Applications of Lattices over Wireless Channels

by

Hossein Najafi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2012

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In wireless networks, reliable communication is a challenging issue due to many attenuation factors such as receiver noise, channel fading, interference and asynchronous delays. Lattice coding and decoding provide efficient solutions to many problems in wireless communications and multiuser information theory. The capability in achieving the fundamental limits, together with simple and efficient transmitter and receiver structures, make the lattice strategy a promising approach. This dissertation deals with problems of lattice detection over fading channels and time asynchronism over the lattice-based compute-and-forward protocol.

In multiple-input multiple-output (MIMO) systems, the use of lattice reduction significantly improves the performance of approximate detection techniques. In the first part of this thesis, by taking advantage of the temporal correlation of a Rayleigh fading channel, low complexity lattice reduction methods are investigated. We show that updating the reduced lattice basis adaptively with a careful use of previous channel realizations yields a significant saving in complexity with a minimal degradation in performance. Considering high data rate MIMO systems, we then investigate soft-output detection methods. Using the list sphere decoder (LSD) algorithm, an adaptive method is proposed to reduce the complexity of generating the list for evaluating the log-likelihood ratio (LLR) values. To form the list of appropriate candidates, the temporal correlation of channel is used beside the lattice structure of the system.

In the second part of this thesis, by applying the lattice coding and decoding schemes over asynchronous networks, we study the impact of asynchronism on the compute-and-forward strategy. While the key idea in compute-and-forward is to decode a linear synchronous combination of transmitted codewords, the distributed relays receive random asynchronous versions of the combinations. Assuming different asynchronous models, we design the receiver structure prior to the decoder of compute-and-forward so that the achievable rates are maximized at any signal-to-noise-ratio (SNR). Finally, we consider symbol-asynchronous X networks with single antenna nodes over time-invariant channels. We exploit the asynchronism among the received signals in order to design the interference alignment scheme. It is shown that the asynchronism provides correlated channel variations which are proved to be sufficient to implement the vector interference alignment over the constant X network.

Acknowledgements

I would like to express my deep gratitude to my supervisor, Professor Mohamed Oussama Damen, for his end-less support, guidance, understanding, and great supervision. I have learned many valuable lessons from him not only about research but also about life.

I wish to express my sincere gratitude and thanks to the members of my dissertation committee, Professors Abbas Yongacoglu, Liang Liang Xie, Murat Uysal, and Changbao Wu and also Professor Guang Gong for attending my Ph.D. examination as the delegate of Professor Murat Uysal.

I would like to express my thanks to my colleagues Mehdi Torbatian, Erfan Danesh Jafari and Walid Abediseid for the interesting discussions and talks. I would also like to thank all my friends and colleagues in University of Waterloo, specially Siamak Fouldi, Ghasem Razavi, Akbar Ghasemi, Babak Alipanahi, Javad Behrouzi, Mahdi Zamani, Hadi Amarloo, from whom I learnt a lot during the years of PhD and shared many happy moments of my life.

Dedicated to my love *Bahar*

and

To my parents

Table of Contents

List of Figures

List of Abbreviations

AWGN	Additive White Gaussian Noise
BF	Beamforming
CDF	Cumulative Distribution Function
CLPS	Closest Lattice Point Search
CPS	Cyclic Prefix Symbols
CSI	Channel State Information
DFE	Decision Feedback Equalizer
DFT	Discrete Fourier Transform
DOF	Degrees of Freedom
IDFT	Inverse Discrete Fourier Transform
ISI	Inter Symbol Interference
LLL	Lenstra-Lenstra-Lovász
LLR	Log-Likelihood Ratio
LOS	Line of Sight
LSD	List Sphere Decoders
MAC	Multiple-Access Channel
MAP	Maximum A Posteriori Probability
MIMO	Multiple-Input Multiple-Output
ML	Maximum-Likelihood

MMSE	Minimum Mean Square Error
OFDM	Orthogonal Frequency Division Multiplexing
PAM	Pulse Amplitude Modulation
QAM	Quadrature Amplitude Modulation
SISO	Soft-Input Soft-Output
SNR	Signal to Noise Ratio
ZF	Zero Forcing

Notations

$\mathbf{h}$	Boldface lower-lase letters denote vectors
$\mathbf{H}$	Boldface upper-case letters denote matrices
$\mathbf{h}^T, \mathbf{H}^T$	Transpose of $\mathbf{h}, \mathbf{H}$
$\mathbf{h}^H, \mathbf{H}^H$	Transpose conjugate of $\mathbf{h}, \mathbf{H}$
$\|\mathbf{h}\|, \|\mathbf{H}\|$	The 2-norm of $\mathbf{h}, \mathbf{H}$
`span` $\mathbf{H}$	Vector space spanned by matrix $\mathbf{H}$
`rank` $\mathbf{H}$	Rank of matrix $\mathbf{H}$
`trace` $\mathbf{H}$	Trace of matrix $\mathbf{H}$
`det` $\mathbf{H}$	Determinant of matrix $\mathbf{H}$
`diag`$\{\cdot\}$	A diagonal matrix of arguments
$\mathbf{I}_n$	Identity matrix of dimension n
$\mathbf{H} \otimes \mathbf{G}$	The Kronecker product of two matrices $\mathbf{H}$ and $\mathbf{G}$
$\mathbb{R}^N$	N-dimensional real space
$\mathbb{Z}^N$	N-dimensional integer space
$\log(\cdot)$	Logarithm in base 2
$\ln(\cdot)$	Natural logarithm
$E\{\cdot\}$	Expectation operation
$\Pr(\cdot)$	Probability of an event
$\Re\{\cdot\}$	The real part of a matrix or a vector

$\Im\{\cdot\}$	The imaginary part of a matrix or a vector
$[x]^+$	$\max\{0, x\}$

Chapter 1

Introduction

Wireless communication is of major importance to the world economy and the way people live in today's interconnected world. The growing demand for high data rate services has urged a massive research efforts to analyze the performance of various wireless communications systems. Considering the limited resources of bandwidth and power, designing efficient techniques for reliable communications is the major motivation for these studies. The central bottlenecks for reliable communication over a wireless network are the additive noise experienced in different receiver nodes across the network and the signal variations produced by the wireless medium in terms of channel fading and interference from the other users communications. Therefore, designing techniques that can increase the efficiency of communications are of major importance. An interesting approach to increase the data rate in a wireless system without increasing bandwidth or power is to use multiple antennas at different nodes. It is well established that the use of multiple antennas at both the transmitter and the receiver side, multiple-input multiple-output (MIMO) system, creates multiple transmission channels and allows for simultaneous increase in data rate (multiplexing gain) and reliability (diversity gain). To be more specific, in a system with M transmitter antennas and N receiver antennas, one can increase the transmission reliability up to MN and the data rate up to $\min(M, N)$

[1, 2]. Signal processing techniques that allow us to exploit the potential of MIMO systems are called space-time codes [2, 3]. In fact, the space-time codes can be used to decrease the error rate and increase the transmission rate while satisfying a fundamental diversity-multiplexing gain tradeoff (DMT) that is specific to the physical medium and the applied protocol [4].

Communication strategies based on lattice codes at the transmitters and/or lattice decoders at the receivers provide solutions to different multiuser problems in information theory. The capability in achieving the fundamental limits, together with simple and efficient transmitter and receiver structures, make the lattice strategy a promising approach [5–7]. Lattice codes were first proposed as an alternative to random Gaussian codes in order to achieve the capacity of the additive white Gaussian noise channel (AWGN) [8–11]. As an important progress, it was shown that lattice code combined with lattice decoding can achieve the capacity of a point to point AWGN channel [12]. Then, applications in a wide variety of channels such as MIMO channel [13] and multiterminal channels [14] were proposed. Note that the capacity achieving results based on the lattice codes are provided in spite of the restrictions that may come from the structure of code compared to the random case. Moreover, it is revealed in several examples recently that structured codes provide strictly better result than random codes ([15] and references therein). Lattice codes are also used for interference management where it was first shown that structured interference space may be more useful [16]. In fact, instead of limiting the desired signal to the interference-free dimensions on a random space, one can exploit the structured interference space and transmit signals such that the desired and the interference levels align differently at each receiver [17–19].

Lattice decoding is an important application of lattices over MIMO systems where the outputs can be represented as a linear combination of the inputs corrupted by an additive noise. In point to point MIMO systems, decoding represents a challenging

problem. Methods from lattice theory has been used as solutions to this challenging problem [20–22]. In fact, the MIMO detection can be translated to the closest lattice point search (CLPS) problem in the lattice theory. On the other hand, lattice reduction methods have proved themselves to be powerful tools in approximating the CLPS problem [23, 24]. Moreover, it is shown that the use of lattice reduction methods significantly improves the performance of suboptimal detection algorithms, [25] and [26]. The LLL algorithm introduced by Lenstra, Lenstra, and Lovász, [27] is the most widely used lattice reduction method due to its efficiency in finding near orthogonal vectors with short norms. Generally, the complexity of the reduction algorithm in a lattice reduction aided detection method can be ignored in a quasi-static scenario where the channel is considered fixed for the whole transmission frame or where the channel variations are slow enough which makes it possible to use the same result for quite a large number of received signals (e.g., [28]). However, in the practical scenarios, channel coefficients vary throughout the frame and have temporal correlation. In other words, the frame length over which the channel can be assumed to be constant is small. Therefore, the complexity of lattice reduction need to be considered in MIMO detection methods. Over correlated fading channels where the channel coefficients slowly vary through the frame, this problem is addressed in Chapter 2. Exploiting the temporal correlation, adaptive methods are proposed to reduce the computational complexity of lattice reduction. Applying high data rate MIMO systems, serially concatenated with a channel code as the outer code ([29], [30] and [31]), we then extend the adaptive algorithms for soft-output detection methods. To form the list of appropriate candidates in an adaptive manner, the temporal correlation of channels is used beside the lattice structure of the MIMO system.

Linear properties of lattices provide many applications for interference management over wireless channels. Dealing with interference and noise over wireless relay networks

are two challenging problems which has been addressed by various relaying strategies. Mostly, the proposed schemes try to overcome the problems by performing one of the following two actions [32–34]. In the first strategy, such as decode-and-forward, the intermediate nodes try to totally remove the noise. Although this solves one of the problems, the network becomes interference-limited. In the second approach, the intermediate nodes try to repeat the transmitted signal (amplify-and-forward) or quantize the observed signal (compress-and-forward) and then pass it towards the destination in order to form a large multi-antenna channel. However, not performing the decoding results in noise accumulation. A new approach referred to as compute-and-forward is proposed in [15] to efficiently manage the interference and remove the noise at the relay nodes. In the compute-and-forward scheme, the same nested lattice codes are applied at the transmitters. With lattice codes, any integer combination of the transmitted codewords is still a codeword. Thus, the intermediate node can decode the combination using a lattice decoder and remove the noise. Therefore the relays are able to recover integer linear functions of codewords and forward a noiseless version of the transmitted signals to the destination. It is shown in [15] that decoding a linear function of transmitted messages provides an opportunity to achieve higher rates over a network with additive white Gaussian noise. On the other hand, the nested lattice implementation of compute-and-forward relies on the algebraic structure of the applied codes to decode a synchronous linear combination of the transmitted messages. Hence, time synchronization is an important assumption in achieving the promising gains. However, because of the distributed nature of the relays across the network, random asynchronous combination of the transmitted signals are received and thus, perfect synchronization is not feasible. Different types of time asynchronism have been studied over AWGN channel that show different performance characteristics. For instance, it is shown in [35] that the symbol-asynchronism does not have an impact on the capacity region of the two-user Gaussian multiple-access channel (MAC) with identical shaping waveform at the

transmitters. In [36], it is shown that higher mutual information is achievable in an asynchronous space-time coded system with appropriate shaping waveforms. In [37], it is verified that the total capacity of memoryless MAC channel under the frame-asynchronous assumption remains unchanged while it is significantly affected for the channels with memory. Moreover, asynchronism can significantly degrade the cooperative system performance if it is not dealt with appropriately [38], [39]. Therefore, the impact of asynchronism on the system performance is an important issue both in practice and in theory and needs to be carefully investigated. Hence, the effect of asynchronous delays on the compute-and-forward rates is studied in Chapter 3.

The interference channel is a network communication model in which pairs of transmitters and receivers use the same communication medium at the same time and frequency. Therefore, each receiver's signal is corrupted by the interference from the transmitted signals intended for other receivers. The X channel is a general interference network in which there is an independent message from every transmitter to each receiver. Characterizing the capacity region of the interference and X channels are very challenging problems in information theory. The problem of locating this region, even in a simple case of two-user Gaussian channel, is still unsolved [40–42]. A fundamental measure to approximate the capacity of wireless channels at high values of SNR is known as the degrees of freedom (DOF) which determines the capacity boundaries as the SNR increases. Since it impacts the design of efficient methods of interference management, investigating the DOF of distributed networks is an important issue in wireless communications. Interference alignment is an intelligent way to manage the interference at the receivers by restricting the undesired signals at some common directions. It was first introduced in [43] wherein its capability in achieving the total number of degrees of freedom of a class of two-user X channels was studied. It is shown in [43] that in a two-user X channel with M antennas at each node, a total of $\lfloor \frac{4M}{3} \rfloor$ degrees of freedom

is achievable. This interesting result was then improved in [44] by employing the idea of channel extensions to achieve the total $4M/3$ degrees of freedom almost surely over channels with constant coefficients. Interference alignment was then applied to the K-user interference channel in [45] and for the $M \times N$ user X network in [46]. It is shown in [46] that by using the symbol extension over the channel with varying coefficients, the upper bound of the total number of degrees of freedom is achievable for the general X network with single antenna nodes. However, the channel variation is a crucial assumption for the mentioned achievable schemes. Thus, these schemes could not trivially be extended to channels with constant coefficients. On the other hand, perfectly synchronized nodes is an important assumption in the aforementioned alignment schemes. As mentioned before, due to the distributed nature of the X network, perfect synchronization is not feasible in many cases. Exploiting the symbol-asynchronous scheme in Chapter 3, we finally investigate the vector interference alignment scheme over time-invariant single antenna X network.

1.1 Summary of the Dissertation

In Chapter 2, we consider the problem of lattice reduction aided MIMO detection over fading channels. Taking advantage of the temporal correlation of a Rayleigh fading channel, new methods are proposed to reduce the computational complexity. Lattice reduction is investigated in the first part of the chapter by adaptively updating the reduced lattice basis: instead of performing lattice reduction on each channel realization, we consider two methods that utilize temporal correlation to reduce complexity. The main idea is to use the results of past channel realizations to perform an efficient reduction of the new one. In the first method, lattice reduction is applied on the nearly reduced channel matrix which is computed from the previous realization. It is shown that this method achieves an error performance the same as the original lattice reduc-

tion method but significantly reduces the computational complexity. For more saving in computational complexity, the part that performs the lattice reduction algorithm on the nearly reduced channel matrix can be removed. In other words, we can use the same transformation matrix to reduce the latter channel matrices if the variations in the channel are small enough. Based on this idea, the second method is introduced by using an updating measure that offers a tradeoff between performance and complexity. It is shown that the second method achieves the maximum receive diversity in an uncoded MIMO system if the updating measure is selected properly. The proposed adaptive methods can be used in conjunction with any lattice reduction algorithm and in any multi-antenna scenario over correlated channels that requires lattice reduction, such as MIMO detection and broadcast precoding. In the second part, we propose an adaptive method for soft-output MIMO detection. It has been shown that a serially concatenated scheme of MIMO channel with a channel code as the outer code, can provide impressive performance for high data rate MIMO communications (e.g., [29], [30], and [31]). In these systems, the iterative joint detection and decoding can be performed by employing a soft-input, soft-output (SISO) decoder for the outer code and exchanging the soft information between the MIMO detector and the SISO decoder. However, the maximum *a posteriori* probability (MAP) detector is not affordable even for moderate MIMO dimensions. To avoid the large complexity, sub-optimal detectors based on the practical list of candidates, have been proposed. List sphere decoders (LSD) are generally used for such a list construction. In [29], the list construction is performed by a modified sphere decoder and the lattice points with the smallest distances to received point are found within a centered sphere on the received point. To have a more stable spherical list decoder, in the sense of the list size and the search radius selection, a new LSD was proposed in [47]. In this case, at the first step, the ML point is found by a sphere decoder. Then, by using the ML point as the center of another sphere decoder, the list of best candidates are formed. In the second part of Chapter 2, we introduce an adap-

tive version of the stable list construction algorithm over slowly varying fading channels. The results of the previous channel realization is utilized for performing the detection in the new channel realization. We use the adaptive hard-output detector as a low complexity near-ML approximation. Then, this near-ML point is used for the initiation of the list construction. After that, for finding the list of appropriate candidates around this point, the temporal correlation of channels is used beside the lattice structure of the MIMO systems and the list is formed in an adaptive manner. Employing a performance/complexity tradeoff measure, the list of best candidates at the MIMO channel output is shifted and updated from a past channel list.

In Chapter 3, we study the impact of asynchronism on the compute-and-forward strategy. The lattice coding and decoding scheme is applied over asynchronous networks. While the key idea in compute-and-forward is to decode a linear synchronous combination of transmitted codewords, the relays receive random asynchronous versions of the combinations. Therefore, we study the effect of asynchronous delays on the compute-and-forward rate. We assume that the transmitters are not aware of the asynchronous delays at different relays and the coder and decoder structures are kept identical to the synchronous compute-and-forward scheme. At the first part, we consider the symbol-asynchronism model in which the delays are assumed to be less than a symbol interval. This model has been studied for the multiple-access channel in [35] and used over simple relay networks in [36] and [48]. We show that the inter-symbol-interference (ISI), resulted from the asynchronism, imposes additional interference at each relay. If this asynchronous interference is considered as noise, it results in an interference-limited system and the channel output scaling in the compute-and-forward scheme is not effective anymore specially at high SNRs. Therefore, it is useful to remove the asynchronous interference from the received signal and provide an equivalent interference-free model. Based on this idea, a whitening filter is used at the output of the channel to provide a

synchronous combination of the transmitted sequences for the decoder of compute-and-forward, but with the cost of reduced channel gain. It is shown that this procedure is equivalent to extracting the synchronous part of the received signal. Using the equalizer output with less channel gain results in a gap compared to the synchronous rates but it vanishes at high SNRs. A numerical example for the MAC channel is also presented and it is shown that a simple 1-bit feedback to one of the transmitters fills almost half of the gap in the compute-and-forward rate for all SNRs.

At the second part, we consider a general asynchronous model called frame-asynchronism where delays are random multiples of a symbol interval. Over the MAC channel, the impact of frame-asynchronism on the capacity region has been investigated in the literature (cf. [37] and references therein). Also, by using the idea of similar channel codes at the transmitters over a three-node network coding scenario in [49], a practical decoder for the frame-asynchronous model is presented. Over fading channels, a similar model is considered in [50] for the interference channel and as a common scheme, it is called a line-of-sight (LOS) interference channel. For the compute-and-forward scenario, to be able to decode a synchronous sum of the transmitted codewords over the frame-asynchronous network, we propose to use multi-antenna receiver with the number of antennas equal to the number of transmitters. Multi-antenna receivers for a synchronous compute-and-forward relaying is studied in [51] where it is shown that one can rotate the channel coefficients toward integers to reduce the impact of the interference from the non-integer parts of the channel. We show that by using extra antennas at the relays, in addition to rotating the channel gains toward integers to reduce the impact of the interference from the non-integer parts of the channel, we can efficiently remove the asynchronous delays. By applying a linear filter whose structure is related to the integer delays prior to the decoder of compute-and-forward, we also maximize the achievable rate at any SNR.

At the third part of Chapter 3, we use the symbol-asynchronous scheme over a

constant X network with arbitrary number of single antenna nodes. By employing the asynchronism in the design of the interference alignment, we achieve the upper bound for the total number of degrees of freedom of this network which is argued to be the same as that of the synchronous case. Interference alignment over asynchronous networks based on propagation delays has been considered before for two user X channel and interference channels. It was first proposed in [45] as an example and then explored in [52] by proper node placement in a network with four nodes to align the interference signals. In [50], a K-user interference channel is modeled by a time indexed graph where the alignment task is associated with finding the maximal independent set of the graph. A signaling scheme for the interference alignment over the asynchronous K user interference channel was also proposed in [53] where it was shown that the total degrees of freedom of $K/2$ is achievable. In this thesis, by exploiting the asynchronous delays, achievable schemes are investigated for the general X network with constant channel coefficients. We consider an $M \times N$ symbol-asynchronous X network with single antenna users and study the effect of the asynchronism among the users on the total number of degrees of freedom of this network and show that is the same as that of the corresponding synchronous network. Then, we achieve the upper bound for the total number of degrees of freedom. The interference alignment scheme is implemented by using the asynchronous delays in the received signals at each receiver which results in ISI among the symbols from different transmitters and hence provides the channel variation required for the vector alignment.

Finally, in Chapter 4 we present a summary of the thesis contributions and discuss future work directions.

1.2 Lattice Backgrounds

In this section, we provide some basic definitions for lattices and the lattice basis reduction algorithms. A lattice , Λ, is a discrete set of points in Euclidean space $\mathbb{R}^n$ such that it is closed under ordinary vector addition and for any point in the lattice, the additive inverse is included in the set. In fact, any translate of the form $\Lambda + \mathbf{x}$ by a lattice point $\mathbf{x}$ is again a lattice. A lattice can be represented by a lattice generator matrix $\mathbf{H} \in \mathbb{R}^{n \times n}$. Let $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_n]$ be an $n \times n$ full-rank generator matrix. Then, the set of all linear combinations of columns by integer coefficients is a lattice, i.e.,

$$\Lambda = \{\mathbf{x} = \mathbf{H}\mathbf{z} : \mathbf{z} \in \mathbb{Z}^n\}. \tag{1.1}$$

Moreover, we call lattice Λ_1, the coarse lattice, is nested in lattice Λ_2, the fine lattice, if $\Lambda_1 \subseteq \Lambda_2$.

From the discrete structure of lattices, we can associate any lattice point with a bounded region in $\mathbb{R}^n$. A fundamental region of a lattice is a block which fills $\mathbb{R}^n$ by repeating it for just one lattice point in each copy. *Voronoi region* of a lattice is an important fundamental region. For each lattice point $\mathbf{x}$ in Λ, its Voronoi region $\mathcal{V}_\mathbf{x}$ consists of all points in the real space which are closer to it than any other lattice point. Specifically,

$$\mathcal{V}_\mathbf{x} = \{\mathbf{r} \in \mathbb{R}^n : \|\mathbf{x} - \mathbf{r}\| \leq \|\mathbf{y} - \mathbf{r}\|, \mathbf{y} \neq \mathbf{x} \in \Lambda\}. \tag{1.2}$$

All fundamental regions of a lattice with generator matrix $\mathbf{H}$ have the same volumes which is given by $\text{Vol}(\mathcal{V}) = \det(\mathbf{H})$ (or for a general non-square basis $\mathbf{H} \in \mathbb{R}^{n \times m}$, $\text{Vol}(\mathcal{V}) = \sqrt{\det(\mathbf{H}^T\mathbf{H})}$).

A *lattice nearest-neighbor quantizer* maps a real point $\mathbf{r} \in \mathbb{R}^n$ to the nearest lattice point. It is defined by

$$Q_\Lambda(\mathbf{r}) = \arg\min_{\boldsymbol{\lambda} \in \Lambda} \|\boldsymbol{\lambda} - \mathbf{r}\|. \tag{1.3}$$

Note that from the symmetry of lattice points, fundamental Voronoi region $\mathcal{V}$ of a lattice can be defined as the set of all points in $\mathbb{R}^n$ that are closest to the zero point as $\mathcal{V} = \mathcal{V}_{\mathbf{0}} = \{\mathbf{r} \in \mathbb{R}^n : Q(\mathbf{r}) = \mathbf{0}\}$.

The *modulo-lattice* function of $\mathbf{r} \in \mathbb{R}^n$ is defined as the nearest-neighbor quantization error:

$$[\mathbf{r}] \bmod \Lambda = \mathbf{r} - Q_\Lambda(\mathbf{r}). \tag{1.4}$$

The *second moment* of a lattice is defined per dimension as

$$\sigma_\Lambda^2 = \frac{1}{n} E[\|\mathbf{u}\|^2] = \frac{1}{n\mathrm{Vol}(\mathcal{V})} \int_{\mathcal{V}} \|\mathbf{r}\|^2 d\mathbf{r}, \tag{1.5}$$

where $\mathbf{u}$ is a random vector uniformly distributed over $\mathcal{V}$. The *normalized second moment* associated with Λ is given by

$$G(\Lambda) = \frac{\sigma_\Lambda^2}{\mathrm{Vol}(\mathcal{V})^{2/n}}. \tag{1.6}$$

1.2.1 Lattice Reduction

The goal of the lattice basis reduction is to find a new basis in which the columns of the new generator matrix $\mathbf{B}$ have small norms and they are as orthogonal as possible. The impact of lattice reduction in two-dimensional space can be seen in Fig. 1.1. This concept was proposed more than a century ago. There is no unique definition for lattice reduction. In fact, the basis for representing a lattice is not unique. The method for finding a basis for a lattice, which is composed of relatively short and nearly orthogonal vectors, is called lattice reduction. Minkowski proposed a definition in 1890s. Minkowski reduction is equivalent to finding the shortest vector in the lattice and there is no polynomial time algorithm known for this method. In 1982, Lenstra, Lenstra, and Lovász (LLL) [27] proposed a breakthrough algorithm for lattice reduction which has complexity of polynomial time in lattice dimension. A further improved version was developed by Schnorr

and Euchner [54] which is called deep insertion LLL (DILLL). This modification gives significantly shorter vector in comparison to the original LLL algorithm. However, the complexity of the deep insertion LLL in worst case can be exponential, but simulations show that on average it does not require much more iterations than the original LLL [55].

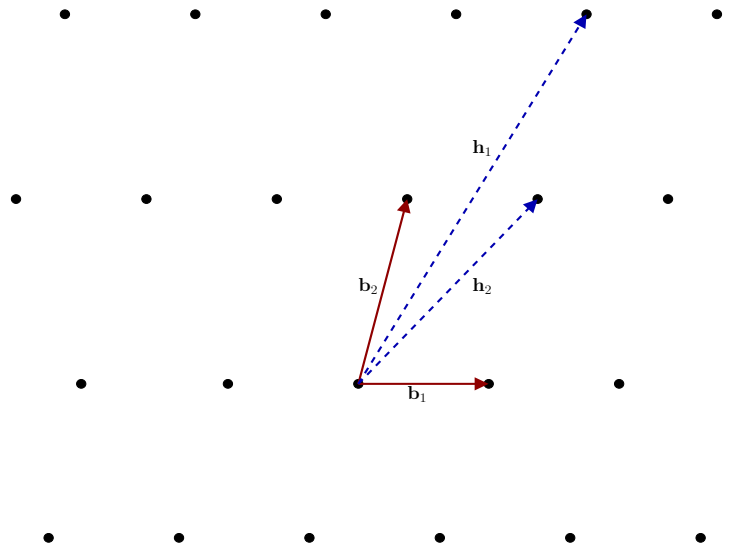


Figure 1.1: Reduction Effect, Solid line: Reduced basis, Dashed Line: Original basis

Definition 1.1. The basis $(\mathbf{b}_1, \cdots, \mathbf{b}_{i-1})$ is called LLL-reduced if

- $||\mu_{ij}|| \leqslant \frac{1}{2}$ for $1 \leqslant i < j \leqslant m$, and
- $p \cdot ||\mathbf{b}_i^*||^2 \leqslant ||\mathbf{b}_{i+1}^* + \mu_{i+1,i}\mathbf{b}_i^*||^2$

where $\mathbf{b}_i^*$s and μ_{ij}s are the Gram-Schmidt mutually orthogonal vectors and coefficients, respectively, and $\frac{1}{4} < p < 1$. Choosing larger values for constant p, results in a better reduction but also a higher complexity.

In the original LLL algorithm, to check if a basis is LLL reduced, only adjacent columns are checked against each other. One can argue that this condition can be extended so that the earlier columns are considered as well. This leads to a non-polynomial algorithm both in theory and practice. However, Schnorr and Euchner proposed the method DILLL to strengthen the condition without losing much practical speed [54].

