{w}_1, \ldots, \mathbf{w}_M)$.

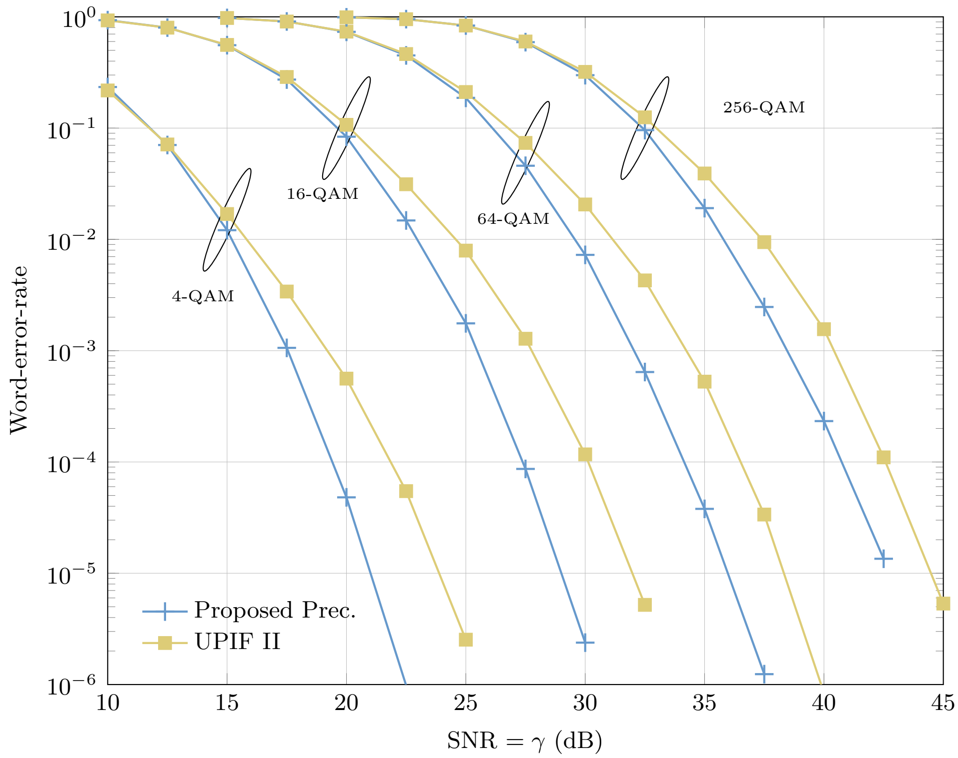


(a) Achievable Rate

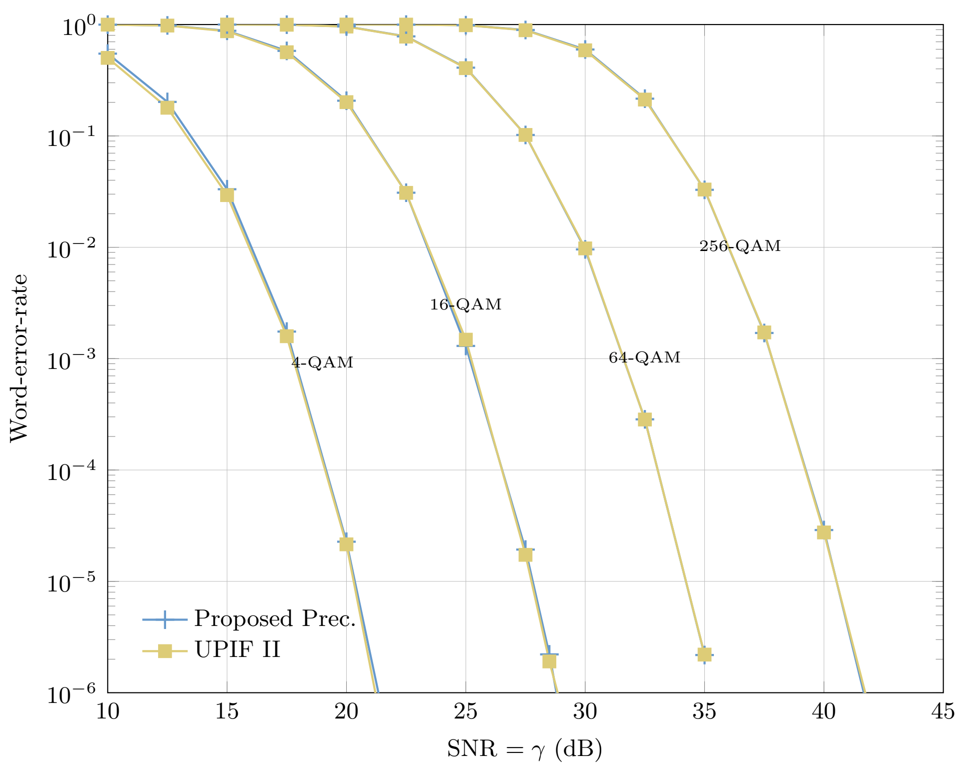


(b) Word-error-rate

Fig. 4.5. Performance of orthogonal precoders using Algorithm 3 and parameterization [3] in 2×2 MIMO: (a) average achievable rate (b) word-error-rate with 4/16/64-QAM.



