ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

A LOW COMPLEXITY DETECTOR FOR VERY LARGE MIMO

M.Sc. THESIS

Ergin ASLAN

Department of Electronics and Communication Engineering

Telecommunication Engineering Programme

JANUARY 2016

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

A LOW COMPLEXITY DETECTOR FOR VERY LARGE MIMO

M.Sc. THESIS

Ergin ASLAN
(504121348)

Department of Electronics and Communication Engineering

Telecommunication Engineering Programme

Thesis Advisor: Prof. Dr. M. Ertuğrul ÇELEBİ

JANUARY 2016

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

GENİŞ MIMO SİSTEMLERDE HIZLI KOD ÇÖZME YÖNTEMLERİ

YÜKSEK LİSANS TEZİ

Ergin ASLAN
(504121348)

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Telekomünikasyon Mühendisliği Programı

Tez Danışmanı: Prof. Dr. M. Ertuğrul ÇELEBİ

OCAK 2016

Ergin ASLAN, a M.Sc. student of ITU Graduate School of Science Engineering and Technology student ID 504121348, successfully defended the thesis entitled "A LOW COMPLEXITY DETECTOR FOR VERY LARGE MIMO", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. M. Ertuğrul ÇELEBİ** İstanbul Technical University	.....
Jury Members :	**Prof. Dr. H. Ümit AYGÖLÜ** İstanbul Technical University	.....
	Asst. Prof. Dr. Hacı İLHAN Yıldız Technical University	.....

Date of Submission : 26 November 2015

Date of Defense : 04 January 2016

To my family,

FOREWORD

Computing resources used in this work were provided by the National Center for High Performance Computing of Turkey (UHeM) under grant number < 4003502015>.

I would like to acknowledge the support of ITU-BAP program under the project "MIMO Sistemlerde Hızlı Kod Çözme Yöntemleri".

I would like to thank my supervisor Professor M. Ertuğrul ÇELEBİ for giving me valuable advice and support always when needed.

November 2015

Ergin ASLAN
(Electrical and Electronics Engineer)

TABLE OF CONTENTS

Page

ABBREVIATIONS

AWGN	: Additive white Gaussian noise
BPSK	: Binary phase shift keying
BER	: Bit error rate
CDMA	: Code division multiple access
FDMA	: Frequency division multiple access
GML	: Global maximum likelihood
HNN	: Hopfield neural network
ILS	: Integer least-squares
i.i.d.	: Independent and identically distributed
LAS	: Likelihood ascent search
LML	: local maximum likelihood
LTE	: Long term evolution
MF	: Matched filter
MF-SIC	: Matched filter successive interference cancellation
MIV-LAS	: Multiple initial vectors likelihood ascent search
MIMO	: Multiple-input Multiple-output
MISO	: Multiple-input Single-output
ML	: Maximum likelihood
MMSE	: Minimum mean-square error
MMSE-SIC	: Minimum mean-square error
MOS-LAS	: Multiple output selection likelihood ascent search
MRC	: Maximum ratio combining
MSCS-LAS	: Multiple search candidate sets likelihood ascent search
MUD	: Multiuser detection
PAM	: Pulse amplitude modulation
PIC	: Parallel interference cancellation
QAM	: Quadrature amplitude modulation
RTS	: Reactive tabu search
SCS	: Search candidate set
SD	: Sphere decoding
SIC	: Successive interference cancellation
SIMO	: Single-input Multiple-output
SISO	: Single-input Single-output
S-LAS	: Sequential likelihood ascent search
SNR	: Signal-to-noise ratio
TDMA	: Time division multiple access
V-BLAST	: Vertical Bell Laboratories Layered Space-Time Architecture
WiMAX	: Worldwide Interoperability for Microwave Access
ZF	: Zero forcing
ZF-SIC	: Zero forcing successive interference cancellation
3G	: Third generation
3GPP	: Third generation partnership project
4G	: Fourth generation

LIST OF FIGURES

Page

A LOW COMPLEXITY DETECTOR FOR VERY LARGE MIMO SYSTEMS

SUMMARY

Multiple input multiple output (MIMO) systems is an antenna technology which uses multiple antennas at the transmitter and the receiver. MIMO systems have several advantages including high date rates and spatial diversity. The capacity of these systems grows linearly with increasing number of antennas. In order to achieve full capacity of MIMO system, large number of antennas should be used at the transmitter and the receiver. This system is referred to as large MIMO or massive MIMO system.

One of the important problems in large MIMO system is the performance and complexity of the receiver. Optimum decoding in the sense of minimizing the average probability of bit error is maximum likelihood (ML) decoding. Bit error rate (BER) performance of ML decoding is best, but computational complexity of this decoding algorithm is exponential, i.e., $O(M^{n_t})$, where M is cardinality of modulation alphabet and n_t is number of transmit antennas. Because of the computational complexity, ML decoding algorithm is not feasible to use in large MIMO systems.

There are some well-known low complexity detectors like zero-forcing (ZF), matched filter (MF), and minimum mean-square error (MMSE). These decoding techniques have low computational complexity, but they achieve poor performance. Therefore, these decoding algorithms cannot be used in large MIMO systems.

Another decoding method is likelihood ascent search (LAS) which is based on local neighborhood search, have been investigated. This method converges to the performance of single input single output (SISO) additive white Gaussian noise (AWGN) for large number of antennas and has lower computational complexity than ML decoding.

In this thesis, a low complexity decoding algorithm based on discrete-time Hopfield neural network (DHNN) is proposed. In this decoding algorithm, asynchronous DHNN is used which updates each neuron serially. The proposed decoding algorithm starts with an initial solution vector which is generated randomly or by well-known low complexity detectors like ZF, MMSE, or MF. Then the algorithm updates neurons until given stopping criteria is satisfied. When network reaches to a stable state, then stopping criteria is satisfied and algorithm terminates.

It is shown that, the proposed detector converges to SISO AWGN performance for large number of antennas. The computational and algorithmic complexity of the proposed detector is much lesser than the complexity of ML decoding. Using the proposed decoding algorithm, there are no necessity to conditional evolution, comparison to multiple thresholds and multi-level quantizations for updating in the search stage.

ÇOK ANTENLİ SİSTEMLERDE HIZLI KOD ÇÖZME TEKNİKLERİ

ÖZET

Çok girişli çok çıkışlı sistemler (MIMO) alıcı ve vericide birden fazla anten kullanan sistemlerdir. Bu sistemler sığa kazancı, daha güvenilir bağlantı ve etkin güç kullanımı gibi avantajlar sağlar. Çok antenli sistemler kablosuz iletişim ağlarının performansını arttırır.

MIMO sistemlerin sağlamış olduğu avantajlardan birisi de çeşitleme kazancıdır. MIMO sistemler çeşitleme kazancını artırarak kanal sönümlemesini azaltmak amacıyla kullanılmaktadır. Bu sistemlerin sağladığı maksimum çeşitleme kazancı $n_t x n_r$ dir. Burada n_t vericideki anten sayısını ve n_r alıcıdaki anten sayısını göstermektedir.

Bu sistemlerin sunmuş olduğu diğer bir avantaj ise yüksek veri hızıdır. Veri hızı bilgi dizisi antenlerden paralel olarak gönderilerek artırılabilir. Bir kanalda gönderilecek veri hızı kanal sığası ile sınırlıdır. Kanal sığası haberleşme sistemleri için bir performans ölçütüdür ve güvenilir bir iletişim için sağlanabilecek en yüksek veri hızını belirtir. Rayleigh sönümlemeli kanallar için kanal sığasının alıcı ve vericideki minimum anten sayısı ile doğrusal olarak arttığı görülmüştür.

Görüldüğü gibi, MIMO sistemlerin sağlamış olduğu avantajlar alıcı ve vericide kullanılan anten sayısı ile orantılı artmaktadır. Dolayısıyla, MIMO sistemlerin potansiyeli kullanılan anten sayısı arttıkça artar. Yüksek potansiyele ulaşmak için, MIMO Sistemlerde alıcı ve vericide çok fazla sayıda anten kullanılmalıdır. Böyle sistemler literatürde çok geniş (very large) MIMO veya yoğun (massive) MIMO olarak adlandırılırlar.

Geniş MIMO sistemlerde en önemli problemlerden biri vericiden gönderilen bilgi dizisinin alıcıda tekrar çözülmesidir. Verici antenlerden eş zamanlı olarak gönderilen işaretlerin alıcıda uzaysal girişim yaratmasından dolayı, geniş MIMO sistemlerde alıcıda alınan işaretin yeniden elde edilmesi tek girişli tek çıkışlı (SISO), tek girişli çok çıkışlı (SIMO) ya da çok girişli tek çıkışlı (MISO) sistemlere göre daha karmaşıktır. Uzaysal girişim olduğunda, alıcıda işaretin yeniden elde edilmesi karışık işaret işleme teknikleri gerektirmektedir.

Ortalama bit hata olasılığını minimize etmede en iyi alıcı en büyük olabilirlikli (ML) alıcıdır. ML alıcı karesel Öklid uzaklığını minimize etme problemini çözer. Anten sayısının çok büyük olduğu durumlarda, sönümlemesiz tek girişli tek çıkışlı (SISO) toplamsal beyaz Gauss gürültülü (AWGN) kanalın performansı ML alıcının performansının üst sınırını vermektedir. ML alıcının sembol hata olasılığı (SER) performansı en iyidir. Buna rağmen bu alıcının hesaplama karmaşıklığı vericideki anten sayısı ile üsteldir. Öyle ki, M modülasyon kümesinin eleman sayısını ve n_t vericide kullanılan anten sayısını gösterdiğinde, ML alıcının hesaplama karmaşıklığı $O(M^{n_t})$ olur. Hesaplama karmaşıklığının bu kadar yüksek olmasından dolayı, ML

alıcı sadece kullanılan anten sayısının küçük olduğu uygulamalarda kullanılabilir ve pratikte geniş MIMO sistemlerde kullanılmaya uygun değildir.

Literatürde, bazı iyi bilinen düşük hesaplama karmaşıklığına sahip alıcılar bulunmaktadır. Bu alıcılara sıfır zorlama (ZF), uyumlu süzgeç (MF), minimum ortalama hata kare (MMSE) örnek olarak verilebilir. Bu kod çözme teknikleri düşük hesaplama karmaşıklığına sahiptirler. MMSE ve ZF alıcıların sembol başına düşen hesaplama karmaşıklığı $O(n_t^2)$ dir. MF alıcının ise sembol başına düşen hesaplama karmaşıklığı $O(n_t)$dir. Burada $n_t = n_r$ olarak farz edilmiş ve hesaplama karmaşıklığı ona göre verilmiştir. Bu alıcıların hesaplama karmaşıklığı ML kod çözme yönteminin hesaplama karmaşıklığına göre oldukça düşüktür. Buna rağmen, bu alıcıların BER performansları MIMO sistemde alıcı ve vericide kullanılan anten sayısı arttıkça düşmektedir. Bu yüzden bu alıcılar geniş MIMO sistemlerde kullanılmaya uygun değildirler.

Literatürde SISO AWGN kanalın BER performansına yaklaşan ve düşük hesaplama karmaşıklığına sahip kod çözme yöntemleri bulunmaktadır. Bunlardan biri de yerel komşuluk aramaya dayalı, olabilirlik yükselen aramadır (LAS). Bu kod çözme algoritmasının performansı, alıcı ve vericide yüksek sayıda anten kullanıldığında, SISO AWGN kanalın performansına yaklaşmaktadır. LAS algoritmasının hesaplama karmaşıklığı yaklaşık olarak ZF ve MMSE gibi alıcıların hesaplama karmaşıklıkları ile aynıdır. Bu da ML alıcıya göre çok daha düşük bir karmaşıklığa sahip olduğunu göstermektedir.

Yapay sinir ağlar çok girişli doğrusal olmayan aygıtlar olarak tanımlanabilir. Bu ağlarda sinirler giriş değerlerine, her bir bağlantının sinirsel ağırlığına ve eşik değerine bağlı olarak çıkış değeri üretirler. Yapay sinir ağlar uzun yıllardır optimizasyon alanında kullanılmaktadır. Yapay sinir ağlarında farklı birçok model bulunmaktadır. Bu modellerden biri de Hopfield sinir ağlarıdır. Hopfield sinir ağlarının optimizasyon alanında kullanımı sinir ağlarının hızlı olmasından ve bu optimizasyon problemlerinin hızlı bir şekilde çözülebilmesinden kaynaklanmaktadır. Hopfield sinir ağlarının optimizasyon problemlerindeki uygulamaları Lyapunov (enerji) fonksiyonuna dayanmaktadır. Bu fonksiyon ağ değişirken azalmaktadır. Bu yüzden ağ durağan duruma geldiği zaman enerji fonksiyonunun yerel minimum noktası bulunmuş olur. Hopfield ağı ile özel bir optimizasyon problemini çözmek için Lyapunov fonksiyonunun hedef problemin maliyet fonksiyonuna göre düzenlenmesi ve buna göre ağın tasarlanması gerekmektedir. Daha sonra Hopfield sinir ağı dinamik olarak optimizasyon probleminin çözümünü bulmaktadır.

Bu tezde, düşük hesaplama karmaşıklığına sahip, ayrık zaman Hopfield sinir ağı (DHNN) tabanlı bir alıcı önerilmektedir. Bu alıcı algoritmasında her seferde tek bir sinir güncellemesi yapan asenkron (seri) DHNN kullanılmıştır. Lyapunov fonksiyonu ML kod çözme probleminin maliyet fonksiyonuna göre düzenlenmiş ve Hopfield sinir ağı buna göre tasarlanmıştır. Her bir sinirin başlangıç değeri ZF, MMSE ve MF gibi bilinen kod çözücüler ile ya da rastgele olarak üretilmiştir. Hopfield ağı, güncelleme fonksiyonuna göre her seferde bir siniri günceller. Sinir ağın her bir sinir güncellemesinde maliyet fonksiyonu azalmaktadır. Böylece başlangıçta kullanılan filtrelere göre daha düşük bir yerel minimum noktası bulunabilmekte ve önerilen kod çözme algoritmasının BER performansı artmaktadır. Algoritma verilen durma ölçütü sağlanana kadar sinirleri güncellemeye devam etmektedir. Ağ durağan duruma geldiği

zaman durma ölçütü sağlanmış olur ve algoritma güncellemeleri durdurur. Böylece algoritma çözüm kümesini bulmuş olur.

Önerilen kod çözme algoritmasının BER performansının, alıcı ve vericide yüksek sayıda anten kullanıldığında, SISO AWGN performansına yaklaştığı görülmektedir. Hesaplama karmaşıklığı ML kod çözme algoritmasına göre daha düşüktür. Bu yöntem kullanıldığında, çok seviyeli kuantalama, çoklu eşik değerleri ile karşılaştırma ve bir takım koşulların kontrol edilmesi gibi işlem yükü gerektiren hesaplamalara gerek kalmamaktadır.

1. INTRODUCTION

Large MIMO refers multiple-input multiple-output system which uses tens to hundreds antennas at the transmitter and the receiver. It has all benefits of conventional MIMO, but on a much greater scale. In order to reveal the potential of MIMO systems, large number of antennas should be used. This can help to improve performance and get full potential of MIMO systems.

Decoding is one of the challenging problems in large MIMO systems. Decoding algorithms should satisfy high BER performance and low computational complexity. In order to satisfy these, complicated algorithms are required.

ML detector is the optimum detector in the sense of minimizing the cost function. However, it is not feasible to use in large MIMO systems due to the exponential computational complexity in transmitter antennas. Sphere decoder [1] is another decoding method which gives BER performance of ML decoder. However, computational complexity is still exponential. Some well-known detectors like ZF, MF, and MMSE have low computational complexity, but they are also not feasible to use in large MIMO systems due to the poor BER performance [2, 3].

There are some other approaches which solve this problem. LAS [4-7] and RTS [8-13] algorithms, which are based on local neighborhood search, converges the performance of SISO AWGN when large number of antennas are used at the transmitter and the receiver. These algorithms have low computational complexity compared to the computational complexity of ML decoder. Probabilistic data association [14-16] is another approach which is also able to achieve good performance at low complexities in large dimensions.

The objective of this thesis work is to show the benefit of using large number of antennas and investigate the decoding in large MIMO systems. Main advantages of MIMO systems and various decoding methods are explained in this thesis.

Specific research goal of this thesis is investigation of a decoding algorithm based on discrete-time Hopfield neural network which has a low computational complexity and high BER performance in large MIMO systems.