To measure how reduced a basis is, *orthogonality defect factor* is used which is defined as follows:

$$\delta(\mathbf{B}) \triangleq \frac{(||\mathbf{b}_1||^2||\mathbf{b}_2||^2 \cdots ||\mathbf{b}_m||^2)}{\det \mathbf{B}^T\mathbf{B}}, \tag{1.7}$$

where $\mathbf{b}_i$'s are the columns of the basis $\mathbf{B}$. Clearly, $\delta(\mathbf{B}) \geq 1$ with equality for an orthogonal basis. In fact, the goal of lattice reduction is to determine a basis with smaller orthogonality defect factor. Therefore, for a lattice with bases $\mathbf{B}_1$ and $\mathbf{B}_2$, we can say that $\mathbf{B}_1$ is better reduced than $\mathbf{B}_2$ if $\delta(\mathbf{B}_1) < \delta(\mathbf{B}_2)$. When different bases of the same lattice are compared, the product of the norms can also be used because the determinant is equal for all of them. We define this product as

$$D(\mathbf{B}) \triangleq ||\mathbf{b}_1||||\mathbf{b}_2|| \cdots ||\mathbf{b}_m||. \tag{1.8}$$

Chapter 2

Adaptive Lattice Detection in MIMO Systems

2.1 Introduction

The application of multiple antenna for communications over wireless fading channels has attracted a large interest as they promise a large capacity increase. In point to point MIMO systems, decoding represents a challenging problem. Many researchers have used some of the methods from lattice theory as solutions to this challenging problem ([20], [21], [22] and [28] and the references therein). In fact, the outputs in MIMO systems can be described as a linear combination of the inputs corrupted by an additive noise. As a result, the detection is translated to the closest lattice point search (CLPS) problem in the lattice theory. Lattice reduction methods have proved themselves to be powerful tools in approximating the CLPS problem [23, 24]. Moreover, the use of lattice reduction methods significantly improves the performance of suboptimal detection algorithms, [25] and [26]. There exist different methods for lattice reduction. Among them, the LLL algorithm due to Lenstra, Lenstra, and Lovász, [27] is the most widely used due to its efficiency in finding near orthogonal vectors with short norms.

Generally, the complexity of using lattice reduction in a lattice reduction aided detection method can be ignored in a quasi-static scenario where the channel is considered fixed for the whole transmission frame or where the channel variations are slow enough which makes it possible to use the same result for quite a large number of received signals (e.g., [28] and references therein). But in this work, the channel is not assumed to be constant throughout the frame. We consider the practical scenario where channel coefficients slowly vary throughout the frame and have temporal correlation. This scenario is justified in many practical situations where the channel slowly varies through the frame. In other words, the frame length over which the channel can be considered constant is small.

Taking advantage of the temporal correlation of a Rayleigh fading channel, new methods are proposed to reduce the computational complexity. Lattice reduction is investigated in the first part of this chapter by adaptively updating the reduced lattice basis: instead of performing lattice reduction on each channel realization, we consider two methods that utilize temporal correlation to reduce complexity. The main idea is to use the results of past channel realizations to perform an efficient reduction of the new one. In the first method, lattice reduction is applied on the nearly reduced channel matrix which is computed from the previous realization. This method achieves an error performance the same as the original lattice reduction method but significantly reduces the computational complexity. For more saving in computational complexity, the part that performs the lattice reduction algorithm on the nearly reduced channel matrix can be removed. In other words, we can use the same transformation matrix to reduce the latter channel matrices if the variations in the channel are small enough (substituting lattice reduction algorithm by a matrix multiplication). Based on this idea, the second method is introduced by using an updating measure that offers a tradeoff between performance and complexity. Using the results of [56], it is shown that this

method achieves the maximum receive diversity in an uncoded MIMO system if the updating measure is selected properly. The proposed adaptive methods can be used in conjunction with any lattice reduction algorithm and in any multi-antenna scenario over correlated channels that requires lattice reduction, such as MIMO detection and broadcast precoding.

In the next part of this chapter, we extend the adaptive schemes to soft-output detection. It has been shown that a serially concatenated scheme of MIMO channel with a channel code as the outer code, can provide impressive performance for high data rate MIMO communications (e.g., [29], [30], and [31]). In these systems, the iterative joint detection and decoding can be performed by employing a soft-input, soft-output (SISO) decoder for the outer code and exchanging the soft information between the MIMO detector and the SISO decoder. However, the maximum *a posteriori* probability (MAP) detector is not affordable even for moderate MIMO dimensions. To avoid the large complexity, suboptimal detectors based on the practical list of candidates, have been proposed. List sphere decoders (LSD) are generally used for such a list construction. In [29], the list construction is performed by a modified sphere decoder and the lattice points with the smallest distances to received point are found within a centered sphere on the received point. To have a more stable spherical list decoder, in the sense of the list size and the search radius selection, a new LSD was proposed in [47]. In this case, at the first step, the ML point is found by a sphere decoder. Then, by using the ML point as the center of another sphere decoder, the list of best candidates are formed. Using this method and also the lattice linearity property, one can construct a shifted list over block fading channels. A list of candidates can be formed only once for each block around the origin (i.e., the all zero point of the corresponding lattice). Then, for the given received signal, the ML point is found at first and then a new list is generated by shifting the original list around the newly found ML point. Based on this idea, we

introduce an adaptive version of the stable list construction algorithm over slowly varying fading channels. Taking advantage of the temporal correlation, an adaptive scheme can be used in which the results of the previous channel realization is utilized for performing the detection in the new channel realization. We use the adaptive hard-output detector as a low complexity near-ML approximation. Then, this near-ML point is used for the initiation of the list construction. After that, for finding the list of appropriate candidates around this point, the temporal correlation of channels is used beside the lattice structure of the MIMO systems and the list is formed in an adaptive manner. Employing a performance/complexity tradeoff measure, the list of best candidates at the MIMO channel output is updated from a past channel list.

2.2 Lattice Reduction Aided Detection

2.2.1 System Model

Consider a MIMO system with M transmit and N receive antennas. If we assume $\mathbf{s}^c = [s_1^c, ..., s_M^c]^T$, $\mathbf{y}^c = [y_1^c, ..., y_N^c]^T$, $\mathbf{w}^c = [w_1^c, ..., w_N^c]^T$ and the $N \times M$ matrix $\mathbf{H}^c$ as the transmitted signal, the received signal, the noise vector and the channel matrix, respectively, then one has[1]

$$\mathbf{y}^c = \mathbf{H}^c \mathbf{s}^c + \mathbf{w}^c, \tag{2.1}$$

where the channel is assumed to be Rayleigh, i.e., the elements of $\mathbf{H}^c$, $h_{i,j}^c$, are independent and identically distributed (i.i.d), with zero mean and unit variance complex Gaussian distribution. The noise is considered to be complex Gaussian. The input signal components are chosen from a Q^2-QAM constellation, χ, with energy $\frac{\rho}{M}$, in which ρ can be interpreted as the SNR observed at any receive antenna. The vector $\mathbf{s}^c$ can

[1]Superscript ^{c} denotes complex values.

be obtained by Gray mapping from the vector of data bits, to the QAM constellation points. Note that the number of transmitted bits in each channel use is $2M\log_2(Q)$.

This system model can be transformed to its real counterpart by using the following transformations defined for vectors and matrices:

$$\begin{aligned} \mathbf{u}^c \mapsto \mathbf{u} &= \begin{bmatrix} \Re\{\mathbf{u}^c\}^T & \Im\{\mathbf{u}^c\}^T \end{bmatrix}, \\ \mathbf{H}^c \mapsto \mathbf{H} &= \begin{bmatrix} \Re\{\mathbf{H}^c\} & -\Im\{\mathbf{H}^c\} \\ \Im\{\mathbf{H}^c\} & \Re\{\mathbf{H}^c\} \end{bmatrix}. \end{aligned} \tag{2.2}$$

In this manner, $\mathbf{H}$ is a $n \times m$ real matrix with $m = 2M$ and $n = 2N$ and the components of vector $\mathbf{s}$ are chosen from a Q-PAM constellation, with energy $\frac{\rho}{2M}$. We can further simplify the model by mapping the equivalent PAM signals to integers using the following mapping:

$$\mathbf{s} = \kappa\mathbf{c} + \mathbf{v}, \tag{2.3}$$

in which the elements of $\mathbf{c}$ are in $\{0, 1, ..., Q-1\}$, κ is a constant scalar related to the PAM constellation energy, and $\mathbf{v}$ is a constant vector. Substituting (2.3) into (2.2), the system model is given by

$$\mathbf{y} = \mathbf{H}(\kappa\mathbf{c} + \mathbf{v}) + \mathbf{w}, \tag{2.4}$$

where vector $\mathbf{c}$ has integer elements. Hence, our problem over a MIMO channel converted to the detection of a lattice point transmitted over a linear channel with additive white Gaussian noise [57].

2.2.2 Hard-Output Detection

Following the system model in Section 2.2.1, the maximum-likelihood (ML) solution to the MIMO system in (2.1) is given by,

$$\hat{\mathbf{s}}^c = \arg \min_{\mathbf{s}^c \in \mathcal{X}^M} ||\mathbf{y}^c - \mathbf{H}^c \mathbf{s}^c||^2, \tag{2.5}$$

where $\mathcal{X}^M$ denotes the M-dimensional hyper-cube with components from Q^2-QAM constellation. Using the mentioned transformations which resulted in (2.4), the minimization in (2.5) can be rewritten as,

$$\begin{aligned} \hat{\mathbf{c}} &= \arg \min_{\mathbf{c} \in \mathcal{U}} ||\mathbf{y} - \mathbf{H}\mathbf{v} - \mathbf{H}\kappa\mathbf{c}||^2 \\ &= \arg \min_{\mathbf{c} \in \mathcal{U}} ||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2, \end{aligned} \tag{2.6}$$

where $\mathcal{U}$ refers to the hyper-cube $\{0, 1, ..., Q-1\}^m$, $\mathbf{y}' = \mathbf{y} - \mathbf{H}\mathbf{v}$, and $\mathbf{H}' = \kappa\mathbf{H}$.

For a general $\mathbf{H}$, finding the optimal solution to this problem has exponential complexity and hence approximate methods are preferred for use in practical systems. A low-complexity approximate solution is the linear zero forcing (ZF) decoder which can be written as

$$\hat{\mathbf{c}} = \lceil (\mathbf{H}')^{-1} \mathbf{y}' \rfloor, \tag{2.7}$$

where $\lceil \rfloor$ denotes the nearest integer function. In this way, the interference is totally suppressed which causes the noise effect to be amplified. Applying the decision feedback equalizer (DFE) can lessen this effect. DFE can be done by performing QR decomposition on the channel matrix to get

$$\mathbf{H}' = [\mathbf{Q}, \mathbf{Q}'] \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}, \tag{2.8}$$

where $\mathbf{R}$ is $m \times m$ upper triangular matrix with positive diagonal elements, $\mathbf{0}$ is $(n - m) \times m$ zero matrix, $\mathbf{Q}$ is a $n \times m$ unitary matrix, and $\mathbf{Q}'$ is a $n \times (n - m)$ unitary matrix. Using the QR decomposition the system model is given by

$$\mathbf{y}'' = \mathbf{R}\mathbf{c} + \mathbf{w}'', \tag{2.9}$$

where $\mathbf{y}'' = \mathbf{Q}^T\mathbf{y}'$. Since $\mathbf{R}$ is upper triangular, the last symbol can be estimated as $\hat{c}_m = \lceil y''_m / R_{m,m} \rfloor$. We can then substitute the estimated value to cancel the noise interference in y''_{m-1}. This approach is continued till the first symbol is detected. The solution can be written as

$$\hat{c}_i = \left\lceil \frac{y''_i - \sum_{j=i+1}^{m} r_{i,j}\hat{c}_j}{r_{i,i}} \right\rfloor \quad \text{for } i = m, m-1, \cdots, 1. \tag{2.10}$$

However, symbol detection in each step depends on the previous ones which causes error propagation in this method. The error performance of ZF and DFE algorithms is far from ML specially at high SNRs. Therefore, further efforts have been done to develop methods with low complexity and error performance close to the ML solution.

If the boundaries of the search region in (2.6) are relaxed to be the whole cubic lattice $\mathbb{Z}^m$ instead of $\mathcal{U}$, *lattice decoding* or *detection* is performed which is related to the Closest Lattice Point Search (CLPS). Solution to this problem is vastly investigated in lattice theory (e.g., [20], [28], [58], [21], and [22]). In fact, efficient lattice detection methods perform *lattice reduction* followed by a closest point search algorithm such as sphere decoder. However, the complexity of the closest point search problem is shown to be NP-hard in general. Therefore, approximate lattice decoders with lower complexity were proposed. Interestingly, lattice reduction methods are powerful tools for improving the error performance of approximate detection techniques. Therefore, they are used in conjunction with ZF and DFE decoders. The idea was first proposed by Babai in [24] and then employed in [25] and [26] for detection and precoding in MIMO systems.

Performing the lattice reduction on $\mathbf{H}'$ results in

$$\mathbf{B} = \mathbf{H}'\mathbf{G}, \tag{2.11}$$

where $\mathbf{B}$ is the reduced basis and $\mathbf{G}$ is a unimodular transformation matrix.[2] Applying the lattice reduction aided detection method, the minimization (2.6) can be approximated as

$$\hat{\mathbf{c}} \simeq \arg \min_{\mathbf{c} \in \mathbb{Z}^m} ||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2. \tag{2.12}$$

Using the reduced matrix in (2.11), one can equivalently solve

$$\begin{aligned} \hat{\mathbf{c}} &\simeq \arg \min_{\mathbf{c} \in \mathbb{Z}^m} ||\mathbf{y}' - \mathbf{H}'\mathbf{G}\mathbf{G}^{-1}\mathbf{c}||^2 \\ &= \mathbf{G} \arg \min_{\mathbf{c}' \in \mathbb{Z}^m} ||\mathbf{y}' - \mathbf{B}\mathbf{c}'||^2, \end{aligned} \tag{2.13}$$

where $\mathbf{c}' = \mathbf{G}^{-1}\mathbf{c}$. Assuming the lattice reduction aided detection over correlated fading channels, we present adaptive methods in the next section in order to reduce the computational complexity.

2.3 Adaptive Lattice Reduction Aided Detection

2.3.1 *Method I*

Here we assume that the Rayleigh fading channel is slowly varying during the transmission frame. Our method hinges on the observation that a lattice basis transformation (i.e., the unimodular matrix associated with the LLL) of the previous channel realization can still reduce the actual channel (though not in the sense of LLL) if the channel variation is slow enough. Then, applying the LLL reduction on the already reduced

[2]Unimodular matrices and their inverses have integer elements $\mathbf{G}\mathbb{Z}^m = \mathbb{Z}^m$.

channel matrix should cost much less than applying the LLL on the original unreduced channel matrix. As mentioned before, the output of the LLL algorithm is a reduced matrix and a transformations matrix. The transformation matrix is used to convert the initial channel matrix to the reduced one. This relation can be expressed as

$$\mathbf{B}_1 = \mathbf{H}_1\mathbf{G}_1, \tag{2.14}$$

where $\mathbf{B}_1, \mathbf{H}_1$ and $\mathbf{G}_1$ are the reduced matrix, channel matrix, and the transformation unimodular matrix, respectively.

Without loss of generality, assume that the fading process is a first-order Gauss-Markov process as

$$\mathbf{H}_k = a\mathbf{H}_{k-1} + \mathbf{Z}_k, \tag{2.15}$$

where $0 \leq a < 1$ shows the channel variations between consecutive transmissions and $\mathbf{Z_k}$ has i.i.d entries distributed as circular complex Gaussian with zero mean and variance equal to $\epsilon^2 = 1 - a^2$, [59]. Assuming small variation in each element of the MIMO fading channel through time, a is close to one and so ϵ is close to zero.

Considering these small changes, it seems quite reasonable to make use of the previous transformation of lattice reduction for $\mathbf{H}_1$ in computing a reduced matrix for the new channel $\mathbf{H}_2$. Therefore, using the previous transformation on the new channel matrix results in,

$$\mathbf{B}_2' \triangleq \mathbf{H}_2\mathbf{G}_1 = (a\mathbf{H}_1 + \mathbf{Z}_2)\mathbf{G}_1 \tag{2.16}$$

$$= a\mathbf{H}_1\mathbf{G}_1 + \mathbf{Z}_2\mathbf{G}_1 = a\mathbf{B}_1 + \mathbf{Z}_2\mathbf{G}_1. \tag{2.17}$$

By the fact that $\mathbf{G}_1$ is a unimodular matrix, $\mathbf{B}_2'$ is still a basis for the space spanned by columns of $\mathbf{H}_2$. Consider the LLL algorithm, as $\mathbf{B}_1$ is already LLL reduced, it satisfies all column swap and size reduction conditions. If the other term in equation (2.17) is

small enough, the Gram-Schmidt coefficients of the resulting matrix in the right-hand side of equation (2.17) are close to those of the $\mathbf{B}_1$. Therefore, performing LLL on $\mathbf{B}_2'$, does not require many more basis updates. In other words, starting from an almost LLL reduced matrix results in less complexity for LLL reduction. Thus, using $\mathbf{B}_2'$ as the initial input for the LLL algorithm in comparison with starting from the original channel matrix $\mathbf{H}_2$, reduces the complexity of the LLL algorithm significantly.

Performing LLL on $\mathbf{B}_2'$ results in

$$\mathbf{B}_2 \triangleq \mathbf{B}_2'\tilde{\mathbf{G}}_2 = (\mathbf{H}_2\mathbf{G}_1)\tilde{\mathbf{G}}_2 = \mathbf{H}_2\mathbf{G}_2 \tag{2.18}$$

in which $\mathbf{G}_2 \triangleq \mathbf{G}_1\tilde{\mathbf{G}}_2$ is the transformation matrix for reducing $\mathbf{H}_2$. Note that the proposed adaptive method of matrix reduction results in a reduced matrix which might be different from the result of applying the LLL algorithm directly on $\mathbf{H}_2$, but as both resulting matrices are LLL reduced, the error performances of the MIMO decoder are close to each other.

In fact, the orthogonality defect factor and the product of the norms (respectively, defined in (1.7) and (1.8), for near reduced basis $\mathbf{B}_2'$ is close to the one for the previous reduced basis, $\mathbf{B}_1$, if the channel slowly changes. Assume $\mathbf{B}_1 = [\mathbf{b}_1|\mathbf{b}_2\cdots|\mathbf{b}_M]$, $\mathbf{B}_2' = [\mathbf{b}_1'|\mathbf{b}_2'\cdots|\mathbf{b}_M']$ and $\mathbf{G}_1 = [\mathbf{g}_1|\mathbf{g}_2\cdots|\mathbf{g}_M]$. The product of the norms for the near reduced basis can be written as

$$\begin{aligned} D(\mathbf{B}_2') &= ||\mathbf{b}_1'||||\mathbf{b}_2'||\cdots||\mathbf{b}_M'|| && (2.19)\\ &= ||\mathbf{H}_2\mathbf{g}_1||||\mathbf{H}_2\mathbf{g}_2||\cdots||\mathbf{H}_2\mathbf{g}_M|| && (2.20)\\ &= ||(a\mathbf{H}_1+\mathbf{Z}_2)\mathbf{g}_1||||(a\mathbf{H}_1+\mathbf{Z}_2)\mathbf{g}_2||\cdots||(a\mathbf{H}_1+\mathbf{Z}_2)\mathbf{g}_M|| && (2.21)\\ &= \prod_{i=1}^{M}||(a\mathbf{H}_1+\mathbf{Z}_2)\mathbf{g}_i||, && (2.22) \end{aligned}$$

where the definition of $\mathbf{B}_2'$ and $\mathbf{H}_2$ are used from (2.17) and (2.15), respectively. By

applying the triangle inequality on (2.19) for the norms, we get

$$D(\mathbf{B}_2') \leq \prod_{i=1}^{M} [a||\mathbf{H}_1\mathbf{g}_i|| + ||\mathbf{Z}_2\mathbf{g}_i||] \tag{2.23}$$

$$= \prod_{i=1}^{M} ||\mathbf{H}_1\mathbf{g}_i|| \left[a + \frac{||\mathbf{Z}_2\mathbf{g}_i||}{||\mathbf{H}_1\mathbf{g}_i||}\right] \tag{2.24}$$

$$= \prod_{i=1}^{M} ||\mathbf{b}_i|| \left[a + \frac{||\mathbf{Z}_2\mathbf{g}_i||}{||\mathbf{H}_1\mathbf{g}_i||}\right] \tag{2.25}$$

$$= D(\mathbf{B}_1) \prod_{i=1}^{M} \left[a + \frac{||\mathbf{Z}_2\mathbf{g}_i||}{||\mathbf{H}_1\mathbf{g}_i||}\right]. \tag{2.26}$$

Furthermore, the ratio of the norms in (2.26) can be written as

$$\frac{||\mathbf{Z}_2\mathbf{g}_i||}{||\mathbf{H}_1\mathbf{g}_i||} = \frac{||\mathbf{Z}_2\mathbf{H}_1^{-1}\mathbf{b}_i||}{||\mathbf{b}_i||} \leq ||\mathbf{Z}_2\mathbf{H}_1^{-1}||, \tag{2.27}$$

where the inequality came from the matrix norm definition, $||\mathbf{A}||_2 = \max_{\mathbf{x}\neq 0} ||\mathbf{A}\mathbf{x}||_2/||\mathbf{x}||_2$. By applying (2.27) in (2.26), we get

$$\begin{aligned} D(\mathbf{B}_2') &\leq D(\mathbf{B}_1) \prod_{i=1}^{M} [a + ||\mathbf{Z}_2\mathbf{H}_1^{-1}||] \\ &= D(\mathbf{B}_1)[a + ||\mathbf{Z}_2\mathbf{H}_1^{-1}||]^M \\ &= D(\mathbf{B}_1)[a + \epsilon||\mathbf{Z}_2'\mathbf{H}_1^{-1}||]^M, \end{aligned} \tag{2.28}$$

where the elements in $\mathbf{Z}_2'$ have unit variance. Although the upper bound in (2.28) might not be tight, it shows that with high probability, the norm products for the near reduced matrix is close to the previously reduced one for small channel variations.

To improve this upper bound, one needs to investigate the norms of the two matrices which are related to the maximum and minimum singular values of a Gaussian distributed matrix ([60] and [61]). Here, for our purpose, the ratios of norm products, $D(\mathbf{B}_2')/D(\mathbf{B}_1)$ and $D(\mathbf{H}_2)/D(\mathbf{B}_1)$, are computed for the LLL algorithm with small

changes in fading parameters. Fig. 2.1 shows the CDF of these ratios in a four dimensional system with $\epsilon = .01$. As it can be seen from the figure and the upper bound (2.28), in *Method I* the norm products for the near reduced matrix is very close to the previous one which has an LLL reduced basis with a small products of the norms. In fact, in the LLL algorithm after each size reduction the product of the norms becomes smaller. Thus, applying the LLL on $\mathbf{B}_2' = \mathbf{H}_2\mathbf{G}_1$ with smaller products of the norms instead of $\mathbf{H}_2$ needs fewer reduction steps. In other words, we have already done most of the reduction steps during the previous reduction.

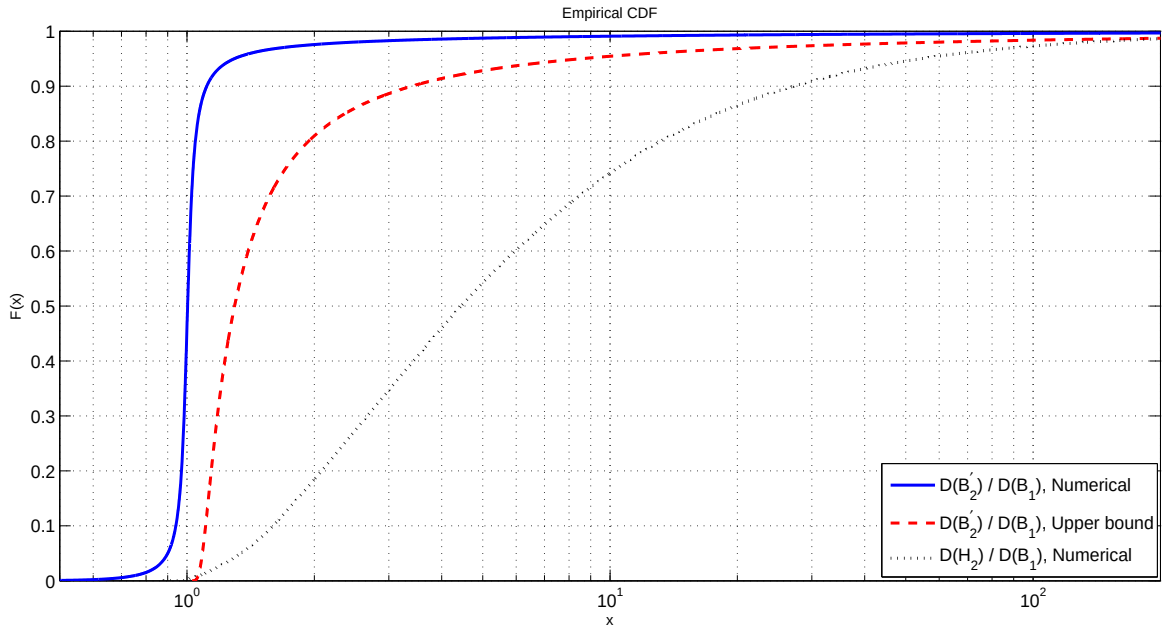


Figure 2.1: CDF of the ratio of the norm products for the channel and the near reduced basis

To bound the number of iterations in the main body of the LLL algorithm (the number of times we check the second test in Definition 1.1), the condition number of the basis, $\kappa(\mathbf{H})$, is used [62, 63]. Number of iterations K in an m-dimensional system is

upper bounded by

$$K \leq m^2 \log_t \left(\frac{A(\mathbf{H})}{a(\mathbf{H})} \right) + m$$
$$\leq m^2 \log_t \kappa(\mathbf{H}) + m \tag{2.29}$$

where $t = \frac{1}{\sqrt{p}}$, $A(\mathbf{H})$ and $a(\mathbf{H})$ are the maximum and the minimum norm of the Gram-Schmidt vectors corresponding to $\mathbf{H}$, respectively. Based on this upper bound, the CDF of the condition number for the channel, the LLL reduced basis and the near reduced basis is depicted in Fig. 2.2 for $\epsilon = .05$. It can be seen that for a small variation in channel, the near reduced basis is better conditioned than the actual channel. Therefore, less number of iterations is needed on average to perform the reduction on the near reduced basis.

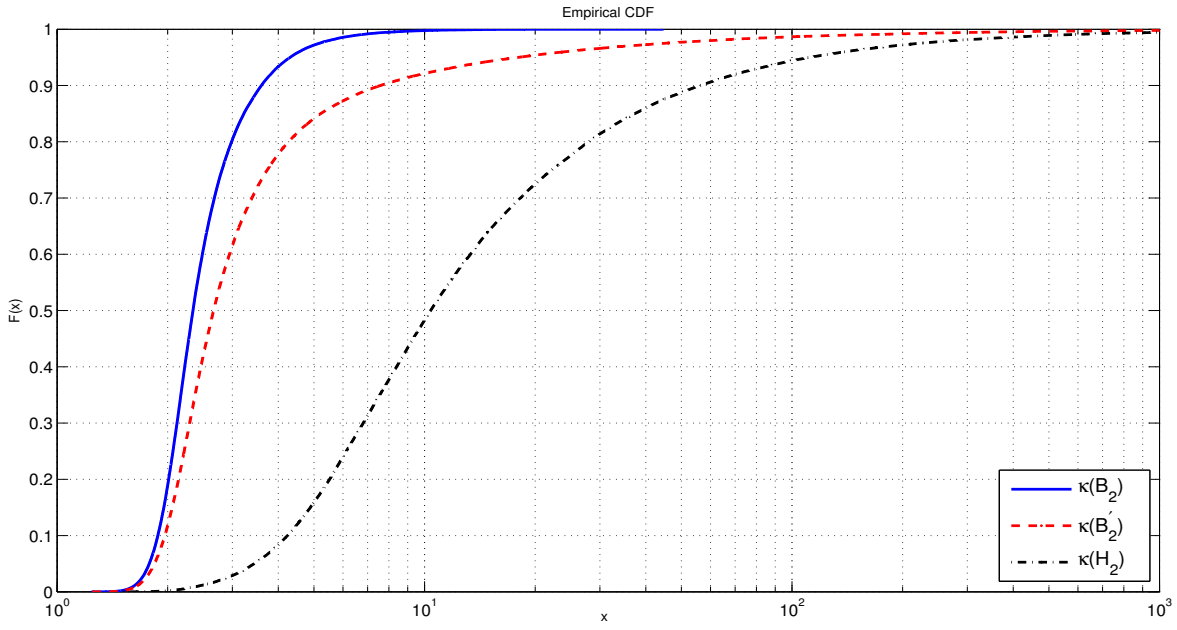


Figure 2.2: CDF of the condition number for the channel, the reduced basis and the near reduced basis

2.3.2 *Method II*

If further saving in computational complexity is required, one can omit the part that performs the LLL reduction algorithm on $\mathbf{B}_2'$, and use $\mathbf{B}_2'$ itself as the reduced basis for $\mathbf{H}_2$. This means that if the variations in the channel are slow enough we can use the same unimodular transformation matrix to reduce the later channel matrices. If we generalize the bounds in (2.28) for small channel variations, we can see that the product of the norms (and orthogonality defect factor) for consecutive realizations are close to each other. In other words, with high probability, the product of the norms and orthogonality defect factor are bounded within a time frame in which the tightness of the bounds is related to the frame length. Considering that the method proposed is an approximate one, as the difference between the current channel and the reference channel gets larger with time, the effect of basis reduction using the unimodular transformation matrix decreases. Therefore, it is required to update the unimodular transformation matrix after it no longer produces acceptable results. Note that, the update rate can be used as a tradeoff between complexity and accuracy.