(a) 4×4 MIMO

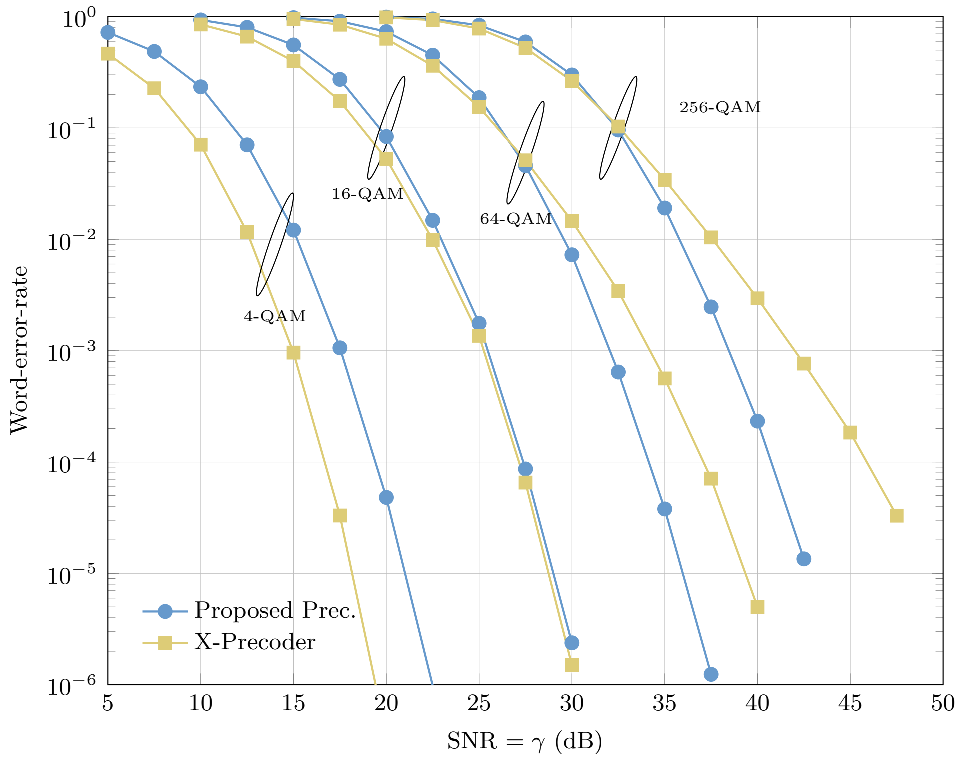


(b) 8×8 MIMO

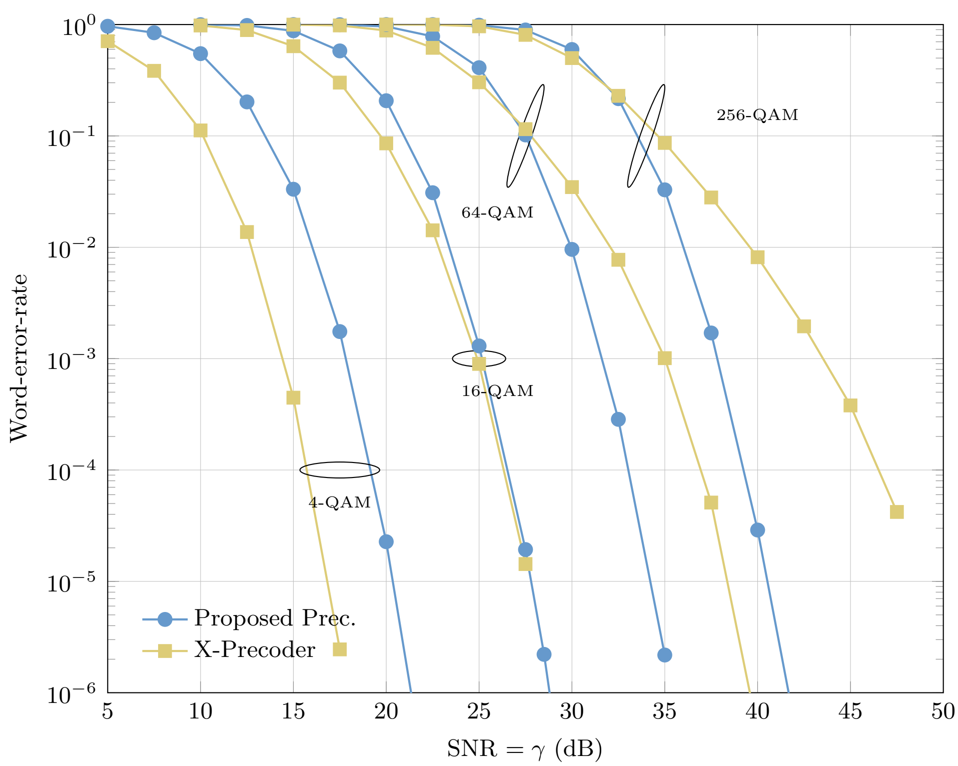
Fig. 4.6. WER of the proposed precoder and UPIF II in: (a) 4×4 MIMO, (b) 8×8 MIMO.