The thesis is organized in seven chapters. Chapter 2 broadly studies MIMO systems. The main advantages of MIMO systems, which are diversity gain and channel capacity, are discussed to get better understanding why large number of antennas are needed. Chapter 3 discusses the decoding methods in the literature.. Their BER performances and computational complexities are given and compared. Chapter 4 discusses general structure of artificial neural networks and application of Hopfield neural networks which are used in optimization problems. Chapter 5 studies the algorithm of the proposed detector and analysis the computational complexity of it. Chapter 6 gives computer simulations of the BER performance of the proposed detector. Chapter 7 gives the conclusion of the thesis work.

2. MIMO SYSTEMS

In this chapter, multiple-input multiple-output (MIMO) system will be discussed. Performance criteria of this system will be given. Why multiple antennas at the transmitter and the receiver should be needed and how potential of using multiple antennas can be realized will be investigated. In order to compare performance of MIMO with performance of other systems, other antenna configurations will be examined.

2.1 Communication Systems

There are several antenna configurations in the communication systems, which are given as follows.

Single input single output systems (SISO) is conventionally used in communication systems: the input is the signal transmitted from a single antenna and the output is the signal received from a single antenna. Antenna configuration of the SISO system is given in Figure 2.1.

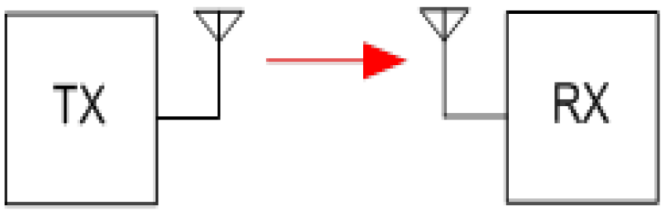

Figure 2.1: SISO Communication System [17].

Multiple input single output systems (MISO) uses multiple antennas at the transmitter, and single antenna at the receiver. Antenna configuration of MISO system is given in Figure 2.2.

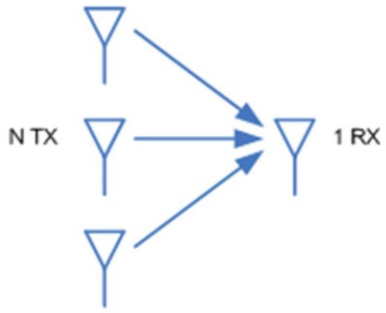

Figure 2.2: MISO communication system [17].

Single input multiple output systems (SIMO) uses single antenna at the transmitter, and multiple antenna at the receiver. Antenna configuration of SIMO system is given in Figure 2.3.

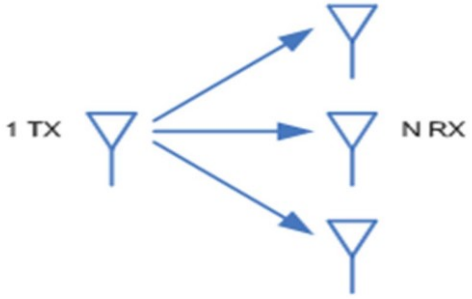

Figure 2.3: SIMO communication system [17].

Multiple input multiple output system (MIMO) is an antenna technology which uses multiple antennas at the transmitter and the receiver. Antenna configuration of MIMO communication system is given in Figure 2.4.

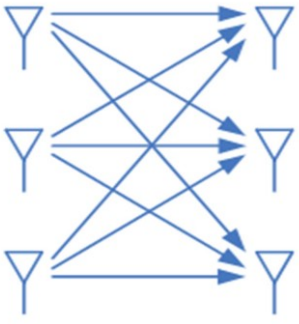

Figure 2.4: MIMO communication system [17].

MIMO systems can be classified as multiuser MIMO (MU-MIMO) and point to point MIMO. In point to point MIMO systems, a single multi-antenna transmitter communicates with a single multi-antenna receiver. On the other hand, in MU-MIMO

systems, a set of users or wireless terminals with one or more antennas communicate with a multi-antenna base station (BS). MU-MIMO systems add multiple access (multiple users) capabilities to MIMO. Block scheme of these systems is given in Figure 2.5 and Figure 2.6 respectively.

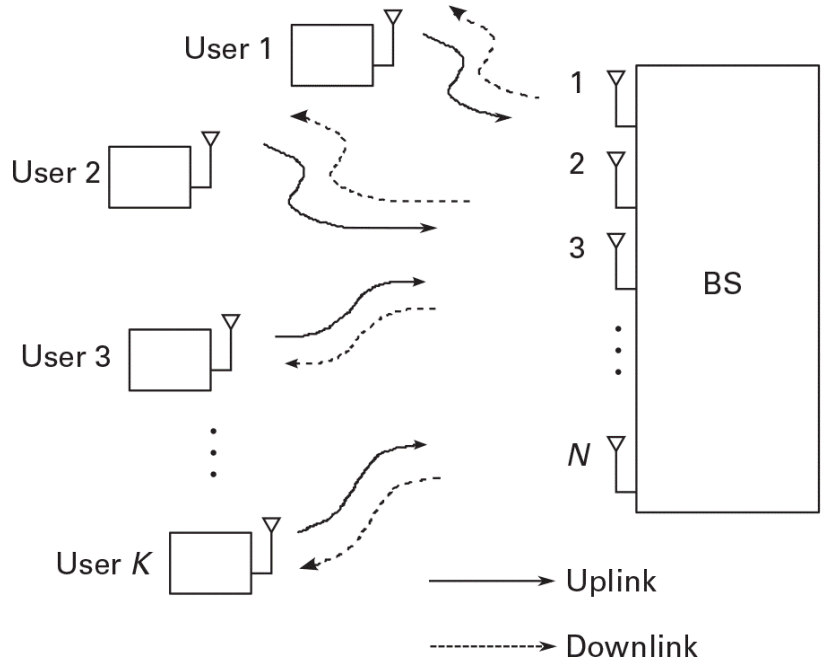

Figure 2.5: Multiuser MIMO systems [18].

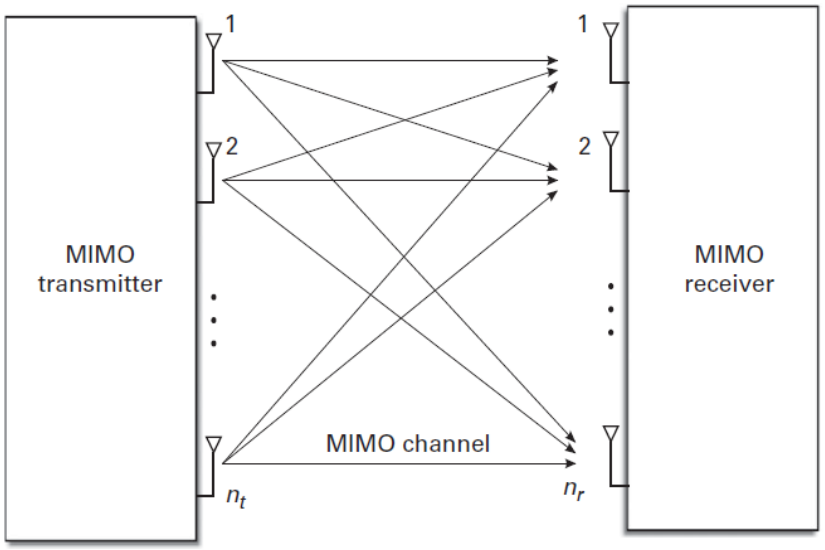

Figure 2.6: Point-to-point MIMO Systems [18].

Because of the advantages of MIMO, it is already being used in current communication systems. Furthermore, MIMO has become an essential element of wireless communication standards including IEEE 802.11n (IEEE 802.11), IEEE 802.16 (WiMAX). MIMO will also be used in the future standards for mobile communications

[19]. Main advantages of MIMO systems, which are diversity and channel capacity, are explained in the following sections.

2.1 Diversity

The signal strength varies over small and large scale as user moves, and at times, the quality of the link is too low to deliver data successfully. This causes unacceptable error rates or radio link failure. A technique known as diversity has been developed in order to combat this problem. Diversity is based on the use of multiple copies of the same signal. The receiver can combine this copies or select from them.

There are different types of diversity techniques. One can generate multiple copies of the same signal by transmitting it multiple times, which gives rise to time diversity, or one can generate multiple copies of the same signal at different parts of the spectrum, which gives rise to frequency diversity. Furthermore, one can exploit the space domain, for example when the same signal is transmitted from several base station antennas and received at a single mobile terminal, or a different copy of the signal is received by spatially separated antennas of the receiver.

Diversity techniques can also be classified according to the way the multiple copies of the signals are exploited. In selection diversity, the best copy of the signal is selected. In equal gain combining, multiple copies of the signal are added. In maximum ratio combining (MRC), multiple copies of the signal are weighted by appropriately selected scaling factors such that the quality of the resulting signal is optimized.

In multiple antenna systems, spatial diversity technique is used. Spatial diversity techniques can be classified as transmit diversity and receive diversity. In transmit diversity techniques, multiple copies of the signal are transmitted from several antennas, and their superposition is received at a single receive antenna. In receive diversity techniques, the signal is transmitted from a single antenna and multiple copies of signal are received at several antennas. This combinations give rise to MISO system, and SIMO systems, respectively.

The evolution of the diversity ideas is the use of multiple antennas at both ends of the communications link. MIMO systems have been used to increase diversity to combat channel fading [20]. Multiple faded replicas of the data symbol is obtained at the receiver by sending same information through different paths. Thus, reliability can be

increased. In a MIMO system with n_t transmit and n_r receive antennas with the assumption that the path gains between individual antenna pairs are independent and identically distributed (i.i.d.), then, the maximal diversity gain of this system is $n_t n_r$.

When the transmission rate R is fixed, the achievable diversity order d for a given scheme can be given as follow [20].

$$d = -\lim_{SNR\to\infty} \frac{logP_e(SNR)}{logSNR} \tag{2.1}$$

where P_e is probability of codeword error. Similarly, (2.1) can be written as

$$P_e(SNR) \propto SNR^{-d} \tag{2.2}$$

Hence, it can be seen that probability of codeword error decreases with increasing SNR and diversity order d. For a V-BLAST system, it can be shown that diversity order is n_r because symbols transmitted from the antennas reach the receiver only through n_r different paths. Space-time block coding can achieve the full diversity gain of $n_r n_t$ [21]. In this coding technique, symbols are coded across both space and time. Hence full diversity can be achieved.

2.2 Capacity

Another benefit of the use of multiple antennas at both sides is spatial multiplexing, that is, the ability to send several independent data streams simultaneously. For a given scheme, the spatial multiplexing gain r is given as follow.

$$r = \lim_{SNR\to\infty} \frac{R(SNR)}{logSNR} \tag{2.3}$$

where $R(SNR)$ (bit/symbol) is the rate of the code $\mathcal{C}(SNR)$. Diversity gain is given previous section. There is a trade-off between the multiplexing gain and diversity gain[20].

The data rate can be increased by sending independent information streams in parallel through spatial channels [22].This rate is limited by the channel capacity.

Capacity is a performance measure for digital communication systems. The capacity of the communication system is the maximal transmission rate for which a reliable

communication can be achieved. A communication at rate R is defined as reliable if one can design a code at rate R that makes the error probability arbitrarily small. If the transmission rate gets larger than the capacity, the system breaks down and receiver makes decoding errors with a non-negligible probability. For any transmission rate R smaller than or equal to the capacity, a reliable communication can be achieved. Capacity is given under a transmit power constraint. If the transmit power is unlimited, the capacity is infinite. Hence, no matter what transmission rate is, the probability error can always be made small by increasing the transmit power.

There are several studies on channel capacity. The Capacity of a fixed channel with additive white Gaussian noise (AWGN) was first derived by Shannon [23]. The theoretical prediction of very high wireless channel capacities in rich scattering environments has been shown by Telatar in [24], and Foschini and Gans in [25]. Vertical Bell laboratories layered space-time architecture (V-BLAST) multiantenna wireless system is investigated by Bell Labs in [26].

In wireless communications, it became apparent that the channel is not a constant quantity, thus, the effects of time-varying channel became the object of theoretical and practical study in [27-30].

Capacity varies according to the nature of the wireless channel and the channel knowledge. A wireless channel varying in time is defined as fading. Fading channels can be classified as slow fading or fast fading depending on the fading rate relative to the signaling rate. The fading is termed slow fading (or time-flat) if the fading remains constant over the signaling duration. If the fading varies within the signaling duration, it is termed fast fading (or time-selective).

The gain information about the channel is called channel state information (CSI). CSI can take different forms including the channel coefficients, the modulus of the channel, the statistic of the channel or the noise variance at the receiver. CSI at the receiver (CSIR) refers to the scenario where the receiver has the knowledge of the channel state information. CSI at the transmitter (CSIT) refers to the scenario where the transmitter has the knowledge of the channel state information.

2.2.1 The capacity of SISO system for time-invariant channel

In this section, the capacity of SISO system [19] will be explained. The channel is assumed to be time-invariant channel.

Let the power of transmit antenna i is $E|x_i|^2$. Thus, the sum of transmit powers is given as

$$\sum_{i=1}^{n_t} E|x_i|^2 = tr(\boldsymbol{R}_{xx}) \leq \bar{P} \tag{2.4}$$

The average total transmit power is constrained to be smaller than a set value $\bar{P}$. The transmit power is set constant to its maximal value $\bar{P}$ in order to maximize the transmit rate.

The input to output relationship of the discrete time time-invariant SISO channel can be written as

$$y(k) = hx(k) + n(k) \tag{2.5}$$

h is the value of channel and it is independent of time.

The signal to noise ratio (SNR) is defined as the ratio between the signal power and the noise power:

$$SNR_{TI} = \frac{\bar{P}|h|^2}{\sigma_n^2} \tag{2.6}$$

where σ_n^2 is the power in noise $n(k)$.

The capacity of SISO system for a time-invariant channel is given below

$$C_{TI}(SNR_{TI}) = \log_2(1 + SNR_{TI}) \tag{2.7}$$

The unit of capacity is bit per transmission. It is the number of bits which are transmitted per symbol transmission time T_s. The capacity can also be defined as $C_{TI}(SNR_{TI})/T_s$ with unit bits per second.

Spectral efficiency is an alternative performance measure to capacity. It is expressed in bit per second per hertz (bits/s/Hz). Spectral efficiency is equal to capacity per second normalized by the system bandwidth W. It can be expressed as

$$S_{TI}(SNR_{TI}) = \frac{C_{TI}(SNR_{TI})}{T_s W} \tag{2.8}$$

2.2.2 The capacity of SIMO system for time-invariant channels

The channel capacity of SISO system is given above. Now, the capacity of SIMO system [19] will be described. The SIMO system model with receiver beamforming is given in Figure 2.7.

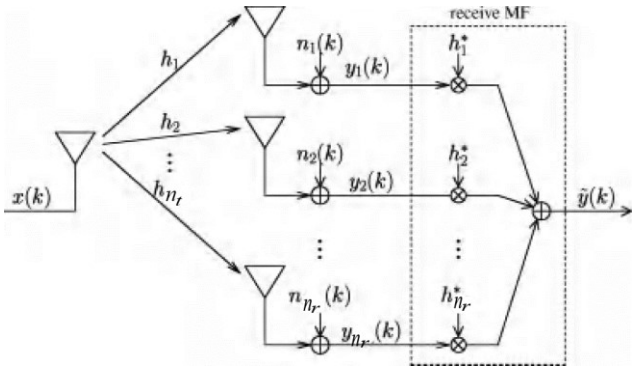

Figure 2.7: SIMO system model with receiver beamforming [19].

In a SIMO system, one input symbol $x(k)$ is sent from a single transmit antenna and received at n_r receive antennas. Optimal receiver should be described, which achieving capacity. This receiver involves a coherent combining of the signals from multiple antennas based on the CSIR.

The SIMO channel is assumed to be time-invariant. The channel coefficients are known at the transmitter. Received signal by *j*th receive antenna is given below

$$y_j(k) = h_j x(k) + n_j(k) \tag{2.9}$$

A proper processing is necessary in order to extract a maximal amount of information about the transmitted symbol. The optimal processing is a linear processing called receive spatial matched filtering (MF) or receive maximum ratio combining (MRC). MF maximizes the post-processing SNR. The received signal at receive antenna j is multiplied by a scalar coefficient h_j^*matched to the channel. The processed signals from all antennas are added up, and it is given as

$$\tilde{y}(k) = \|\boldsymbol{h}\|^2 x(k) + \tilde{n}(k) \tag{2.10}$$

where $\boldsymbol{h} = \left[h_1, h_2, \ldots, h_{n_r}\right]^T$ and $\tilde{n}(k) = \sum_{j=1}^{n_r} h_j^* \, n_j(k)$. The SNR of SIMO channel can be given as

$$SNR_{SIMO} = \frac{\bar{P}\|\boldsymbol{h}\|^2}{\sigma_n^2} \tag{2.11}$$

The capacity of time-invariant SIMO system is given as

$$C_{SIMO}^{TI} = \log_2\left(1 + \frac{\bar{P}\|\boldsymbol{h}\|^2}{\sigma_n^2}\right) \tag{2.12}$$

The benefit provided by multiple antennas at the receiver can be seen from the SNR expression given in (2.11).