We propose to update the unimodular transformation matrix using the orthogonality defect factor defined in (1.7). To adaptively track the channel variations and be sure that the orthogonality defect factor for the adaptive reduction stays bounded, we use a lower and an upper bound for the orthogonality defect factor in *Method II*. Therefore, in each channel realization we check the ratio of the near-reduced basis to the reference one (the last time the LLL was performed). If it is within the assumed bound, the reduced basis is computed just by multiplying the reference unimodular matrix to the new channel matrix, otherwise the reduction is performed on the new channel and the reference basis and its corresponding unimodular are reset. If we denote the last time we performed LLL as the reference point, the channel at that time by $\mathbf{H}_{ref}$, and the defect factor of the output of the LLL as $\delta(\mathbf{B}_{ref})$, the condition for keeping the unimodular

transformation matrix is defined as

$$\frac{1}{\alpha} \leq \frac{\delta\left(\mathbf{H}_i \mathbf{G}_{ref}\right)}{\delta(\mathbf{B}_{ref})} \leq \alpha \ , \tag{2.30}$$

where α is chosen according to the fading parameters, computational complexity constraints and the desired error performance. The minimum value for α is 1 which denotes the non-adaptive reduction method. The greater α provides less complexity but worse performance. Note that, by performing QR decomposition for detection at the end, the orthogonality defect factor can be computed simply by using the upper triangular matrix, R, with $O(n^2)$ complexity.

To investigate when the adaptive lattice reduction in *Method II* achieves the maximum diversity, the techniques from [56] are used. It was shown that lattice reduction aided detection achieves the maximum receive diversity which is the number of receive antennas. In fact, for a point-to-point MIMO system with the V-BLAST transmission with M transmit antennas and N receive antennas, when we use the LLL lattice aided decoding,

$$\lim_{SNR \to \infty} \frac{-\log P_e}{\log SNR} = N, \tag{2.31}$$

where P_e is the average error probability. Considering the updating condition in (2.30), the orthogonality defect factor for adaptive reduction *Method II* is upper bounded by $\alpha\delta(\mathbf{B}_{ref})$. On the other hand, $\mathbf{B}_{ref}$ is an LLL reduced matrix and $\delta(\mathbf{B}_{ref})$ is less than $C^{M(M-1)}$ in which C is a constant(see [64]). Therefore, for all channel matrices, the adaptively reduced basis has an upper bound as

$$\delta(B)_{Adp-LLL} \leq \alpha\delta(B)_{LLL} \leq \alpha C^{M(M-1)}. \tag{2.32}$$

Applying the above bound, we have the following proposition for for adaptive reduction *Method II*:

Proposition 2.1. *The LLL reduction aided detection by the adaptive Method II with N receive antennas achieves the maximum receive diversity if* $\lim_{SNR\to\infty} \frac{\log \alpha}{\log SNR} = 0$.

Proof. Using Theorem 2 in [56], it can be shown that

$$P_e \leq \frac{c_1 \alpha^N}{SNR^N} \tag{2.33}$$

which c_1 in a constant. Therefore, we have

$$\begin{aligned} &\lim_{SNR\to\infty} \frac{-\log P_e}{\log SNR} \geq \\ &\lim_{SNR\to\infty} \frac{N \log SNR - \log c_1 - N \log \alpha}{\log SNR} = N. \end{aligned} \tag{2.34}$$

□

Remark 2.1. Note that there is always a gap in performance between the basic reduction and the adaptive *Method II* because the adaptive method is an approximate one with lower complexity. In fact, we have a a tradeoff between detection complexity and performance in the adaptive reduction *Method II* which can be controlled by α for different SNRs.

Remark 2.2. It is also possible to combine *Method I* and *Method II* to get even more efficient adaptive algorithm. This means that when it is required to update the unimodular transformation matrix in *Method II*, we can perform it using the adaptive algorithm we proposed in *Method I*, i.e. at the time instant i, that it is required to update the unimodular matrix we can perform the LLL algorithm on $\mathbf{H}_i\mathbf{G}_{ref}$ instead of $\mathbf{H}_i$. This method makes it possible to save more in computational complexity. In the simulation results, this method is named as *Method III*.

2.4 Soft-Output Detection for Multiple Antenna System

2.4.1 MAP Detection with List sphere decoder

A soft-output detection for MIMO system is used in this work as in [29] and [30]. Using a soft-input, soft-output (SISO) decoder for the outer code and exchanging the soft information between the MIMO detector and the SISO decoder, iterative joint detection and decoding can be performed (see Fig. 2.3). In this system and at the transmitter side, $2M\log_2(Q)$ coded bits are transmitted per each channel use by means of an M-dimensional vector of QAM symbols. Let x_j, $j = 0, ..., 2M\log_2(Q) - 1$ be the jth bit in transmitted vector $\mathbf{x}$. The maximum *a posteriori* (MAP) probability bit detection for x_j conditioned on received vector $\mathbf{y}$ can be expressed in the log-likelihood ratio (LLR) form as

$$L(x_j|\mathbf{y}) = \ln \frac{P[x_j = +1|\mathbf{y}]}{P[x_j = -1|\mathbf{y}]} \tag{2.35}$$

where $x_j \in \{-1, +1\}$ is the representation for the logical zero and one, respectively. The LLR values are used to verify the reliability of decisions.

Using the interleaver, we can assume that the bits within the transmitted vector are statistically independent of each other. Following our system model in (2.4) and using Bayes' theorem and likelihood function for computing the LLRs as in [65] and [29], soft output values $L(x_j|\mathbf{y})$ can be written as

$$L(x_j|\mathbf{y}) = \ln \frac{\sum_{\mathbf{x}\in\mathbb{X}_j^{+1}} \exp(-\frac{1}{2\sigma^2}||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2 + \frac{1}{2}\mathbf{x}_{[j]}^T \cdot \mathbf{L}_{A,[j]})}{\sum_{\mathbf{x}\in\mathbb{X}_j^{-1}} \exp(-\frac{1}{2\sigma^2}||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2 + \frac{1}{2}\mathbf{x}_{[j]}^T \cdot \mathbf{L}_{A,[j]})} \tag{2.36}$$

in which $\mathbb{X}_j^{+1}$ and $\mathbb{X}_j^{-1}$ are the set of $2^{2M\log(Q)-1}$ bit vectors having $x_j = +1$ and $x_j = -1$, respectively, i.e. $\mathbb{X}_j^{+1} = \{\mathbf{x}|x_j = +1\}$, $\mathbb{X}_j^{-1} = \{\mathbf{x}|x_j = -1\}$. Vector $\mathbf{x}_{[j]}$ is obtained from

$\mathbf{x}$ and $\mathbf{L}_{A,[j]}$ is the vector of *a priori* LLR values of $\mathbf{x}$, both by omitting x_j. Note that, $\mathbf{c}$ is the vector of translated symbols from $\mathbf{s}$ by (2.3) and $\mathbf{s} = \text{map}(\mathbf{x})$. Moreover, for the first iteration between the MIMO detector and SISO decoder, all symbols are assumed to be equally likely and thus $\mathbf{L}_A$ is a zero vector. To simplify equation (2.36), the *a posteriori* LLR associated with each transmitted bit can be approximated by using the max-log approximation [66] as

$$\begin{aligned} L(x_j|\mathbf{y}) \approx & \frac{1}{2} \max_{x \in \mathbb{X}_j^{+1}} \left\{ -\frac{1}{\sigma^2} ||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2 + \mathbf{x}_{[j]}^T \cdot \mathbf{L}_{A,[j]} \right\} \\ & - \frac{1}{2} \max_{x \in \mathbb{X}_j^{-1}} \left\{ -\frac{1}{\sigma^2} ||\mathbf{y}' - \mathbf{H}'\mathbf{c}||^2 + \mathbf{x}_{[j]}^T \cdot \mathbf{L}_{A,[j]} \right\}. \end{aligned} \tag{2.37}$$

After computing L, it is deinterleaved to get the *a priori* input L' to the SISO decoder.

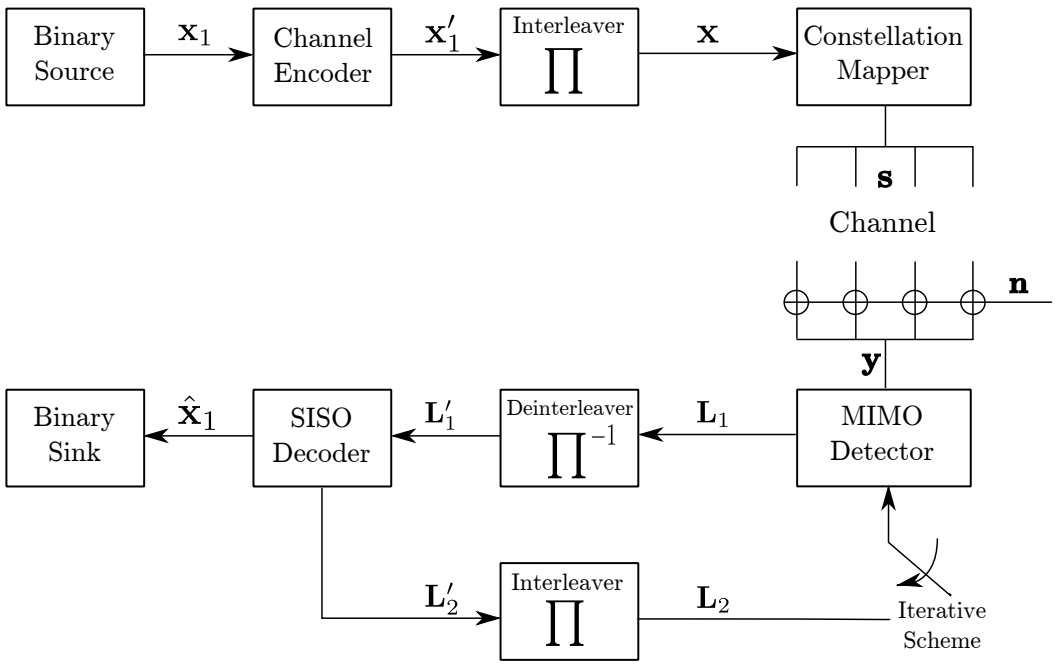


Figure 2.3: Transmitter and receiver in iterative detection/decoding for MIMO systems

The MAP detector, even for the simplified LLR computation by the max-log approximation (2.37) has an exponential complexity in the length of bit vectors or the number of transmit antennas and the bits in each constellation symbols. Therefore, searching over all possible transmitted signals is not an efficient way of solving this problem and to avoid this high complexity procedure, sub-optimal detectors based on the practical list of candidates, have been proposed, (e.g., [29], [30] and [31]). To have a reliable result, by means of lattice representation of the system, the most likely lattice points must be included in the list with a practical size. List sphere decoders (LSD) are used commonly to provide the list of best candidates of the transmitted symbols and their corresponding Euclidean distances as the output. In this search algorithm, the list construction is performed by a sphere decoder and the lattice points with the smallest distances to the received point are placed in the list. For example, in [29], the list sphere decoder is in fact an ML estimator with a fixed radius of searching sphere which fills the list of size L, with the closest lattice points. When $\mathbf{y}'$ is outside this finite lattice constellation, which is highly probable (specially in lower SNRs or in higher MIMO dimensions), the sphere centered on the received point, searches within a large number of points to find a small number of constellation points. Considering this fact, a new LSD was proposed in [47]. In this case, as the first step, the ML point can be found by a sphere decoder and then the center of spherical list L is selected on $\mathbf{H}'\hat{\mathbf{c}}_{ML}$ instead of $\mathbf{y}'$. A double Pohst recursion [67] is used to construct the list. One of them is used to find the points at a distance less than R from the ML point and the other recursion computes the squared Euclidean distances between the valid points in the search sphere and the received point, $||\mathbf{y}' - \mathbf{H}'\hat{\mathbf{c}}_{list}||^2$, in a parallel manner. In the next part, based on this LSD technique, we propose the adaptive list detector for slow fading channels.

2.4.2 Adaptive Soft-Output Detection

To generate the list, we modify the algorithm in [47] by finding a near-ML point in the first step using our adaptive hard-output detection method developed in Section 2.3. Considering the adaptive near-ML point as the beginning point of the LSD and as the center of spherical list L, the list sphere decoding algorithm with a fixed radius is used to construct the list. One can see by the numerical results that by choosing a large list size, the performance of the soft-output detector which uses the near-ML point as the center of the sphere, is nearly the same as the one with the ML point at the center.

In the second step and for arranging the list of appropriate candidates, if the temporal correlation of channels is used beside the lattice structure of the MIMO systems, the list can also be formed in an adaptive manner. To perform it, a reference list is constructed by finding the points inside the sphere centered at the zero point of the lattice and then shifting it to the estimated near-ML point. In the reference list, the lattice points are sorted by their distance to the zero point. For the subsequent channel realizations, we check if the variation between the new channel and old one is within a certain bound or not. If that is the case, we simply shift the reference list around the new estimated near-ML point and just update the Euclidian distances, $||\mathbf{y}'_{new} - \mathbf{H}'_{new}\hat{\mathbf{c}}_{list}||^2$, where $\hat{\mathbf{c}}_{list}$ are the shifted lattice points. This procedure continues until the variation from the reference channel to the new one is greater than the selected bound. In this case, the reference lattice is updated by the new channel basis and the new reference list is constructed.

As a measure of the channel variation, the orthogonality defect factor (1.7) is used as follows. Using this measure as an overall measure for tracking the basis variation, the aforementioned bound for updating the reference list is defined as

$$\frac{1}{\beta} < \frac{\delta(\mathbf{H}_{new})}{\delta(\mathbf{H}_{ref})} < \beta \tag{2.38}$$

where β is a measure of the performance/complexity tradeoff for the adaptive list construction. Note that the intersection between the shifted reference list and the finite lattice constellation may be small. Thus, we can construct a larger reference list to assure that we always have a sufficient number of points in the shifted list around the estimated lattice point. In the next section, the simulation results are provided for different channel codes.

2.5 Simulation Results

2.5.1 Hard-Output Detection

In this part, the performance and complexity of the adaptive reduction methods for hard-output detection are studied and compared with the conventional reduction methods. Correlated Rayleigh fading coefficients are generated according to [68]. The motivation for this was the ability to generate the channel samples sequentially with accurate statistical properties and low computational complexity. The fading parameters of the channels for all the simulations are $f_d = 100$Hz and $f_s = 270$ksps where f_d is the maximum Doppler frequency and $f_s = 1/T_s$ is the sampling rate. We consider the MIMO channel with $M = N$ transmit and receive antennas. A 4-QAM constellation is used to investigate the error performance of the adaptive methods. The bit error performances are figured versus E_b/N_0 in which E_b is the average energy of a bit at each receiver antenna. Figs. 2.4 and 2.5 show the bit error rates of different detection methods. In these simulations, the adaptive methods are used to modify the LLL and the deep insertion LLL algorithms. It can be seen that using *Method I* does not affect the error performance of the detection method, where the degradation in performance is small for different value of α for *Method III*. Note that to provide near-ML performance, the MMSE-DFE detection method is used.

To study the complexity, we compare the average number of required flops for different reduction algorithms where a flop is either a multiplication, a division, an addition or a subtraction. Figs. 2.6 and 2.7 show the computational complexity savings by using the adaptive methods. Fig. 2.6 shows the average number of flop counts for the LLL and the adaptive methods. The results for the deep insertion LLL are depicted in Fig. 2.7. In these figures, the average number of flops are sketched for different number of antennas where in all cases $M = N$ is considered. Fig. 2.8 compares the average number of basis updates for the LLL algorithm and the adaptive algorithms. As you can see in this figure, there exist an obvious gain in using the adaptive methods. Also, Fig. 2.9 shows the average number of required size reductions and insertions for both deep insertion LLL and adaptive methods. Note that reduction, swap and insertion are the basic basis updates used in LLL and deep insertion LLL. As it was expected from the adaptive nature of the proposed methods, there is a significant gain in terms of number of basis updates. This admits that for the channel we have considered in these simulations, using the transformation matrix of the previous channel realization gives, a quite good reduction for the current channel realizations. In fact, multiplication of the transformation matrix by the new channel matrix becomes the main part of computational complexity in the adaptive methods.

2.5.2 Soft-Output Detection

The performances of using the near-ML point and the adaptive method for soft-output detection are studied. We consider a MIMO channel with $M = N = 4$ transmit and receive antennas. The coded data stream with length of 9216-bits is modulated by a 16-QAM constellation using the Gray mapping. At first, a half rate convolutional code with generator polynomials $(5, 7)$ is used. For the sphere detection, a list with the maximal length of $L = 512$ is computed. Fig. 2.10 shows the bit error performance for

the soft-output detection and the iterative scheme for the LSD with ML and near-ML as the starting point, respectively. It can be seen that although the performance of ML estimation is better than the adaptive near-ML, using the near-ML point as the center of the spherical list construction does not affect the performance of the soft-output detection method.

Fig. 2.11 shows the error performance of the adaptive scheme for the soft-output detection. Here, β is selected as 1.5, 3, and 5. The average updating time of the reference list construction for these selections of adaptive measures is 36, 117 and 170, respectively (i.e., for a chosen $\beta = 3$ we just need to construct the reference list every 117 channel realization, on average, and simply shift the list for the rest of channel realizations). As discussed before, there is a tradeoff between the performance and the complexity for the soft-output detection. Based on the fading speed, the appropriate β should be selected for a desired performance and complexity.

Finally, the performance of the proposed adaptive soft-output detection method investigated by using a turbo code as the outer code. The parallel concatenated turbo code with two memory and feedback polynomial $1 + D + D^2$ and feedforward polynomial $1 + D^2$ with β selected as 1.5 and 3 is used over the slow fading channel. For the sphere detection, a list with the maximal length of $L = 128$ is employed. Fig. 2.12 shows the bit error rate performance and the tradeoff between the performance and the complexity for the adaptive scheme.

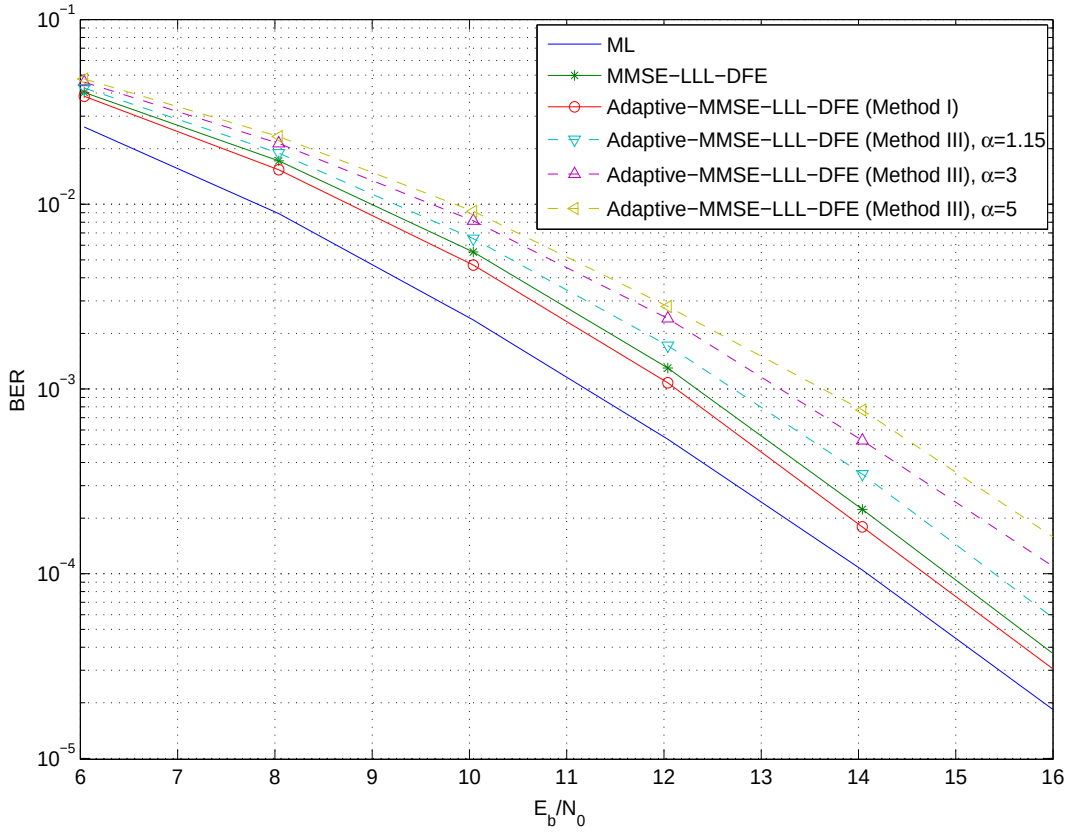


Figure 2.4: Bit error performance of the adaptive hard-output detection methods using the LLL reduction for a 4×4, 4-QAM MIMO system.

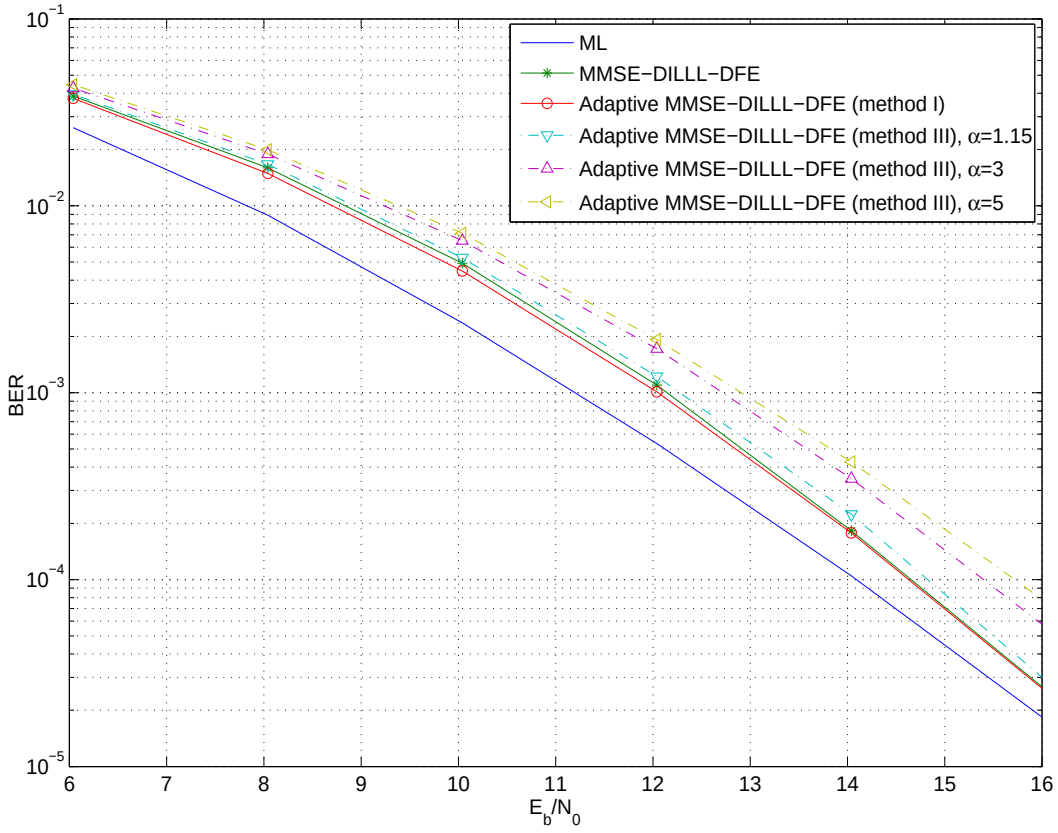


Figure 2.5: Bit error performance of the adaptive hard-output detection methods using the DILLL reduction for a 4×4, 4-QAM MIMO system.

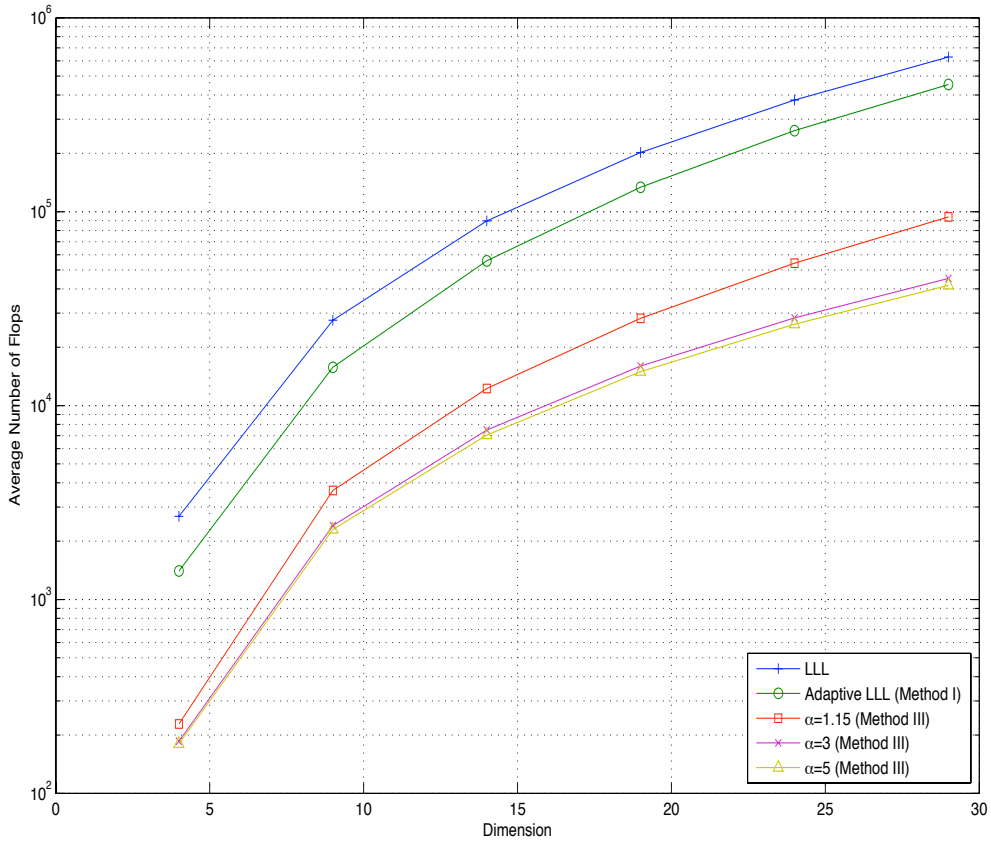


Figure 2.6: Average number of flops for the LLL and the adaptive methods for different number of antennas.