such that it has the largest minimum product distance [58]. However, finding the minimum product distance of a lattice is a hard problem, especially for unitary matrices. To the best of our knowledge, currently there is no optimal unitary matrix with respect to minimum product distance known. However, there are some available *orthogonal* matrices having good minimum product distance properties listed in [75]. We used these matrices for the UPIF II simulations. Fig. 4.6 shows the results of WER for 4×4 and 8×8 MIMO configurations each with 4/16/64/256-QAM. One can see that the proposed precoder and UPIF II yield nearly the same performance in the 8×8 MIMO case. While in the 4×4 MIMO case, the proposed precoder outperforms UPIF II for all 4/16/64/256-QAM. Even though we cannot confirm that the proposed precoder is better than UPIF II for all MIMO configurations, we can say that the proposed precoder can perform better in some scenarios. Moreover, the proposed precoder can be employed for any MIMO dimension, while for dimension beyond 30, it is hard to realize UPIF II because no "good" orthogonal matrix for UPIF II with dimension beyond 30 is currently available in literature.

Lastly, we compare the proposed precoder to the X-precoder [30], an ML- and QAM-based precoding scheme. In Fig. 4.7, we present WER performance for 4×4 and 8×8 MIMO configurations with various QAM constellations. In both MIMO configurations, the behavior of WER curves is similar. One can see that the X-precoder is better than the proposed precoder for 4-QAM case, while for 16-QAM, both schemes achieve almost the same performance in high SNR regime. However, for 64- and 256-QAM, we can clearly see the significant advantage of the proposed precoder over the X-precoder in terms of WER. This advantage comes from the fact that the error performance of the X-precoder is characterized by the minimum distance of received QAM constellations which gets smaller as the constellations size increases. Therefore, the error performance degrades as the constellation size increases. On the other hand, the error performance of the proposed precoder is characterized by the effective SNR, and thus, it is not significantly affected by the constellation size. Moreover, it is known that the X-precoder does not achieve full diversity gain, while similar to UPIF I [3], the proposed precoder achieves full diversity gain. We conclude that the proposed orthogonal precoder is superior to the X-precoder for high order QAM.



(a) 4×4 MIMO



(b) 8×8 MIMO

Fig. 4.7. WER of the proposed precoder and X-precoder in: (a) 4×4 MIMO, (b) 8×8 MIMO.

4.8 Summary

We have considered an orthogonal precoding scheme for MIMO with integer-forcing receivers (IF-MIMO). We showed that the proposed orthogonal precoder is better than its unitary counterpart in terms of both performance and complexity. We then proposed methods based on the steepest gradient algorithm on Lie groups and a random search algorithm for finding good orthogonal matrices for the proposed precoder. These methods exhibit lower complexity than the parameterization technique, and can be applied to any MIMO configuration. The numerical results confirmed that the proposed precoder outperforms UPIF II and the X-precoder in some scenarios. Even though the X-precoder is designed specifically for QAM constellations, the proposed precoder yields better error performance in high order QAM cases, e.g., 64/256-QAM.

Chapter 5

Conclusions

5.1 General Conclusions

In this dissertation, we have investigated applications of compute-and-forward methods to two different types of wireless communications systems, namely multiple access relay channels and precoded MIMO systems. For each system, we identified the problems that arise in the existing schemes and proposed solutions to solve them and to improve their performance.

The first main results dealt with the main issue of applying compute-and-forward methods the MARC; that is the linear combinations received by the final destination must be linearly independent. We proposed two cooperation strategies where the transmission is performed as efficiently as possible while keeping the outage probability as low as possible. We showed the performance improvement in terms of outage probability and network throughput achieved by our proposed strategies over existing strategies. It is also shown that the second strategy achieves full-diversity gain of the MARC.

As the second results, we developed an orthogonal precoding scheme for IF-MIMO. Our scheme was proposed based on UPIF [3] which uses a precoder matrix from unitary groups for IF-MIMO. It is quite obvious that the proposed orthogonal precoding has lower complexity than the unitary precoding. One may expect that this complexity advantage is obtained with the cost of performance degradation. However, we showed that the opposite is true — that the proposed precoding outperforms unitary precoding in terms of achievable rate and outage probability. We further proposed an efficient algorithm for finding a "good" precoder matrix based on the steepest gradient algorithm that exploits the geometrical properties of orthogonal matrices as a Lie group.

5.2 Future Work

Having intensively investigated applications of compute-and-forward methods in some wireless communication systems, we found several interesting problems that may be addressed as future work. We conclude this dissertation with a short discussion of some research directions related to the problems considered in this dissertation.