2.2.3 The capacity of MISO system for time-invariant channels

In a MISO system several antennas are used at the transmitter and single antenna is used at the receiver. It is assumed that the channel state information is known at the transmitter (CSIT). In order to achieve capacity of the channel, pre-processing should be performed. In this scheme, sending signal is a simple scaled of the transmitted symbol from each antenna. Then, all the signals are added up at the receiver. The system model is given in Figure 2.8.

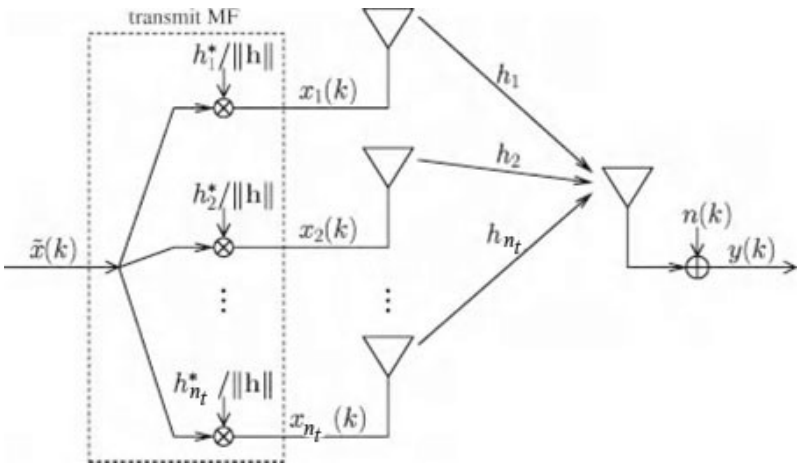

Figure 2.8: MISO system model with transmit beamforming [19].

$x_i(k)$ is sent from antenna i and received from time-invariant channel h_i in the time-invariant MISO system. The channel coefficients are known at the transmitter and denoted as $h_i, i = 1,2,\dots,n_t$. The received signal is given as

$$y(k) = \sum_{i=1}^{n_t} h_i x_i(k) + n(k) = \boldsymbol{h^T x}(k) + n(k) \tag{2.13}$$

where $\boldsymbol{h} = \left[h_1 \dots h_{n_t}\right]^T$ and $\boldsymbol{x(k)} = \left[x_1(k) \dots x_{n_t}(k)\right]^T$.

A preprocessing called transmit spatial matched filtering or transmit MRC is implemented at the transmitter in order to achieve capacity. This preprocessing technique also referred as to maximum ratio transmission[31]. Only one symbol $\tilde{x}(k)$ is sent at time k in transmit MF. The transmit symbol $x_i(k)$ is scaled by the coefficient h_i^* matched to the channel h_i. The output of the pre-processing is normalized to comply with the transmit power constraint. The signal sent from antenna i is given below.

$$x_i(k) = \frac{h_i^*}{\|\boldsymbol{h}\|}\tilde{x}(k) \tag{2.14}$$

The system output is given in (2.15). This system is a scalar AWGN channel.

$$y(k) = \|\boldsymbol{h}\|\tilde{x}(k) + n(k) \tag{2.15}$$

The SNR at the receive antenna is given in (2.16).

$$SNR_{MISO} = \frac{\bar{P}\|\boldsymbol{h}\|^2}{\sigma_n^2} \tag{2.16}$$

The capacity of time-invariant MISO channel is given in (2.17).

$$C_{MISO}^{TI} = \log_2\left(1 + \frac{\bar{P}\|\boldsymbol{h}\|^2}{\sigma_n^2}\right) \tag{2.17}$$

The optimal pre-processing transforms the MISO channel into an equivalent SISO channel. The capacity of SIMO and MISO system is same for time-invariant channels when channel state information is known at the transmitter (CSIT). However, it is not true for fading channels when CSIT is not known at the transmitter.

2.2.4 The capacity of MIMO system for time-invariant channels

In MIMO systems, multiple antennas are used simultaneously at both the transmitter and the receiver. The transmission and reception of each symbol relies on pre-processing and post-processing. This processes are matched to the underlying structure

of the channel based on their singular value decomposition (SVD). Multiple pairs of pre-processing and post processing create multiple spatial routes which are independent. Hence, MIMO system becomes equivalent to multiple SISO channels. The capacity of MIMO becomes the sum of the capacity of each SISO channel.

The SVD of the channel matrix $\boldsymbol{H}$ is given as

$$\boldsymbol{H} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{H}} \tag{2.18}$$

where $n_r x n_r$ matrix $\boldsymbol{U}$ and $n_t x n_t$ matrix $\boldsymbol{V}$ are unitary matrices. $\boldsymbol{\Lambda}$ is $n_r x n_t$ diagonal matrix with nonnegative singular values $\lambda_k, k = 1,2,\dots,n_{min}$, where $n_{min} = \min(n_t, n_r)$. The singular values λ_k are ordered decreasingly and called the eigenmodes of the channel.

r_H is the rank of $\boldsymbol{H}$ which defined as the number of nonzero singular values of **H**. The rank is also defined as the minimal value between the number of independent rows and the number of independent columns of $\boldsymbol{H}$. Hence, it can be seen that the rank $r_H \leq \min(n_t, n_r)$. The rank of the channel matrix determines the maximum number of independent streams which can be multiplexed simultaneously.

The energy in the channel can be written as

$$tr(HH^H) = \sum_{i=1}^{n_t}\sum_{j=1}^{n_r}\left|h_{ji}\right|^2 = \sum_{k=1}^{r_H}\lambda_i^2 \tag{2.19}$$

The equivalent MIMO model with pre-processing and post-processing is given in Figure 2.9.

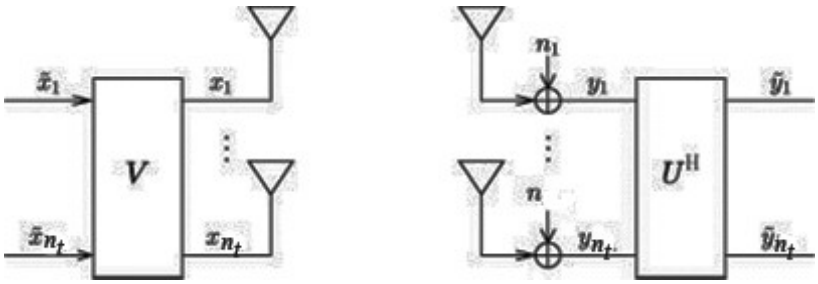

Figure 2.9: Equivalent MIMO model [19].

The matrix $\boldsymbol{V}$ is transmit pre-processing matrix which are matched to the conjugate of the right singular vectors of $\boldsymbol{H}$. The signal can be given in (2.20), after being transferred through the channel.

$$\boldsymbol{Hx} = \boldsymbol{U\Lambda V^H V\tilde{x}} = \boldsymbol{U\Lambda\tilde{x}} \tag{2.20}$$

The vector $\tilde{\boldsymbol{y}}$ is obtained by post processing the received signal **y** by $\boldsymbol{U^H}$. The vector input-output relationship of the equivalent system can be given as

$$\tilde{\boldsymbol{y}} = \boldsymbol{\Lambda\tilde{x}} + \tilde{\boldsymbol{n}} \tag{2.21}$$

where $\tilde{\boldsymbol{n}} = \boldsymbol{U^H n}$ is a linear combination of Gaussian variables and keeps the same statistics as $\boldsymbol{n}$: $\tilde{\boldsymbol{n}} \sim \mathcal{CN}(0, \sigma^2 \boldsymbol{I})$. $\mathbb{E}(\tilde{\boldsymbol{x}}\tilde{\boldsymbol{x}}^H) = \mathbb{E}(\boldsymbol{xx}^H)$ and $\mathbb{E}(\tilde{\boldsymbol{n}}\tilde{\boldsymbol{n}}^H) = \mathbb{E}(\boldsymbol{nn}^H)$ due to the unitary matrices $\boldsymbol{U}$ and $\boldsymbol{V}$.

Each output of the equivalent system in (2.21) is given below

$$\tilde{y}_k = \begin{cases} \lambda_k \tilde{x}_k + \tilde{n}_k & \text{for } k = 1,2,\dots,r_H \\ \tilde{n}_k & \text{for } k = r_H + 1,\dots,n_r, \text{when } r_H < n_r \end{cases} \tag{2.22}$$

Each independent channel is referred as eigenchannel or alternatively subchannel. Each eigenchannel has different SNR. The associated channel coefficient is an eigenvalue of the channel matrix. Each input-output relationship in (2.22) describes an AWGN channel. Because the additive noises are all independent from each other, those AWGN subchannels are all independent. On the other words, an optimal coding can be implemented independently for each AWGN subchannel. Hence, the capacity of the MIMO system is the sum of the individual capacities. General structure of the eigenmodes transmission is given in Figure 2.10.

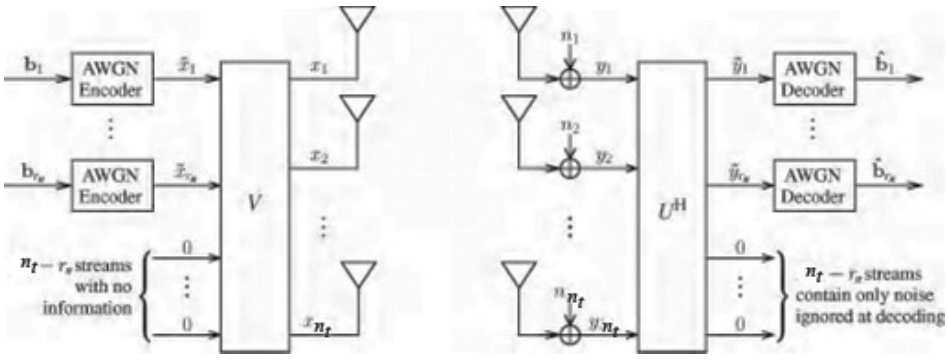

Figure 2.10: General structure of eigenmodes transmission, achieving capacity [19].

The following quantities are defined for each eigenchannel as

$$\gamma_k = \frac{\lambda_k^2}{\sigma_n^2} \tag{2.23}$$

When $P_k$is defined as the transmit power of eigenchannel k, $P_k\gamma_k$ is the SNR of the kth eigenchannel. While complying with the overall transmit power constraint with $\sum_{k=1}^{r_H} P_k \leq \bar{P}$, P_k is adjusted to maximize the capacity of the MIMO system. The capacity of the MIMO system is given in (2.24). It is the sum of the individual capacities with optimized transmit power per eigenchannel.

$$C_{MIMO}^{TI} = \max_{P_k \, s.t. \sum_{k=1}^{r_H} P_k \leq \bar{P}} \left\{ \sum_{i=1}^{r_H} \log_2(1 + P_k\gamma_k) \right\} \tag{2.24}$$

If the covariance matrix $\boldsymbol{R}_{xx}$ is fixed, the capacity of the time-invariant MIMO system can be rearranged as

$$C_{MIMO}^{TI} = \max_{\boldsymbol{R}_{xx}, tr(\boldsymbol{R}_{xx}) \leq \bar{P}} \log_2 \det\left[I + \frac{1}{\sigma_n^2} \boldsymbol{H}\boldsymbol{R}_{xx}\boldsymbol{H}^H \right] \tag{2.25}$$

2.2.5 Capacity for fading channels

Consider a narrowband MIMO channel with vector output at time k.

$$\boldsymbol{y}(k) = \boldsymbol{H}(k)\boldsymbol{x}(k) + \boldsymbol{n}(k) \tag{2.26}$$

where $\boldsymbol{H}(k)$ is fading channel process. It is assumed stationary and ergodic. The fading and noise processes are independent. Distribution of CSIT is known at the transmitter. CSIR is known at the receiver. Coding delay is large compared to the channel coherence time. The transmit power constraint is given in (2.27).

$$\sum_{i=1}^{n_t} \mathbb{E}[|x_i|^2] = tr(\boldsymbol{R}_{xx}) \leq \bar{P} \tag{2.27}$$

Even if the instantaneous value of the CSIT is not known, a maximal rate for reliable communication can be determined based on the CSIT distribution. This rate is constant over the channel fading and is the ergodic capacity.

Consider a narrowband SISO channel at time k:

$$y(k) = h(k)x(k) + n(k) \tag{2.28}$$

The transmit power constraint is $\mathbb{E}[|x_i|^2] \leq \bar{P}$. The instantaneous SNR at time k is given below:

$$SNR(k) = \frac{\bar{P}|h(k)|^2}{\sigma_n^2} \tag{2.29}$$

The ergodic capacity for a fast fading SISO channel with CSIT distribution can be written as

$$C_{SISO}^{FF} = \mathbb{E}\left[\log_2\left(1 + \frac{\bar{P}|h|^2}{\sigma_n^2}\right)\right] \tag{2.30}$$

Consider the input-output relationship of a fast fading SIMO channel given in **(2.**26**)**. The instantaneous SNR is given in (2.31) and the ergodic capacity of a fast fading SIMO channel with CSIT distribution can be given in (2.32).

$$SNR(k) = \frac{\bar{P}\|\boldsymbol{h}(k)\|^2}{\sigma_n^2} \tag{2.31}$$

$$C_{SIMO}^{FF} = \mathbb{E}\left[\log_2\left(1 + \frac{\bar{P}\|\boldsymbol{h}(k)\|^2}{\sigma_n^2}\right)\right] \tag{2.32}$$

The ergodic capacity of a fast fading MISO i.i.d. Rayleigh channel $\boldsymbol{y}(k) = \boldsymbol{h^T}(k)\boldsymbol{x}(k) + n(k)$ with CSIT distribution can be written as

$$C_{MISO}^{FF} = \mathbb{E}\left[\log_2\left(1 + \frac{1}{n_t}\frac{\bar{P}\|\boldsymbol{h}(k)\|^2}{\sigma_n^2}\right)\right] \tag{2.33}$$

In (2.33), it can be seen that the MISO channel exhibits a performance loss of $\frac{1}{n_t}$. The input-output relationship for the fast fading narrowband MIMO channel is given in (2.26). The capacity of a fast fading MIMO channel with CSIT distribution can be written as

$$C_{MIMO}^{FF} = \max_{\boldsymbol{R}_{xx}, tr(\boldsymbol{R}_{xx}) \leq \bar{P}} \mathbb{E}\left[\log_2 \det\left[I + \frac{1}{\sigma_n^2}\boldsymbol{H}\boldsymbol{R}_{xx}\boldsymbol{H}^H\right]\right] \tag{2.34}$$

Consider a Rayleigh i.i.d. MIMO channel and assume that the symbols transmitted from different antennas are uncorrelated and are assigned the same transmit power. That is, the optimum input correlation matrix is $\boldsymbol{R}_{xx} = \frac{\bar{P}}{n_t}\boldsymbol{I}$. Capacity for an i.i.d. Rayleigh fast fading MIMO channel is given below:

$$C_{MIMO}^{FF,Rayleigh} = \mathbb{E}\left[\log_2 det\left[\boldsymbol{I} + \frac{\bar{P}}{n_t\sigma_n^2}\boldsymbol{H}\boldsymbol{H}^H\right]\right] \tag{2.35}$$

For performance analysis of fast fading i.i.d. Rayleigh MIMO channels, it is assumed that $\mathbb{E}\left[\left|h_{ji}\right|^2\right] = 1$ for each SISO link h_{ji}. The instantaneous receive SNR at receive antenna j at time k is $\frac{\bar{P}\left|h_{ji}\right|^2}{\sigma_n^2}$ when only *i*th transmit antenna is active. Therefore, the average SNR per SISO link is $\rho = \frac{\bar{P}}{\sigma_n^2}$.