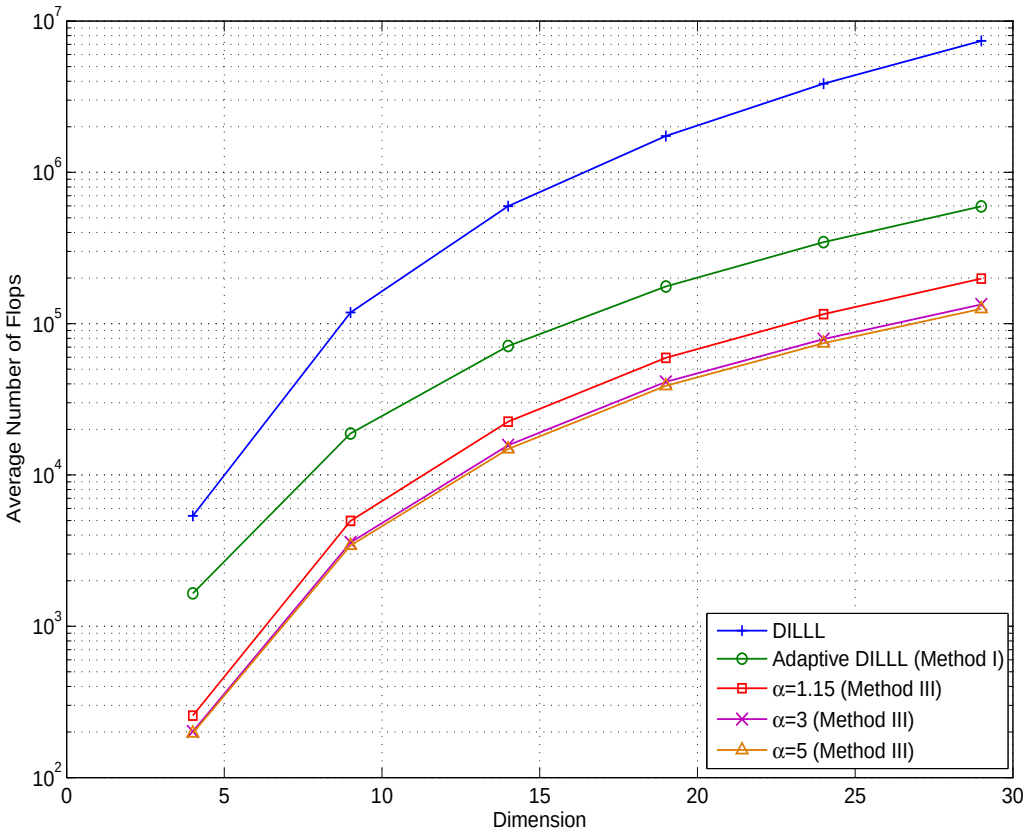
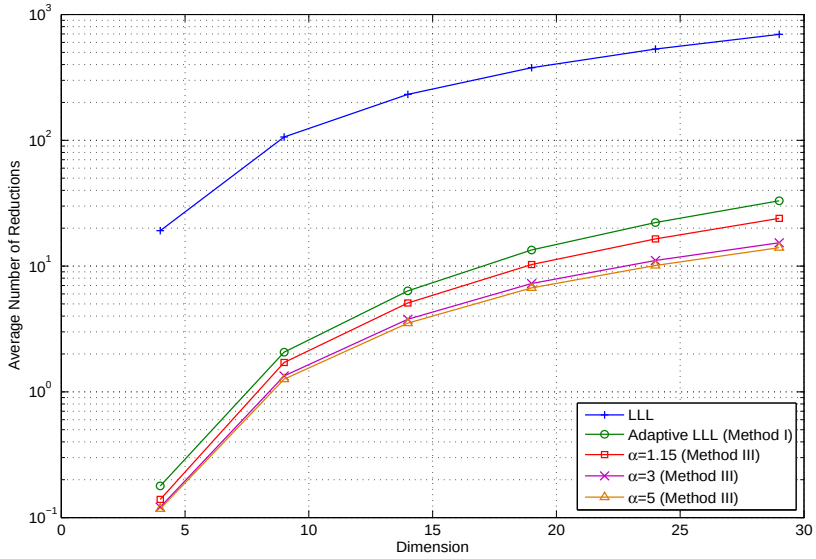
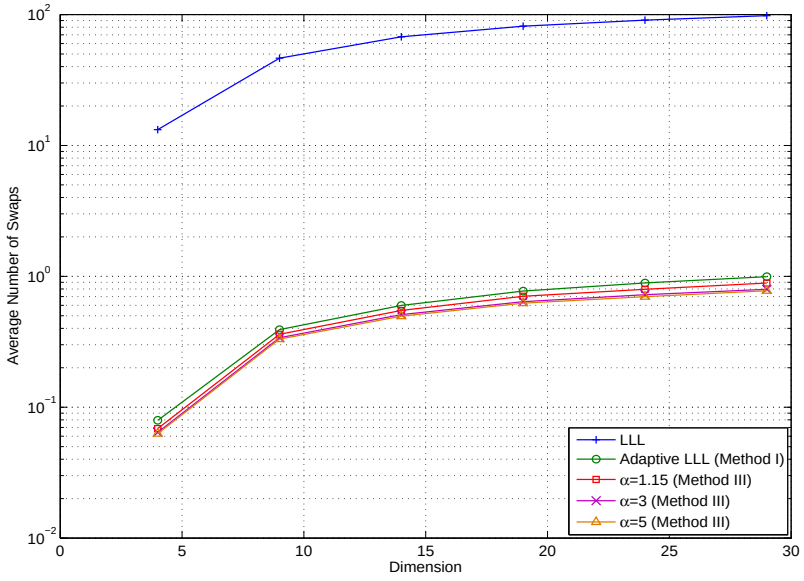


Figure 2.7: Average number of flops for the DILLL and the adaptive methods for different number of antennas.

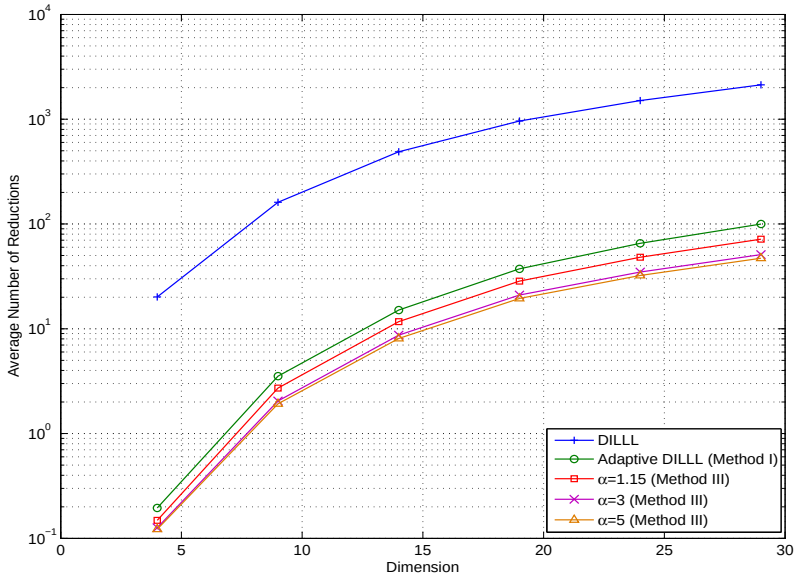


(a) Average Number of Size Reductions

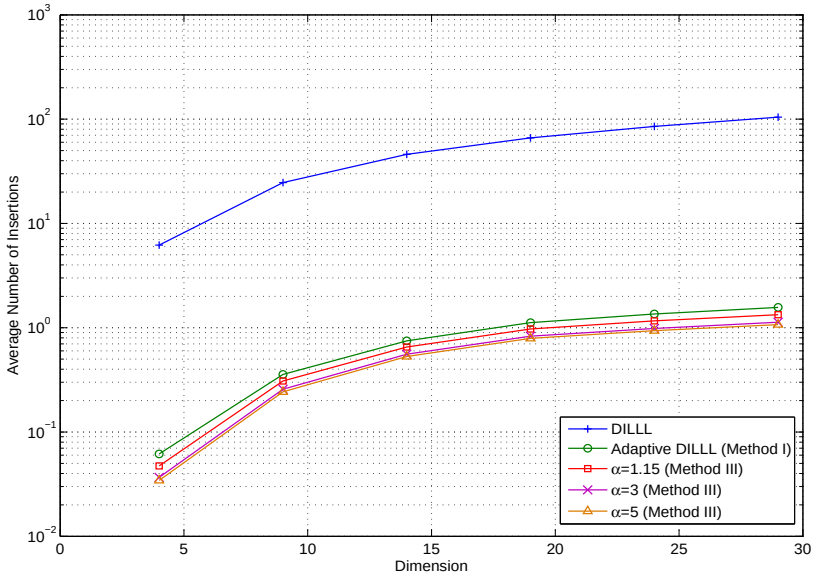


(b) Average Number of Swaps

Figure 2.8: Average number of basis updates for the LLL and the adaptive methods for different number of antennas.



(a) Average Number of Size Reductions



(b) Average Number of Insertions

Figure 2.9: Average number of basis updates for the DILLL and the adaptive methods for different number of antennas.

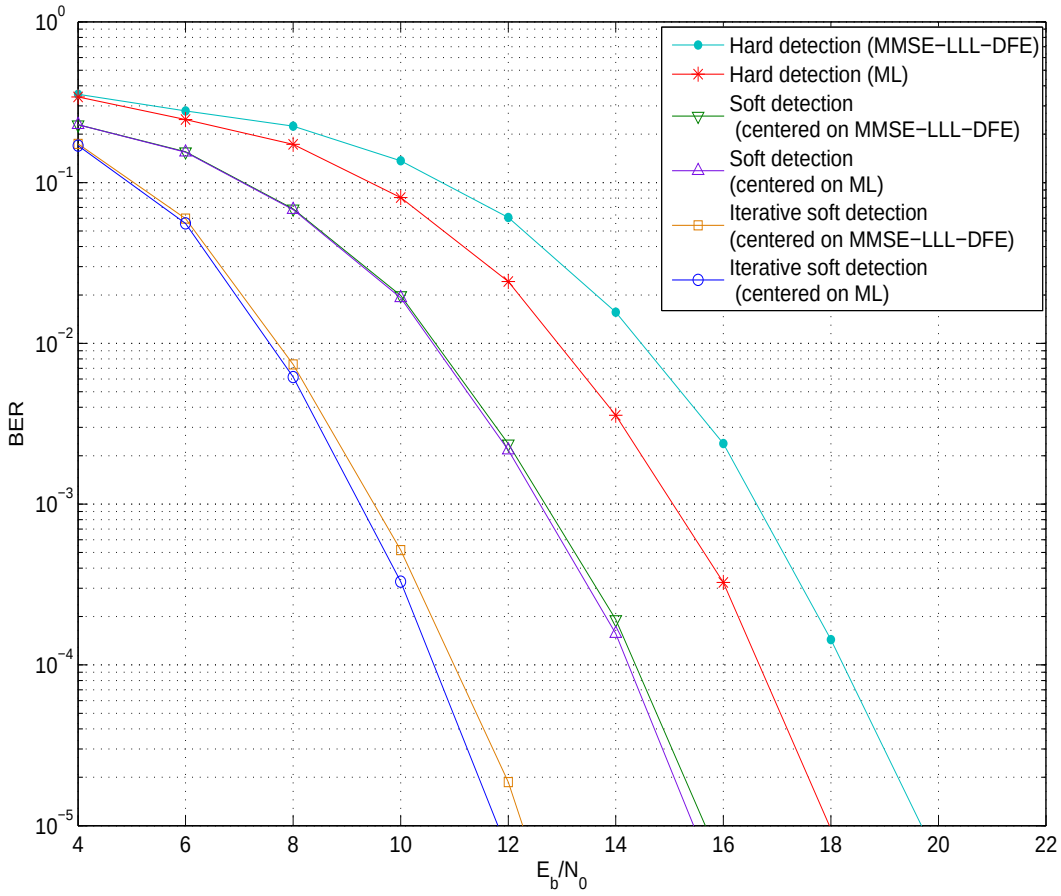


Figure 2.10: Error performance of the ML and the adaptive near-ML detection and their corresponding soft-output detection (with no iteration) and iterative soft-output detection with four iterations between the detector and decoder by using a convolutional code with generator polynomials $(5, 7)$ for 16-QAM.

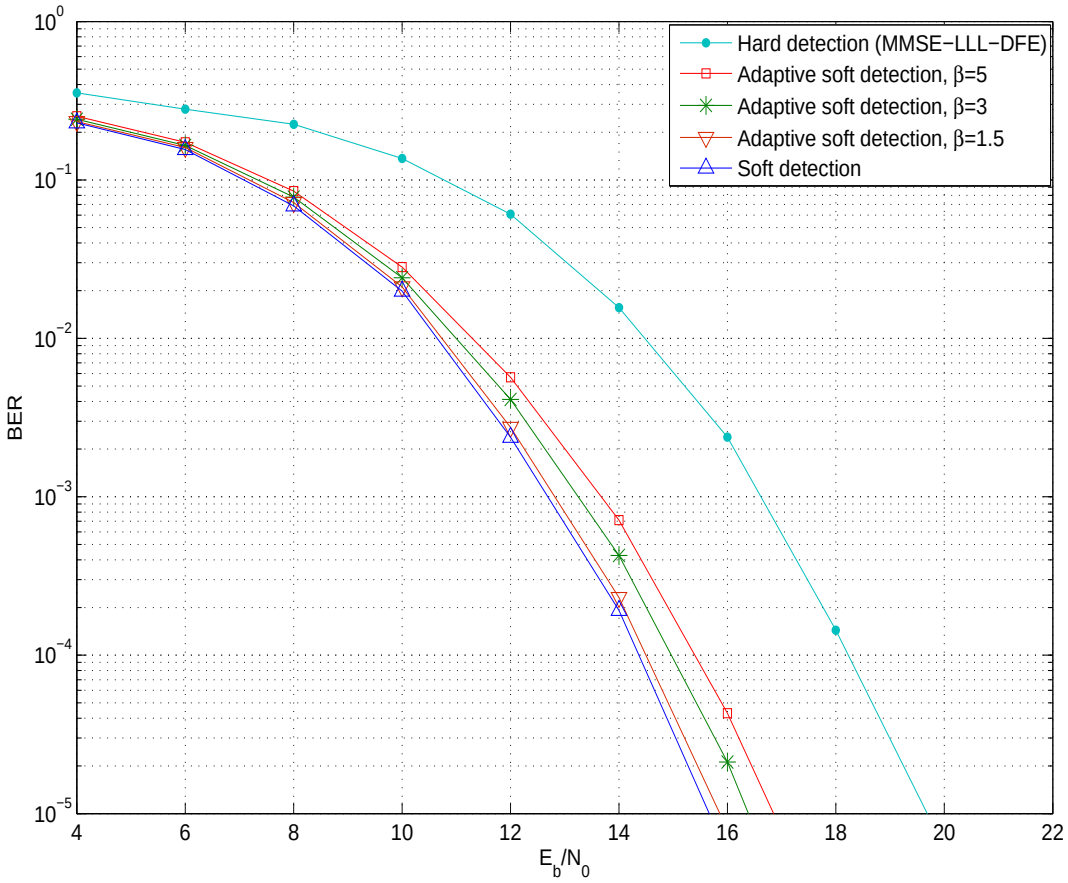


Figure 2.11: Error performance for the adaptive soft-output sphere detection centered on the near-ML point (with no iteration between the detector and the decoder) for different values of β by using a convolutional code with generator polynomials $(5, 7)$ for 16-QAM.

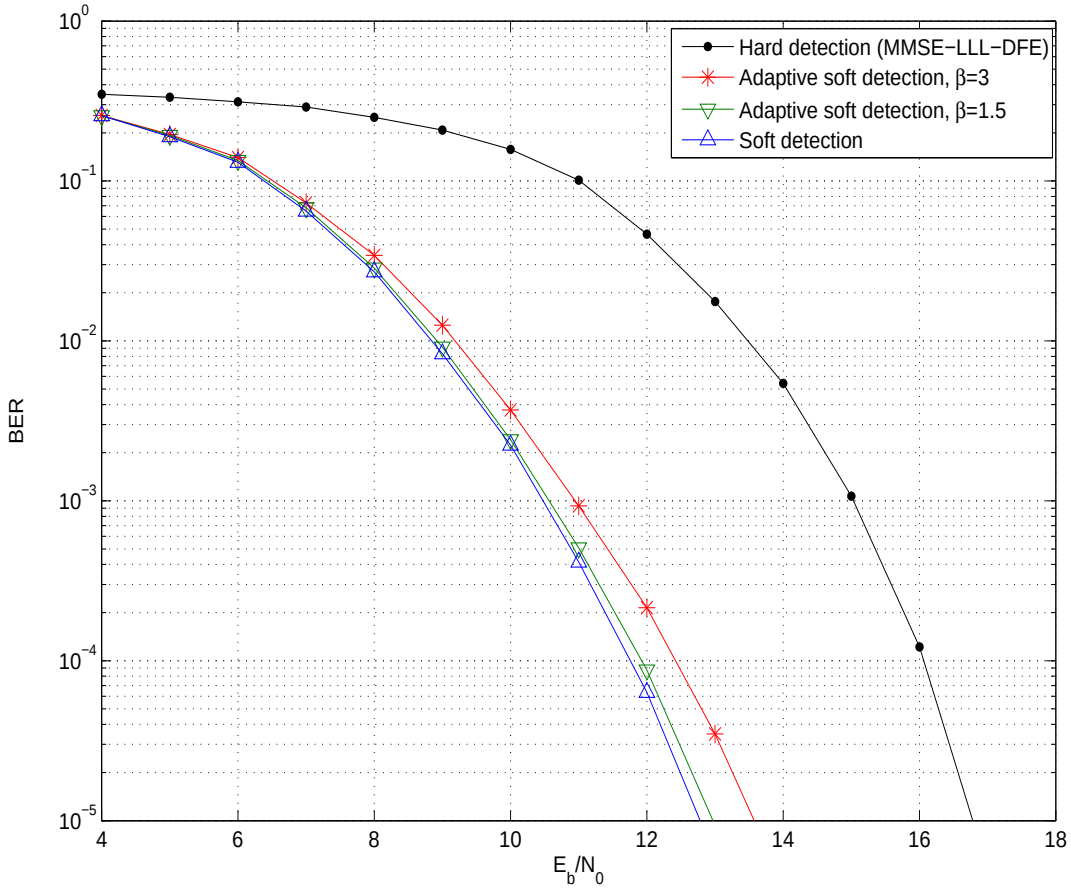


Figure 2.12: Error performance for the adaptive soft-output sphere detection centered on the near-ML point (with no iteration between the detector and the decoder) for different values of β by using a two memory parallel concatenated turbo code (eight iterations inside turbo decoder) for 16-QAM.

2.6 Conclusion and Discussion

Adaptive lattice reduction methods were studied over slowly varying fading channels. Two adaptive methods which improve the complexity of lattice reduction algorithms were introduced and analyzed. The first method achieves the same error performance as the original lattice reduction aided detector. In the second method which is an approximate one, a measure for tradeoff between detection complexity and performance was proposed. It was shown that if the variations in the channel are slow enough, we can use the past transformation matrices to reduce the future channel matrices and convert lattice reduction algorithm to just performing a matrix multiplication without sacrificing the diversity order achieved in an uncoded MIMO system. Next, a reduced complexity list decoder for soft-output detection in MIMO systems was introduced over the slow fading channels. Using the adaptive near-ML estimated point as the center of the sphere in the list decoder, an adaptive list is constructed by means of the past channel information. It was shown that one can save significantly in computational complexity if a shifted list from a past realization is used. Employing an appropriate value for the performance/complexity factor, the soft-output channel detector with different channel decoders was used to investigate the performance of the adaptive list construction.

The adaptive lattice reduction methods can be used in conjunction with any MIMO scenario that requires lattice reduction. In the MIMO broadcast systems, search based precoding scheme is a straightforward method which requires a lattice decoder, [69]. To have a less complex precoder, approximate methods have been proposed in [70] by applying the lattice reduction. When the channel realizations are correlated in time, as the one we are using here, it is possible to take advantage of this correlation to reduce the complexity of the precoding stage in the system. Therefore, the proposed adaptive lattice reduction method can be used in the lattice reduction aided broadcast precoding to reduce the computational complexity.

Chapter 3

Asynchronous Compute-and-Forward

3.1 Introduction

Dealing with interference and noise over wireless relay networks are two challenging problems which have been addressed by different relaying strategies. Mostly, the proposed schemes try to overcome the problems by performing one of the following two actions [32–34]. In the first strategy, such as decode-and-forward, the intermediate nodes try to totally remove the noise. Although this solves one of the problems, the network becomes interference-limited. In the second approach, the intermediate nodes try to repeat the transmitted signal (amplify-and-forward) or quantize the observed signal (compress-and-forward) and then pass it towards the destination in order to form a large multi-antenna channel. However, not performing the decoding results in noise accumulation. A new approach referred to as compute-and-forward was proposed in [15] to efficiently manage the interference and remove the noise at the relay nodes. Although the idea has been proposed before in the two-way relay channels as physical-layer network coding (e.g. [71]), the compute-and-forward scheme assumes a general multiple-access

channel at the intermediate nodes. Furthermore, reliable physical layer network coding is also an application of compute-and-forward [72], [73].

In the compute-and-forward scheme, the same structured codes are applied at the transmitters. Therefore the relays are able to recover integer linear functions of codewords and forward a noiseless version of the transmitted signals to the destination. It is shown in [15] that decoding a linear function of transmitted messages provides an opportunity to achieve higher rates over a network with additive white Gaussian noise. Nested lattice codes from [12] are used to develop such a protocol. In fact, with lattice code, any integer combination of the transmitted codewords is still a codeword. Thus, the intermediate node can decode the combination using a lattice decoder and remove the noise. However, the combination provided by the fading channel has real coefficients rather than integer ones. Scaling the output towards integers and optimizing the problem for maximum achievable rates is proposed in [15]. Note that, while the scaling can reduce the impact of non-integer parts of the channel, it may increase the noise power. It is shown in [74] that lattice implementation of compute-and-forward is interference-limited at high SNR values. Therefore, there is a significant loss in achievable rates compared to the MIMO upper bound resulting from full cooperation at the transmitter and the receiver side. To overcome this problem and match the MIMO upper bound, an asymptomatic compute-and-forward design is presented in [74] based on the real alignment scheme [75].

On the other hand, the lattice implementation of compute-and-forward relaying strategy relies on the algebraic structure of the applied codes to decode a synchronous linear combination of the transmitted messages. However, because of the distributed nature of the relays across the network, random asynchronous combination of the transmitted signals are received and hence, perfect synchronization is not feasible. Therefore, the impact of asynchronism on the performance of this scheme is an important issue both in practice

and in theory and needs to be carefully investigated. In general, time synchronization is an important assumption in achieving the promising gains in many communication systems. Different types of asynchronism have been studied over AWGN channel that show different performance characteristics. For instance, it is shown in [35] that the symbol-asynchronism does not have an impact on the capacity region of the two-user Gaussian multiple-access channel (MAC) with identical shaping waveform at the transmitters. In [36], it is shown that higher mutual information is achievable in an asynchronous space-time coded system with appropriate shaping waveforms. In [37], it is verified that the total capacity of memoryless MAC channel under the frame-asynchronous assumption remains unchanged while it is significantly affected for the channels with memory. Moreover, asynchronism can significantly degrade the cooperative system performance if it is not dealt with appropriately [38], [39]. In this chapter, we study the effect of asynchronous delays on the compute-and-forward rate. We assume that the transmitters are not aware of the asynchronous delays at different relays and the coder and decoder structures are kept identical to the synchronous compute-and-forward scheme.

In Section 3.3, we consider the symbol-asynchronism model in which the delays are assumed to be less than a symbol interval. This model was studied for the multiple-access channel in [35] and used over simple relay networks in [36] and [48]. We show that the ISI, resulted from the asynchronism, imposes additional interference at each relay. If this asynchronous interference is considered as noise, it results in an interference-limited system and the channel output scaling in the compute-and-forward scheme is not effective anymore specially at high SNRs. Therefore, it is useful to remove the asynchronous interference from the received signal and provide an equivalent interference-free model. Based on this idea, a whitening filter is used at the output of the channel to provide a synchronous combination of the transmitted sequences for the decoder of compute-and-forward, but with the cost of reduced channel gain. It is shown that this procedure is

equivalent to extracting the synchronous part of the received signal. However, using the equalizer output with less channel gain results in a gap compared to the synchronous rates, which vanishes at high SNRs. A numerical example for the MAC channel is also presented and it is shown that a simple 1-bit feedback to one of the transmitters fills almost half of the gap in the compute-and-forward rate for all SNRs.

In Section 3.4, we consider an asynchronous model called frame-asynchronism where delays are random multiples of a symbol interval. Over the MAC channel, the impact of frame-asynchronism on the capacity region has been investigated in the literature (cf. [37] and references therein). Also, by using the idea of similar channel codes at the transmitters over a three-node network coding scenario in [49], a practical decoder for the frame-asynchronous model is presented. Over fading channels, a similar model is considered in [50] for the interference channel and as a common scheme, it is called a line-of-sight (LOS) interference channel. For the compute-and-forward scenario, to be able to decode a synchronous sum of the transmitted codewords over the frame-asynchronous network, we propose to use multi-antenna relays with the number of antennas equal to the number of transmitters. Multi-antenna receivers for a synchronous compute-and-forward relaying is studied in [51] where it is shown that one can rotate the channel coefficients toward integers to reduce the impact of the interference from the non-integer parts of the channel. We show that by using extra antennas at the relays, in addition to rotating the channel gains, we can efficiently remove the asynchronous delays. By applying a linear filter whose structure is related to the integer delays prior to the decoder of compute-and-forward, we maximize the achievable rate at any SNR.

3.2 Asynchronous System Models

Consider a relay network with M transmitters and K relays. Because of the distributed structure of this network and the effect of propagation delay, the received signals from different transmitters at each relay are not synchronous. We consider two types of asynchronism in the received signals. At the first part, we assume that the received signals at each relay are frame-synchronous but not symbol-synchronous. Therefore, the transmitted signals from distributed transmitters are aligned up to random delays of length less than a symbol interval T_s. This type of asynchronism may also be resulted from the small amount of time synchronization mismatch in the assumed channel model (using the continuous-time shaping waveforms). At the second part, we consider frame-asynchronous scheme as a general asynchronous model for the delays in propagation. In this manner, each relay receives the transmitted codewords from different transmitters with random delays as finite integer values of the symbol interval.

Assume the continuous-time channel where each transmitter uses a unit energy shaping waveform $\psi_m(t)$ to transmit its codeword symbols to the relays. Thus, the transmitted baseband signal from the m-th transmitter, $m = 1, 2, \ldots, M$, is given by

$$x_m(t) = \sum_{i=1}^{n} x_m(i)\psi_m(t - iT_s), \tag{3.1}$$

where $x_m(i)$ is the transmitted symbol from the m-th transmitter at the i-th symbol interval and n is the length of the resulting transmitted vector $\mathbf{x}_m$. Moreover, each transmitted vector is subject to the power constraint given by

$$\frac{1}{n}\|\mathbf{x}_m\|^2 \leq \text{SNR}, \quad m = 1, 2, \ldots, M. \tag{3.2}$$

The transmitted signals are received asynchronously at each relay. Let τ_{km} denote the delay of the received signal from the m-th transmitter at the k-th relay, $k = 1, 2, \ldots, K$.

The received signal is specified by

$$y_k(t) = \sum_{m=1}^{M} h_{km} x_m(t - \tau_{km}) + z_k(t), \tag{3.3}$$

where h_{km} is the channel coefficient between the m-th transmitter and the k-th relay and $z_k(t)$ is the additive white Gaussian noise signal at the k-th receiver node with zero mean and power spectral density equal to σ^2. We assume that the channel coefficients are constant during the transmission of a codeword (frame). Furthermore, the transmitters do not have channel state information (CSI) and each relay knows its channel coefficients to the transmitters. The same assumption is made for the delay information at both sides.

Consider the symbol-asynchronous channel model. We assume that $0 < \tau_{km} < T_s$ for $k = 1, 2, \ldots, K,\ m = 1, 2, \ldots, M$. The asynchronous delays τ_{km} are continuous-time random variables which depend on the propagation medium. We assume that τ_{km}'s are constant during the transmission of one frame. Assume that the transmitters use the shaping waveforms with a time support equal to pT_s, i.e., $\psi_m(t) = 0, \forall\, t \notin [0, pT_s]$. Using the signal models in (3.1) and (3.3), the received signal at the k-th relay can be obtained as

$$y_k(t) = \sum_{m=1}^{M} h_{km} \sum_{i=1}^{n} x_m(i) \psi_m(t - iT_s - \tau_{km}) + z_k(t). \tag{3.4}$$

To provide the equivalent discrete channel model, the above received signal $y_k(t)$ is passed through M matched filters at each relay sampled at $t = (i+1)T_s + \tau_{km}$, for $m = 1, 2, \ldots, M$. Therefore, the matched filter outputs at each relay result in M streams which can be written as follows:

$$y_{km}(i) = \int_{iT_s + \tau_{km}}^{(i+p)T_s + \tau_{km}} y_k(t) \psi_m^*(t - iT_s - \tau_{km}) dt, \tag{3.5}$$

for $i = 1, \ldots, n$ and $m = 1, 2, \ldots, M$. The integration is performed over $p \geq 1$ symbol intervals which is equal to the length of the shaping waveforms. It can be seen from (3.4) and (3.5) that every symbol $y_{km}(i)$ is interfered by at most $(p-1)$ past and $(p-1)$ future symbols from transmitter m and also by $2p-1$ symbols of past and future of each of the other transmitters. In the next section, we will discuss the output of matched filters in more details for the case when there are two transmitters.