1. **Diversity-Multiplexing Tradeoff (DMT) Analysis**. In this dissertation, we have focused on the design of cooperation strategies for multiple-access relay channels employing compute-and-forward with a main objective of enhancing outage probability performance. The considered MARC utilizes full multiplexing gain where each source transmits message streams independent of the others. However, there may be cases where only a certain level of multiplexing gain is required in exchange for higher order diversity gain. Therefore, the optimal diversity-multiplexing tradeoff in the multiple-access relay channels with compute-and-forward should be investigated. This is a quite challenging problem because the diversity gain of the M best linear equations of a compute-and-forward scheme has to be devised — this problem is related to the successive minima problem of a lattice.

   While DMT analysis is commonly characterized with the asymptotic (infinite) SNR assumption, one may also have an interest in the analysis where SNR is finite. This analysis is particularly useful because in practice, wireless communication systems often use sufficiently low or moderate SNR. The analysis can reveal the diversity gain and multiplexing behavior of a system with realistic (finite) SNR. To the best of our knowledge, there is no work in the literature that addresses the finite-SNR DMT analysis for either the MARC with compute-and-forward or IF-MIMO.

2. **MARC-CF with Imperfect Feedback**. Our results in Chapter 3 are based on assumption that the feedback from the destination are always received perfectly. In practice, however, it may not be always the case. It would be of

interest to investigate the performance of MARC with compute-and-forward assuming noisy or imperfect feedback channels. In particular, we think that assuming the same channel model for both feedforward and feedback channels is an interesting open problem. Interested readers are referred to [76–78].

3. **ARQ-based Compute-and-Forward Schemes**. We have shown that with the help of feedback we can improve the outage performance of the MARC-CF significantly. Even though the lim-FB strategy could not achieve the full-diversity gain of the MARC, it can be further improved by introducing an automatic repeat request (ARQ) protocol. In this dissertation, we only limit the destination to request for help to the relay only one time. It is also of interest to investigate the network performance in the case multiple requests are allowed. See [79] for the performance of an ARQ protocol in a MARC with decode-and-forward. To the best of our knowledge, there is no work in the literature that investigates the performance of a compute-and-forward scheme with an ARQ protocol.

4. **IF-MIMO Precoding Without CSIT**. We have developed a precoding scheme for IF-MIMO with assumption that all channel state information (CSI) is known by all nodes including transmitters (CSIT). However, letting the transmitters know CSI requires a large amount of communication overhead which is less practical. Therefore, designing a precoding scheme without CSIT would be of interest. One insight may be obtained from the *blind compute-and-forward* [80] where the need of CSI is eliminated for employing compute-and-forward.

Appendix A

Proof of Proposition 4.3.2

Recall that a bijective mapping $\mathcal{E}$ is employed to map $\mathbf{w}_m$ to a codeword $\mathbf{x}_m$. Further, given a full rank matrix $\mathbf{A}$, all $\mathbf{x}_m$'s can be decoded correctly if and only if all sub-channels decode their linear combination $\mathbf{c}_m$ correctly. Therefore, (4.37) is equivalent to

$$P_{\mathrm{e}} = \Pr\big((\hat{\mathbf{w}}_1, ..., \hat{\mathbf{w}}_M) \neq (\mathbf{w}_1, ..., \mathbf{w}_M)\big) \tag{A.1}$$

$$= \Pr\big((\hat{\mathbf{x}}_1, ..., \hat{\mathbf{x}}_M) \neq (\mathbf{x}_1, ..., \mathbf{x}_M)\big) \tag{A.2}$$

$$= \Pr\big((\hat{\mathbf{c}}_1, ..., \hat{\mathbf{c}}_M) \neq (\mathbf{c}_1, ..., \mathbf{c}_M)\big). \tag{A.3}$$

Define the error probability at sub-channel m as

$$P_{\mathrm{e},m} = \Pr(\hat{\mathbf{c}}_m \neq \mathbf{c}_m). \tag{A.4}$$

Because $\Lambda_{\mathrm{c}} = \alpha\mathbb{Z}[i]$ and $\Lambda_{\mathrm{s}} = 2^{2q}\Lambda_{\mathrm{c}}$, the resulting linear combination and effective noise in (4.9) respectively become $\mathbf{c}_m \in \alpha\mathbb{Z}[i]$ and $\mathbf{z}_{\mathrm{eff},m} \in \mathbb{C}$, i.e., they are one-dimensional complex-valued vectors. Thus,

$$\begin{aligned}
P_{\mathrm{e},m} &= \Pr\Big(\{\Re(\hat{\mathbf{c}}_m) \neq \Re(\mathbf{c}_m)\} \cup \{\Im(\hat{\mathbf{c}}_m) \neq \Im(\mathbf{c}_m)\}\Big) \\
&\leq 2\Pr\big(\Re(\hat{\mathbf{c}}_m) \neq \Re(\mathbf{c}_m)\big) && \text{(A.5)} \\
&= 2\Pr\left(|\Re(\mathbf{z}_{\mathrm{eff},m})| \geq \frac{\alpha}{2}\right) && \text{(A.6)} \\
&= 4\Pr\left(\Re(\mathbf{z}_{\mathrm{eff},m}) \geq \frac{\alpha}{2}\right), && \text{(A.7)}
\end{aligned}$$