Performance of fast fading i.i.d. Rayleigh MIMO channels can be summarized as

- MIMO capacity grows linearly with respect to the minimum number of transmit or receive antennas.
- For $n_t = n_r = M$, MIMO capacity is approximately *M* times larger than SISO capacity.
- When the number of transmit antennas increases, the performance gain of MISO is negligible.
- At high SNR, capacity is approximated as

$$C_{MIMO}^{FF,Rayleigh} \sim M_{min}\log_2\rho \tag{2.36}$$

2.3 Large MIMO Systems

Large MIMO (also called large-scale antenna systems, large-scale MIMO, very large MIMO, ARGOS, full-dimension MIMO, or hyper-MIMO) [32-34] refers multiple-input multiple-output system which uses tens to hundreds antennas at the transmitter and the receiver. Some possible antenna configurations of large MIMO are given in Figure 2.11. Each antenna unit is small and active.

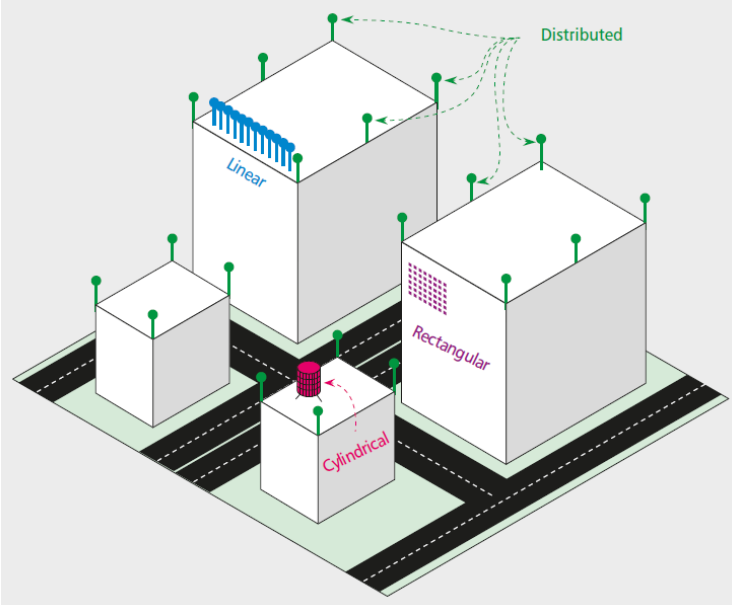

Figure 2.11: Some possible antenna configurations for a large MIMO [35].

Large MIMO systems can be classified as Type-I (point-to-point) large MIMO and Type-2 (multi-user) MIMO according to antenna configurations. Figure 2.12 shows an example of Type-I large MIMO systems and Figure 2.13 shows an example of Type-II large MIMO systems. In the point-to-point large MIMO system, a large number of antennas are used on the receiver, and also a large number of antennas are used at the transmitter. In the multi-user large MIMO system, only the receiver is equipped with a large number of antennas, while the total number of antennas at the transmit side is smaller [36].

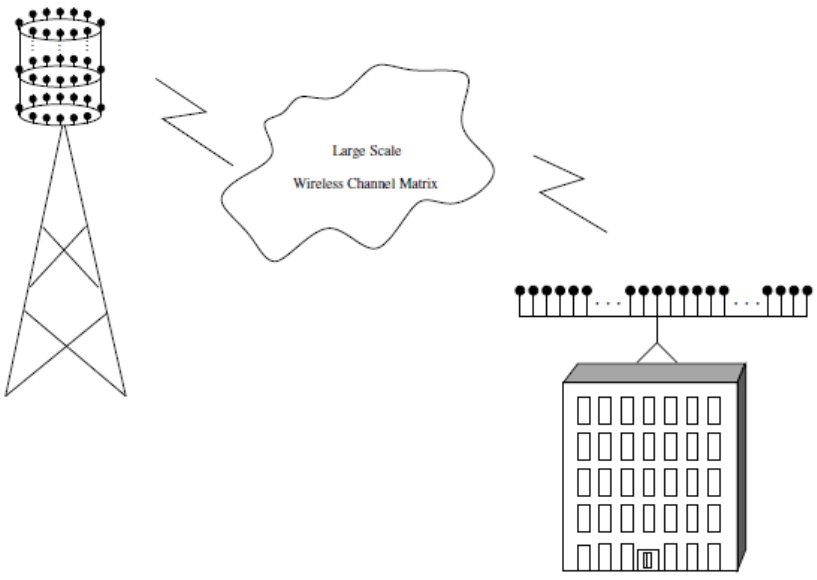

Figure 2.12: An example of Type-I (point-to-point) Large MIMO systems [36].

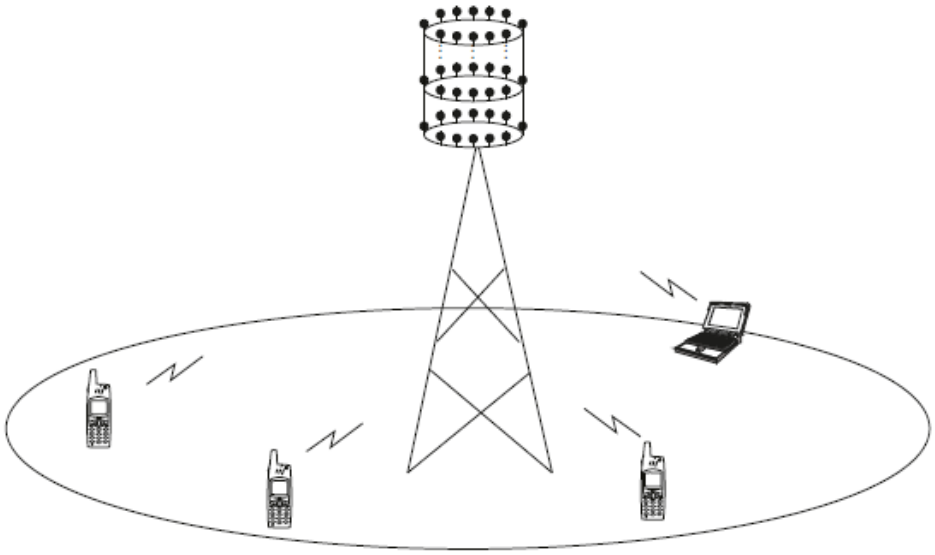

Figure 2.13: An example of Type-II (multiuser) Large MIMO systems [36].

Large MIMO systems are energy-efficient, secure, robust, and use the spectrum efficiently [35]. It has all benefits of conventional MIMO, but on a much greater scale. First, the capacity can be theoretically increased by using large number of antennas. The achievable rate in MIMO systems scales as

$$\min(\mathrm{n_t}, \mathrm{n_r}) \log_2(1 + SNR) \tag{2.37}$$

Current wireless standards have only some of the potential benefits of MIMO, because they use only a small number of antennas. They can achieve spectral efficiencies of only about 15 bps/Hz or less. However, using large number of antennas, multigigabit rate transmissions at high spectral efficiencies of the order of tens to hundreds of bps/Hz can be achieved. This indicates that large MIMO can achieve very high data rates without increasing the bandwidth, using large $\mathrm{n_t}$, $\mathrm{n_r}$.

Second, large MIMO systems can increase link reliability through spatial diversity [3]. The probability of link outage is given in (2.38) in a point-to-point MIMO system with n_t transmit and n_r receive antennas. It can be seen that in (2.38), the probability of link outage decreases, when number of antennas used at the transmitter and the receiver increases. In the other words, MIMO systems have a potential to achieve a diversity order of $\mathrm{n_t n_r}$. Namely, with large n_t and n_r, the MIMO link performance in terms of error rate can approach an exponential fall in error rate with increase in SNR.

$$P_{outage} \propto SNR^{-n_t n_r} \quad (2.38)$$

It can be seen that, diversity gain and channel capacity increase with using large number of antennas. Besides, large MIMO system give other advantages like using inexpensive component and robustness against jamming or unintended man-made interference. [35].

Large MIMO can be built with inexpensive, low-power component. With large MIMO, expensive ultra-linear high power amplifiers used in conventional systems are replaced by hundreds of low cost milli-watt amplifiers. Like large coaxial cable, some expensive items can be eliminated.

Large MIMO increases the robustness against both unintended man-made interference and intentional jamming. Spreading information over frequency is not feasible due to the scarcity of bandwidth. Therefore, using multiple antennas is better way of improving robustness of wireless communications.

Another advantage of large MIMO is energy efficiency. In order to get same performance with SISO system, each single-antenna user in a large MIMO system can scale down its transmit power proportional to the number of antennas at the BS [37].

When large number of antennas are used, the distribution of the singular values of the channel matrix approaches a deterministic function [38]. Additionally, very tall or very wide matrices tend to be very well conditioned. Linear detectors like ZF and MMSE detectors need to perform matrix inversions. While dimensions of matrix are getting large, some matrix operations such as inversions can be done fast, by using series expansion techniques.

Consequently, in order to reveal the potential of MIMO systems, large number of antennas should be used. This can help to improve performance and get full potential of MIMO systems. However, algorithms for large MIMO systems are required to keep the complexity low in order to make such a system practical. The spacing between the antennas in the communication terminal is another problem. Inadequate spacing between antennas causes spatial correlation. Therefore, channel capacity of large MIMO can decrease.

3. MIMO DECODING

Decoding in MIMO systems is one of the challenging operation on the receiver[3]. Because the receive antennas encounter spatial interference due to simultaneous transmission from multiple transmit antennas, decoding in MIMO communication system is more complicated than decoding in SISO or SIMO communication systems. Sophisticated signal processing algorithms are needed, in order to detect signals in the presence of this spatial interference.

MIMO communication systems can be characterized by the following linear vector channel model:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{3.1}$$

where $\boldsymbol{y} \in \mathbb{C}^{n_r}$, $\boldsymbol{H} \in \mathbb{C}^{n_r\, x\, n_t}$, $\boldsymbol{x} \in \mathbb{C}^{n_t}$, $\boldsymbol{n} \in \mathbb{C}^{n_r}$, and n_t and n_r are the number of transmit and receive antennas, respectively. In MIMO communication system, $\boldsymbol{x}$ is the vector transmitted from n_t transmit antennas, $\boldsymbol{H}$ is $n_r\, x\, n_t$ channel gain matrix whose entries are modeled as i.i.d. complex Gaussian with zero mean and unit variance , $\boldsymbol{y}$ is the received vector by the n_r receive antennas, and $\boldsymbol{n}$ is a complex AWGN vector where $E\{nn^H\} = \sigma^2 \boldsymbol{I}_{n_r}$. Here, $(.)^H$ denotes the Hermitian (complex conjugation and transposition) operation.

Any decoding algorithm obtains an estimate of the transmit vector $\boldsymbol{x}$ from received vector $\boldsymbol{y}$, given channel matrix $\boldsymbol{H}$ [39]. It is assumed that $\boldsymbol{H}$ is known perfectly at the receiver but is unknown at the transmitter. The elements of $\boldsymbol{x}$ generally come from a known modulation alphabet $\mathbb{A}$ with discrete valued symbols, i.e., $x \in \mathbb{A}^{n_t}$.

Several decoding algorithms work with the complex-valued system model in (3.1). On the other hand, some decoding algorithms may work with an equivalent real-valued system model below:

$$\boldsymbol{y_r} = \boldsymbol{H_r}\boldsymbol{x_r} + \boldsymbol{n_r} \tag{3.2}$$

where

$$\boldsymbol{H_r} \triangleq \begin{bmatrix} \Re(\boldsymbol{H}) & -\Im(\boldsymbol{H}) \\ \Im(\boldsymbol{H}) & \Re(\boldsymbol{H}) \end{bmatrix} \in \mathbb{R}^{2n_r x 2n_t}, \qquad \boldsymbol{y_r} \triangleq [\Re(\boldsymbol{y})^T \quad \Im(\boldsymbol{y})^T]^T \in \mathbb{R}^{2n_r},$$

$$\boldsymbol{x_r} \triangleq [\Re(\boldsymbol{x})^T \quad \Im(\boldsymbol{x})^T]^T \in \mathbb{R}^{2n_t}, \qquad \boldsymbol{n_r} \triangleq [\Re(\boldsymbol{n})^T \quad \Im(\boldsymbol{n})^T]^T \in \mathbb{R}^{2n_r},$$

Here, $\Re(.)$ and $\Im(.)$ are the real and imaginary parts of the complex argument, respectively. $(.)^T$ denotes transpose operation.

In this chapter, we will present several well-known MIMO decoding algorithms in a spatially multiplexed (V-BLAST) MIMO system with n_t transmit and n_r receive antennas with $n_t \leq n_r$. The n_t symbols are transmitted simultaneously from n_t transmit antennas.

3.1 Optimum Decoding

The optimal detector minimizes the average probability of error, $p(\hat{\boldsymbol{x}} \neq \boldsymbol{x})$, where $\hat{\boldsymbol{x}}$ is the estimate of the transmitted symbols. ML detector solves the quadratic optimization problem of minimizing the squared Euclidean distance between the actual received vector $\boldsymbol{y}$ and the hypothesized received signal $\boldsymbol{Hx}$. The ML solution is given by,

$$\hat{\boldsymbol{x}}_{ML} = \underset{x \in \mathbb{A}^{n_t}}{\operatorname{argmin}} \|\boldsymbol{y} - \boldsymbol{Hx}\|^2 \tag{3.3}$$

ML decoding requires exhaustive search with exponential complexity in the solution space. Thus, ML solution is only feasible for small n_t. Computing the exact solution is impractical, when n_t is large. Sphere decoding also gives ML solution. However, its complexity is exponential at low and medium SNRs. Because of this, Sphere decoding is also impractical for large n_t.

The non-faded SISO AWGN performance is a lower bound on the ML performance, for large n_t. This bound is easy to compute for large n_t at high SNRs. It is generally use in simulation to compare BER performance results. For various modulation, the BER performance of non-faded SISO AWGN is given in Figure 3.1.

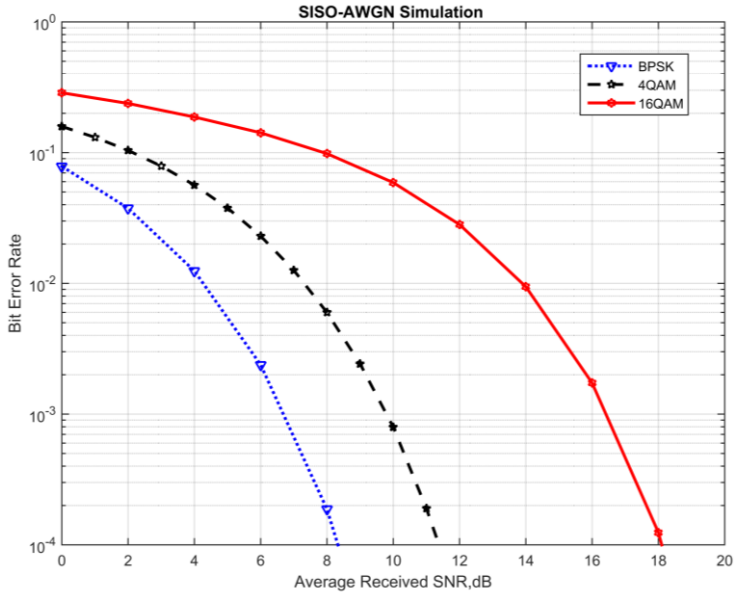

Figure 3.1: The BER performance of non-faded SISO AWGN with BPSK, 4QAM, and 16QAM.

3.2 Linear Decoding Techniques

Linear detectors have the advantage of low complexity. These decoding algorithms generate soft estimates of transmitted symbols. In order to obtain hard estimate, linear decoding methods use a slicer, which quantizes each entry to the nearest symbol in the modulation alphabet to obtain $\hat{x}$.

3.2.1 MF detector

Equation **(3.1)** can be written in the form

$$\boldsymbol{y} = \boldsymbol{Hx} + \boldsymbol{n} = \sum_{i=1}^{n_t} \boldsymbol{h}_i x_i + \boldsymbol{n}$$

$$= \boldsymbol{h}_k x_k + \sum_{i=1, i\neq k}^{n_t} \boldsymbol{h}_i x_i + \boldsymbol{n} \qquad (3.4)$$

where $\boldsymbol{h_i}$, $i = 1, 2, \ldots, n_t$, is the ith column of the channel matrix.

In (3.4), first term is the component due to the *k*th stream, and second term is due to all streams other than *k*th stream. This term is the interference component when *k*th stream is concerned.

In detecting the *k*th stream symbol x_k, matched filter deals with the interference from other streams as a noise and obtains a soft estimate of x_k as

$$\tilde{x}_k = \boldsymbol{h}_k^* \boldsymbol{y} \tag{3.5}$$

MF detector obtains hard estimate $\hat{x}_k$ by mapping $\tilde{x}_k$ to the nearest element in the alphabet in terms of Euclidean distance.