Note that the aforementioned signaling scheme is a general framework and it can be used over any asynchronous network. For example, we will consider it at the end of this chapter over an X network with constant channel coefficients. In the next section, we apply it to the symbol-asynchronous compute-and-forward scheme and for simplicity we assume that all transmitters use the same shaping waveform $\psi(t)$, with a single time support, $p = 1$, i.e., $\psi(t) = 0, \forall\, t \notin [0, T_s]$.

Now consider the frame-asynchronous network where the random delays at each receiver are integer multiples of the symbol interval. The discrete-time channel model at each relay can be written as

$$y_k(i) = \sum_{m=1}^{M} h_{km} x_m(i - d_{km}) + z_k(i). \tag{3.6}$$

where d_{km} is the delay of the received signal from the m-th transmitter at the k-th relay. We assume that d_{km} are drawn i.i.d according to a uniform distribution from the integer interval $\{0, 1, \ldots, D\}$ where D is the maximum delay in the network. Note that each node can be equipped with multiple antennas. In Section 3.4, we consider the frame-asynchronous channel model with single-antenna transmitters and multi-antenna receivers in which the delays are the same across each receiver antennas.

Remark 3.1. When the system is synchronous and hence the delays are equal to zero, then the channel model is simply given by a noisy synchronous combination of the

transmitted vector as

$$\mathbf{y}_k = \sum_{m=1}^{M} h_{km}\mathbf{x}_m + \mathbf{z}_k. \tag{3.7}$$

for $k = 1, 2, \ldots, K$.

3.2.1 Compute-and-Forward

A relay strategy referred to as compute-and-forward is proposed in [15], where each relay decodes a linear equation of the transmitted messages and forward it to a destination. At the destination node, the original messages can be recovered if sufficient equations are available. The key idea is using the same structured codebook at all transmitters so that a linear combination of codewords at each relay is also a codeword. Over a synchronous network, each transmitter uses the same encoder which maps the finite field messages to the real (or complex) field. Each relay receives a linear combination of the transmitted vectors provided by the channel plus additive noise. The goal is to reliably compute a linear combination of the messages at each relay. Let κ_m be the length of the message vector at transmitter m and $\kappa = \max_m \kappa_m$. Vector $\mathbf{w}_m$ of length κ is the zero-padded message of transmitter m over a prime-sized finite field $\mathbb{F}_p$. The encoder at transmitter m maps $\mathbf{w}_m$ to a codeword of length n to provide the channel input $\mathbf{x}_m$. The k-th relay exploits the channel output to decode the following message:

$$\mathbf{u}_k = \sum_{m=1}^{M} q_{km}\mathbf{w}_m,$$

where q_{km} are coefficients in the finite field and mapped to the desired integer coefficients a_{km} at relay k over the channel. In Fig. 3.1, the system between the transmitters and the relays are depicted.

Using nested lattice codes, it is shown in [15] that for a coefficient vector $\mathbf{a}_k \in \mathbb{Z}^M$,

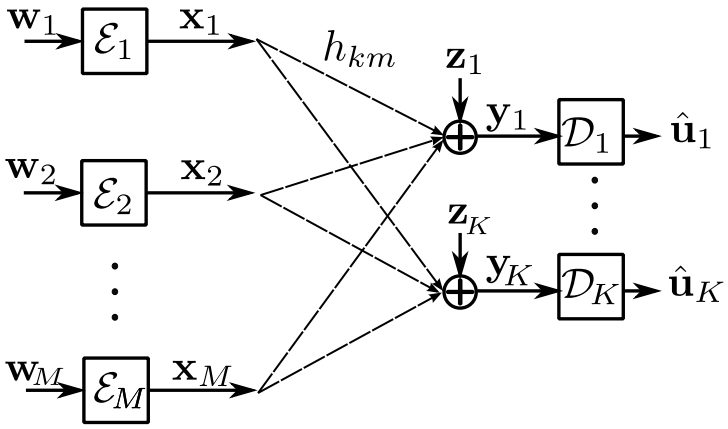


Figure 3.1: Compute-and-forward: computing integer linear functions over the channel

one can achieve the following computation rates:

$$R(\mathbf{h}_k, \mathbf{a}_k) = \left[\frac{1}{2}\log\left(\frac{\mathrm{SNR}}{\alpha_k^2 + \mathrm{SNR}\|\alpha_k\mathbf{h}_k - \mathbf{a}_k\|^2}\right)\right]^+, \tag{3.8}$$

where $\mathbf{h}_k = [h_{k1}, \ldots, h_{km}]^T$, $[R]^+ \triangleq \max(R, 0)$ and $\alpha_k \in \mathbb{R}$. For the choice α_k as the MMSE coefficient, the computation rate in (3.8) is uniquely maximized and given by

$$R(\mathbf{h}_k, \mathbf{a}_k) = \left[\frac{1}{2}\log\left(\left(\|\mathbf{a}_k\|^2 - \frac{\mathrm{SNR}(\mathbf{h}_k^T\mathbf{a}_k)^2}{1 + \mathrm{SNR}\|\mathbf{h}_k\|^2}\right)^{-1}\right)\right]^+. \tag{3.9}$$

The computation rates are achievable if for sufficiently large n, there exist encoders and decoders such that all relays can recover the desired combinations with arbitrarily small average probability of error while the message rates satisfy

$$r_m < \min_{\{k:a_{km}\neq 0\}} R(\mathbf{h}_k, \mathbf{a}_k), \tag{3.10}$$

where $r_m \triangleq (\kappa_m \log q)/n$. Moreover, to achieve the highest computation rates for a given channel vector at each relay, a maximization over integer coefficients can be performed.

3.3 Symbol-Asynchronous Compute-and-Forward

To study the impact of asynchronism on the compute-and-forward scheme, the signaling scheme in Section 3.2 is applied. Consider two transmitters that are using the same shaping waveform in order to send their corresponding codewords to the relays. At each relay, the sum of the delayed transmitted signals multiplied by the channel coefficients plus an additive noise is observed as in (3.3). Note that for the choice of real shaping waveform, the asynchronous coefficients affect the real and the imaginary parts of the signal separately. Thus, all relations are valid for both real and complex system models. For simplicity, we consider a real system model here. Following the system equation in (3.5), by using two matched filters at the k-th relay we obtain

$$y_{k1}(i) = h_{k1}x_1(i) + \lambda_{k1}h_{k2}x_2(i) + \lambda_{k2}h_{k2}x_2(i-1) + z_{k1}(i), \tag{3.11}$$

$$y_{k2}(i) = h_{k2}x_2(i) + \lambda_{k1}h_{k1}x_1(i) + \lambda_{k2}h_{k1}x_1(i+1) + z_{k2}(i), \tag{3.12}$$

where $\lambda_{k1}, \lambda_{k2}$ are defined in (3.13) and $z_{km}(i)$, $m = 1, 2$, is a discrete white Gaussian noise with zero mean and unit variance.

$$\begin{aligned} \lambda_{k1} &= \int_0^{T_s} \psi(t)\psi(t-\bar{\tau}_k)dt, \\ \lambda_{k2} &= \int_0^{T_s} \psi(t)\psi(t+T_s-\bar{\tau}_k)dt, \end{aligned} \tag{3.13}$$

where it is assumed that $\bar{\tau}_k = \tau_{k2} - \tau_{k1} \geq 0$. Furthermore, assume that the rectangular shaping waveform is used, i.e., $\lambda_{k1} + \lambda_{k2} = 1$.

As can be seen in (3.11) and (3.12), the symbol-asynchronous channel at each relay is converted to a multi-antenna multiple-access channel with memory and a correlated noise sequence, $[z_1(i),\ z_2(i)]^T$. Note that the asynchronous signals are added together with different random delays at each relay and the transmitters do not have the delay

information. Thus, no variation at the transmitter side of the synchronous compute-and-forward scheme is considered. To remove the memory and provide a linear sum of the transmitted vectors over a channel plus an additive white noise, a whitening filter is used as follows. By rearranging the channel outputs in (3.11) and (3.12), and writing the system model in a vector form as in [35], we get

$$\tilde{\mathbf{y}}_k = \mathbf{\Gamma}_k \tilde{\mathbf{x}}_k + \tilde{\mathbf{z}}_k, \tag{3.14}$$

where $\tilde{\mathbf{y}}_k$, $\tilde{\mathbf{x}}_k$, $\tilde{\mathbf{z}}_k$ are $2n \times 1$ vectors and $\mathbf{\Gamma}_k$ is a $2n \times 2n$ block diagonal matrix, which are given by

$$\begin{aligned}
\tilde{\mathbf{y}}_k &= [y_{k1}(1), y_{k2}(1), y_{k1}(2), \cdots, y_{k1}(n), y_{k2}(n)]^T, \\
\tilde{\mathbf{x}}_k &= [h_{k1}x_1(1), h_{k2}x_2(1), \cdots, h_{k1}x_1(n), h_{k2}x_2(n)]^T, \\
\tilde{\mathbf{z}}_k &= [z_{k1}(1), z_{k2}(1), z_{k1}(2), \cdots, z_{k1}(n), z_{k2}(n)]^T,
\end{aligned}$$

$$\mathbf{\Gamma}_k = \begin{bmatrix}
1 & \lambda_{k1} & 0 & \cdots & 0 & 0 & 0 \\
\lambda_{k1} & 1 & \lambda_{k2} & 0 & \cdots & 0 & 0 \\
0 & \lambda_{k2} & 1 & \lambda_{k1} & 0 & \cdots & 0 \\
\vdots & & & \ddots & & & \vdots \\
0 & 0 & \cdots & 0 & \lambda_{k2} & 1 & \lambda_{k1} \\
0 & 0 & 0 & \cdots & 0 & \lambda_{k1} & 1
\end{bmatrix}. \tag{3.15}$$

The noise vector is zero mean Gaussian with covariance matrix equal to $\mathbf{\Gamma}_k$. In order to whiten the noise at the output of the matched filters, the Cholesky factorization of the covariance matrix is used as $\mathbf{\Gamma}_k = \mathbf{W}_k^T \mathbf{W}_k$ in which $\mathbf{W}_k$ is an upper triangular matrix [76]. Multiplying the vectors in (3.14) by the whitening filter $\mathbf{W}_k^{-T}$ gives the

equivalent system model with white unit variance noise as

$$\bar{\mathbf{y}}_k = \mathbf{W}_k \tilde{\mathbf{x}}_k + \bar{\mathbf{z}}_k, \tag{3.16}$$

where the new channel matrix is the upper triangular matrix resulted from the Cholesky factorization. Using the definition of Cholesky factorization, it is shown in Appendix A.1 that there are only two non-zero entries at each row of $\mathbf{W}_k$ (except for the last row which has one non-zero entry). The sequence of the entries on odd and even rows converge to two different numbers as n goes to infinity. For our choice of shaping waveform, the limits are $\sqrt{\lambda_{k1}}$ and $\sqrt{\lambda_{k2}}$ for odd and even rows, respectively. Moreover, it is discussed that as far as the decoder of the compute-and-forward scheme with nested lattice codes is used for the channel model in (3.16), one can replace the entries of the varying channel matrix $\mathbf{W}_k$ with the convergence limits and hence obtain the following equivalent equation at each relay:

$$\bar{y}_{k1}(i) = \sqrt{\lambda_{k1}} h_{k1} x_1(i) + \sqrt{\lambda_{k1}} h_{k2} x_2(i) + \bar{z}_{k1}(i), \tag{3.17}$$

where $\bar{z}_{k1}(i)$ is a zero mean unit variance noise vector.

Note that if we are able to change the general structure for the matched filter defined in Section 3.2 and adapt it to the asynchronous delays, the above equations can be obtained directly from a new matched filter output. In this manner, the synchronous and the asynchronous parts of the received sequences are considered separately. For example, for the first equation, one can define

$$y'_{k1}(i) = \int_{iT_s+\tau_{k2}}^{(i+1)T_s+\tau_{k1}} y_k(t)\psi(t - iT_s - \tau_{k1})dt, \tag{3.18}$$

and get the output as

$$y'_{k1}(i) = \lambda_{k1} h_{k1} x_1(i) + \lambda_{k1} h_{k2} x_2(i) + z'_{k1}(i), \tag{3.19}$$

where the noise sequence has a variance equal to λ_{k1}. This procedure is depicted in Fig. 3.2 for the i-th section of the received signal at the k-th relay.

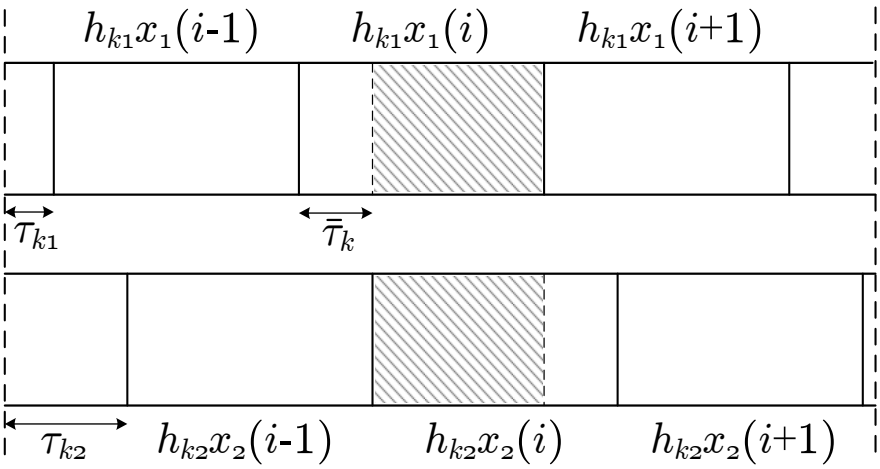


Figure 3.2: The synchronous part of the received signal at the k-th relay.

3.3.1 Achievable Rates

The sensitivity of the computation rates (which relies on the structure of the applied codes) are considered here. By using the aforementioned asynchronous model and the rate regions in [15], the results for the synchronous and the asynchronous receiver structures are given in this part. Note that in comparison with the synchronous scheme, a combination of matched filters output and a whitening filter is added to the asynchronous relay structure. In fact, the filters are applied just as a part of the signaling scheme at each relay to provide useful sequences for the decoder of compute-and-forward.

At first, we compute the achievable rates of the symbol-asynchronous compute-and-

forward with synchronous receiver structure (without the whitening filter). Consider the matched filter outputs in (3.11) and (3.12) at the decoder input. Assume $\mathbf{h}_k^1 = [h_{k1}, \lambda_{k1}h_{k2}]^T$ as the channel gain vector for the signal part in (3.11) and similarly, $\mathbf{h}_k^2 = [\lambda_{k1}h_{k1}, h_{k2}]^T$ in (3.12). By considering another source of interference other than the one from the non-integer part of the channels and computing the MMSE coefficient for maximizing the rate for a given integer vector as in (3.9), the corresponding computation rates can be obtained at each relay. Then, the best synchronous equation from (3.11) and (3.12) is applied to the decoder of compute-and-forward to get the highest computation rate at each relay. This result for two transmitters is summarized in the following theorem:

Theorem 3.1. *The symbol-asynchronous compute-and-forward with synchronous receiver structure achieves the computation rate*

$$R(\mathbf{h}_k, \mathbf{a}_k, \boldsymbol{\lambda}_k) = \max[R_k^1, R_k^2]. \tag{3.20}$$

where, for $m = 1, 2$,

$$R_k^m = \left[\frac{1}{2}\log\left(\left(\|\mathbf{a}_k\|^2 - \frac{SNR(\mathbf{h}_k^{mT}\mathbf{a}_k)^2}{1 + SNR\|\mathbf{h}_k^m\|^2 + SNR(\lambda_{k2}h_{km})^2}\right)^{-1}\right)\right]^+.$$

Now, consider the equalizer at each relay which was introduced in this section. A synchronous equation is provided to the decoder and hence the equivalent channel output in (3.17) is applied which is simply a noisy sum of the two faded signals. Therefore, the computation rate for two transmitters is given by:

Theorem 3.2. *The symbol-asynchronous compute-and-forward with the whitening filter achieves the computation rate* $R'(\mathbf{h}_k, \mathbf{a}_k, \boldsymbol{\lambda}_k)$ *as*

$$\left[\frac{1}{2}\log\left(\left(\|\mathbf{a}_k\|^2 - \frac{\lambda_{k1}SNR(\mathbf{h}_k^T\mathbf{a}_k)^2}{1 + \lambda_{k1}SNR\|\mathbf{h}_k\|^2}\right)^{-1}\right)\right]^+. \tag{3.21}$$

Note that since only the synchronous part of the received signal is exploited, the equivalent channel gains are reduced compared to the synchronous case. Thus, the resulted computation rate is strictly less than the one for the synchronous system in (3.8), but as it can be seen in (3.21), the gap vanishes as SNR increases.

Remark 3.2. The above results can be generalized to more than two transmitters with different choices of shaping waveforms. The synchronous part and different combinations of the asynchronous part of the received signal are provided to the decoder accordingly. In the general form, the matched filter output for the synchronous part can be written as

$$y'_{k1}(i) = \sum_{m=1}^{M} \alpha_{km} h_{km} x_m(i) + z'_{k1}(i), \tag{3.22}$$

where α_{km} is related to the selected shaping waveforms at the transmitters and the asynchronous delays at each relay.

3.3.2 Numerical Results

The symbol-asynchronous compute-and-forward is considered for the two-user multiple-access channel. It is assumed that the transmitters do not have any information about the channel or the computation rates which leads to an outage probability. For a given outage probability, the achievable computation rates are computed for different SNRs. The computation rates are maximized over the integer vectors. Moreover, it is assumed that none of the integer coefficients are zero and the receiver always have a linear combination of signals from both transmitters.

As it can be seen from (3.21), a larger relative delay results in less computation rate. To lessen the effect of a large delay, a 1-bit feedback scheme is considered here in which

the receiver asks one of the transmitters to send its data with one symbol delay if it results in a higher computation rates. In this way, the system with close to one symbol delay will be converted to an almost synchronous system. Note that for more than two transmitters, different combinations of relative asynchronous delays can be examined for the 1-bit feedback scenario to maximize the computation rate. The performance for synchronous compute-and-forward and asynchronous compute-and-forward with different receiver structures for an outage probability equal to 0.3 are depicted in Fig. 3.3. It can be seen that with the synchronous receiver, the asynchronous interference results in an interference-limited system. Moreover, by applying the whitening filter, one can see the loss in the achievable rate of the asynchronous compute-and-forward compared to the synchronous case. However, the resulting gap vanishes as the SNR increases. Additional gains for the 1-bit feedback model is also noticeable.

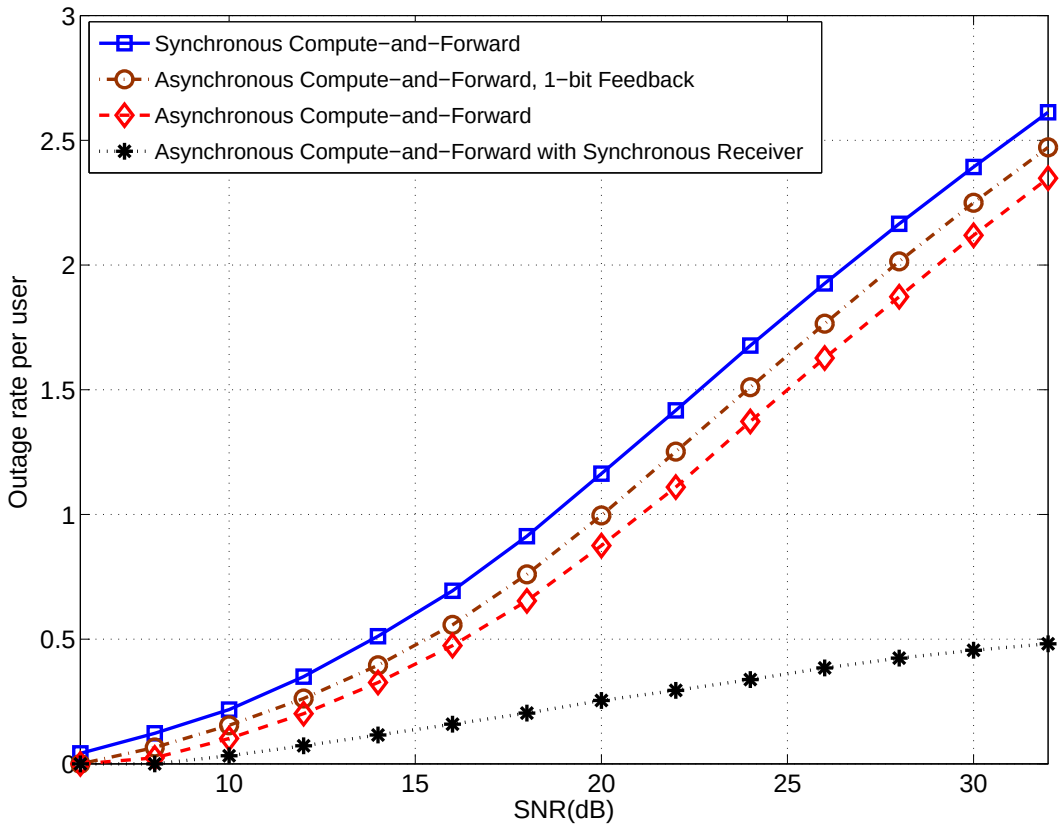


Figure 3.3: Outage rates of the symbol-asynchronous compute-and-forward maximized over non-zero integer coefficients for an outage probability of 0.3.

3.4 Frame-Asynchronous Compute-and-Forward

In the previous section, we considered an asynchronous network where delays are less than a symbol interval. In order to study the impact of asynchronism with larger delays on the lattice coding implementation of compute-and-forward, in this section, we assume a basic asynchronous model where delays are arbitrary (finite) multiples of a symbol interval. To be able to decode a synchronous sum of the transmitted codewords on a frame-asynchronous network with fixed channel gains, we propose to use multi-antenna receivers. Over a network with M single-antenna transmitters, we assume that each receiver equipped with M antennas, equal to the number of transmitters. Multi-antenna receivers in compute-and-forward relaying is studied in [51] for a synchronous network where it is shown that one can rotate the channel coefficients toward integers to reduce the impact of the interference from the non-integer parts of the channel on the achievable rates. Here, we show that by using extra antennas at the receivers, in addition to rotating the channel gains we can efficiently remove the asynchronous delays.

To be more specific, consider the frame-asynchronous system model in (3.6) for the case of multi-antenna receivers. The received signal at receiver k at time i is specified by

$$\mathbf{y}_k(i) = \mathbf{H}_k \mathbf{x}_k(i) + \mathbf{z}_k(i), \tag{3.23}$$

where $\mathbf{H}_k = [\mathbf{h}_{k1}, \mathbf{h}_{k2}, \cdots, \mathbf{h}_{kM}]$ and $\mathbf{h}_{km} \in R^M$ is the channel vector between the transmitter m and the receiver k antennas, and $\mathbf{x}_k(i)$ is the asynchronous vector of transmitted symbols with different delays as

$$\mathbf{x}_k(i) = [x_1(i - d_{k1}), x_2(i - d_{k2}), \cdots, x_M(i - d_{kM})]^T. \tag{3.24}$$

Note that we assume that all antennas at the receiver obtain the symbols with the same

delay from a specific transmitter. We can also add cyclic prefix symbols (CPS) to the beginning of each transmitted vectors with length equal to the maximum delay in the system, D. In this case, regardless of the value of delays, each receiver can initiate the decoding after D symbols and take the next n symbols for processing. Since the delays are finite, the rate loss due to the adding CPS is negligible as the codeword length goes to infinity. In this manner, the total n-block system equation is given by

$$\tilde{\mathbf{y}}_k = \mathbf{G}_k \tilde{\mathbf{x}}_k + \tilde{\mathbf{z}}_k, \tag{3.25}$$

where $\mathbf{G}_k$ is a block circulant $Mn \times Mn$ matrix generated from $\mathbf{H}_k$ based on the asynchronous delays, $\tilde{\mathbf{x}}_k$ is the sequence of transmitted symbols as

$$\tilde{\mathbf{x}}_k = [x_1(1), x_2(1), \cdots, x_M(1), \cdots, x_1(n), x_2(n), \cdots, x_M(n)]^T, \tag{3.26}$$

and $\tilde{\mathbf{y}}_k$, $\tilde{\mathbf{z}}_k$ are $Mn \times 1$ vectors of received symbols and noise at different antennas respectively which, by using (3.23), are defined as follows:

$$\begin{aligned} \tilde{\mathbf{y}}_k &= [\mathbf{y}_k^T(1), \mathbf{y}_k^T(2), \cdots, \mathbf{y}_k^T(n)]^T, \\ \tilde{\mathbf{z}}_k &= [\mathbf{z}_k^T(1), \mathbf{z}_k^T(2), \cdots, \mathbf{z}_k^T(n)]^T. \end{aligned} \tag{3.27}$$

Considering the above system model, the receivers are able to convert the asynchronous channel with real coefficients to a synchronous channel with integer coefficients by multiplying the matrix $\mathbf{B}_k$ before the decoding. In this manner, we have $\mathbf{B}_k \mathbf{G}_k = \mathbf{A}_k$ where $\mathbf{A}_k = \mathbf{I}_n \bigotimes \mathbf{a}_k^T$ and $\bigotimes$ is the kronecker product. Therefore, the resulted channel output is the synchronous linear sum of the transmitted lattice points and can be applied to the lattice decoder. However, although we are able to remove the non-integer parts of the channel and the asynchronous delays simultaneously, a higher rate at finite SNR may be achievable if we just remove the asynchronous delays and try to minimize the the sum

of the noise and the non-integer interference. In other words, if we put $\mathbf{B}_k = \mathbf{Q}_k\mathbf{G}_k^{-1}$, where $\mathbf{Q}_k = \mathbf{I}_n \bigotimes \mathbf{q}_k^T$ and $\mathbf{q}_k$ is a $M \times 1$ real vector, then for a given channel matrix $\mathbf{G}_k$ and a desired integer vector $\mathbf{a}_k$, one can minimize the total noise plus interference over the decoder multiplying matrix $\mathbf{B}_k$ (or $\mathbf{b}_k$ since the structure of $\mathbf{B}_k$ is fixed for a given delay vector $\mathbf{d}_k$). In this manner, the synchronous system equation after the filter is given by

$$\mathbf{y}'_k = \sum_{m=1}^{M} q_{km}\mathbf{x}_m + \mathbf{z}'_k. \tag{3.28}$$

Note that the additive noise $\mathbf{z}'_k$ is now a self-correlated Gaussian noise with power equal to $||\mathbf{b}_k||^2$, where $\mathbf{b}_k^T$ is a row of $\mathbf{B}_k$ with at most M^2 nonzero elements in case of distinct delays, and covariance matrix $\mathbf{B}_k\mathbf{B}_k^T$. Moreover, the density of the non-integer interference term is upper bounded in [15] by the density of a white Gaussian vector with variance $\mathrm{SNR}||\mathbf{q}_k - \mathbf{a}_k||^2$. Since the noise is independent of the signal, we can still use the nearest lattice point in Euclidean distance at the decoder. Thus, it is shown in Lemma A.3 that over the channel with colored noise, we can achieve the same rates as the channel with white Gaussian noise at the same power. Therefore, the equivalent noise variance at the decoder of compute-and-forward is

$$||\mathbf{b}_k||^2 + \mathrm{SNR}||\mathbf{q}_k - \mathbf{a}_k||^2. \tag{3.29}$$

Using this result, we have the following theorem for the achievable rates:

Theorem 3.3. *Applying the above filter before decoding, the frame-asynchronous compute-and-forward with M antennas at each receiver and no CSI at M transmitters achieves the computation rate*

$$R(\mathbf{H}_k, \mathbf{a}_k, \mathbf{d}_k) = \max_{\mathbf{b}_k} \left[\frac{1}{2}\log\left(\frac{SNR}{||\mathbf{b}_k||^2 + SNR||\mathbf{q}_k - \mathbf{a}_k||^2}\right)\right]^+, \tag{3.30}$$

and for given $\mathbf{H}_k, \mathbf{a}_k$ and $\mathbf{d}_k$, this rate is uniquely maximized by choosing $\mathbf{b}_k$ as the

solution to the equality constrained quadratic programming problem

$$\begin{aligned} &\min_{\mathbf{b}} \quad (||\mathbf{b}||^2 + SNR||\mathbf{G}_1\mathbf{b} - \mathbf{a}||^2) \\ &s.t. \quad \mathbf{G}_2\mathbf{b} = \mathbf{0}, \end{aligned} \tag{3.31}$$

where $\mathbf{G}_1$ *and* $\mathbf{G}_2$ *are specified by the channel matrix and asynchronous delays at receiver* k*.*

The proof is given in Appendix A.3.