where (A.5) is due to union bound and the fact that $\Re(\mathbf{c}_m)$ and $\Im(\mathbf{c}_m)$ have an identical probability distribution, (A.6) is because $\Re(\mathbf{c}_m)$ and $\Im(\mathbf{c}_m)$ are decoded using the nearest-neighbor quantizer with respect to $\alpha\mathbb{Z}[i]$, and (A.7) follows the symmetry of probability density function of $\Re(\mathbf{z}_{\text{eff},m})$ around zero. Using [28, Lemma 4], we have

$$P_{\text{e},m} \leq 4\Pr\left(\Re(\mathbf{z}_{\text{eff},m}) \geq \frac{\alpha}{2}\right) \tag{A.8}$$

$$\leq 4\exp\left(-\frac{\alpha^2}{4\sigma^2_{\text{eff},m}}\right) = 4\exp\left(-\frac{\alpha^2}{4\gamma\left\|\mathbf{L}_{\text{P}}^T\mathbf{a}_m\right\|^2}\right). \tag{A.9}$$

If $\mathbf{A}_{\text{opt}}$ is employed, then

$$P_{\text{e},m} \leq 4\exp\left(-\frac{\alpha^2}{4\gamma\lambda_m^2(\mathbf{L}_{\text{P}}^T)}\right) = 4\exp\left(-\frac{3}{2^{4q+1}\lambda_m^2(\mathbf{L}_{\text{P}}^T)}\right) \tag{A.10}$$

Now, due to (2.17), for all $m = \{1, ..., M\}$, we have

$$P_{\text{e},m} \leq 4\exp\left(-\frac{3}{2^{4q+1}\lambda_M^2(\mathbf{L}_{\text{P}}^T)}\right) \tag{A.11}$$

$$\leq 4\exp\left(-\frac{3\lambda_1^2(\mathbf{L}_{\text{P}}^{-1})}{2^{4q+1}M^2}\right), \tag{A.12}$$

where (A.12) follows (4.29).

With union bound, we derive the total error probability of the system as

$$P_{\text{e}} = \Pr\left((\hat{\mathbf{c}}_1, ..., \hat{\mathbf{c}}_M) \neq (\mathbf{c}_1, ..., \mathbf{c}_M)\right) \tag{A.13}$$

$$\leq \sum_{m=1}^{M} P_{\text{e},m} = 4M\exp\left(-\frac{3\lambda_1^2(\mathbf{L}_{\text{P}}^{-1})}{2^{4q+1}M^2}\right), \tag{A.14}$$

which completes the proof.

Bibliography

[1] J. Richter, "Compute-and-forward in multi-user relay networks: optimization, implementation, and secrecy," Ph.D. dissertation, Technische Universität Dresden, Faculty of Electrical and Computer Engineering, 4 2017.

[2] M. D. Plumbley, "Geometrical methods for non-negative ICA: manifolds, Lie groups and toral subalgebras," *Neurocomputing*, vol. 67, pp. 161–197, 2005.

[3] A. Sakzad and E. Viterbo, "Full diversity unitary precoded integer-forcing," *IEEE Trans. on Wireless Comm.*, vol. 14, no. 8, pp. 4316–4327, Aug 2015.

[4] "Cisco visual networking index: Global mobile data traffic forecast update, 2017–2022," *Cisco Inc. White Paper*, Feb 2019.

[5] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. on Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul 2000.

[6] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proc. of the 12th Annual Int. Conf. Mobile Computing and Networking*, ser. MobiCom '06. New York, NY, USA: ACM, 2006, pp. 358–365.

[7] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 438–460, Mar. 2011.

[8] P. Popovski and H. Yomo, "Physical network coding in two-way wireless relay channels," in *2007 IEEE Int. Conf. on Comm.*, June 2007, pp. 707–712.

[9] J. Sykora and A. Burr, *Wireless Physical Layer Network Coding*, 1st ed. Cambridge University Press, 2018.

[10] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: Analog network coding," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, p. 397–408, Aug. 2007. [Online]. Available: https://doi.org/10.1145/1282427.1282425

[11] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "Xors in the air: Practical wireless network coding," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, p. 497–510, Jun. 2008. [Online]. Available: https://doi.org/10.1109/TNET.2008.923722

[12] S. C. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Physical Communication*, vol. 6, pp. 4 – 42, 2013, network Coding and its Applications to Wireless Communications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874490712000419

[13] C. Feng, D. Silva, and F. R. Kschischang, "An algebraic approach to physical-layer network coding," *IEEE Trans. on Inf. Theory*, vol. 59, no. 11, pp. 7576–7596, Nov. 2013.