The vector form of the MF solution is below

$$\tilde{x}_k = \boldsymbol{H}^H \boldsymbol{y} \tag{3.6}$$

i.e., the transformation matrix $\boldsymbol{G}_{MF} = \boldsymbol{H}^H$. Computing the MF solution is very simple. The complexity order is just $n_t n_r$. Performance of MF detector degrades with increasing n_t, because of increased levels of uncanceled interference from other streams.

3.2.2 ZF detector

The ZF detector, also named as interference-nulling detector, cancels the interference from other streams. The linear transformation $\boldsymbol{G}_{ZF} = \boldsymbol{Q}$ is the pseudo-inverse of the $\boldsymbol{H}$ matrix.

$$\boldsymbol{Q} = (\boldsymbol{H}^H \boldsymbol{H})^{-1} \boldsymbol{H}^H \tag{3.7}$$

where $\boldsymbol{Q}$ is $n_t x n_r$ matrix and $\boldsymbol{QH} = \boldsymbol{I}_{\boldsymbol{n_t}}$.

A soft estimate of the symbol x_k can be obtained as

$$\begin{aligned} \tilde{x}_k &= \boldsymbol{q}_k \boldsymbol{y} = \boldsymbol{q}_k \boldsymbol{H} \boldsymbol{x} + \boldsymbol{q}_k \boldsymbol{n} \\ &= x_k + \boldsymbol{q}_k \boldsymbol{n} \end{aligned} \tag{3.8}$$

where $\boldsymbol{q}_k, k = 1,2,\dots,n_t$, denotes the kth row of $\boldsymbol{Q}$. $\boldsymbol{q}_k \boldsymbol{H}$ is a row vector of length n_t. Its entries are all zero except for a 1 in the kth coordinate. ZF detector obtains hard estimate $\hat{x}_k$ by mapping $\tilde{x}_k$ to the nearest symbol in the modulation alphabet in terms of Euclidean distance. The SNR at the kth stream zero-forced output is given by

$$SNR_k = \frac{|x_k|^2}{\|\boldsymbol{q}_k\|^2 \sigma^2} \tag{3.9}$$

The noise variance is increased by a factor of $\|\boldsymbol{q}_k\|^2$. Because of this, ZF detector is worse than the MF detector at low SNRs (large σ). Its performance also degrades large n_t at moderate to full loads.

The vector form of ZF solution is given below

$$\tilde{x}_{ZF} = \boldsymbol{Q}y \tag{3.10}$$

The computation complexity of ZF detector is cubic in n_t due to the computation of the matrix inverse. The per-symbol complexity is ${n_t}^2$, which one order is more than that of the MF detector.

3.2.3 MMSE detector

The transformation matrix of MMSE detector minimizes the mean square error between the transmit vector and the estimated vector. The transformation matrix $\boldsymbol{G}_{MMSE}$ is the solution to the following minimization problem:

$$\min_{\mathbf{G}} \mathbb{E}[\|\boldsymbol{x} - \boldsymbol{G}\boldsymbol{y}\|^2] \tag{3.11}$$

The solution to (3.11) is given by:

$$\boldsymbol{G}_{MMSE} = (\boldsymbol{H}^H\boldsymbol{H} + \sigma^2\boldsymbol{I}_{n_t)})^{-1}\boldsymbol{H}^H \tag{3.12}$$

and MMSE solution is given by:

$$\tilde{\boldsymbol{x}}_{\boldsymbol{MMSE}} = \boldsymbol{G}_{MMSE}\boldsymbol{y} \tag{3.13}$$

The BER performance of the MMSE detector is better than both the MF and the ZF detector over the entire range of SNRs. The MMSE detector combines the best performance attributes of MF and ZF detectors. It behaves like MF because of the prominence of the diagonal entries of $\boldsymbol{H}^H\boldsymbol{H}$ as $\sigma \rightarrow \infty$, at low SNRs. MMSE behaves like ZF since the second term inside the inverse operation in (3.12) becomes negligible, at high SNRs. MMSE solution needs knowledge of the noise variance σ^2, but MF and ZF solutions do not need it. Like the ZF detector, the per-symbol complexity of the MMSE detector is ${n_t}^2$ due to the matrix inversion involved in (3.12). Like the MF and ZF performances, the MMSE performance also degrades for increasing n_t.

3.3 Interference Cancellation

Interference cancellation techniques, including successive interference cancellation (SIC) and parallel interference cancellation (PIC), estimate interference and remove it in multiple stages. These techniques belong to the class of non-linear detectors.

Successive interference cancellation algorithm is given below:

1) The symbol transmitted in a data stream (preferably in the strongest data stream) is detected using any of the MF, ZF, and MMSE detector. SIC is named as MF-SIC, ZF-SIC, and MMSE-SIC according to the component detector used. The data streams can be ordered based on their received SNRs in (3.9), or, equivalently based on norm of $\boldsymbol{q}_k, k = 1,2, \dots, n_t$, if ZF-SIC technique is used.
2) The interference is estimated using the detected data symbol and knowledge of the channel matrix.
3) The estimated interference is cancelled from the received signal.
4) The symbol in the next strongest data stream is detected using the interference cancelled output signal. The interference of this stream is estimated and cancelled. Until the last stream (weakest) is detected, this procedure is continued.

Symbol decoding algorithm using ZF-SIC is given below:

1) Set $\boldsymbol{y}^{(1)} = \boldsymbol{y}, \boldsymbol{H}^{(1)} = \boldsymbol{H}$, where the superscripts denotes the stage index. $\boldsymbol{Q}^{(m)}$ is the pseudo-inverse of the channel matrix $\boldsymbol{H}^{(m)}$, and $\boldsymbol{q_l}^{(m)}$ is the *l*th row of $\boldsymbol{Q}^{(m)}$. Set stage index m = 1.
2) The symbol of the kth data stream is detected using the ZF detector.

$$\hat{x}_k = f\left(\boldsymbol{q_k}^{(m)} \boldsymbol{y}^{(m)}\right) \tag{3.14}$$

where f(.) is the slicing function. The algorithm is ended, when m reaches to n_t.

3) The interference vector due to the *k*th stream is estimated using $\hat{x}_k$.

$$\hat{\boldsymbol{a}}_k = \boldsymbol{h_k} \hat{x}_k \tag{3.15}$$

4) The estimated interference vector $\hat{\boldsymbol{a}}_k$ is subtracted from $\boldsymbol{y}^{(m)}$ in order to get the cancelled output $\boldsymbol{y}^{(m+1)}$.

$$\boldsymbol{y}^{(m+1)} = \boldsymbol{y}^{(m)} - \hat{\boldsymbol{a}}_k = \boldsymbol{y}^{(m)} - \boldsymbol{h}_k \hat{x}_k \tag{3.16}$$

5) kth column of $\boldsymbol{H}^{(m)}$ is set to zero to obtain $\boldsymbol{H}^{(m+1)}$

The complexity of ZF-SIC is $O(n_t^4)$. Its complexity is an order more than the complexity of the ZF detector. In each stage, ZF-SIC needs to do a matrix inversion. However, ZF-SIC has better performance than linear detectors. In large MIMO systems, it does not scale well.

The BER performance result of ZF-SIC, MF, and ZF detectors are given in Figure 3.2. 4QAM modulation and for $n_t = 8$ and $n_r = 12$ V-BLAST MIMO system is used in simulation. It can be seen that BER performance of MF detector is worse than BER performance of other detectors. ZF-SIC detectors performs better than linear detectors. However, its performance is far from performance of optimum detector. Hence, It does not scale well for large MIMO systems.

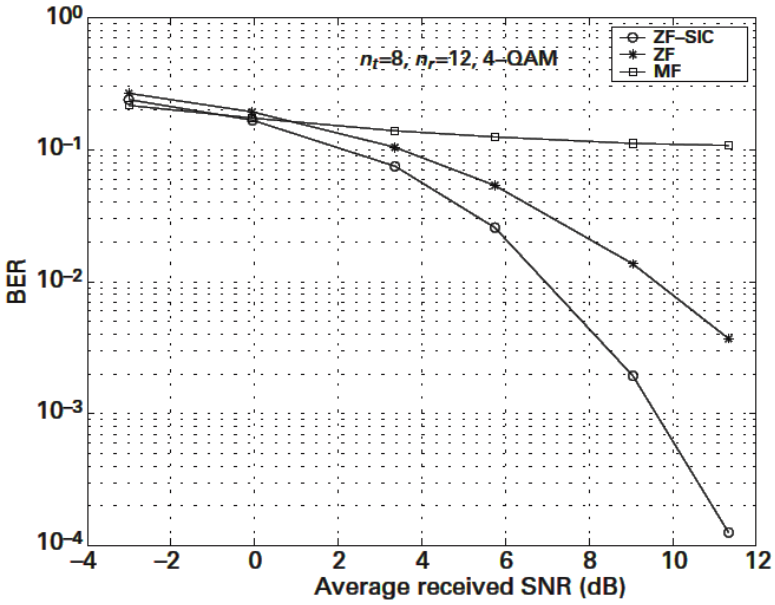

Figure 3.2: BER performance comparison between MF, ZF, and ZF-SIC detectors [18].

3.4 Sphere Decoding

Sphere decoding obtains the exact ML solution. It is faster than a brute-force exhaustive search. The ML optimization problem, referred to as an integer least-squares (ILS) problem, can be written using the real-valued system model as

$$\min_{\boldsymbol{x}_r \in \mathbb{B}^{2n_t}} \|\boldsymbol{y}_r - \boldsymbol{H}_r \boldsymbol{x}_r\|^2 \tag{3.17}$$

where $\mathbb{B}$ is the underlying pulse amplitude modulation (PAM) alphabet corresponding to the QAM alphabet $\mathbb{A}$. $\boldsymbol{x_r}$ spans a rectangular $2n_t$-dimensional lattice because the elements of $\boldsymbol{x_r}$ are point from a PAM constellation. Sphere decoding technique finds the closest lattice point to $\boldsymbol{y_r}$ in terms of the Euclidean distance, when the skewed lattice $\boldsymbol{H_r x_r}$ and the vector $\boldsymbol{y_r} \epsilon\, \mathbb{R}^{2n_r}$ are given. This technique search only lattice points that lie in a $2n_t$-dimensional sphere of radius d around $\boldsymbol{y_r}$, instead of an exhaustive search over the entire lattice. The complexity is reduced due to the reduced search space.

The computational complexity of sphere decoding algoritm is reduced at high SNRs due to the small search radius. On the other hand, SD is inefficient in terms of complexity at low to moderate SNRs due to the increased search radius [39]. Several variants of the SD have also been proposed to reduce complexity. Still, although the SD and several of its low complexity variants achieve ML performance, their complexity in low to moderate SNRs becomes prohibitive beyond 32 real dimensions [40, 41].Because of this, sphere decoding algorithm is inadequate for large MIMO systems.

In Figure 3.3, the BER performance of MMSE detector and SD detector is given. It can be seen that, the BER performance of SD algorithm performs better than MMSE. Though SD algorithm gives ML performance, it is not feasible to use in large MIMO system due to high computational complexity.

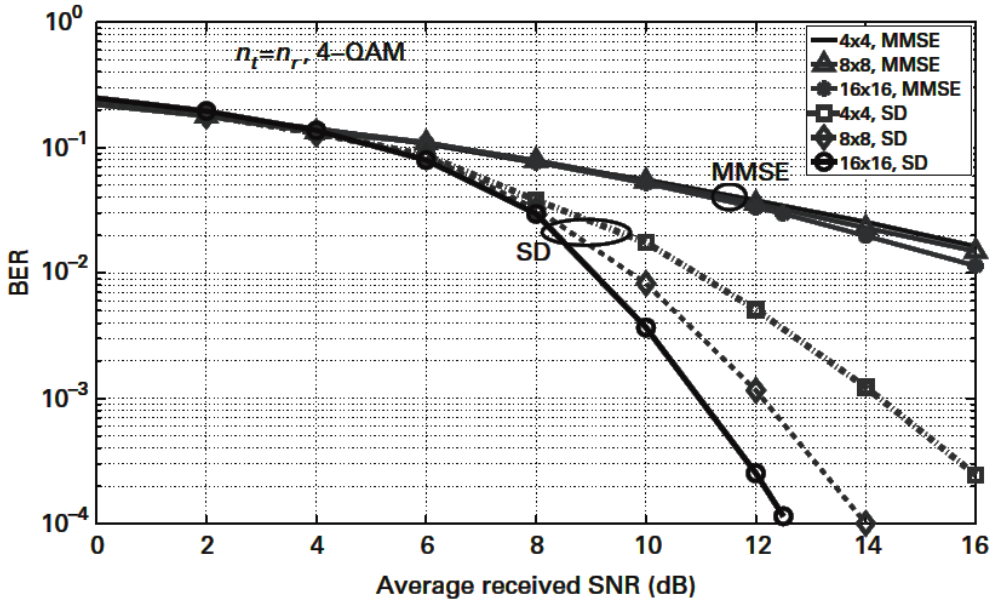

Figure 3.3: The BER performance of sphere decoding and MMSE decoding for V-BLAST MIMO [18].

3.5 Likelihood Ascent Search

In this section, conventional likelihood ascent search (LAS) algorithm [6], LAS with multiple symbol update [5], and multiple output selection-LAS algorithm [7] will be discussed.

3.5.1 Conventional LAS

Consider a V-BLAST system with $n_t \leq n_r$, where n_t symbols are transmitted from n_t transmit antenna simultaneously. The signal received at antenna k is given below

$$y_k = \sum_{j=1}^{n_t} h_{kj} x_j + n_k \tag{3.18}$$

where $x_j \in \{-1, +1\}$ is the BPSK symbol transmitted by the *j*th transmit antenna. h_{kj} is the path gain from transmit antenna j to receive antenna k ($\forall k \in \{1,2, \dots, n_r\}, \forall j \in \{1,2, \dots, n_t\}$). It is modeled as i.i.d. Complex Gaussian random variables with zero mean and $E\left[\left(h_{kj}^I\right)^2\right] = E\left[\left(h_{kj}^Q\right)^2\right] = 0.5$, where h_{kj}^I and h_{kj}^Q real and imaginary part of h_{kj}, respectively. n_k is the noise sample of kth receive antenna modeled as the complex white Gaussian noise (AWGN) with mean zero and variance $E[{n_k}^2] = N_0 = \frac{N_t E_s}{\gamma}$, where E_s is the average energy of the transmitted symbols and γ is the average received SNR per receive antenna[21]. The vector form of (3.18) is given as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{3.19}$$

where

$\boldsymbol{y} = \left[y_1 y_2 \dots y_{n_r}\right]^T, \boldsymbol{x} = \left[x_1 x_2 \dots x_{n_t}\right]^T, \boldsymbol{n} = \left[n_1 n_2 \dots n_{n_r}\right]^T$. $\boldsymbol{H}$ is $n_r x n_t$ channel gain matrix ant it is known perfectly at the receiver.