To clarify the above expression, consider an example of two-user frame-asynchronous MAC where the receiver is equipped with two antennas. Assume that the asynchronous delays are $d_1 = 0$ and $d_2 = 1$. Thus, the symbols from the second transmitter is received with one symbol delay with respect to the ones from the first transmitter. Consider the channel matrix $\mathbf{H}$ in (3.23) with columns $\mathbf{h}_1 = [h_{11}, h_{21}]^T$ and $\mathbf{h}_2 = [h_{12}, h_{22}]^T$. Applying the aforementioned structure at the decoder, the first row of matrix $\mathbf{B}$ has first four nonzero entries which can be defined by $\mathbf{b} = [b_1, b_2, b_3, b_4]^T$ and the next rows are simply the cyclic shift of $\mathbf{b}$ by two entries to the right. To obtain the desired vector $\mathbf{q}$ before the decoder of compute-and-forward, there is a set of four equations which needed to be satisfied:

$$\begin{cases} h_{11}b_1 + h_{21}b_2 = q_1 \\ h_{12}b_3 + h_{22}b_4 = q_2 \end{cases}, \quad \begin{cases} h_{11}b_3 + h_{21}b_4 = 0 \\ h_{12}b_1 + h_{22}b_2 = 0 \end{cases}. \tag{3.32}$$

Therefore, to uniquely minimize $||\mathbf{b}||^2 + \text{SNR}||\mathbf{q} - \mathbf{a}||^2$ over $\mathbf{b} \in \mathbb{R}^4$ in (3.30), one can use the above conditions and form the optimization problem in (3.31) with the matrices $\mathbf{G}_1$ and $\mathbf{G}_2$ given by

$$\mathbf{G}_1 = \begin{bmatrix} h_{11} & h_{21} & 0 & 0 \\ 0 & 0 & h_{12} & h_{22} \end{bmatrix}, \quad \mathbf{G}_2 = \begin{bmatrix} 0 & 0 & h_{11} & h_{21} \\ h_{12} & h_{22} & 0 & 0 \end{bmatrix}. \tag{3.33}$$

Note that with the synchronous receiver, the optimization is in fact an unconstrained problem and the optimal solution will be simplified to $\mathbf{b}_{\text{opt}} = (\frac{1}{\text{SNR}}\mathbf{I} + \mathbf{H}^T\mathbf{H})^{-1}\mathbf{Ha}$ in $\mathbb{R}^2$ as in synchronous MIMO compute-and-forward.

3.4.1 Numerical Results

The achievable rates for the frame-asynchronous compute-and-forward with a multiple-antenna receiver is considered here. In Fig. 3.4 and Fig. 3.5, we compare the average rate over the channel states for the synchronous and the frame-asynchronous case with a two-antenna and a three-antenna receiver (two transmitters and three transmitters), respectively. It can be seen that forcing the structure of the filter to remove the asynchronism results in a gap in the achievable rates which becomes larger as the number of users increases. Note that at high SNR values, the optimal solution results in removing all the non-integer part of the channel. Therefore, the gap between the synchronous and the asynchronous rates for a given number of users becomes constant as the SNR increases.

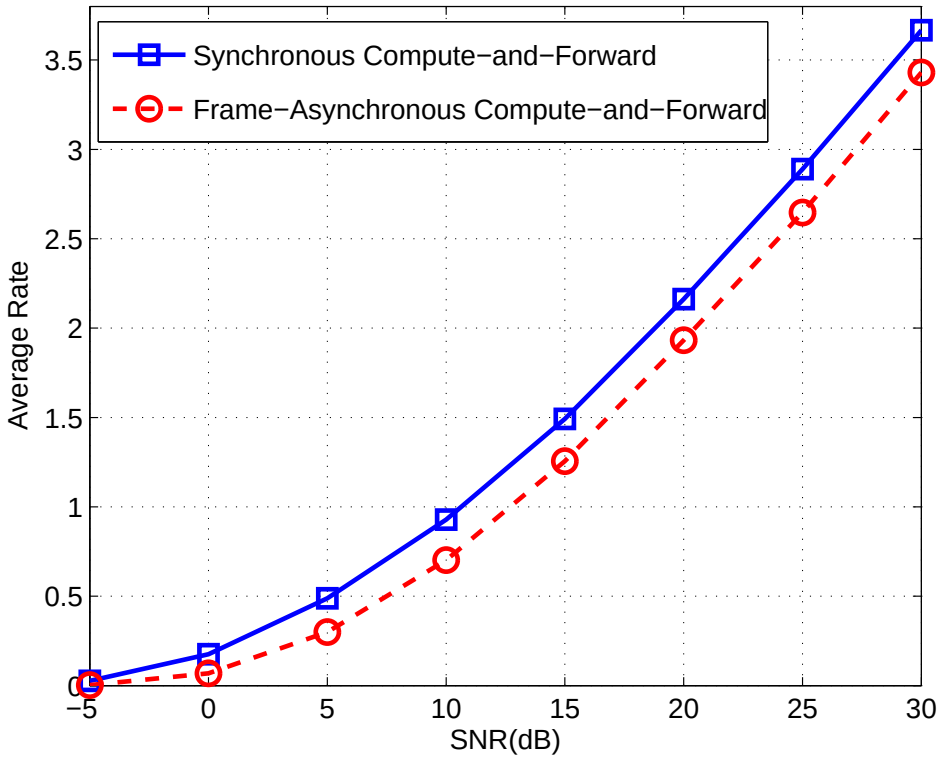


Figure 3.4: Average rate of the frame-asynchronous compute-and-forward with a two-antenna receiver, integer vector $\mathbf{a} = [1, 1]^T$ and delay vector $\mathbf{d} = [0, 1]^T$.

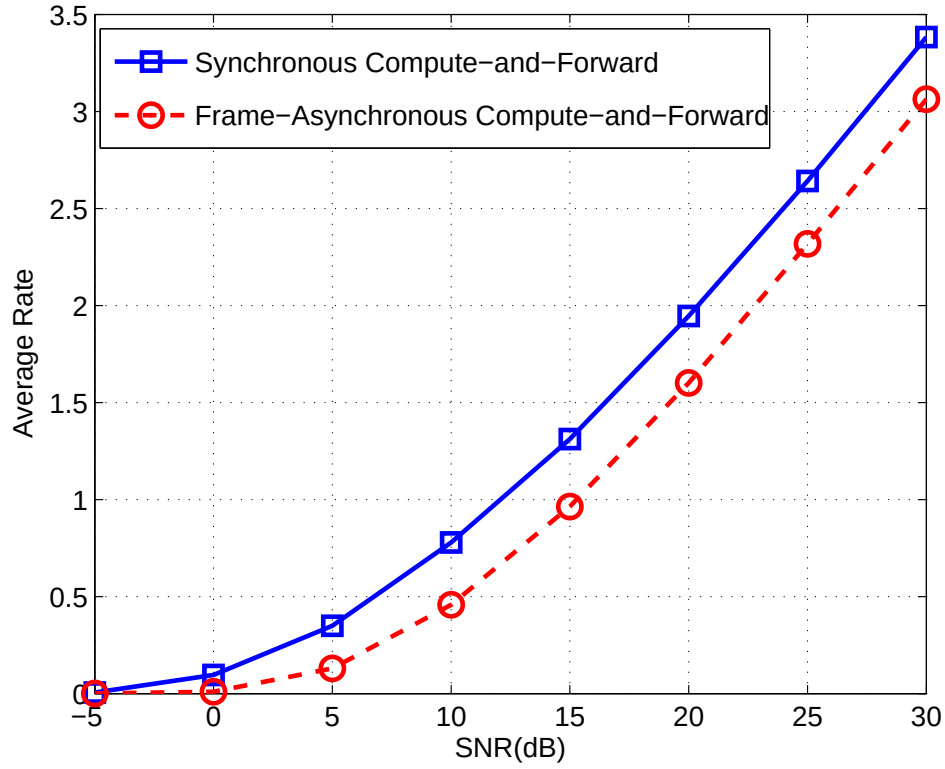


Figure 3.5: Average rate of the frame-asynchronous compute-and-forward with a three-antenna receiver, integer vector $\mathbf{a} = [1, 1, 1]^T$ and delay vector $\mathbf{d} = [0, 1, 2]^T$.

3.5 Asynchronous Application in Interference Alignment over X Networks

The X channel is a general interference network in which there is an independent message from every transmitter to every receiver. Characterizing the capacity region of the interference and the X channel is an important and challenging problem in information theory. The problem of locating this region, even in a simple case of two-user Gaussian interference channel, is still unsolved [40–42]. A fundamental measure to approximate the capacity of wireless channels at high values of SNR is known as the degrees of freedom which determines the capacity boundaries as the SNR increases. The the degrees of freedom or the multiplexing gain is defined as the ratio of the capacity to the logarithm of the SNR as SNR tends to infinity. Since it impacts the design of efficient methods of interference management, investigating the DOF of distributed networks is an important issue in wireless communications. Interference alignment is an intelligent way to manage the interference at the receivers by restricting the undesired signals at some common directions. It was first introduced in [43] wherein its capability in achieving the total number of degrees of freedom of a class of two-user X channels was studied. It is shown in [43] that in a two-user X channel with M antennas at each node, a total of $\lfloor \frac{4M}{3} \rfloor$ degrees of freedom is achievable. This interesting result was then improved in [44] by employing the idea of channel extensions to achieve the total $\frac{4M}{3}$ degrees of freedom almost surely over channels with constant coefficients. Interference alignment was then applied to the K-user interference channel in [45] and for the $M \times N$ user X network in [46]. It is shown in [46] that by using the symbol extension over the channel with varying coefficients, the upper bound of the total number of degrees of freedom is achievable for the general X network with single antenna nodes. In [77], for the K-user interference channel with time-varying fading, it is shown that by pairing proper channels one can achieve at least half the interference-free ergodic capacity at any SNR.

However, the channel variation is a crucial assumption for the mentioned achievable schemes. It is argued in [44] that in a single antenna two-user X channel, at least one link must have the variations in time/frequency in order to perform the alignment task. Thus, these schemes could not trivially be extended to channels with constant coefficients. Interference alignment over a time-invariant two-user X channel with single antenna nodes is addressed recently in [18] and [78] where the upper bound of $\frac{4}{3}$ for the total number of degrees of freedom was shown to be achieved. In [18], by using the properties of real numbers and incorporating rational dimensions in the transmission, the authors have shown that it is possible to perform an alignment scheme called real interference alignment in the single dimensional systems. Also, [78] shows that with asymmetric complex signaling, the upper bound of $\frac{4}{3}$ is achievable for the complex Gaussian channel with constant fading coefficients almost surely.

On the other hand, perfectly synchronized nodes is an important assumption in the aforementioned alignment schemes. Mostly, it has been assumed that the nodes are synchronous such that the received signals at each receiver are aligned with respect to their symbols. However, due to the distributed nature of the X network, perfect synchronization may not be feasible in many cases. In fact, asynchronism inherently exists in distributed communication systems and the receivers obtain an asynchronous combination of the transmitted signals. Interference alignment over asynchronous networks based on propagation delays has been considered before. It was first proposed in [45] as an example and then explored in [52] by proper node placement in a network with four nodes to align the interference signals. In [50], a K-user interference channel is modeled by a time indexed graph where the alignment task is associated with finding the maximal independent set of the graph. A signaling scheme for the interference alignment over the asynchronous K user interference channel was also proposed in [53] where it was shown that the total degrees of freedom of $K/2$ is achievable.

In this section, by exploiting the asynchronous delays, achievable schemes are investigated for the X network with constant channel coefficients. We consider an $M \times N$ symbol-asynchronous X network with single antenna users and study the effect of the asynchronism among the users on the total number of degrees of freedom of this network and show that is the same as that of the corresponding synchronous network. Then, by using the symbol asynchronism over the network with time-invariant channel coefficients, we achieve the upper bound for the total number of degrees of freedom. In fact, asynchronous delays provide the variation required for the interference alignment over the constant channel. The interference alignment scheme is implemented by using the asynchronous delays in the received signals at each receiver which results in ISI among the symbols from different transmitters and hence provides the channel variation required for the vector alignment. Moreover, the channel state information of the links has no impact on the design of the alignment scheme. The relative delays among the received signals at each receiver only need to be globally known to all nodes.

3.5.1 Asynchronous X Networks

Consider an $M \times N$ user X network with M transmitters and N receivers where each node is equipped with a single antenna. Over the X network, there is a total MN independent messages, one from every transmitter to every receiver. Considering the distributed nature of the network and the effect of propagation delay, the received signals at each receiver are not synchronous. In this section, we assume that the received signals are frame-synchronous but not symbol-synchronous.

Using the symbol-asynchronous model, at each receiver, the transmitted signals are aligned up to delays of length less than a symbol interval, T_s. To be more specific, let τ_{km} denote the delay of the received signal from the m-th transmitter at the k-th receiver node. We assume $0 < \tau_{km} < T_s$, $k = 1, 2, \ldots, N$ and $m = 1, 2, \ldots, M$. The

delays are continuous-time random variables which depend on the propagation medium. We assume that τ_{km}'s are constant during the transmission of a frame. We define the relative delay between the transmitted signal from the u-th and the m-th transmitters at the k-th receiver as

$$\tau_{u,m}^{[k]} \triangleq \tau_{ku} - \tau_{km}, \tag{3.34}$$

where $k\{1, 2, \ldots, N\}$ and $u, m \in \{1, 2, \ldots, M\}$. Since the asynchronous delays, τ_{km}'s, are continuous independent random variables spanned over a symbol interval, $\tau_{u,m}^{[k]}$'s are distinct with probability one. To perform the alignment task, the relative delays among the received signals at each receiver need to be globally known to all nodes.

We assume that each transmitter uses a unit energy shaping waveform $\psi_m(t)$, $m = 1, 2, \ldots, M$, to transmit its signal. Therefore, the transmitted signal from the m-th transmitter is given by

$$x_m(t) = \sum_{i=0}^{\ell-1} x_m(i)\psi_m(t - iT_s), \tag{3.35}$$

where $x_m(i)$ is the transmitted symbol from the m-th transmitter at the i-th symbol interval and ℓ is the length of the transmitted vector over the channel. The transmitted signals are received asynchronously at each receiver node. Thus, the received signal at the k-th receiver is modeled as

$$y_k(t) = \sum_{m=1}^{M} h_{km}x_m(t - \tau_{km}) + z_k(t), \ k = 1, 2, \ldots, N, \tag{3.36}$$

where h_{km} is the channel coefficient between the m-th transmitter and the k-th receiver and $z_k(t)$ is the noise signal at the k-th receiver node. It is assumed that the channel coefficients are non-zero finite random variables and they are constant during the transmission of a codeword.

Assume that the average power at each transmitter is equal to ρ. Let $R_{km}(\rho)$, for $k \in \{1, 2, \ldots, N\}$ and $m \in \{1, 2, \ldots, M\}$, be the transmission rate for the corresponding

message from transmitter m to receiver k and $\mathcal{C}(\rho)$ be the capacity region of the X network (i.e., the set of all achievable rates at this SNR). The region of degrees of freedom, $\mathcal{D}$, for the $M \times N$ user X network is defined as the set of all real positive matrices $[(d_{km})] \in \mathbb{R}_+^{M\times N}$ such that $\forall[(\alpha_{km})] \in \mathbb{R}_+^{M\times N}$,

$$\sum_{\substack{k\in\{1,2,\ldots,N\},\\ m\in\{1,2,\ldots,M\}}} \alpha_{km} d_{km} \leq \limsup_{\rho\to\infty} \left[\sup_{[(R_{km})](\rho)\in\mathcal{C}(\rho)} \frac{1}{\log\rho} \sum_{\substack{k\in\{1,2,\ldots,N\},\\ m\in\{1,2,\ldots,M\}}} \alpha_{km} R_{km}(\rho) \right], \tag{3.37}$$

where $[(R_{km})](\rho)$ is an achievable rate-matrix [46]. Finally, the total number of degree of freedom defined as

$$\max_{[(d_{km})]\in\mathcal{D}} \sum_{\substack{k\in\{1,2,\ldots,N\},\\ m\in\{1,2,\ldots,M\}}} d_{km}. \tag{3.38}$$

We assume that all transmitters use the same shaping waveform, $\psi(t)$, that has a time support equal to pT_s, i.e., $\psi(t) = 0, \forall\, t \notin [0, pT_s]$. At the m-th transmitter, a codeword of length n, $\mathbf{x}_m = [x_m(0), x_m(1), \ldots, x_m(n-1)]^T$, is supported by cyclic prefix and cyclic suffix symbols (CPS) each of length $p+1$ such that the first and the last $p+1$ symbols of $\mathbf{x}_m$ are repeated at the end and at the beginning of this vector, respectively. Resulted vector $\mathbf{x}_m^{cps} = [x_m(n-p-1), x_m(n-p), \ldots, x_m(n-1), x_m(0), x_m(1), \ldots, x_m(n-1), x_m(0), x_m(1), \ldots, x_m(p)]^T$ of length $\ell = n + 2(p+1)$ is then transmitted over the channel. Using the signal model in (3.35) and (3.36), the received signal at the k-th receiver can be written as

$$y_k(t) = \sum_{m=1}^{M} h_{km} \sum_{i=0}^{\ell-1} x_m^{cps}(i)\psi(t - iT_s - \tau_{km}) + z_k(t), \tag{3.39}$$

where $x_m^{cps}(i)$ is the i-th entry of the vector $\mathbf{x}_m^{cps}$.

This signal is then passed through a matched filter at each receiver. Without loss of generality, we assume that the output of the matched filter at the k-th receiver is

sampled at $t = (i+1)T_s + \tau_{k1}$, where $i = 0, \ldots, \ell - 1$. Note that unlike the relay structure at the Section 3.3 where M matched filters were used, here, we just use one matched filter at each receiver. Thus, we get

$$\begin{aligned} y_k(i) &= \int_{iT_s+\tau_{k1}}^{(i+p)T_s+\tau_{k1}} y_k(t)\psi^*(t - iT_s - \tau_{k1})dt \\ &= \sum_{m=1}^{M} h_{km} \sum_{q=-p}^{p} \gamma_{km}(q) x_m^{cps}(i+q) + z_k(i), \end{aligned} \tag{3.40}$$

where $x_m^{cps}(q) = 0$, $\forall q < 0$ and

$$\begin{aligned} \gamma_{km}(q) &= \int_0^{pT_s} \psi(t - qT_s + \tau_{1,m}^{[k]})\psi^*(t)dt, \\ z_k(i) &= \int_{iT_s+\tau_{k1}}^{(i+p)T_s+\tau_{k1}} z_k(t)\psi^*(t - iT_s - \tau_{k1})dt. \end{aligned} \tag{3.41}$$

Note that the shaping waveform has a length equal to $p \geq 1$ symbol intervals. Therefore, in each transmitted stream every symbol is interfered by $(p-1)$ previous and $(p-1)$ future symbols of the same stream and also by $2p-1$ symbols of each of the other streams.

By discarding CPS symbols at the output of the matched filter at each receiver, we obtain

$$\mathbf{y}_k = \sum_{m=1}^{M} h_{km} \mathbf{\Gamma}_{km} \mathbf{x}_m + \mathbf{z}_k, \quad k = 1, 2, \ldots, N, \tag{3.42}$$

where

$$\begin{aligned} \mathbf{x}_m &= [x_m(0), x_m(1), \ldots, x_m(n-1)]^T, \\ \mathbf{y}_k &= [y_k(p+1), y_k(p+2), \ldots, y_k(n+p)]^T, \\ \mathbf{z}_k &= [z_k(p+1), z_k(p+2), \ldots, z_k(n+p)]^T, \end{aligned} \tag{3.43}$$

Matrix $\mathbf{\Gamma}_{km}$ is the circulant convolution matrix of generator sequence $\hat{\boldsymbol{\gamma}}_{km}$ where $\hat{\boldsymbol{\gamma}}_{km} =$

$[\gamma_{km}(0), \gamma_{km}(1), \ldots, \gamma_{km}(p), 0, \ldots, 0, \gamma_{km}(-p), \ldots, \gamma_{km}(-1)]^T$ of length n and $\mathbf{z}_k$ is a colored noise vector. For all k, m, matrix $\mathbf{\Gamma}_{km}$ is given in (3.44). It can simply be verified that n should be large enough such that $n \geq 2p$. Note that adding cyclic prefix and cyclic suffix, makes the processed channel, $h_{km}\mathbf{\Gamma}_{km}$, circulant. In fact, by using the effect of asynchronism among the users, the original quasi-static links with constant coefficients $h_{km}\mathbf{I}_N$ are converted into variable links with correlated coefficients over time as $h_{km}\mathbf{\Gamma}_{km}$.

$$\begin{bmatrix} \gamma_{km}(0) & \cdots & \gamma_{km}(-p+1) & \gamma_{km}(-p) & 0 & \ldots & 0 & \gamma_{km}(p) & \gamma_{km}(p-1) & \ldots & \gamma_{km}(1) \\ \gamma_{km}(1) & \cdots & \cdots & \gamma_{km}(-p+1) & \gamma_{km}(-p) & 0 & \ldots & 0 & \gamma_{km}(p) & \ldots & \gamma_{km}(2) \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \gamma_{km}(p) & \cdots & \gamma_{km}(1) & \gamma_{km}(0) & \cdots & \gamma_{km}(-p) \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \gamma_{km}(-1) & \cdots & \gamma_{km}(-p) & 0 & \cdots & \cdots & 0 & \cdots & \gamma_{km}(p) & \cdots & \gamma_{km}(0) \end{bmatrix}. \tag{3.44}$$

Considering the eigenvalue decomposition of a circulant matrix, [79], we obtain

$$\mathbf{\Gamma}_{km} = \mathbf{U}^H \mathbf{\Lambda}_{km} \mathbf{U}, \tag{3.45}$$

where $\mathbf{U}$ is the discrete Fourier transform (DFT) matrix of dimension n given by

$$\mathbf{U}(q, s) = \frac{1}{\sqrt{n}} e^{-j\frac{2\pi(q-1)(s-1)}{n}}, \ q, s = 1, 2, \ldots, n, \tag{3.46}$$

and $\mathbf{\Lambda}_{km}$ is a diagonal matrix containing the elements of the DFT of the generator sequence of $\mathbf{\Gamma}_{km}$. i.e., $\mathbf{\Lambda}_{km} = \mathtt{diag}\{\lambda_{km}(0), \lambda_{km}(1), \ldots, \lambda_{km}(n-1)\}$, where

$$\lambda_{km}(i) = \sum_{q=0}^{n-1} \hat{\gamma}_{km}(q) e^{-j\frac{2\pi}{n}qi}, \quad i = 0, \ldots, n-1, \tag{3.47}$$

and $\hat{\gamma}_{km}(q)$ is the q-th element of $\hat{\boldsymbol{\gamma}}_{km}$. Using (3.45), we get

$$\mathbf{y}'_k = \sum_{m=1}^{M} h_{km}\boldsymbol{\Lambda}_{km}\mathbf{x}'_m + \mathbf{z}'_k, \quad k = 1, 2, \ldots, N, \tag{3.48}$$

where $\mathbf{y}'_k, \mathbf{x}'_m, \mathbf{z}'_k$ are the linear transformations of $\mathbf{y}_k, \mathbf{x}_m, \mathbf{z}_k$ by the DFT matrix $\mathbf{U}$, respectively. We refer to the channel model in (3.48) as the transformed channel throughout the paper.

It is shown in [53] that for a well-designed waveform, $\boldsymbol{\Lambda}_{km}$'s are diagonal matrices with non-zero bounded diagonal elements. Therefore, one can interpret equation (3.48) as a model of the received signal at the k-th receiver of a synchronous X network with varying fading coefficients. Moreover, it is demonstrated that by using an appropriate shaping waveform with sub-linear decaying rate in time, $\boldsymbol{\Lambda}_{km}$ can be approximated as follows

$$\boldsymbol{\Lambda}_{km} = \boldsymbol{\Lambda}_0 \mathbf{E}\Big(\hat{\tau}_{1,m}^{[k]}\Big) + \epsilon, \tag{3.49}$$

where the approximation error, ϵ, is a bounded value and goes to zero as p increases, $\hat{\tau}_{1,m}^{[k]} = \frac{\tau_{1,m}^{[k]}}{T_s}$, $\mathbf{E}(\hat{\tau}_{1,m}^{[k]}) = \texttt{diag}\{1, e^{-j\frac{2\pi}{n}\hat{\tau}_{1,m}^{[k]}}, \ldots, e^{-j\frac{2\pi(n-1)}{n}\hat{\tau}_{1,m}^{[k]}}\}$ and $\boldsymbol{\Lambda}_0$ is defined similar to $\boldsymbol{\Lambda}_{km}$ when $\hat{\tau}_{1,m}^{[k]} = 0$. In this case, the received signal at the k-th receiver can be simplified as

$$\mathbf{y}'_k = \boldsymbol{\Lambda}_0 \sum_{m=1}^{M} h_{km}\mathbf{E}\Big(\hat{\tau}_{1,m}^{[k]}\Big)\mathbf{x}'_m + \mathbf{z}'_k, \quad k = 1, 2, \ldots, N. \tag{3.50}$$

Employing this signaling scheme, in the next parts we perform the vector interference alignment over X networks.

3.5.2 Degrees of Freedom and Interference Alignment

Consider the $M \times N$ user asynchronous X network with single antenna nodes. In order to show that this network has the same total degrees of freedom as that of the corresponding

synchronous network, we first argue that the total degrees of freedom of this channel is upper bounded by $\frac{MN}{M+N-1}$. Then, we present the interference alignment scheme which employs the asynchronous delays among the users to achieve the total $\frac{MN}{M+N-1}$ degrees of freedom of the network. Finally, the achievable scheme with finite channel extension is considered for the $M \times 2$ user network. In the asynchronous X network, the vectors are transmitted over constant channels and received with random delays. Considering the presented model in the previous part, at first, the model of the underlying asynchronous network is converted into a synchronous network with varying fading coefficients and then the interference alignment is performed. In fact, we apply the vector interference alignment schemes from [46] on the transformed channel. However, note that in the introduced asynchronous network model, the channel coefficients are *correlated* in a way that the assumptions in [46] are no longer satisfied. Therefore, it is required to show that the interference alignment is still applicable here. The results are summarized in the following theorem:

Theorem 3.4. *The total number of degrees of freedom for the $M \times N$ user asynchronous X network with single antenna nodes is $\frac{MN}{M+N-1}$.*

The converse of Theorem 3.4 is proved here by considering the asynchronous X network in the frequency domain. Assume that all transmitters use $\psi(t)$ as the shaping waveform with bandwidth W. At each receiver, by taking the Fourier transform of both sides of equation (3.36), we get

$$Y_k(f) = \sum_{m=1}^{M} h_{km} e^{-j2\pi f \tau_{km}} X_m(f) + Z_k(f), \quad k = 1, 2, \ldots, N. \tag{3.51}$$

Using equation (3.35) and Fourier transform of the shaping waveform $\Psi(f)$ to compute

$X_m(f)$, the above relation is simplified to

$$\begin{aligned} Y_k(f) &= \sum_{m=1}^{M} h_{km}\Psi(f)e^{-j2\pi f\tau_{km}} \sum_i x_m(i)e^{-j2\pi fiT_s} + Z_k(f) \\ &= \sum_{m=1}^{M} H_{km}(f)X'_m(f) + Z_k(f), \quad k = 1, 2, \ldots, N, \end{aligned} \tag{3.52}$$

where $X'_m(f)$ is the discrete time Fourier transform (DTFT) of the sequence transmitted by the m-th transmitter and $H_{km}(f)$ is defined as follows:

$$H_{km}(f) = h_{km}\Psi(f)e^{-j2\pi f\tau_{km}}.$$

Equation (3.52) represents a synchronous X network with varying channel coefficients, $H_{km}(f)$, in the frequency domain. Since $\psi(t)$ has the bandwidth equal to W, the corresponding channel has W degrees of freedom per second, each of them represents a constant synchronous X network. Applying the upper bound for the total number of degrees of freedom of the synchronous X network in [46], one can see that the total number of degrees of freedom of the asynchronous X network is upper bounded by $\frac{MN}{M+N-1}$ per orthogonal time and frequency dimension.