[14] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. on Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.

[15] B. M. Kurkoski, "Encoding and indexing of lattice codes," *IEEE Trans. on Inf. Theory*, vol. 64, no. 9, pp. 6320 – 6332, Sept. 2018.

[16] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. on Inf. Theory*, vol. 48, no. 6, pp. 1250–1276, Jun. 2002.

[17] U. Erez and R. Zamir, "Achieving 1/2 log (1+SNR) on the AWGN channel with lattice encoding and decoding," *IEEE Trans. on Inf. Theory*, vol. 50, no. 10, pp. 2293–2314, Oct. 2004.

[18] L. Wei and W. Chen, "Efficient compute-and-forward network codes search for two-way relay channel," *IEEE Comm. Letters*, vol. 16, no. 8, pp. 1204–1207, August 2012.

[19] M. P. Wilson, K. Narayanan, H. D. Pfister, and A. Sprintson, "Joint physical layer coding and network coding for bidirectional relaying," *IEEE Trans. on Inf. Theory*, vol. 56, no. 11, pp. 5641–5654, Nov. 2010.

[20] M. E. Soussi, A. Zaidi, and L. Vandendorpe, "Compute-and-forward on a multiaccess relay channel: Coding and symmetric-rate optimization," *IEEE Trans. on Wireless Comm.*, vol. 13, no. 4, pp. 1932–1947, Apr. 2014.

[21] M. N. Hasan and B. M. Kurkoski, "Practical compute-and-forward approaches for the multiple access relay channel," in *2017 IEEE Int. Conf. on Comm. (ICC)*, May 2017, pp. 1–6.

[22] X. Insausti, A. Sáez, and P. M. Crespo, "A novel scheme inspired by the compute-and-forward relaying strategy for the multiple access relay channel,"

Wireless Networks, vol. 25, no. 2, pp. 665–673, Feb 2019. [Online]. Available: https://doi.org/10.1007/s11276-017-1583-1

[23] J. Zhu and M. Gastpar, "Gaussian multiple access via compute-and-forward," *IEEE Trans. on Inf. Theory*, vol. PP, no. 99, pp. 1–1, 2016.

[24] R. H. Y. Louie, Y. Li, and B. Vucetic, "Practical physical layer network coding for two-way relay channels: performance analysis and comparison," *IEEE Trans. on Wireless Comm.*, vol. 9, no. 2, pp. 764–777, February 2010.

[25] I. . Baik and S. . Chung, "Network coding for two-way relay channels using lattices," in *2008 IEEE Int. Conf. on Comm.*, May 2008, pp. 3898–3902.

[26] J. Zhan, B. Nazer, U. Erez, and M. Gastpar, "Integer-forcing linear receivers," *IEEE Trans. on Inf. Theo.*, vol. 60, pp. 7661–7685, 2014.

[27] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*, 1st ed., ser. Graduate Texts in Mathematics. Cambridge University Press, 2005.

[28] O. Ordentlich and U. Erez, "Precoded integer-forcing universally achieves the MIMO capacity to within a constant gap," *IEEE Trans. on Inf. Theory*, vol. 61, no. 1, pp. 323–340, Jan 2015.

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

[30] S. K. Mohammed, E. Viterbo, Y. Hong, and A. Chockalingam, "MIMO precoding with X- and Y-codes," *IEEE Trans. on Inf. Theory*, vol. 57, no. 6, pp. 3542–3566, June 2011.

[31] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, 3rd ed. New York, NY, USA: Springer-Verlag, 1999, ISBN 0-387-98585-9.

[32] V. Ntranos, V. R. Cadambe, B. Nazer, and G. Caire, "Asymmetric compute-and-forward," in *2013 51st Annual Allerton Conf. on Comm., Control, and Computing (Allerton)*, Oct 2013, pp. 1174–1181.

[33] B. Nazer, V. R. Cadambe, V. Ntranos, and G. Caire, "Expanding the compute-and-forward framework: Unequal powers, signal levels, and multiple linear combinations," *IEEE Trans. on Inf. Theory*, vol. 62, no. 9, pp. 4879–4909, Sept. 2016.

[34] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proceedings of the 12th Annual Int. Conf. on Mobile Computing and Networking*, ser. MobiCom '06. New York, NY, USA: ACM, 2006, pp. 358–365. [Online]. Available: http://doi.acm.org/10.1145/1161089.1161129

[35] T. Cover and A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. on Inf. Theory*, vol. 25, no. 5, pp. 572–584, Sept. 1979.

[36] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. on Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[37] A. E. Gamal and Y.-H. Kim, *Network Information Theory.* New York, NY, USA: Cambridge University Press, 2012.

[38] G. Kramer and A. J. van Wijngaarden, "On the white Gaussian multiple-access relay channel," in *Inf. Theory, 2000. Proc. IEEE Int. Symp. on*, 2000, p. 40.