Until a fixed point is reached, LAS algorithm searches a sequence of bit vectors which maximizes likelihood function [2] in V-BLAST system. The likelihood function can be written as

$$\Lambda(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{y} + \boldsymbol{x}^{\boldsymbol{T}}(\boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{y})^{*} - \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{H}\boldsymbol{x} \tag{3.20}$$

The likelihood function given in (3.20) can be rewritten

$$\Lambda(\boldsymbol{x}) = \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{y}_{eff} - \boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{H}_{eff}\boldsymbol{x} \tag{3.21}$$

where

$$\boldsymbol{y}_{eff} = \boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{y} + (\boldsymbol{H}^{\boldsymbol{H}}\boldsymbol{y})^{*} \tag{3.22}$$

$$\boldsymbol{H}_{eff} = \boldsymbol{H}^{H}\boldsymbol{H} \tag{3.23}$$

In search step *n*, LAS algorithm search the bit vector $\boldsymbol{x}^{(n)}$. The initial vector $\boldsymbol{x}^{(0)}$ is generated by known detectors, i.e., MF, ZF, MMSE. When the initial vector is generated by MF detector, the resulting LAS detector is called as MF-LAS. ZF-LAS and MMSE-LAS are defined likewise. The LAS algorithm obtains $\boldsymbol{x}^{(n+1)}$ from $\boldsymbol{x}^{(n)}$ using an update rule until a fixed point is reached. The LAS algorithm updates bits if the change in likelihood $\Delta\Lambda\left(\boldsymbol{x}^{(n)}\right)$ is positive, where it is given in (3.24).

$$\Delta\Lambda\left(\boldsymbol{x}^{(n)}\right) \triangleq \Lambda\left(\boldsymbol{x}^{(n+1)}\right) - \Lambda\left(\boldsymbol{x}^{(n)}\right) \geq 0 \tag{3.24}$$

Gradient of the likelihood function can be given below.

$$\boldsymbol{g}^{(n)} = \frac{\partial\left(\Lambda\left(\boldsymbol{x}^{(n)}\right)\right)}{\partial(\boldsymbol{x}^{(n)})} = \boldsymbol{y}_{eff} - \boldsymbol{H}_{real}\boldsymbol{x}^{(n)} \tag{3.25}$$

Using (3.21), (3.24) can be written as

$$\begin{aligned}\Delta\Lambda\left(\boldsymbol{x}^{(n)}\right) &= \boldsymbol{x}^{(n+1)^{T}}\boldsymbol{y}_{eff} - \boldsymbol{x}^{(n+1)^{T}}\boldsymbol{H}_{eff}\boldsymbol{x}^{(n+1)} \\ &\quad - \left(\boldsymbol{x}^{(n)^{T}}\boldsymbol{y}_{eff} - \boldsymbol{x}^{(n)^{T}}\boldsymbol{H}_{eff}\boldsymbol{x}^{(n)}\right) \\ &= \left(\boldsymbol{x}^{(n+1)^{T}} - \boldsymbol{x}^{(n)^{T}}\right)\left(\boldsymbol{y}_{eff} - \boldsymbol{H}_{real}\boldsymbol{x}^{(n)}\right) \\ &\quad - \left(\boldsymbol{x}^{(n+1)^{T}} - \boldsymbol{x}^{(n)^{T}}\right)\left(-\boldsymbol{H}_{real}\boldsymbol{x}^{(n)}\right) \\ &\quad - \boldsymbol{x}^{(n+1)^{T}}\boldsymbol{H}_{eff}\boldsymbol{x}^{(n+1)} + \boldsymbol{x}^{(n)^{T}}\boldsymbol{H}_{eff}\boldsymbol{x}^{(n)}\end{aligned} \tag{3.26}$$

where

$$\boldsymbol{H}_{real} = \boldsymbol{H}_{eff} + \left(\boldsymbol{H}_{eff}\right)^{*} = 2\Re\left(\boldsymbol{H}_{eff}\right) \tag{3.27}$$

It can be seen that,

$$\boldsymbol{x}^{(n)^T}\boldsymbol{H}_{real}\boldsymbol{x}^{(n)} = 2\boldsymbol{x}^{(n)^T}\boldsymbol{H}_{eff}\boldsymbol{x}^{(n)} \tag{3.28}$$

where

$$\Delta\boldsymbol{x}^{(n)} = \boldsymbol{x}^{(n+1)} - \boldsymbol{x}^{(n)} \tag{3.29}$$

If we add and subtract the term $\frac{1}{2}\boldsymbol{x}^{(n)^T}\boldsymbol{H}_{real}\boldsymbol{x}^{(n+1)}$ to the right hand side of (3.26) and,

$$\boldsymbol{x}^{(n)^T}\boldsymbol{H}_{real}\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n+1)^T}\boldsymbol{H}_{real}\boldsymbol{x}^{(n)} \tag{3.30}$$

(3.26) can be given as

$$\begin{aligned}\Delta\Lambda\left(\boldsymbol{x}^{(n)}\right) &= \Delta\boldsymbol{x}^{(n)^T}\left(\boldsymbol{y}_{eff} - \boldsymbol{H}_{real}\boldsymbol{x}^{(n)}\right) - \frac{1}{2}\Delta\boldsymbol{x}^{(n)^T}\boldsymbol{H}_{real}\Delta\boldsymbol{x}^{(n)} \\ &= \Delta\boldsymbol{x}^{(n)^T}\left(\boldsymbol{g}^{(n)} + \frac{1}{2}\boldsymbol{z}^{(n)}\right)\end{aligned} \tag{3.31}$$

where

$$\boldsymbol{z}^{(n)} = -\boldsymbol{H}_{real}\Delta\boldsymbol{x}^{(n)} \tag{3.32}$$

$L^{(n)}$ is the set of bits which is checked for possible change in a step n, i.e., $L^{(n)} \subseteq \{1,2,\ldots,n_t\}$.

Likelihood change given by (3.31) can be written as

$$\Delta\Lambda\left(\boldsymbol{x}^{(n)}\right) = \sum_{j\in L^{(n)}} \Delta x_j^{(n)}\left[g_j^{(n)} + \frac{1}{2}z_j^{(n)}\right] \tag{3.33}$$

where $x_j^{(n)}, g_j^{(n)}$, and $z_j^{(n)}$ are the jth element of the vectors $\boldsymbol{x}^{(n)}, \boldsymbol{g}^{(n)}$,and $\boldsymbol{z}^{(n)}$, respectively.

The update rule is given below [6]:

$$x_j^{(n+1)} = \begin{cases} +1 \ , & \text{if } j \in L^{(n)}, x_j^{(n)} = -1, \text{and } g_j^{(n)} > \ t_j^{(n)} \\ -1 \ , & \text{if } j \in L^{(n)}, x_j^{(n)} = +1, \text{and } g_j^{(n)} < -\ t_j^{(n)} \\ x_j^{(n)}, & \text{otherwise} \end{cases} \tag{3.34}$$

where $t_j^{(n)}$ is a threshold for the *j*th bit in the *n*th step which is given below.

$$t_j^{(n)} = \sum_{i \epsilon L^{(n)}} \left| (\boldsymbol{H_{real}})_{\boldsymbol{j,i}} \right|, \forall j \in L^{(n)} \tag{3.35}$$

where $(\boldsymbol{H_{real}})_{\boldsymbol{j,i}}$ is the element in the jth row and ith column of the matrix $\boldsymbol{H_{real}}$.

Different choices can be made to specify the sequence of $L^{(n)}$. One of these is to checking one bit in each step. It is referred as a sequential LAS (SLAS) algorithm with constant threshold, $t_j = \left| (\boldsymbol{H_{real}})_{\boldsymbol{j,j}} \right|$. The indices of bits which will be checked can be chosen circularly or randomly. $L_f^{(n)} \subseteq L^{(n)}$ denotes the set of indices of the bits changed by the update rule in (3.34) at step n. The updated bit vector can be written as

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} - 2 \sum_{i \epsilon \, L_f^{(n)}} x_i^{(n)} \boldsymbol{e}_i \tag{3.36}$$

where $\boldsymbol{e}_i$ is the *i*th coordinate vector. The *i*th element of $\boldsymbol{e}_i$ is one and other elements are zero. The gradient vector for the next step is given below,

$$\boldsymbol{g}^{(n+1)} = \boldsymbol{y}_{eff} - \boldsymbol{H}_{real} \boldsymbol{x}^{(n+1)} = \boldsymbol{g}^{(n)} + 2 \sum_{i \epsilon \, L_f^{(n)}} x_i^{(n)} (\boldsymbol{H_{real}})_i \tag{3.37}$$

where $(\boldsymbol{H_{real}})_i$ denotes the *i*th column of the matrix $\boldsymbol{H_{real}}$.The LAS algorithm keeps updating the bits in each step until a fixed point is reached.

The algorithm of conventional LAS is given below [6].

1) Input: $\boldsymbol{H}$ and $\boldsymbol{y}$. Obtain $\boldsymbol{y}_{eff} = \boldsymbol{H^H y} + (\boldsymbol{H^H y})^*, \boldsymbol{H}_{eff} = \boldsymbol{H}^H \boldsymbol{H}$, and $\boldsymbol{H}_{real} = \boldsymbol{H}_{eff} + \left(\boldsymbol{H}_{eff}\right)^* = 2\Re\left(\boldsymbol{H}_{eff}\right)$
2) Obtain initial vector $\boldsymbol{x}^{(0)}$ using known decoding algorithm, i.e., ZF, MF, or MMSE.

3) Compute $\boldsymbol{g}^{(0)}$ as

$$\boldsymbol{g}^{(0)} = \boldsymbol{y}_{eff} - \boldsymbol{H}_{real}\boldsymbol{x}^{(0)} \tag{3.38}$$

4) Compute threshold using equation (3.35).
5) Update symbols sequentially (for sequential LAS) using search algorithm in (3.36) and (3.37) until a fixed point is reached.

The computational complexity of LAS algorithm consists four component

i) The generation of initial vector using known decoding algorithm requires an average per-bit complexity $O(n_r), O(n_t n_r), and\ O(n_t n_r)$ for MF, ZF, and MMSE, respectively.

ii) Initial computation of $\boldsymbol{g}^{(0)}$ in (3.38) requires the computation of $\boldsymbol{H}^H\boldsymbol{H}$.This computation requires a per-bit complexity of order $O(n_t n_r)$.

iii) Update of $\boldsymbol{g}^{(n)}$ given in equation (3.37) in each step. This computation using sequential LAS requires a complexity of $O(n_t)$. It is a constant per-bit complexity.

iv) The average number of steps required to reach a fixed point is obtained through simulations. The average number of steps requires is linear in n_t, i.e., constant per-bit complexity where the constant depends on SNR, n_t, n_r, and the initial vector. The complexity of the LAS algorithm in terms of average number of steps per transmit antenna till fixed point is reached is given in Figure 3.4.

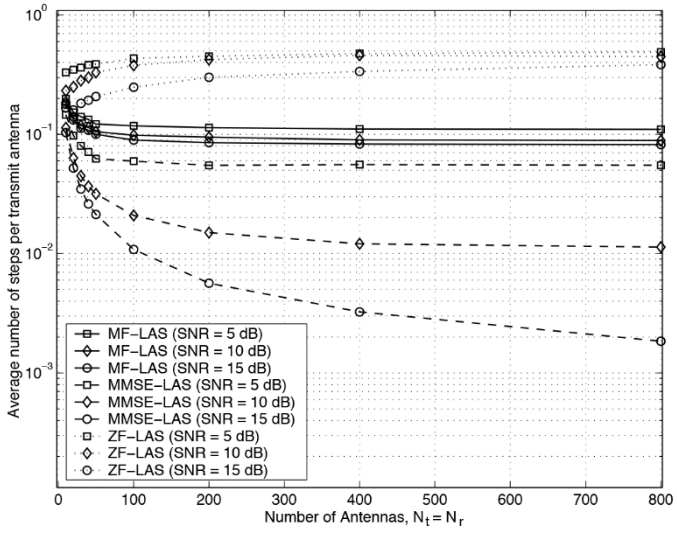

Figure 3.4: The computational complexity of the LAS algorithm in terms of average number of steps per transmit antenna till fixed points is reached in V-BLAST [6].

The overall average per-bit complexity of the LAS detector is $O(n_t n_r)$.

The BER performance of LAS algorithm is given in Figure 3.5. It can be seen that orders of diversity achieved by LAS algorithm increases for increasing number of antennas for ZF-LAS. The performance of LAS algorithm converges to SISO AWGN performance for 200 and 400 antennas. Chosen initial vector does not affect the performance of the LAS algorithm for large number of antennas.

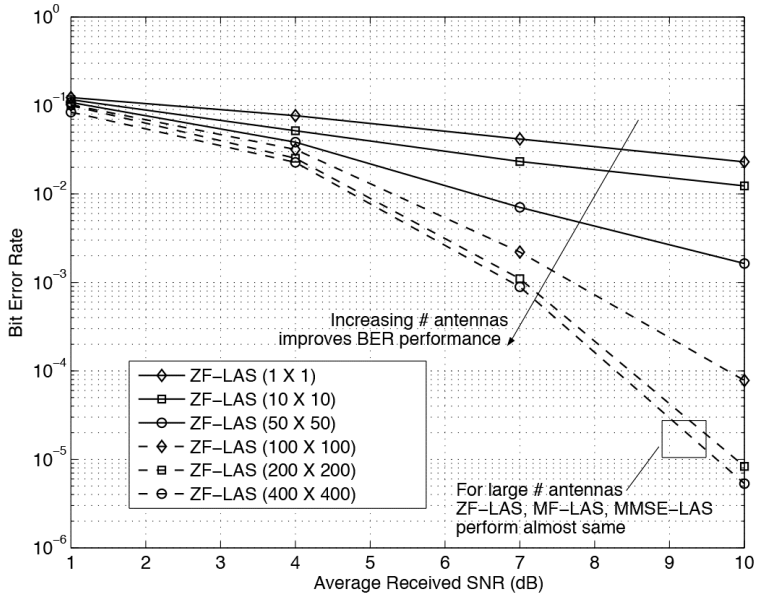

Figure 3.5: The BER performance of ZF-LAS for V-BLAST as a function of average received SNR, BPSK [6].

In Figure 3.6, the performance of the known detector (MF, ZF, ZF-SIC), MF-LAS, and ZF-LAS are compared. The BER at $n_t = n_r = 1$ is the SISO flat Rayleigh fading BER for BPSK. The performances of MF and ZF degrades with increasing number of transmit and receive antennas. The performance of ZF-SIC increases for antennas up to $n_t = n_r = 15$. Flooring effect occurs for $n_t \geq 15$ because the interference is large. It is beyond the cancellation ability of the ZF-SIC. The BER performance of the LAS detector improves when number of antennas are increased. MF-LAS performs better than MF detector. The performance of ZF-SIC is better than ZF-LAS for antennas in the range 4 to 24 .For $n_t \geq 50$, performance of MF-LAS is better than ZF-LAS because MF detector performs better than ZF.

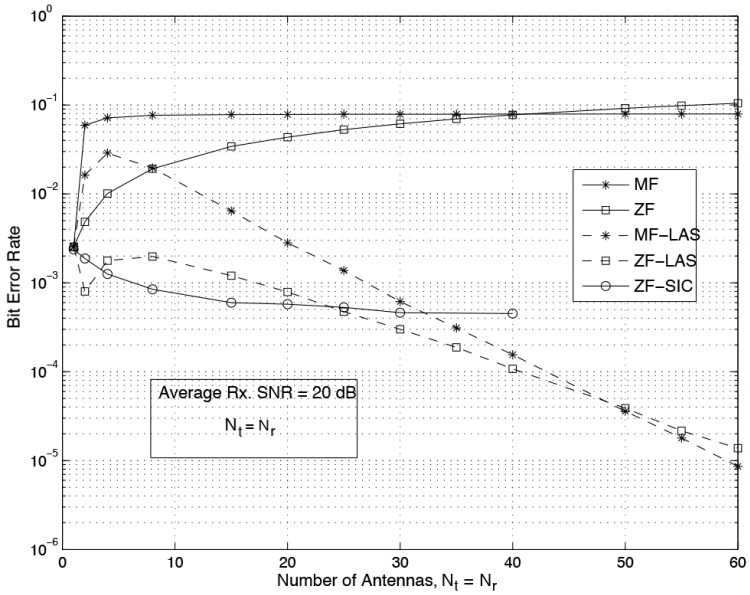

Figure 3.6: The BER performance of MF/ZF-LAS for V-BLAST as a function of number of transmit/receive antennas at an average received SNR=20 dB, BPSK [6].

3.5.2 LAS with multiple symbol update

The motivation of multiple symbol update can be explained as follows [5]. Let define a set of data vectors $\mathcal{L}_k \subseteq \mathbb{S}$ where any $\boldsymbol{x} \in \mathcal{L}_k$ and $\mathbb{S}$ denotes the signal space. If a K-symbol update is performed on $\boldsymbol{x}$ resulting in a vector $\boldsymbol{x}'$, then $\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}'\| \geq \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|$. It can be seen that $\boldsymbol{x_{ML}} \in \mathcal{L}_K, \forall K = 1,2,\dots,2n_t$, because any number of symbol updates on $\boldsymbol{x_{ML}}$ will not decrease the cost function. Let define another set $\mathcal{M}_K \bigcap_{j=1}^{K} \mathcal{L}_j$. It can be seen that $\boldsymbol{x_{ML}} \in \mathcal{M}_K, \forall K = 1,2,\dots,2n_t$, and $\mathcal{M}_{2n_t} = \{\boldsymbol{x_{ML}}\}$. If the updates are done optimally, then the output of the K-LAS algorithm converges to a vector in $\mathcal{M}_K$. Also, $|\mathcal{M}_{K+1}| \leq |\mathcal{M}_K|, K = 1,2,\dots,2n_t - 1$. It can be seen that $\boldsymbol{x}$ and $\boldsymbol{x_{ML}}$ will differ in $K+1$ or more locations for any $\boldsymbol{x_{ML}} \in \mathcal{M}_K, K = 1,2,\dots,2n_t \; and \; \boldsymbol{x} \neq \boldsymbol{x_{ML}}$. K-symbol updates with large K could approach ML performance with increasing complexity for increasing K. The overall complexity of K-symbol update simultaneously is $O(n_t^K)$. The system model is given as follows.