3.5.3 Achievable Scheme for the $M \times N$ Network

In this part, the $M \times N$ user asynchronous X network is considered. The coefficients in the transformed channel model for the asynchronous case are highly correlated and we need to show that the interference alignment is still feasible. The interference alignment scheme achieves a total $N(u+1)^K+(M-1)Nu^K$ degrees of freedom over $n = N(u+1)^K+(M-1)u^K$ symbol extensions on the transformed channel where $K = (M-1)(N-1)$ and u is an arbitrary positive integer. In fact, for every message in the first transmitter, $(u+1)^K$ degrees of freedom and for every message in the other transmitters, u^K degrees

of freedom are achievable. By taking the supremum over all u and for negligible length of CPS symbols, the total number of degrees of freedom is achieved arbitrary close to the upper bound.

In order to simplify the model and apply the transformed channel model presented in (3.48), an inverse discrete Fourier transform (IDFT) filter at each transmitter and a DFT filter at each receiver are used. The transmitted signals can be written as

$$\underline{x}'_m = \sum_{k=1}^{N} \mathbf{V}_{km}\mathbf{x}_{km} = \begin{cases} \sum_{k=1}^{N}\sum_{i=1}^{(u+1)^K} x_{k1}(i)\mathbf{v}_{k1}(i), & m = 1 \\ \sum_{k=1}^{N}\sum_{i=1}^{u^K} x_{km}(i)\mathbf{v}_{km}(i), & m \neq 1 \end{cases} \tag{3.53}$$

where $\mathbf{x}_{k1}$ is the $(u+1)^K \times 1$ vector of symbols for the k-th receiver and $\mathbf{V}_{k1}$ is the corresponding $n \times (u+1)^K$ matrix containing all directions as its columns. Similarly, for $m = 2, \ldots, M$, $\mathbf{x}_{km}$ is a $u^K \times 1$ vector and $\mathbf{V}_{km}$ is a $n \times u^K$ matrix. Using the transformed channel model, we can write the received signal at the k-th receiver as

$$\underline{y}'_k = \sum_{m=1}^{M} h_{km}\mathbf{\Lambda}_{km}\left(\sum_{k=1}^{N}\mathbf{V}_{km}\mathbf{x}_{km}\right) + \underline{z}'_k, \quad k = 1, 2, \ldots, N, \tag{3.54}$$

Note that the constant channel coefficients, h_{km}, just scale the vectors and do not have any effect on the interference alignment scheme. Therefore, we neglect them for describing the achievable scheme.

To perform the interference alignment by beamforming at the transmitters and zero-forcing at the receivers, the beamforming matrices, $\mathbf{V}_{km}$, are chosen such that at each receiver all the interfering signals from transmitters $2, 3, \ldots, M$, are aligned to the interference space from the first transmitter. Therefore, the dimension of the interference from the first transmitter, $(N-1)(u+1)^K$, is the total interference dimension. Considering total dimension n, the available space for desired signal has $n - (N-1)(u+1)^K = (M-1)u^K + (u+1)^K$ dimensions at each receiver.

To choose the beamforming matrices, we write $K = (M-1)(N-1)$ relations for each receiver resulting in a total KN relations for the network. At the k-th receiver, $k = 1, 2, \ldots, N$, the K relations can be written as

$$\text{span}(\mathbf{\Lambda}_{km}\mathbf{V}_{rm}) \subset \text{span}(\mathbf{\Lambda}_{k1}\mathbf{V}_{r1}), \quad r \neq k, \quad m = 2, 3, \ldots, M, \tag{3.55}$$

where $\text{span}(\mathbf{P})$ represents the vector space spanned by the columns of $\mathbf{P}$. By these choices of alignment equations, at the each receiver all the interferences are aligned to the space of interference from the first transmitter as required. Note that since $\mathbf{\Lambda}_{km}\mathbf{V}_{km}$ is the desired signal space, it is not included in the above relations. Rearranging the above relations at the k-th receiver, $k = 1, 2, \ldots, N$, we get

$$\text{span}(\mathbf{\Lambda}_{rm}\mathbf{V}_{km}) \subset \text{span}(\mathbf{\Lambda}_{r1}\mathbf{V}_{k1}), \quad r \neq k, \quad m = 2, 3, \ldots, M. \tag{3.56}$$

To simplify the relations in (3.56), the following selection is used for the beamforming matrices at the left-hand side of this equation.

$$\mathbf{V}_{km} = \mathbf{V}_{k2}, \quad k = 1, 2, \ldots, N, \quad m = 3, \ldots, M. \tag{3.57}$$

By applying the above selections, equation (3.56) at the kth receiver, $k = 1, 2, \ldots, N$, can be written as

$$\text{span}(\mathbf{T}_{rm}\mathbf{V}_{k2}) \subset \text{span}(\mathbf{V}_{k1}), \quad r \neq k, \quad m = 2, 3, \ldots, M. \tag{3.58}$$

where $\mathbf{T}_{rm}$'s are defined as

$$\mathbf{T}_{rm} \triangleq \mathbf{\Lambda}_{r1}^{-1}\mathbf{\Lambda}_{rm}.$$

Using (3.49) from the system model, $\mathbf{T}_{km}$'s are generally expressed as

$$\begin{aligned}\mathbf{T}_{km} &= \mathtt{diag}\left\{1, e^{-j\frac{2\pi}{n}\tilde{\tau}_{km}}, \ldots, e^{-j\frac{2\pi(n-1)}{n}\tilde{\tau}_{km}}\right\} \\ &= \mathtt{diag}\left\{1, \phi_{km}, \ldots, \phi_{km}^{n-1}\right\}, \end{aligned} \tag{3.59}$$

where $\tilde{\tau}_{km} = \hat{\tau}_{1,m}^{[k]}$ and $\phi_{km} \triangleq e^{-j\frac{2\pi}{n}\tilde{\tau}_{km}}$ for $k = 1, 2, \ldots, N$ and $m = 2, \ldots, M$.

To construct the resulted beamforming matrices for each receiver, $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$, we first generate a vector $\mathbf{w}_k$. Let $\tilde{\tau}_{k0}$ be a finite random variable drawn from a continuous distribution. Vector $\mathbf{w}_k$ has $n \times 1$ dimension and it is selected as

$$\mathbf{w}_k = [1, \phi_{k0}, \cdots, \phi_{k0}^{n-1}]^T, \tag{3.60}$$

where $\phi_{k0} \triangleq e^{-j\frac{2\pi}{n}\tilde{\tau}_{k0}}$. Using $\mathbf{w}_k$ and applying the K relations for the k-th receiver in (3.58), $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$ are constructed by two sets of columns vectors which are generated as following:

$$\begin{aligned}\mathbf{V}_{k2} &\triangleq \left\{ \left(\prod_{\substack{m=2,3,\ldots,M \\ r\neq k}} \mathbf{T}_{rm}^{\alpha_{rm}} \right) \mathbf{w}_k : \ \alpha_{rm} \in \{1, 2, \ldots, u\} \right\}, \\ \mathbf{V}_{k1} &\triangleq \left\{ \left(\prod_{\substack{m=2,3,\ldots,M \\ r\neq k}} \mathbf{T}_{rm}^{\beta_{rm}} \right) \mathbf{w}_k : \ \beta_{rm} \in \{1, 2, \ldots, u+1\} \right\}. \end{aligned} \tag{3.61}$$

where $k = 1, 2, \ldots, N$. From this construction, one can check that the alignment conditions in (3.58) are satisfied. After selecting $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$, the other beamforming matrices, $\mathbf{V}_{km},\ m = 3, \ldots, M$, are generated by using (3.57). To show that $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$ are full-rank matrices, we use the following Lemma:

Lemma 3.1. *Let $\mathbf{T} = [\tau_1, \tau_2, \ldots, \tau_u]^T$ be a vector of length m in which the elements are drawn independently from a continuous distribution and $\mathbf{\Phi} = [\phi_1, \phi_2, \ldots, \phi_u]^T$ be the*

corresponding column vector where $\phi_k = e^{-j\frac{2\pi}{n}\tau_k}$. *Let* $\mathbf{z}$ *be a vector of length* n *with the elements of the form* $z_k = \prod_i (\phi_i)^{q_i}$ *where* ϕ_i*'s are entries of* $\mathbf{\Phi}$, $q_i \in \{0, 1, \ldots, n-1\}$ *and* $n \in \mathbb{N}$. *If all pairs with different elements of* $\mathbf{z}$ *(any two elements) have at least one component,* ϕ_k, *with different exponent,* q_k, *then all the elements of* $\mathbf{z}$ *are distinct with probability one.*

Proof. Writing the equality of two different elements of $\mathbf{z}$, we get

$$(q_1 - s_1)\tau_1 + (q_2 - s_2)\tau_2 + \ldots (q_u - s_u)\tau_u + in = 0,$$

where $i \in \mathbb{Z}$ and q_k, s_k are the corresponding exponents of the two elements. Having at least one base with different exponent ensures that the equation in τ_k is not trivially equivalent to zero. Moreover, by integer coefficients, $(q_k - s_k)$, the probability that the τ_k's with continuous distributions are the roots of the equation is zero. Therefore, the elements of $\mathbf{z}$ are distinct almost surely. □

Using (3.59) and (3.60) for generating $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$ in (3.61), one can see that these matrices have a Vandermonde structure in which the k-th row is the $(k-1)$-th exponent of the elements of the vectors $\mathbf{z}_{k2}$ and $\mathbf{z}_{k1}$, respectively. For $m = 2, 3, \ldots, M$ and $k = 1, 2, \ldots, N$, we get

$$\begin{aligned}
\mathbf{z}_{k2} &= [\ldots, \phi_{k0} \prod_{\substack{m \\ r \neq k}} \phi_{rm}^{\alpha_{rm}}, \ldots], \quad \alpha_{rm} \in \{1, 2, \ldots, u\}, \\
\mathbf{z}_{k1} &= [\ldots, \phi_{k0} \prod_{\substack{m \\ r \neq k}} \phi_{rm}^{\beta_{rm}}, \ldots], \quad \beta_{rm} \in \{1, 2, \ldots, u+1\}.
\end{aligned} \tag{3.62}$$

Matrices $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$ have full rank if the elements of the above vectors are distinct. This condition is satisfied by construction. Therefore, by applying Lemma 3.1, it can be shown that both matrices are full-rank.

Finally, we need to show that the space spanned by all the desired signals and the interferences at each receiver has full rank. In other words, the desired and the interference direction vectors at each receiver should be linearly independent. Consider the first receiver. The desired signals are received along the column vectors of $\mathbf{S}_1$ as in (3.63) where $\mathbf{\Lambda}_{1m}\mathbf{V}_{1m}$ is the vector space spanned by the desired signal from the m-th transmitter. $\mathbf{V}_{1m}, m = 3, \ldots, M$ have been replaced by $\mathbf{V}_{12}$ as required in (3.57). Moreover, the beamforming matrices are designed such that all the interferences are along those from the first transmitter. Therefore, the interference matrix can be written as $\mathbf{I}_1$ in (3.64) where $\mathbf{\Lambda}_{11}\mathbf{V}_{r1}, r = 2, \ldots, N$, refers to the one of $N-1$ undesired signals from the first transmitter intended for the other receivers.

$$\mathbf{S}_1 = [\mathbf{\Lambda}_{11}\mathbf{V}_{11} \;\; \mathbf{\Lambda}_{12}\mathbf{V}_{12} \;\; \ldots \mathbf{\Lambda}_{1M}\mathbf{V}_{12}], \tag{3.63}$$

$$\mathbf{I}_1 = [\mathbf{\Lambda}_{11}\mathbf{V}_{21} \;\; \mathbf{\Lambda}_{11}\mathbf{V}_{31} \;\; \ldots \mathbf{\Lambda}_{11}\mathbf{V}_{N1}]. \tag{3.64}$$

Thus, the vector space spanned by the desired and the interfering signals at the first receiver can be written as $\mathbf{T}_1 = [\mathbf{S}_1 \;\; \mathbf{I}_1]$. Multiplying by $\mathbf{\Lambda}_{11}$, we now show that the rank of the following $n \times n$ matrix is equal to n with probability one.

$$\tilde{\mathbf{T}}_1 = [\mathbf{V}_{11} \;\; \mathbf{T}_{12}\mathbf{V}_{12} \;\; \ldots \mathbf{T}_{1M}\mathbf{V}_{12} \;\; \mathbf{V}_{21} \;\; \mathbf{V}_{31} \;\; \ldots \mathbf{V}_{N1}],$$

where $\mathbf{T}_{1m} = \mathbf{\Lambda}_{11}^{-1}\mathbf{\Lambda}_{1m}$, $m = 2, 3, \ldots, M$, is defined in (3.59). Rewriting the above matrix by replacing $\mathbf{V}_{k2}$ and $\mathbf{V}_{k1}$ vectors, we get an $n \times n$ vandermonde matrix where the k-th row is the $(k-1)$-th exponent of the elements of the vector $\mathbf{z}$ given as follows

$$\mathbf{z} = [\mathbf{z}_{11} \;\; \phi_{12}\mathbf{z}_{12} \;\; \ldots \;\; \phi_{1M}\mathbf{z}_{12} \;\; \mathbf{z}_{21} \;\; \mathbf{z}_{31} \;\; \ldots \;\; \mathbf{z}_{N1}],$$

where $\mathbf{z}_{k1}$ and $\mathbf{z}_{12}$ are defined in (3.62) and each of them has distinct elements inside. Note that $\mathbf{z}_{k1}$ has a unique base of $\phi_{k0}, k = 1, 2, \ldots, N$, by construction. Furthermore,

there is a unique base, ϕ_{1m}, for every $\phi_{1m}\mathbf{z}_{12}, m = 2, 3, \ldots, M$, term in the vector. Therefore, one can see that all the elements in $\mathbf{z}$ have at least one different base with different exponent and according to Lemma 1 they are all distinct. Considering $\mathbf{z}$ with length n as the second row in the $n \times n$ Vandermonde matrix $\tilde{\mathbf{T}}_1$, we conclude that it has full rank almost surely. The same argument can be made for the vector space spanned at the other receivers.

This scheme achieves a total $N(u+1)^K + (M-1)Nu^K$ degrees of freedom over $N(u+1)^K + (M-1)u^K + 2(p+1)$ symbols on the actual channel. By taking the supremum over all u and for negligible length of CPS symbols, $2(p+1)$, the total number of degrees of freedom is achieved arbitrary close to the upper bound. Together with the converse result, this completes the proof of Theorem 1.

3.5.4 Achievable Scheme for the $M \times 2$ Network

While the total number of degrees of freedom of the X network is asymptotically achieved, in the special case when $N = 2$, the total $\frac{2M}{M+1}$ can be achieved with a finite channel extension [46]. To limit the rate loss due to adding the CPS symbols, we combine $Q > 1$ supersymbols each of length $M + 1$ and add the CPS symbols to the resulted vector of length $Q(M + 1)$. Thus, the actual transmitted frame has a length of $\ell = Q(M + 1) + 2(p + 1)$. In this manner, for each of the $2M$ messages, one degrees of freedom is achieved over every $M + 1$ channel use on the transformed channel model. Therefore, for this transmission scheme the total number of degrees of freedom is equal to $\frac{Q(2M)}{Q(M+1)+2(p+1)}$. For large values of Q, one can achieve the total degrees of freedom arbitrary close to $\frac{2M}{M+1}$.

At each transmitter, beamforming is performed over every $M+1$ symbol extension of the transformed channel. Then, Q supersymbols are combined together to construct the vector of length $Q(M + 1)$ which is the input to the transformed channel. The resulting

vector is then passed through the IDFT filter and supported by the CPS symbols to form the transmitted vector. At each receiver, the received vector is passed through the matched filter and then the CPS symbols are discarded. Finally, the DFT filter is used to provide the required output vector. Fig. 3.6 shows the block diagram for the interference alignment scheme over the 2×2 asynchronous X network in which $\mathbf{Q}$ represents the procedure for constructing the beamforming vectors and the combined vector of length $Q(M+1)$ for the transmitting stream.

Consider the channel with the extended symbols for $q = 0, 1, \ldots, Q-1$, as

$$\underline{y}'_k(q) = \sum_{m=1}^{M} h_{km} \mathbf{\Delta}_{km}(q) \underline{x}'_m(q) + \underline{z}'_k(q), \quad k = 1, 2, \tag{3.65}$$

where $\mathbf{\Delta}_{km}(q)$, $\underline{x}'_m(q)$ and $\underline{y}'_k(q)$ are the q-th $M+1$ symbol extension of the transformed channel and the corresponding input and output vectors, shown as follows, respectively.

$$\begin{aligned}
\mathbf{\Delta}_{km}(q) &\triangleq \mathtt{diag}\{\lambda_{km}(q(M+1)+1), \lambda_{km}(q(M+1)+2), \ldots, \lambda_{km}((q+1)(M+1))\}, \\
\underline{x}'_m(q) &\triangleq [x'_m(q(M+1)+1), x'_m(q(M+1)+2), \ldots, x'_m((q+1)(M+1))]^T, \\
\underline{y}'_k(q) &\triangleq [y'_k(q(M+1)+1), y'_k(q(M+1)+2), \ldots, y'_k((q+1)(M+1))]^T.
\end{aligned} \tag{3.66}$$

In the $M \times 2$ user X network, the interference alignment scheme uses two directions at every transmitter node to transmit the two independent streams x_{1m} and x_{2m} in $M+1$ symbol extensions, resulting in the total $\frac{2M}{M+1}$ degrees of freedom. Over one extended channel use, the input vector at the m-th transmitter can be written as

$$\underline{x}'_m = x_{1m}\mathbf{v}_{1m} + x_{2m}\mathbf{v}_{2m}, \tag{3.67}$$

where $\mathbf{v}_{1m}$ and $\mathbf{v}_{2m}$ are the transmission directions at each transmitter for the first and

second receiver, respectively. Replacing the input vector in the channel model, we obtain

$$\underline{y}'_k = \sum_{m=1}^{M} \mathbf{\Delta}_{km}(x_{1m}\mathbf{v}_{1m} + x_{2m}\mathbf{v}_{2m}) + \underline{z}'_k, \quad k = 1, 2.$$

Then, at the receivers side, each receiver detects its M desired signals by ZF all the interference vectors which are intended for the other receiver. To be able to recover M desired vectors out of the total $M + 1$ available direction by using ZF, the set of all interfering vectors should be squeezed to a single dimension. Consider the first receiver, we choose the beamforming vectors $\mathbf{v}_{2m}$ such that the resulting interference vectors at this receiver, $\mathbf{\Delta}_{1m}\mathbf{v}_{2m}$, align at the same dimension. Similarly, at the second receiver, all the interference vectors $\mathbf{\Delta}_{2m}\mathbf{v}_{1m}$ should occupy only one dimension. These two conditions can be written as

$$\begin{aligned} &\text{At Receiver1:} \;\; \mathbf{\Delta}_{1m}\mathbf{v}_{2m} = \mathbf{\Delta}_{11}\mathbf{v}_{21}, \\ &\text{At Receiver2:} \;\; \mathbf{\Delta}_{2m}\mathbf{v}_{1m} = \mathbf{\Delta}_{21}\mathbf{v}_{11}, \end{aligned} \tag{3.68}$$

where $m = 2, \ldots, M$.

We need to show that from the total $M+1$ available dimensions at each receiver, all the interference signals aligned in a single dimension and the desired signals span the other M dimensions. In other words, the desired vectors should be linearly independent of the interference vector. For this sake, the beamforming vectors should be chosen properly such that the following $(M + 1) \times (M + 1)$ matrices have linearly independent column vectors almost surely.

$$\begin{aligned} &\text{At Receiver1:} \;\; \mathbf{T}_1 = \left[\mathbf{\Delta}_{11}\mathbf{v}_{11} \;\; \mathbf{\Delta}_{12}\mathbf{v}_{12} \;\; \ldots \mathbf{\Delta}_{1M}\mathbf{v}_{1M} \;\; \mathbf{\Delta}_{11}\mathbf{v}_{21}\right], \\ &\text{At Receiver2:} \;\; \mathbf{T}_2 = \left[\mathbf{\Delta}_{21}\mathbf{v}_{21} \;\; \mathbf{\Delta}_{22}\mathbf{v}_{22} \;\; \ldots \mathbf{\Delta}_{2M}\mathbf{v}_{2M} \;\; \mathbf{\Delta}_{21}\mathbf{v}_{11}\right], \end{aligned} \tag{3.69}$$

where $\mathbf{\Delta}_{11}\mathbf{v}_{21}$ and $\mathbf{\Delta}_{21}\mathbf{v}_{11}$ are the common interference directions at the first and the

second receiver, respectively.

Let $\hat{\tau}_0$ and $\hat{\tau}_1$ be two independent continuous random variables and $n = Q(M+1)$. The two vectors $\mathbf{v}_{11}$ and $\mathbf{v}_{21}$ are selected as follows

$$\begin{aligned}\mathbf{v}_{11} &= [1, \phi_0, \cdots, \phi_0^M]^T, \\ \mathbf{v}_{21} &= [1, \phi_1, \cdots, \phi_1^M]^T,\end{aligned} \tag{3.70}$$

where $\phi_0 \triangleq e^{-j\frac{2\pi}{n}\hat{\tau}_0}$ and $\phi_1 \triangleq e^{-j\frac{2\pi}{n}\hat{\tau}_1}$. Applying these selection and then using (3.68) to choose the other beamforming vectors, we now show that the matrices in (3.69) have full ranks.

Consider the first receiver. Equivalently, by multiplying $\mathbf{T}_1$ by the full-rank matrix $\mathbf{\Delta}_{11}^{-1}$ and replacing $\mathbf{v}_{1m}$, $m = 2, 3, \ldots, M$, by the corresponding values from (3.68), we show that the following matrix, $\tilde{\mathbf{T}}_1$, is full rank.

$$\tilde{\mathbf{T}}_1 = \left[\mathbf{v}_{11} \;\; \mathbf{F}_2\mathbf{v}_{11} \;\; \ldots \mathbf{F}_M\mathbf{v}_{11} \;\; \mathbf{v}_{21}\right], \tag{3.71}$$

where the product channel matrix $\mathbf{F}_m$ is defined as

$$\mathbf{F}_m \triangleq \mathbf{\Delta}_{11}^{-1}\mathbf{\Delta}_{1m}\mathbf{\Delta}_{2m}^{-1}\mathbf{\Delta}_{21}, \quad m = 2, \ldots, M.$$

By using well-designed shaping waveforms, (3.49) is applicable with a negligible approximation error. Therefore, the diagonal product matrices can be written as

$$\mathbf{F}_m = \mathtt{diag}\{1, e^{-j\frac{2\pi}{n}\hat{\tau}_m}, \ldots, e^{-j\frac{2\pi(M)}{n}\hat{\tau}_m}\}, \tag{3.72}$$

where $\hat{\tau}_m \triangleq \hat{\tau}_{1,m}^{[1]} - \hat{\tau}_{1,m}^{[2]}$, $m = 2, \ldots, M$ and $\hat{\tau}_{1,m}^{[k]} \triangleq \frac{\tau_{1,m}^{[k]}}{T_s}$, $k = 1, 2$. Note that $\hat{\tau}_m$'s are continuous independent random variables for all $m = 2, 3, \ldots, M$. Replacing (3.70) and (3.72) in (3.71) yields

$$\tilde{\mathbf{T}}_1 = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ \phi_0 & \phi_2 & \phi_3 & \cdots & \phi_M & \phi_1 \\ \phi_0^2 & \phi_2^2 & \phi_3^2 & \cdots & \phi_M^2 & \phi_1^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \phi_0^M & \phi_2^M & \phi_3^M & \cdots & \phi_M^M & \phi_1^M \end{bmatrix}, \tag{3.73}$$

where $\phi_m \triangleq e^{-j\frac{2\pi}{n}(\hat{\tau}_m+\hat{\tau}_0)}$, $m = 2, \ldots, M$, and ϕ_0, ϕ_1 are defined in (3.70). Since $\hat{\tau}_m$'s are drawn independently from a continuous distribution, ϕ_m's are distinct almost surely. Therefore, the Vandermonde matrix $\tilde{\mathbf{T}}_1$ has full rank of $M+1$ with probability one.

Similarly, one can show that at the second receiver the corresponding matrix, $\mathbf{T}_2$ in (3.69), has full rank almost surely. Therefore, the proposed asynchronous scheme achieves $\frac{2M}{M+1}$ degrees of freedom over the transformed channel, or equivalently, $\frac{Q(2M)}{Q(M+1)+2(p+1)}$ degrees of freedom over the actual channel. For large values of Q, the total degrees of freedom can be achieved arbitrary close to $\frac{2M}{M+1}$.

Remark 3.3. The random nature of the asynchronous delays provides the feasibility of interference alignment. Having independent random delays is necessary to apply vector interference alignment in this work. However, inserting artificial delays at the transmitters will not have the same effect because it is the delay differences at each receiver that makes matrix $\tilde{\mathbf{T}}_k$ full rank.

3.6 Conclusion

In this chapter, the compute-and-forward relaying strategy over asynchronous networks was considered. By applying the nested lattice coder and decoder over a network with distributed nodes, the challenge arising from time asynchronism was addressed. With

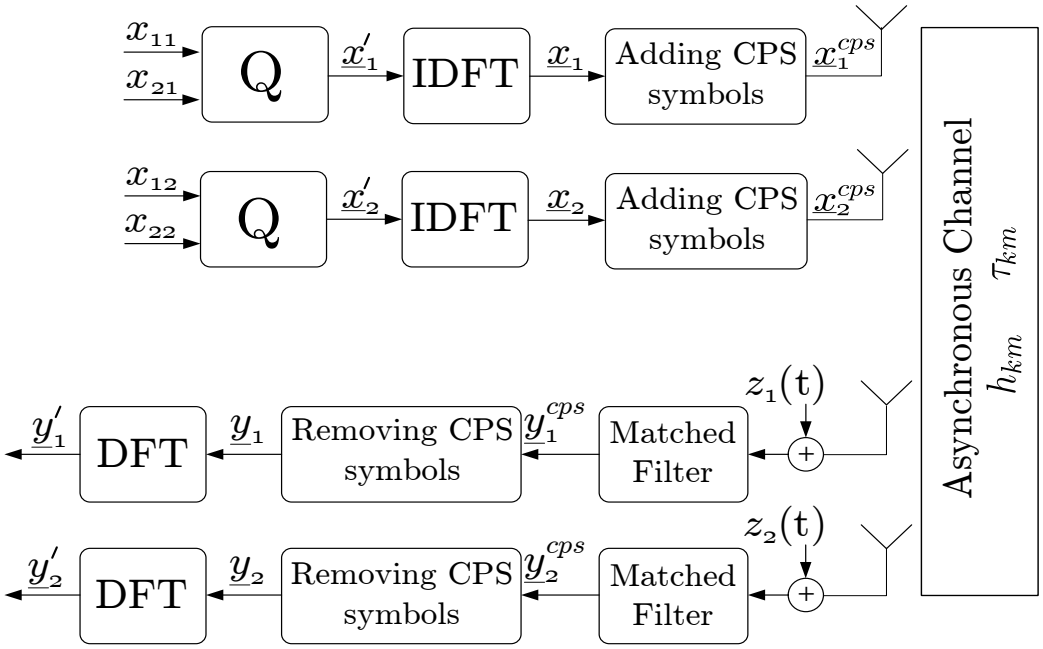


Figure 3.6: Block diagram of the interference alignment scheme over the 2×2 asynchronous X network

symbol-asynchronous network model, matched filters followed by a whitening filter were used at each relay as part of a signaling scheme to extract appropriate sequences for the decoder of compute-and-forward. Over a frame-asynchronous network, multiple antennas at the each relay were used to remove the asynchronous delays along with rotating the channel coefficients. Using a linear filter with structure related to the delays, the computation rate at each receiver is maximized at any SNR. At last, we extended the symbol-asynchronous network model to the $M \times N$ X network with single antenna users over time-invariant channel. By taking advantage of the asynchronous delays in the received signals, achievable schemes were studied. It was shown that the asynchronism causes ISI among received symbols from different transmitters and hence provides correlated channel variations which were proved to be sufficient to implement the vector interference alignment in the network with single antenna nodes.