[39] C. Hausl and P. Dupraz, "Joint network-channel coding for the multiple-access relay channel," in *2006 3rd Annual IEEE Comm. Society on Sensor and Ad Hoc Comm. and Net.*, vol. 3, Sept. 2006, pp. 817–822.

[40] D. H. Woldegebreal and H. Karl, "Multiple-access relay channel with network coding and non-ideal source-relay channels," in *2007 4th Int. Symp. on Wireless Comm. Sys.*, Oct 2007, pp. 732–736.

[41] S. Wei, J. Li, W. Chen, L. Zheng, and H. Su, "Design of generalized analog network coding for a multiple-access relay channel," *IEEE Trans. on Comm.*, vol. 63, no. 1, pp. 170–185, Jan 2015.

[42] J. He, V. Tervo, S. Qian, Q. Xue, M. Juntti, and T. Matsumoto, "Performance analysis of lossy decode-and-forward for non-orthogonal MARCs," *IEEE Trans. on Wireless Comm.*, vol. 17, no. 3, pp. 1545–1558, March 2018.

[43] J. Wannstrom, "LTE-advanced," 2013. [Online]. Available: http://www.3gpp.org/technologies/keywords-acronyms/97-lte-advanced

[44] H. Holma and A. Toskala, *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*, 1st ed. USA: John Wiley & Sons, Inc., 2000.

[45] R. Youssef and A. G. i. Amat, "Distributed serially concatenated codes for multi-source cooperative relay networks," *IEEE Trans. on Wireless Comm.*, vol. 10, no. 1, pp. 253–263, January 2011.

[46] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, vol. 44, pp. 463–471, 1985.

[47] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Program.*, vol. 66, no. 2, pp. 181–199, Sep. 1994.

[48] A. K. Lenstra, H. W. Lenstra, and L. Lovasz, “Factoring polynomials with rational coefficients,” *MATH. ANN*, vol. 261, pp. 515–534, 1982.

[49] E. Viterbo and J. Boutros, “A universal lattice code decoder for fading channels,” *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1639–1642, Jul 1999.

[50] L. Ding, K. Kansanen, Y. Wang, and J. Zhang, “Exact SMP algorithms for integer-forcing linear MIMO receivers,” *IEEE Trans. on Wireless Comm.*, vol. 14, no. 12, pp. 6955–6966, Dec 2015.

[51] J. Wen, L. Li, X. Tang, and W. H. Mow, “An efficient optimal algorithm for the successive minima problem,” *IEEE Trans. on Comm.*, vol. 67, no. 2, pp. 1424–1436, Feb 2019.

[52] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, “Closest point search in lattices,” *IEEE Trans. on Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug 2002.

[53] W. Liu and C. Ling, “Efficient integer coefficient search for compute-and-forward,” *IEEE Trans. on Wireless Comm.*, vol. 15, no. 12, pp. 8039–8050, Dec. 2016.

[54] A. Sakzad, J. Harshan, and E. Viterbo, “Integer-forcing MIMO linear receivers based on lattice reduction,” *IEEE Trans. on Wireless Comm.*, vol. 12, no. 10, pp. 4905–4915, Oct. 2013.

[55] B. Bai, W. Chen, K. B. Letaief, and Z. Cao, “Outage exponent: A unified performance metric for parallel fading channels,” *IEEE Transactions on Information Theory*, vol. 59, no. 3, pp. 1657–1677, 2013.

[56] M. Vu and A. Paulraj, “MIMO wireless linear precoding,” *IEEE Sig. Proc. Magazine*, vol. 24, no. 5, pp. 86–105, Sept 2007.

[57] B. Vrigneau, J. Letessier, P. Rostaing, L. Collin, and G. Burel, “Extension of the MIMO precoder based on the minimum Euclidean distance: A cross-form matrix,” *IEEE Jour. of Sel. Topics in Sig. Proc.*, vol. 2, no. 2, pp. 135–146, April 2008.

[58] E. Bayer-Fluckiger, F. Oggier, and E. Viterbo, “New algebraic constructions of rotated Zn-lattice constellations for the Rayleigh fading channel,” *IEEE Trans. Inf. Theory*, vol. 50, no. 4, pp. 702–714, 2006.

[59] A. Edelman, T. A. Arias, and S. T. Smith, “The geometry of algorithms with orthogonality constraints,” *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, pp. 303–353, Apr. 1999.

[60] M. R. Bremner, *Lattice Basis Reduction: An Introduction to the LLL Algorithm and Its Applications*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2011.

[61] T. E. Abrudan, J. Eriksson, and V. Koivunen, "Steepest descent algorithms for optimization under unitary matrix constraint," *IEEE Trans. on Sig. Proc.*, vol. 56, no. 3, pp. 1134–1147, March 2008.

[62] R. C. Raffenetti and K. Ruedenberg, "Parametrization of an orthogonal matrix in terms of generalized eulerian angles," *Int. Journal of Quantum Chemistry*, vol. III S, pp. 625–634, 1970.