The received signal vector can be given as

$$\boldsymbol{y_c} = \boldsymbol{H_c}\boldsymbol{x_c} + \boldsymbol{n_c} \tag{3.39}$$

where $\boldsymbol{y_c} \in \mathbb{C}^{n_r}$, $\boldsymbol{H_c} \in \mathbb{C}^{n_r \, x \, n_t}$, $\boldsymbol{x_c} \in \mathbb{C}^{n_t}$, $\boldsymbol{n_c} \in \mathbb{C}^{n_r}$, and n_t and n_r are the number of transmit and receive antennas, respectively. Each element of $\boldsymbol{x_c}$ is an M-PAM or M-QAM symbol. M-PAM symbols take discrete values from $\{A_m, m = 1,2,\dots,M\}$,

where $A_m = (2m - 1 - M)$, and M-QAM is two PAMs in quadrature [3]. The channel gain matrix $\boldsymbol{H_c}$ is modeled as i.i.d. $\mathcal{CN}(0,1)$. The noise vector $\boldsymbol{n_c}$ is modeled as $\mathcal{CN}(0, \sigma^2)$.

$\boldsymbol{y_c}, \boldsymbol{H_c}, \boldsymbol{x_c}$, and $\boldsymbol{n_c}$ can be decomposed into real and imaginary parts as

$$\boldsymbol{y_c} = \boldsymbol{y_I} + j\boldsymbol{y_Q}, \quad \boldsymbol{x_c} = \boldsymbol{x_I} + j\boldsymbol{x_Q}, \quad \boldsymbol{n_c} = \boldsymbol{n_I} + j\boldsymbol{n_Q}, \quad \boldsymbol{H_c} = \boldsymbol{H_I} + j\boldsymbol{H_Q} \tag{3.40}$$

The complex system in (3.39) can be converted into a real-valued system model as in [5],

$$\boldsymbol{y_r} = \boldsymbol{H_r}\boldsymbol{x_r} + \boldsymbol{n_r} \tag{3.41}$$

where

$$\boldsymbol{H_r} \triangleq \begin{bmatrix} \Re(\boldsymbol{H_c}) & -\Im(\boldsymbol{H_c}) \\ \Im(\boldsymbol{H_c}) & \Re(\boldsymbol{H_c}) \end{bmatrix} \in \mathbb{R}^{2n_r x 2n_t}, \qquad \boldsymbol{y_r} \triangleq [\Re(\boldsymbol{y_c})^T \quad \Im(\boldsymbol{y_c})^T]^T \in \mathbb{R}^{2n_r},$$

$$\boldsymbol{x_r} \triangleq [\Re(\boldsymbol{x_c})^T \quad \Im(\boldsymbol{x_c})^T]^T \in \mathbb{R}^{2n_t}, \qquad \boldsymbol{n_r} \triangleq [\Re(\boldsymbol{n_c})^T \quad \Im(\boldsymbol{n_c})^T]^T \in \mathbb{R}^{2n_r},$$

If the subscript r is dropped for notational simplicity, the real-valued system model can be written as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{3.42}$$

The real part of the complex symbols will be mapped to $[x_1, \dots, x_{N_t}]$ and imaginary part will be mapped into $[x_{N_t+1}, \dots, x_{2N_t}]$. $[x_{N_t+1}, \dots, x_{2N_t}]$ will be zeroes for M-PAM because imaginary part of these symbols are zeroes.

$2n_t$-dimensional signal space $\mathbb{S}$ is Cartesian product of $\mathbb{A}_1$ to $\mathbb{A}_{2N_t}$, where $\mathbb{A}_i$ is the signal set of M-PAM or M-QAM. For example, $\mathbb{A}_i = \{-1,1\}$ for 4-QAM modulation. The ML solution vector can be given as,

$$\boldsymbol{x}_{ML} = \arg\min_{\boldsymbol{x}\in\mathbb{S}} \| \boldsymbol{y} - \boldsymbol{H}\boldsymbol{x} \|^2 = \arg\min_{\boldsymbol{x}\in\mathbb{S}} \boldsymbol{x}^T\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^T\boldsymbol{H}\boldsymbol{x} \tag{3.43}$$

The K-LAS algorithm search sequentially and the likelihood of the solution increases monotonically with every search stage. Each search stage consists of several iteration. The algorithm first updates one symbol per iteration. The likelihood monotonically increases from one search to next search until a local minima is reached. When a local minima is reached, algorithm tries a 2-symbol and/or 3-symbol update in order to

further increase the likelihood. Algorithm uses this new point for the next search stage if this likelihood increase happen.

The K-LAS algorithm starts with an initial solution $\boldsymbol{x}^{(0)}$, given by $\boldsymbol{x}^{(0)} = \mathcal{Q}(\boldsymbol{B}\boldsymbol{y})$, where $\boldsymbol{B}$ is the initial solution is filter and $\mathcal{Q}(.)$ is the quantizer which rounds the element to nearest-neighbor. Initial solution filter can be chosen as MF or ZF or MMSE filter. The index n in $\boldsymbol{x}^{(n)}$ is the iteration number in a given search stage. After the *k*th iteration given a search stage, the ML cost function can be given as

$$C^{(k)} = \boldsymbol{x}^{(k)^T}\boldsymbol{H}^T\boldsymbol{H}\boldsymbol{x}^{(k)} - \mathbf{2}\boldsymbol{y}^T\boldsymbol{H}\boldsymbol{x}^{(k)} \tag{3.44}$$

Each search stage starts with 1-symbol update and continues with 2 and/or 3-symbol update.

3.5.2.1 One-symbol update

Assuming the *p*th symbol in the $(k+1)$th iteration is updated, the update rule can be written as

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \lambda_p^{(k)}\boldsymbol{e}_p \tag{3.45}$$

where $\boldsymbol{e}_p$ is the unit vector. Its pth entry is 1 and all other entries are zero. p can take values from $1,2,\dots,n_t$ for M-PAM and $1,2,\dots,2n_t$ for M-QAM. $\boldsymbol{x}^{(k)}$ should belong to the signal space $\mathbb{S}$ for any iteration *k*. Hence, $\lambda_p^{(k)}$ can take only certain integer values. For example, $\lambda_p^{(k)}$ can take values from $\{-6,-4,-2,0,2,4,6\}$ in the case of 4-PAM and 16-QAM. Both modulation technique have the same signal set $\mathbb{A}_p = \{-3,-1,1,3\}$

The cost difference can be written as

$$\Delta C_p^{k+1} \triangleq C^{k+1} - C^k = \lambda_p^{(k)^2}(\boldsymbol{G})_{p,p} - 2\lambda_p^{(k)}z_p^{(k)} \tag{3.46}$$

where $\boldsymbol{z}^{(k)} = \boldsymbol{H}^T\left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}^{(k)}\right)$, $z_p^{(k)}$ is the *p*th entry of the $\boldsymbol{z}^{(k)}$ vector, and $(\boldsymbol{G})_{p,p}$ is the (p,p)th entry of the $\boldsymbol{G}$ is Grammian matrix which it is given below.

$$\boldsymbol{G} = \boldsymbol{H}^T\boldsymbol{H} \tag{3.47}$$

Equation (3.46) can be written as

$$\Delta C_p^{k+1} = {l_p^{(k)}}^2 a_p - 2l_p^{(k)} \left| z_p^{(k)} \right| sgn\left(\lambda_p^{(k)}\right) sgn(z_p^{(k)}) \tag{3.48}$$

where $sgn(.)$ is the signum function. a_p and $l_p^{(k)}$ are non-negative quantities, and they are given below.

$$a_p = (\boldsymbol{G})_{p,p}, l_p^{(k)} = \left| \lambda_p^{(k)} \right| \tag{3.49}$$

The cost difference should be negative in order to reduce the ML cost function from the kth iteration to the $(k+1)$th iteration. it can be seen that the sign of $\lambda_p^{(k)}$ must satisfy

$$sgn\left(\lambda_p^{(k)}\right) = sgn(z_p^{(k)}) \tag{3.50}$$

Hence, the ML cost difference in (3.20) can be written as

$$\mathcal{F}\left(l_p^{(k)}\right) \triangleq \Delta C_p^{k+1} = {l_p^{(k)}}^2 a_p - 2l_p^{(k)} \left| z_p^{(k)} \right| \tag{3.51}$$

For $\mathcal{F}\left(l_p^{(k)}\right)$ to be non-positive, the necessary and sufficient condition from (3.51) is given below

$$l_p^{(k)} < \frac{2\left| z_p^{(k)} \right|}{a_p} \tag{3.52}$$

The value of $l_p^{(k)}$ satisfying (3.52) gives the largest descent in the ML cost function from the kth to the $(k+1)$th iteration when symbol p is updated. $l_p^{(k)}$ takes only certain integer values. Hence, the brute-force way to get the optimum $l_p^{(k)}$ is to evaluate $\mathcal{F}\left(l_p^{(k)}\right)$ at all possible values of $l_p^{(k)}$. This method becomes computationally expensive when the constellation size M increases. However, for one-symbol update, a closed-form expression for the optimum $l_p^{(k)}$ which minimizes $\mathcal{F}\left(l_p^{(k)}\right)$ can be given by

$$l_{p,opt}^{(k)} = 2 \left\lfloor \frac{\left| z_p^{(k)} \right|}{2a_p} \right\rfloor \tag{3.53}$$

where $\lfloor . \rceil$ denotes the rounding operation, i.e., $\lfloor x \rceil$ is the integer closest to x for real numbers. When the pth symbol in $\boldsymbol{x}^{(k)}$, i.e., $x_p^{(k)}$, is updated, the new value of the symbol can be given as

$$\tilde{x}_p^{(k+1)} = x_p^{(k)} + l_{p,opt}^{(k)} sgn(z_p^{(k)}) \tag{3.54}$$

However, $\tilde{x}_p^{(k+1)}$ can take values only in the set $\mathbb{A}_p$. Hence, $\tilde{x}_p^{(k+1)}$ should be checked whether it is greater than $(M-1)$ or less than $-(M-1)$. The adjustment equations are given below

$$\text{If } \tilde{x}_p^{(k+1)} > (M-1), \qquad l_p^{(k)} = l_p^{(k)} - sgn(z_p^{(k)})\left(\tilde{x}_p^{(k+1)} - (\mathrm{M}-1)\right) \tag{3.55}$$

$$\text{if } \tilde{x}_p^{(k+1)} < -(M-1), \qquad l_p^{(k)} = l_p^{(k)} - sgn(z_p^{(k)})\left(\tilde{x}_p^{(k+1)} + (\mathrm{M}-1)\right) \tag{3.56}$$

After these adjustments, $\tilde{l}_{p,opt}^{(k)}$ is obtained. It can be shown that $\mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right)$ is non-positive if $\mathcal{F}\left(l_{p,opt}^{(k)}\right)$ is non-positive. Then, compute $\mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right)$ for $\forall\, p = 1,2,\ldots,2n_t$.

In (3.57), the algorithm chooses symbols s which minimize $\mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right)$.

$$s = \underset{p}{arg\,min}\, \mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right) \tag{3.57}$$

If $\mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right) < 0$, the update for $(k+1)$th iteration is given as

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \tilde{l}_{s,opt}^{(k)} sgn(z_s^{(k)})\boldsymbol{e_s} \tag{3.58}$$

$$\boldsymbol{z}^{(k+1)} = \boldsymbol{z}^{(k)} - \tilde{l}_{s,opt}^{(k)} sgn(z_s^{(k)})\boldsymbol{g_s} \tag{3.59}$$

where $\boldsymbol{g_s}$ is the sth column of $\boldsymbol{G}$. If $\mathcal{F}\left(\tilde{l}_{p,opt}^{(k)}\right) \geq 0$, then the one-symbol update search terminates. The data vector generated by one-symbol update is referred as "one-symbol update local minima". After the one-symbol local minima is reached , K-LAS algorithm updates multiple symbols simultaneously in order to further decrease the cost function.

For one symbol update, the BER performance of LAS algorithm with BPSK modulation is given in Figure 3.5. The BER performance of LAS algorithm with 4-QAM and 16-QAM are given in Figure 3.7 and Figure 3.8, respectively [4].

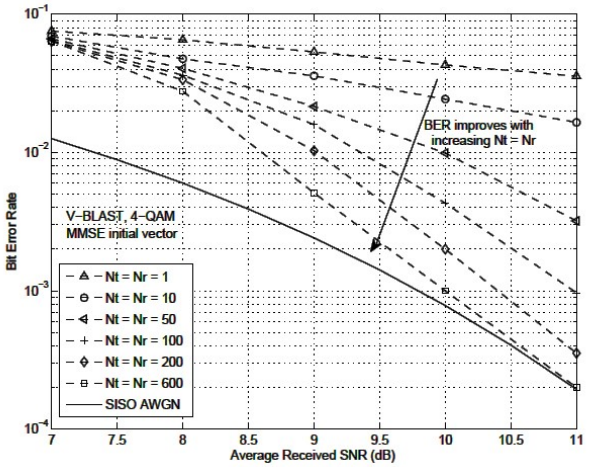

Figure 3.7: The BER performance of the LAS detector for V-BLAST with MMSE initial vector, 4-QAM [4].

In Figure 3.7, the BER performance of MMSE-LAS detector filter is given. LAS algorithm achieve SISO AWGN performance, when large number of antennas are used. Compared to BPSK modulation, more SNR is required in order to achieve same target BER for 4-QAM. Figure 3.8 shows the performance of LAS algorithm with 16-QAM modulation. The performance of LAS algorithm converges to the performance of SISO AWGN, when large number of antennas are used. It can be seen that, for 10^{-4}given target BER, required SNR decreases with increasing number of antennas.

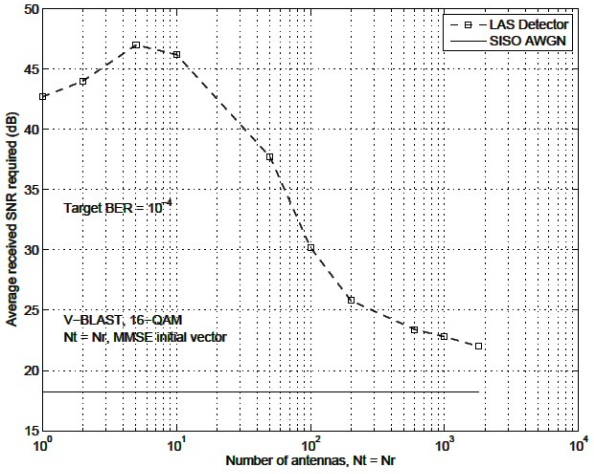

Figure 3.8: Average received SNR required to achieve a target BER of 10^{-4} in V-BLAST for 16-QAM LAS detector with MMSE initial vector [4].