Chapter 4

Conclusion and Future Work

4.1 Conclusion

Communication strategies based on lattice codes at the transmitters and/or lattice decoder at the receivers provide solution to various multiuser problems in wireless communications. Over correlated fading channels, adaptive lattice reduction aided detection methods were studied in the first part. Two adaptive methods which improve the complexity of lattice reduction algorithms were introduced and analyzed. The proposed methods can be used in conjunction with any multi-antenna scenario that requires lattice reduction, such as MIMO decoders and broadcast precoders to reduce the computational complexity. A reduced complexity list decoder for soft-output detection in MIMO systems was then introduced over the slow fading channels. Using the adaptive hard-output estimated point as the center of the sphere in the list decoder, an adaptive list was constructed. It was shown that one can save significantly in computational complexity if a shifted list generated from a past realization of the correlated channel is used. Employing an appropriate value for the performance/complexity tradeoff, the soft-output channel detector with different channel codes was used to investigate the performance of the adaptive list construction.

Considering asynchronous nodes in a wireless network, the compute-and-forward relaying strategy was studied in the second part. By applying the nested lattice coder and decoder over a network with distributed nodes, the challenge arising from the time asynchronism was addressed. With symbol-asynchronous network model, matched filters followed by a whitening filter were used at each relay as part of a signaling scheme to extract appropriate sequences for the decoder of compute-and-forward. Over a frame-asynchronous network, multiple antennas at the each relay were used to remove the asynchronous delays along with rotating the channel coefficients. Using a linear filter with structure related to the delays, the computation rate at each receiver was maximized at any SNR. At the end, by extending the application of asynchronous model, we considered a general symbol-asynchronous X network with single antenna nodes over time-invariant channels. It was shown that the upper bound for the total number of degrees of freedom of the asynchronous network is the same as the corresponding synchronous network. Furthermore, taking advantage of the asynchronous delays in the received signals, achievable schemes were studied. It was shown that the asynchronism provides correlated channel variations which were proved to be sufficient to implement the vector interference alignment over the constant X network.

In summary, in this thesis, communication schemes with lattice codes and lattice decoders were considered over the wireless channels. Exploiting the channel properties, new algorithms were designed to save in computational complexity and provide feasible communication methods over asynchronous networks.

4.2 Future Works

Adaptive Lattice Reduction

The proposed adaptive lattice reduction technique in this thesis is a general framework that can be used with any lattice reduction algorithm and in any lattice reduction aided MIMO scenario. However, one can select a reduction algorithm and also adjust the algorithm steps according to the temporal correlation of the fading channel. Moreover, the impact of spatial correlation in the MIMO channel coefficients can be considered in the detection model. For example, unlike the LLL reduction which performs a reduction step on two adjacent vectors in an ordered manner, the Seyen lattice reduction [80] considers all basis vectors simultaneously and chooses the vectors that reduce its reduction measure. Therefore, using such an algorithm on a correlated channel matrix may provide more saving in the computational complexity.

Lattice Interference Alignment and Decoding Techniques at the Receiver

The use of lattice codes in interference and X networks will allow us to benefit from the structured interference created by these kinds of codes. In fact, the discrete space structure can be exploited to transmit the desired signals over the interference-free dimensions. Considering the compute-and-forward strategy which uses lattice codes at any SNRs, one can design and investigate lattice based interference alignment methods with regard to practical assumptions on the channel such as synchronization, availability of channel state information at the transmitters and the scalability of the achieved performance with the SNR. Additionally, the lattice alignment schemes should take into account the use of practical lattice decoders. Considering the nature of the interference network, each receiver is interested in only one of the received signals, all the others are unwanted. Therefore, there is no need to decode all the received signals from interfering

transmitters individually. It is more efficient to just decode the sum of the interferers, especially if they are designed to align along certain dimensions. In this way, the challenging part of the alignment method can be converted into designing an efficient decoder that separates the sum of interferers from the desired signal.

Structured Codes in Downlink Systems

Compute-and-forward offers protecting against noise at the same time with possibility of exploiting the interference for cooperation in the network. Utilizing the algebraic structure in the codes and designing an invertible precoder, compute-and-forward can be used in downlink systems at any SNR. Note that the main idea in the compute-and-forward is to decode a linear combination of the transmitted signals. To make this strategy useful over broadcast channels, one need to exchange the routine at the transmitters and receivers. Moreover, exploiting the powerful structures in compute-and-forward, opportunistic broadcast strategies can be investigated in which good performances are provided with limited (integer) feedback properties. On the other hand, considering the distributed structures in compute-and-forward model and utilizing the asynchronous schemes, one can investigate it a potential candidate for multi-cell broadcast channels.

Appendix A

Some Proofs of Chapter 3

A.1 Cholesky factorization in the symbol-asynchronous compute-and-forward

Consider the $n \times n$ covariance matrix $\mathbf{\Gamma} = (\gamma_{j,i})$ at one of the relays which is a positive definite matrix (the index k is dropped for simplicity). The corresponding Cholesky decomposition is in the form of $\mathbf{\Gamma} = \mathbf{W}^T\mathbf{W}$, where $\mathbf{W}$ is an upper triangular matrix. Assuming the covariance matrix in (3.15) for two transmitters and the choice of shaping waveform in this work, the entries of the upper triangular matrix are given according to the the following lemma:

Lemma A.1. *There are only two non-zero entries in each row of matrix $\mathbf{\Gamma}$ which for $n \to \infty$ converge to $\sqrt{\lambda_1}$ and $\sqrt{\lambda_2}$ at the odd and the even rows, respectively (except for the last row which has one nonzero entry). Specifically, for $s \in \mathbb{N}$, we obtain*

$$\lim_{j\to\infty} w_{j,j} = \lim_{j\to\infty} w_{j,j+1} = \begin{cases} \sqrt{\lambda_1}, & \text{if } j = 2s-1, \\ \sqrt{\lambda_2}, & \text{if } j = 2s. \end{cases}$$

Proof. Following the definition of the Cholesky decomposition, the matrix $\mathbf{W} = (w_{q,j})$

satisfies

$$\gamma_{j,i} = \sum_{q=1}^{i} w_{q,j} w_{q,i}, \quad 1 \leq i \leq j \leq n.$$

It can be shown that there are only two non-zero entries at the main and upper diagonal. Simplifying the above relation results in

$$\begin{aligned}
\gamma_{j,j} &= w_{j,j}^2 + w_{j-1,j}^2 = 1, \\
\gamma_{j,j+1} &= w_{j,j+1}.w_{j,j} = \lambda_1, \quad j = 2s-1, \\
\gamma_{j,j+1} &= w_{j,j+1}.w_{j,j} = \lambda_2, \quad j = 2s.
\end{aligned}$$

Solving for $w_{j,j}$ and $w_{j,j+1}$ provides continued fractions, where for $\lambda_1 = 1 - \lambda_2$ they converge to $\lambda_1/\sqrt{\lambda_1}$ and $\sqrt{\lambda_1}$ at the odd rows and to $\lambda_2/\sqrt{\lambda_2}$ and $\sqrt{\lambda_2}$ at the even rows, respectively. □

Note that for the general system model with more than two transmitters and different shaping waveforms, one can check that the continued fractions in the upper-triangular matrix convergence to different values depending on the system parameters.

Finally, by considering the equations in the odd rows, it remains to be shown that the varying channel coefficients can be replaced by the convergence limit λ_1, and the equivalent system equation can be written as in (3.17). Assume $\boldsymbol{\delta}_m$'s are the difference vectors between the varying coefficients and the limit. The convergence of the continued fractions results in $\|\boldsymbol{\delta}_m\|^2 = o(n)$. Therefore, as $n \to \infty$ the variance of the additional noise goes to zero and the system equation can be shown with the constant channel coefficients. This argument can be summarized as follows:

Lemma A.2. *As $n \to \infty$, the odd rows of $\mathbf{W}$ result in the following system equation:*

$$\bar{y}(i) = \sqrt{\lambda_1} h_1 x_1(i) + \sqrt{\lambda_1} h_2 x_2(i) + \bar{z}(i),$$

where $\bar{\mathbf{z}}$ has unit variance.

A.2 Closest point decoding in the frame-asynchronous compute-and-forward

Considering the receiver filter in Section 3.4, we use the following lemma to show that in the achievable computation rates, the noise power is the sum of the self-correlated additive Gaussian noise and the non-integer part of the equation at the output of the filter.

Lemma A.3. *For the n-dimensional nested lattice codes over the frame-asynchronous compute-and-forward scheme, the decoding error probability approaches zero as n goes to infinity if the rate is below the computation rates of the equivalent channel with white Gaussian noise.*

Proof. Consider the ensemble of lattices resulted from Construction A as in [12] with the closest point decoding. The decoding error probability P_k at receiver k can be upper bounded as

$$P_k \leq \Pr(\mathbf{z}'_k \notin \mathcal{V}_k), \tag{A.1}$$

where $\mathcal{V}_k$ is the fundamental Voronoi region of the fine lattice and $\mathbf{z}'_k$ is the total noise at the output of the filter with variance given in (3.29). For white Gaussian noise, it is shown in [15] that the probability of error in (A.1) goes to zero as n tends to infinity. Moreover, it can be verified that P_k only depends on the norm of the noise. Following in the footsteps of [81], we need to show that the error probability is monotonically nondecreasing with the length of the noise (in a specific direction). In other word, for a positive δ, if for some lattice point $\lambda_i \neq \mathbf{0}$ we have $\|\mathbf{z}'_k - \lambda_i\| \leq \|\mathbf{z}'_k\|$ then $\|(1+\delta)\mathbf{z}'_k - \lambda_i\| \leq$

$(1+\delta)\|\mathbf{z}'_k\|$. Using the triangular inequality we obtain

$$\begin{aligned}\|(1+\delta)\mathbf{z}'_k - \lambda_i\| &\leq \|\mathbf{z}'_k - \lambda_i\| + \|\delta\mathbf{z}'_k\| \\ &\leq \|\mathbf{z}'_k\| + \|\delta\mathbf{z}'_k\| \\ &= (1+\delta)\|\mathbf{z}'_k\|.\end{aligned}$$

Therefore, the error probability can be smaller only when the noise vector become shorter. In this case, proving that the error probability can be arbitrary small for white Gaussian noise with variance N is sufficient to show that it is small for any noise with variance less than or equal to N. □

A.3 Proof of theorem 3.3

Considering the receiver filter in Section 3.4, to maximize the computation rate at receiver k at any SNR, the equivalent noise power $||\mathbf{b}_k||^2 + \text{SNR}||\mathbf{q}_k - \mathbf{a}_k||^2$ needs to be minimized. From the structure of the filter, we have an equality constrained quadratic programming problem which gives the optimal solution. For simplicity, we drop the index k here. Depending on the number of distinct integer delays, there is a set of equations that needs to be satisfied in order to remove the asynchronism before decoding a linear sum of the transmitted vectors. The total number of equations is equal to $\tilde{d}M$ where $\tilde{d}$ is the number of distinct delays, from which M equations form the matrix $\mathbf{G}_1$ in the objective function and the remaining generate $\mathbf{G}_2$ as the conditions. Therefore, the optimization problem is given by

$$\begin{aligned} \min_{\mathbf{b}} \quad & (||\mathbf{b}||^2 + \text{SNR}||\mathbf{G}_1\mathbf{b} - \mathbf{a}||^2) \\ \text{s.t.} \quad & \mathbf{G}_2\mathbf{b} = \mathbf{0}, \end{aligned} \tag{A.2}$$

where $\mathbf{b} \in \mathbb{R}^{\tilde{d}M}$. Using the standard quadratic form, the above problem can be written as

$$\begin{aligned} \min_{\mathbf{b}} \quad & (\frac{1}{2}\mathbf{b}^T\mathbf{C}\mathbf{b} + \mathbf{e}^T\mathbf{b} + c) \\ \text{s.t.} \quad & \mathbf{G}_2\mathbf{b} = \mathbf{0} \end{aligned} \tag{A.3}$$

where the associated parameters are given by

$$\begin{aligned} \mathbf{C} &= 2(\mathbf{I} + \text{SNR}\mathbf{G}_1^T\mathbf{G}_1), \\ \mathbf{e}^T &= -2\text{SNR}\mathbf{a}^T\mathbf{G}_1, \\ c &= \text{SNR}\mathbf{a}^T\mathbf{a}. \end{aligned} \tag{A.4}$$

The solution to the above equality constrained quadratic programming problem is given by the Lagrange multiplier method as

$$\begin{bmatrix} \mathbf{C} & \mathbf{G}_2^T \\ \mathbf{G}_2 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{b}_{\text{opt}} \\ \boldsymbol{\lambda}_{\text{opt}} \end{bmatrix} = \begin{bmatrix} -\mathbf{e} \\ \mathbf{0} \end{bmatrix}. \tag{A.5}$$

where $\mathbf{b}_{\text{opt}}$ is the optimal solution and $\boldsymbol{\lambda}_{\text{opt}}$ is the set of associated Lagrange multipliers [82].

In order to have a unique solution to the above problem, the matrix of coefficients in (A.5) needs to be nonsingular. Let $\mathbf{Z}$ be the matrix whose columns span $\text{Ker}(\mathbf{G}_2)$, i.e., $\mathbf{G}_2\mathbf{Z} = 0$. It can be verified that if $\mathbf{G}_2$ has full row rank and $\mathbf{Z}^T\mathbf{C}\mathbf{Z}$ is positive definite, then the matrix of coefficients is invertible and hence, there exists a unique solution to the above equality constrained quadratic problem. Considering the asynchronous channel with random channel coefficients, $\mathbf{G}_1$ and $\mathbf{G}_2$ have full row rank with probability one. Moreover, from the definition of $\mathbf{C}$ in (A.4) and the fact that $\mathbf{Z}$ is the kernel of $\mathbf{G}_2$ with full column rank, it is easy to show that $\mathbf{u}^T\mathbf{Z}^T\mathbf{C}\mathbf{Z}\mathbf{u} > 0$ for all nonzero vector $\mathbf{u} \in \mathbb{R}^{(\tilde{d}-1)M}$ and hence, matrix $\mathbf{Z}^T\mathbf{C}\mathbf{Z}$ is positive definite which yields the desired result. □

Bibliography

[1] G. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless personal communications*, vol. 6, no. 3, 1998.

[2] V. Tarokh, N. Seshadri, and A. Calderbank, "Space-time codes for high data rate wireless communications: Performance analysis and code construction," *IEEE Transactions on Information Theory*, vol. 44, pp. 744–765, Mar. 1998.

[3] H. El Gamal and M. Damen, "Universal space-time coding," *IEEE Transactions on Information Theory*, vol. 49, pp. 1097–1119, May 2003.

[4] L. Zheng and D. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Transactions on Information Theory*, vol. 49, pp. 1073–1096, May 2003.

[5] J. Conway, N. Sloane, and E. Bannai, "Sphere packings, lattices, and groups," 1999.

[6] R. Zamir, "Lattices are everywhere," in *Information Theory and Applications Workshop*, pp. 392–421, 2009.

[7] U. Erez, S. Litsyn, and R. Zamir, "Lattices which are good for (almost) everything," *IEEE Transactions on Information Theory*, vol. 51, pp. 3401–3416, Oct. 2005.

[8] R. de Buda, "Some optimal codes have structure," *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 893–899, Aug. 1989.

[9] G. Poltyrev, "On coding without restrictions for the AWGN channel," *IEEE Transactions on Information Theory*, vol. 40, pp. 409–417, Mar. 1994.

[10] H. Loeliger, "Averaging bounds for lattices and linear codes," *IEEE Transactions on Information Theory*, vol. 43, pp. 1767–1773, Nov. 1997.

[11] R. Urbanke and B. Rimoldi, "Lattice codes can achieve capacity on the AWGN channel," *IEEE Transactions on Information Theory*, vol. 44, pp. 273–278, Jan. 1998.

[12] U. Erez and R. Zamir, "Achieving $\frac{1}{2}\log(1 + \mathrm{SNR})$ on the AWGN channel with lattice encoding and decoding," *IEEE Transactions on Information Theory*, vol. 50, pp. 2293–2314, Oct. 2004.

[13] H. El Gamal, G. Caire, and M. Damen, "Lattice coding and decoding achieve the optimal diversity-multiplexing tradeoff of MIMO channels," *IEEE Transactions on Information Theory*, vol. 50, pp. 968–985, June 2004.

[14] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Transactions on Information Theory*, vol. 48, pp. 1250–1276, June 2002.

[15] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Transactions on Information Theory*, vol. 57, pp. 6463–6486, Oct. 2011.

[16] G. Bresler, A. Parekh, and D. Tse, "The approximate capacity of the many-to-one and one-to-many gaussian interference channels," *IEEE Transactions on Information Theory*, vol. 56, pp. 4566–4592, Sep. 2010.

[17] S. Sridharan, A. Jafarian, S. Vishwanath, S. Jafar, and S. Shamai, "A layered lattice coding scheme for a class of three user gaussian interference channels," in *Communication, Control, and Computing, 2008 46th Annual Allerton Conference on*, pp. 531–538, Sep. 2008.

[18] A. Motahari, S. Gharan, and A. Khandani, "Real interference alignment with real numbers," *Arxiv preprint arxiv:0908.1208*, 2009.

[19] R. Etkin and E. Ordentlich, "The degrees-of-freedom of the-user gaussian interference channel is discontinuous at rational channel coefficients," *IEEE Transactions on Information Theory*, vol. 55, pp. 4932–4946, Nov. 2009.

[20] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum likelihood detection and the search for the closest lattice point," *IEEE Transactions on Information Theory*, vol. 49, no 2, pp. 2389–2402, Oct. 2003.

[21] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Transactions on Information Theory*, vol. 45, pp. 1369–1642, July 1999.

[22] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, pp. 2201–2214, July 2002.

[23] W. H. Mow, "Universal lattice decoding: Principle and recent advances," *Wireless Communications and Mobile Computing, Special Issue on Coding and Its Applications in Wireless CDMA Systems*, vol. 3, pp. 553–569, Aug. 2003.

[24] L. Babai, "On lovász' lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6 no. 1, pp. 1–13, 1986.

[25] H. Yao and G. Wornel, "Lattice-reduction-aided detectors for MIMO communication systems," in *Proceedings of IEEE Global Communications (GLOBECOM)*, pp. 424–428, NOV. 2002.

[26] C. Windpassinger and R. Fischer, "Low-complexity near maximum-likelihood detection and precoding for MIMO systems using lattice reduction," in *Proceedings of IEEE Information Theory Workshop (ITW)*, pp. 345–348, Mar. 2003.

[27] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Math. Ann*, vol. 261, pp. 515–534, 1982.

[28] A. D. Murugan, H. E. Gamal, M. O. Damen, and G. Caire, "A unified framework for tree search decoding: redisovering the sequential decoder," *IEEE Transactions on Information Theory*, vol. 52 no.3, pp. 933–953, Mar. 2006.

[29] B. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Transactions on Communications*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[30] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoder," *IEEE Transactions on Wireless Commununications*, vol. 3, pp. 2299–2311, Nov. 2004.

[31] J. Hagenauer and C. Kuhn, "The list-sequential (LISS) algorithm and its application," *IEEE Transactions on Communications*, vol. 55, pp. 918–928, May 2007.

[32] T. Cover and A. Gamal, "Capacity theorems for the relay channel," *IEEE Transactions on Information Theory*, vol. 25, pp. 572–584, Sep. 1979.

[33] J. Laneman, D. Tse, and G. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Transactions on Information Theory*, vol. 50, pp. 3062–3080, Dec. 2004.

[34] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *IEEE Transactions on Information Theory*, vol. 51, pp. 3037–3063, Sep. 2005.

[35] S. Verdu, "The capacity region of symbol-asynchronous gaussian multiple-access channel," *IEEE Transactions on Information Theory*, vol. 35, pp. 733–751, Jul. 1989.

[36] S. Wei, "Diversity multiplexing tradeoff of asynchronous cooperative diversity in wireless networks," *IEEE Transactions on Information Theory*, vol. 53, pp. 4150–4172, Nov. 2007.

[37] S. Verdu, "Multiple-access channels with memory with and without frame synchronism," *IEEE Transactions on Information Theory*, vol. 35, pp. 605–619, May 1989.

[38] H. Zhang, N. B. Mehta, A. F. Molisch, J. Zhang, and H. Dai, "Asynchronous interference mitigation in cooperative base station systems," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 155–165, Jan. 2008.

[39] S. Zhang, S. C. Liew, and P. P. Lam, "On the synchronization of physical layer network coding," in *Proceedings of IEEE Information Theory Workshop (ITW)*, pp. 404–408, Oct. 2006.

[40] T. Han and K. Kobayashi, "A new achievable rate region for the interference channel," *IEEE Transactions on Information Theory*, vol. 27, pp. 49–60, Jan. 1981.

[41] R. Etkin, D. Tse, and H. Wang, "Gaussian interference channel capacity to within one bit," *IEEE Transactions on Information Theory*, vol. 54, pp. 5534–5562, Dec. 2008.

[42] A. Motahari and A. Khandani, "Capacity bounds for the gaussian interference channel," *IEEE Transactions on Information Theory*, vol. 55, pp. 620–643, Feb. 2009.

[43] M. Maddah-Ali, A. Motahari, and A. Khandani, "Communication over MIMO X channels: interference alignment, decomposition. and performance analysis," *IEEE Transactions on Information Theory*, vol. 54, pp. 3457–3470, Aug. 2008.

[44] S. Jafar and S. Shamai, "Degrees of freedom region of the MIMO X channel," *IEEE Transactions on Information Theory*, vol. 54, pp. 151–170, Jan. 2008.

[45] V. Cadambe and S. Jafar, "Interference alignment and degrees of freedom of the K user interference channel," *IEEE Transactions on Information Theory*, vol. 54, pp. 3425–3441, Aug. 2008.

[46] V. Cadambe and S. Jafar, "Interference alignment and the degrees of freedom of wireless X networks," *IEEE Transactions on Information Theory*, vol. 55, pp. 3893–3908, Sep. 2009.

[47] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, "Soft-input soft-output lattice sphere decoder for linear channels," in *Proceedings of IEEE Global Communications (GLOBECOM)*, pp. 1583–1587, Dec. 2003.

[48] C. Choudhuri and U. Mitra, "On the capacity of the symbol-asynchronous relay channel," in *Proceedings of 47th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1411–1418, Sep. 2009.

[49] D. Wang, S. Fu, and K. Lu, "Channel coding design to support asynchronous physical layer network coding," in *Proceedings of IEEE Global Communications (GLOBECOM)*, pp. 1–6, Dec. 2009.

[50] L. Grokop, D. Tse, and R. D. Yates, "Interference alignment for line-of-sight channels," *IEEE Transactions on Information Theory*, vol. 57, pp. 5820–5839, Sep. 2011.

[51] J. Zhan, B. Nazer, M. Gastpar, and U. Erez, "MIMO compute-and-forward," in *Proceedings of IEEE International Symposium on Information Theory*, pp. 2848–2852, June 2009.

[52] V. Cadambe and S. Jafar, "Degrees of freedom of wireless networks - what a difference delay makes," in *Proceedings of the Forty-First Asilomar Conference on Signals, Systems and Computers, 2007 (ACSSC 2007)*, pp. 133–137, Nov. 2007.

[53] M. Torbatian, H. Najafi, and M. O. Damen, "Asynchronous interference alignment," *IEEE Transactions on Wireless Communications*, 2012. (to appear).

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

[55] H. Cohen, *A course in computational algebraic number theory*. Springer, 2000.

[56] M. Taherzadeh, A. Mobasher, and A. Khandani, "LLL reduction achieves the receive diversity in MIMO decoding," *IEEE Transactions on Information Theory*, vol. 53, pp. 4801–4805, Dec. 2007.

[57] M. O. Damen, A. Chkeif, and J. C. Belfiore, "Lattice codes decoder for space-time codes," *IEEE Communications Letters*, vol. 4, pp. 161–163, May 2000.

[58] W. H. Mow, "Maximum likelihood sequence estimation from the lattice viewpoint," *IEEE Transactions on Information Theory*, vol. 40, pp. 1591–1600, Sep. 1994.

[59] I. Abou-Faycal, M. Medard, and U. Madhow, "Binary adaptive coded pilot symbol assisted modulation over rayleigh fading channels without feedback," *IEEE Transactions on Communications*, vol. 53, pp. 1036–1046, Jun. 2005.

[60] A. Edelman, "Eigenvalues and condition numbers of random matrices," *SIAM J. Matrix Anal. Appl.*, vol. 9, pp. 543–560, Oct. 1988.

[61] S. Szarek, "Condition numbers of random matrices," *J. Complexity*, vol. 7, no. 2, pp. 131–149, 1991.

[62] H. Daudé and B. Vallée, "An upper bound on the average number of iterations of the LLL algorithm," *Theoretical Computer Science*, vol. 123, no. 1, pp. 95–115, 1994.

[63] J. Jaldén, D. Seethaler, and G. Matz, "Worst-and average-case complexity of LLL lattice reduction in MIMO wireless systems," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2685–2688, 2008.

[64] H. Napias, "A generalization of the LLL algorithm over euclidean rings or orders," *J. Theorie Des Nombres de Bordeaux*, vol. 8, pp. 387–396, 1996.

[65] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, Mar. 1996.

[66] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and suboptimal MAP decoding algorithms operating in the log domain," in *Proceedings of International Conference on Communications*, pp. 1009–1013, June 1995.

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

[68] K. E. Baddour and N. C. Beaulieu, "Autoregressive modeling for fading channel simulations," *IEEE Transactions on Wireless Communications*, vol. 4 no. 4, pp. 1650–1662, July 2005.

[69] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst, "A vector-perturbation technique for near capacity multiantanna multiuser communication-part II," *IEEE Transactions on Information Theory*, pp. 537–544, Mar. 2005.

[70] C. Windpassinger, R. F. H. Fischer, and J. B. Huber, "Lattice-reduction-aided broadcast precoding," *IEEE Transactions on Communications*, pp. 2057–2060, Dec. 2004.

[71] S. Zhang, S. Liew, and P. Lam, "Hot topic: physical-layer network coding," in *Proceedings of the 12th annual international conference on Mobile computing and networking (MobiCom 2006)*, pp. 358–365, Sep. 2006.

[72] B. Nazer and M. Gastpar, "Reliable physical layer network coding," *Proceedings of the IEEE*, vol. 99, pp. 438–460, Mar. 2011.

[73] C. Feng, D. Silva, and F. Kschischang, "An algebraic approach to physical-layer network coding," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, pp. 1017–1021, June 2010.

[74] U. Niesen and P. Whiting, "The degrees of freedom of compute-and-forward," *arXiv:1101.2182 [cs.IT]*, Jan. 2011.

[75] A. S. Motahari, S. O. Gharan, M. A. Maddah-Ali, and A. K. Khandani, "Real interference alignment: Exploiting the potential of single antenna systems," *arXiv:0908.2282 [cs.IT]*, Aug. 2009.

[76] G. H. Golub and C. F. V. Loan, *Matrix Computations*. The Johns Hopkins University Press, 1996.

[77] B. Nazer, S. Jafar, M. Gastpar, and S. Vishwanath, "Ergodic interference alignment," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, pp. 1769–1773, July 2009.

[78] V. Cadambe, S. Jafar, and C. Wang, "Interference alignment with asymmetric complex signaling - settling the Høst–Madsen–Nosratinia conjecture," *IEEE Transactions on Information Theory*, vol. 56, pp. 4552–4565, Sep. 2010.

[79] R. Gray, *Toeplitz and circulant matrices: A review*. Now Pub, 2006.

[80] M. Seysen, "Simultaneous reduction of a lattice basis and its reciprocal basis," *Combinatorica*, vol. 13, pp. 363–376, 1993.

[81] A. Lapidoth, "Nearest neighbor decoding for additive non-Gaussian noise channels," *IEEE Transactions on Information Theory*, vol. 42, pp. 1520–1529, Sep. 1996.

[82] J. Nocedal and S. Wright, *Numerical optimization*. Springer verlag, 1999.