[63] Y. H. Gan, C. Ling, and W. H. Mow, "Complex lattice reduction algorithm for low-complexity full-diversity MIMO detection," *IEEE Trans. on Signal Processing*, vol. 57, no. 7, pp. 2701–2710, Jul. 2009.

[64] S. Lyu and C. Ling, "Boosted KZ and LLL algorithms," *IEEE Trans. on Sig. Proc.*, vol. 65, no. 18, pp. 4784–4796, Sep. 2017.

[65] J. Wen, L. Li, X. Tang, W. H. Mow, and C. Tellambura, "An efficient optimal algorithm for integer-forcing linear MIMO receivers design," in *2017 IEEE Int. Conf. on Comm. (ICC)*, May 2017, pp. 1–6.

[66] W. Banaszczyk, "New bounds in some transference theorems in the geometry of numbers," *Mathematische Annalen*, vol. 296, no. 4, p. 625–635, 1993.

[67] B. C. Hall, *Lie Groups, Lie Algebras, and Representations*, 2nd ed., ser. Grad. Texts in Math. Springer Int. Publishing, 2015, vol. 222.

[68] C. Moler and C. V. Loan, "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later," *SIAM Review*, vol. 45, no. 1, pp. 3–000, 2003.

[69] A. Fischer, "Structure of Fourier exponents of almost periodic functions and periodicity of almost periodic functions," *Mathematica Bohemica*, vol. 121, pp. 249–262, 1996.

[70] T. Abrudan, J. Eriksson, and V. Koivunen, "Conjugate gradient algorithm for optimization under unitary matrix constraint," *Signal Processing*, vol. 89, no. 9, pp. 1704 – 1714, 2009.

[71] J. Wen, B. Zhou, W. H. Mow, and X. W. Chang, "An efficient algorithm for optimally solving a shortest vector problem in compute-and-forward design," *IEEE Trans. on Wireless Comm.*, vol. 15, no. 10, pp. 6541–6555, Oct. 2016.

[72] Z. B. Zabinsky, "Random search algorithms," *Technical Report*, 2009.

[73] R. J.-B. W. Francisco J. Solis, "Minimization by random search techniques," *Math. of Operations Res.*, vol. 6, no. 1, pp. 19–30, Feb 1981.

[74] F. Mezzadri, "How to generate random matrices from the classical compact groups," *Notices of the AMS*, vol. 54, pp. 592–604, 2007.

[75] "List of full diversity algebraic rotations." [Online]. Available: https://ecse.monash.edu/staff/eviterbo/rotations/rotations.html

[76] A. Lapidoth and M. Wigger, "On the AWGN MAC with imperfect feedback," *IEEE Trans. on Inf. Theory*, vol. 56, no. 11, pp. 5432–5476, 2010.

[77] C. Li, "Capacity-achieving codes for noisy feedback channels: A necessary condition," *IEEE Comm. Letters*, vol. 18, no. 6, pp. 1007–1010, 2014.

[78] M. Gastpar and G. Kramer, "On cooperation via noisy feedback," in *2006 Int. Zurich Seminar on Comm.*, 2006, pp. 146–149.

[79] K. Azarian, H. El Gamal, and P. Schniter, "On the optimality of the arq-ddf protocol," *IEEE Trans. on Inf. Theory*, vol. 54, no. 4, pp. 1718–1724, 2008.

[80] C. Feng, D. Silva, and F. R. Kschischang, "Blind compute-and-forward," *IEEE Trans. on Comm.*, vol. 64, no. 4, pp. 1451–1463, April 2016.

List of Publications

1. **M. N. Hasan**, B. M. Kurkoski, A. Sakzad, and E. Viterbo, "Steepest gradient-based orthogonal precoder for integer-forcing MIMO," *IEEE Transactions on Wireless Communications*, Vol. 2, no. 2, pp. 942-955, Feb. 2020.

2. **M. N. Hasan**, B. M. Kurkoski, "Opportunistic Cooperation Strategies for Multiple Access Relay Channels with Compute-and-Forward," *IEEE Transactions on Vehicular Technology*, Aug. 2020. (submitted)

3. **M. N. Hasan**, B. M. Kurkoski, A. Sakzad, and E. Viterbo, "Orthogonal precoder for integer-forcing MIMO," in *2019 IEEE International Symposium on Information Theory (ISIT)*, July 2019, pp. 1187–1191.

4. **M. N. Hasan** and B. M. Kurkoski, "An efficient strategy for applying compute-and-forward to the MARC," in *2018 International Symposium on Information Theory and Its Applications (ISITA)*, Oct 2018, pp. 149–153.

5. **M. N. Hasan** and B. M. Kurkoski, "Practical compute-and-forward approaches for the multiple access relay channel," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

6. **M. N. Hasan** and B. M. Kurkoski, "Compute-and-forward for the multiple access relay channel with feedback," in *2016 Symposium on Information Theory and Its Applications (SITA)*, no. 914, Dec 2016, pp. 538–543.