3.5.2.2 Multiple symbol update

K-symbol updates can be done in $\binom{2n_t}{K}$ ways. The K-symbol update rule for $(k+1)$th iteration can be written as

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \sum\nolimits_{j=1}^{K} \lambda_{i_j}^{(k)} \boldsymbol{e}_{i_j} \tag{3.60}$$

where the symbol indices $i_j, j = 1,2,\dots,K$ can take values from $1,2,\dots,n_t$ for M-PAM and $1,2,\dots,2n_t$ for M-QAM. Define the set of indices as $\mathcal{U} \triangleq \{i_1, i_2, \dots, i_K\}$. $\boldsymbol{x}^{(k)}$ belongs to signal space $\mathbb{S}$ for any iteration k. Therefore, $\lambda_{i_j}^{(k)}$ can take only certain integer values. $\lambda_{i_j}^{(k)} \in \boldsymbol{\mathcal{A}}_{i_j}^{(k)}$, where $\boldsymbol{\mathcal{A}}_{i_j}^{(k)} \triangleq \left\{x | (x + x_{i_j}^{(k)}) \in \boldsymbol{\mathcal{A}}_{i_j}, x \neq 0\right\}$. It can be written as $\left(\lambda_{i_1}^{(k)}, \lambda_{i_2}^{(k)}, \dots, \lambda_{i_K}^{(k)}\right) \in \boldsymbol{\mathcal{A}}_{\mathcal{U}}^{(k)}$. $\boldsymbol{\mathcal{A}}_{\mathcal{U}}^{(k)}$ denotes the Cartesian product of $\boldsymbol{\mathcal{A}}_{i_1}^{(k)}, \boldsymbol{\mathcal{A}}_{i_2}^{(k)} \dots, \boldsymbol{\mathcal{A}}_{i_k}^{(k)}$. For 16-QAM, $\boldsymbol{\mathcal{A}}_{i_j} = \{-3,-1,1,3\}$, and if $x_{i_j}^{(k)}$ is -1, then $\boldsymbol{\mathcal{A}}_{i_j}^{(k)} = \{-2,2,4\}$. Using equation (3.57), the cost difference function can be written as

$$\begin{aligned}
&\Delta C_{\mathcal{U}}^{k+1}\left(\lambda_{i_1}^{(k)}, \lambda_{i_2}^{(k)}, \dots, \lambda_{i_K}^{(k)}\right) \triangleq C^{(k+1)} - C^{(k)} \\
&= \sum_{j=1}^{K} {\lambda_{i_j}^{(k)}}^2 (\boldsymbol{G})_{i_j,i_j} + 2\sum_{q=1}^{K}\sum_{p=q+1}^{K} \lambda_{i_p}^{(k)} \lambda_{i_q}^{(k)} (\boldsymbol{G})_{i_p,i_q} \\
&- 2\sum_{j=1}^{K} \lambda_{i_j}^{(k)} z_{i_j}^{(k)}
\end{aligned} \tag{3.61}$$

For a given $\mathcal{U}$, in order to decrease the ML cost function, the value of $\left(\lambda_{i_1}^{(k)}, \lambda_{i_2}^{(k)}, \dots, \lambda_{i_K}^{(k)}\right)$ should be chosen such that the cost difference in (3.61) is negative. If multiple values exist, then the value is chosen which gives the most negative cost difference.

Unlike for one-symbol updates, a closed-form expression for $\left(\lambda_{i_{1,opt}}^{(k)}, \lambda_{i_{2,opt}}^{(k)}, \dots, \lambda_{i_{K,opt}}^{(k)}\right)$ which minimizes the cost difference over $\boldsymbol{\mathcal{A}}_{\mathcal{U}}^{(k)}$ is difficult to obtain. A brute-force method can be evaluated or approximate methods can be adopted to solve this problem with less complexity. Using one of these method based

on ZF an approximate method can be given as follows. The cost difference function in (3.61) can be written as

$$\Delta C_{\mathcal{U}}^{k+1}\left(\lambda_{i_1}^{(k)}, \lambda_{i_2}^{(k)}, \dots, \lambda_{i_K}^{(k)}\right) = \boldsymbol{\Lambda}_{\mathcal{U}}^{(k)^T} \boldsymbol{F}_{\mathcal{U}} \boldsymbol{\Lambda}_{\mathcal{U}}^{(k)} - 2\boldsymbol{\Lambda}_{\mathcal{U}}^{(k)^T} \mathbf{z}_{\mathcal{U}}^{(k)} \tag{3.62}$$

where $\boldsymbol{\Lambda}_{\mathcal{U}}^{(k)} \triangleq \left[\lambda_{i_1}^{(k)}, \lambda_{i_2}^{(k)}, \dots, \lambda_{i_K}^{(k)}\right]^T$, $\mathbf{z}_{\mathcal{U}}^{(k)} \triangleq \left[z_{i_1}^{(k)}, z_{i_2}^{(k)}, \dots, z_{i_K}^{(k)}\right]^T$, and $\boldsymbol{F}_{\mathcal{U}} \in \mathbb{R}^{KxK}$, where $(\boldsymbol{F}_{\mathcal{U}})_{p,q} = (\boldsymbol{G})_{i_p,i_q}$ and p, q $\in \{1,2, \dots, K\}$. A unique global minimum is given by

$$\widetilde{\boldsymbol{\Lambda}}_{\mathcal{U}}^{(k)} = \boldsymbol{F}_{\mathcal{U}}{}^{-\mathbf{1}} \mathbf{z}_{\mathcal{U}}^{(k)} \tag{3.63}$$

However, the solution in (3.63) need not lie in $\mathbb{A}_{\mathcal{U}}^{(k)}$**.** So, the solution should be rounded to the nearest element as

$$\widehat{\boldsymbol{\Lambda}}_{\mathcal{U}}^{(k)} = 2\left\lfloor 0.5\widetilde{\boldsymbol{\Lambda}}_{\mathcal{U}}^{(k)} \right\rceil \tag{3.64}$$

It is possible that the solution $\widehat{\boldsymbol{\Lambda}}_{\mathcal{U}}^{(k)} = \left[\hat{\lambda}_{i_1}^{(k)}, \hat{\lambda}_{i_2}^{(k)}, \dots, \hat{\lambda}_{i_K}^{(k)}\right]$ given by (3.64) need not lie in $\boldsymbol{\mathcal{A}}_{\mathcal{U}}^{(k)}$ due to $x_{i_j}^{(k+1)} \notin \boldsymbol{\mathcal{A}}_{i_j}$ for some j.In such cases, the following adjustment to $\hat{\lambda}_{i_j}^{(k)}$ should be used

$$\hat{\lambda}_{i_j}^{(k)} = \begin{cases} (M-1) - x_{i_j}^{(k+1)}, & \text{when } x_{i_j}^{(k+1)} + \hat{\lambda}_{i_j}^{(k)} > (M-1) \\ -(M-1) - x_{i_j}^{(k+1)}, & \text{when } x_{i_j}^{(k+1)} + \hat{\lambda}_{i_j}^{(k)} < -(M-1) \end{cases} \tag{3.65}$$

The optimum $\mathcal{U}$ for approximate method is given below

$$\hat{\mathcal{U}} \triangleq \hat{\imath}_1, \hat{\imath}_2, \dots, \hat{\imath}_K = \arg\min_{\mathcal{U}} \Delta C_{\mathcal{U}}^{k+1}\left(\hat{\lambda}_{i_1}^{(k)}, \hat{\lambda}_{i_2}^{(k)}, \dots, \hat{\lambda}_{i_K}^{(k)}\right) \tag{3.66}$$

The K-symbol update algorithm is done only if

$$\Delta C_{\hat{\mathcal{U}}}^{k+1}\left(\hat{\lambda}_{\hat{\imath}_1}^{(k)}, \hat{\lambda}_{i_2}^{(k)}, \dots, \hat{\lambda}_{\hat{\imath}_K}^{(k)}\right) < 0 \tag{3.67}$$

The K-symbol update rule is given as

$$\boldsymbol{z}^{(k+1)} = \boldsymbol{z}^{(k)} - \sum_{j=1}^{K} \hat{\lambda}_{\hat{\imath}_j}^{(k)} \boldsymbol{g}_{\hat{\imath}_j} \tag{3.68}$$

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \sum_{j=1}^{K} \hat{\lambda}_{\hat{\imath}_j}^{(k)} \boldsymbol{e}_{\hat{\imath}_j} \tag{3.69}$$

3.5.2.3 Performance of K-LAS

In Figure 3.9, it can be seen that the BER performance of 3-LAS algorithm approaches unfaded SISO AWGN performance for increasing antenna. For $n_t = n_r = 64$, 3-LAS achieves an uncoded BER of 10^{-3} at 1 dB away from SISO AWGN performance. For $n_t = n_r = 128$ and 256, the 3-LAS performance moves closer to the AWGN performance.

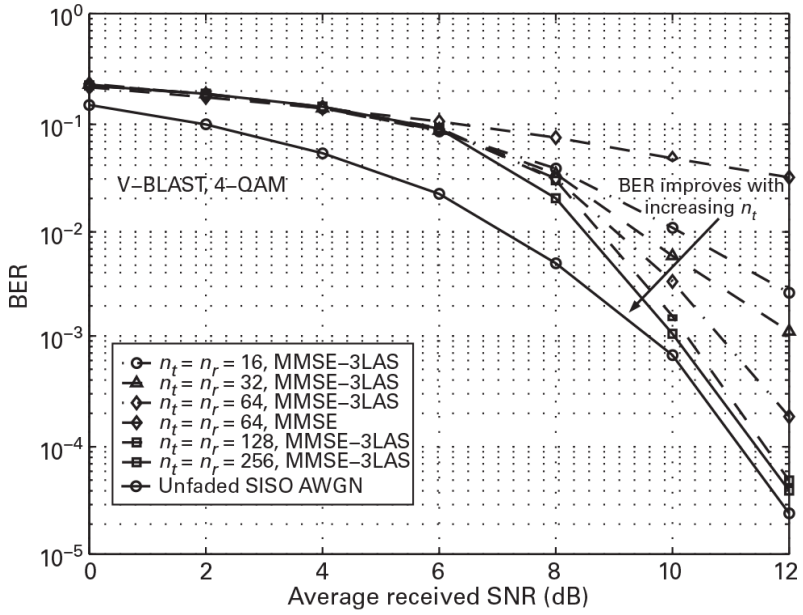

Figure 3.9: The BER performance of 3-LAS in large V-BLAST MIMO systems with 4-QAM [5].

In Figure 3.10, BER performance of 1-LAS and 3-LAS are compared. 3-LAS achieves a better performance than 1-LAS. When the number of antennas are very large, 1-LAS also achieves near-AWGN performance. However, 3-LAS algorithm achieves near-AWGN performance with tens of antennas.

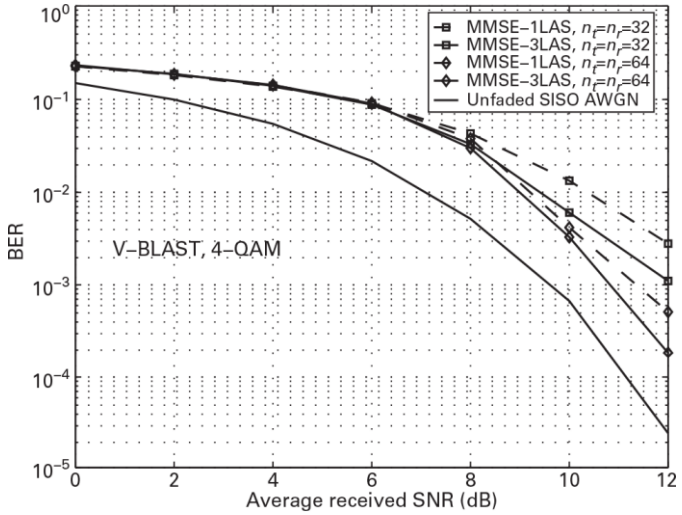

Figure 3.10: The BER performance of 3-LAS vs. 1-LAS in large V-BLAST MIMO systems with 4-QAM [5].

3.5.3 Multiple output selection-LAS algorithm

Multiple output selection LAS (MOS-LAS) [7] is a low complexity decoding algorithm based on likelihood ascent search in large MIMO systems. The main idea of this algorithm is to generate multiple possible solutions or outputs and select the best one from them. The system model is V-BLAST MIMO with n_t transmit and n_r receive antenna.

In order to improve the BER performance of conventional LAS algorithms, MOS-LAS generates different local maximum likelihood (LML) points and then select best one as the final output. It is expected the probability that the output is a global maximum likelihood (GML) is increased. There is two MOS-LAS approaches to achieve this goal.

3.5.3.1 Multiple initial vectors (MIV)-LAS algorithm

The procedure of multiple initial vector LAS (MIV-LAS) is given as follows:

1. Generate K initial vectors $p_1, p_2, \dots, p_K$.
2. Obtain K LML points $x_1, x_2, \dots, x_K$ by using LAS algorithm with K different initial vectors generated in (1)
3. Select the LML points with the minimum metric, i.e.,

$$\hat{\boldsymbol{x}} = arg \min_{i=1,2,\dots,K} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x_i}\|^2 \tag{3.70}$$

Initial vectors are not necessarily generated from known detectors. It could be random vectors. Generating initial vectors randomly, there is no need channel inversion.

Further, using random initial vectors, MIV-LAS can achieve better BER performance and less complexity at the same time compared to conventional LAS.

3.5.3.2 Multiple search candidate sets (MSCS)-LAS algorithm

MSCS-LAS algorithm is another approach which uses only one initial vector. MMSE is used to generate initial vector because it generally gives best results. The algorithm update one bit each step with different orders.

The algorithm can be given as

1) Generate an initial vector p using MMSE detector.
2) Obtain K LML points $x_1, x_2, \dots, x_K$ by using MMSE-LAS algorithm with K different sequence of SCS as $L_1(n), L_2(n), \dots, L_K(n)$.
3) Select the LML point with the minimum metric given in (3.70).

The complexity of MIV-LAS is about K times the conventional LAS. The complexity of MSCS-LAS algorithm is comparable to conventional LAS with small K (K<10).

3.5.3.3 Performance of MOS-LAS algorithm

In figures, KR-MIV-LAS denotes the MIV-LAS algorithm with K random vectors, ZF/MF/MMSE-MIV-LAS denotes the MIV-LAS algorithm with 3 initial vectors using ZF, MF, and MMSE, K-MSCS-LAS denotes the MSCS-LAS algorithm with K SCS. In Figure 3.11, it can be seen that MIV-LAS algorithm performs similarly to MSCS-LAS. Using 20 random initial vectors, MIV-LAS can achieve ML performance for less than 10 antennas.

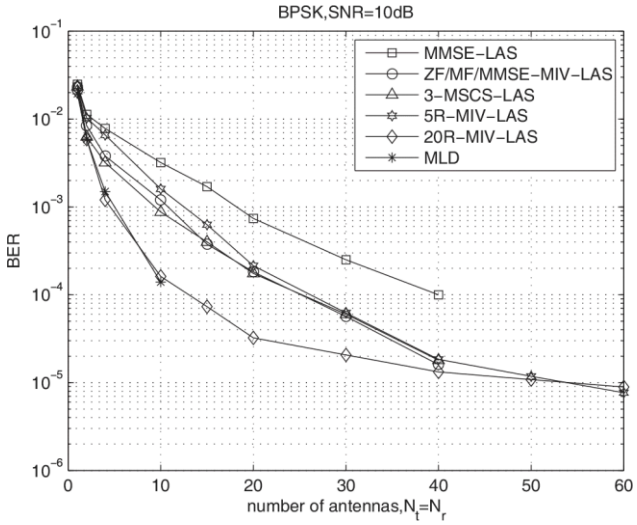

Figure 3.11: The BER performance of MOS-LAS as a function of antennas at 10 dB SNR, BPSK [7].

In Figure 3.12, one can see that ZF-LAS and MF-LAS algorithms perform poorly in high order modulations. MSCS-LAS algorithm has better BER performance than MMSE-LAS and MIV-LAS with almost the same or less complexity.

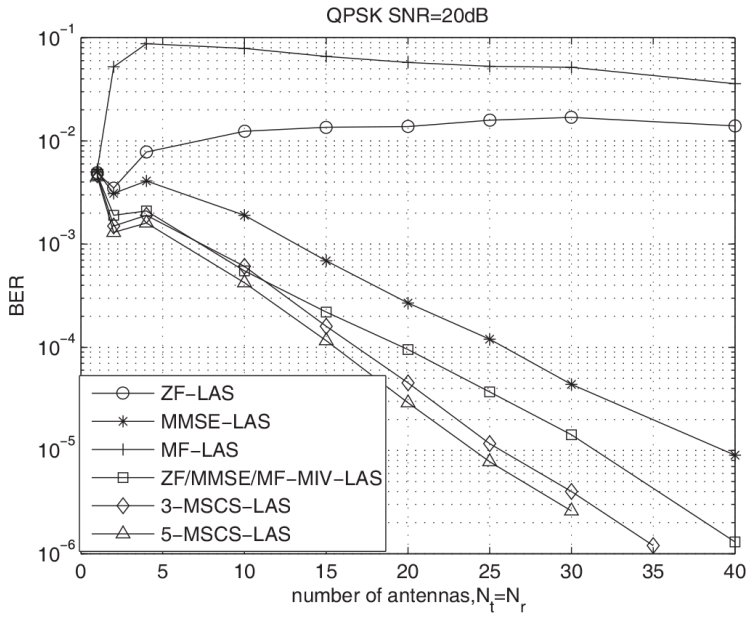

Figure 3.12: The BER performance of MOS-LAS as a function of antennas at 20 dB SNR, QPSK [7].

In Figure 3.13, it can be seen that ZF/MMSE/MF-MIV-LAS algorithm has about 2dB SNR gain over the MMSE-LAS algorithm. On the other hand, 3-MSCS-LAS algorithm has 4dB SNR gain.

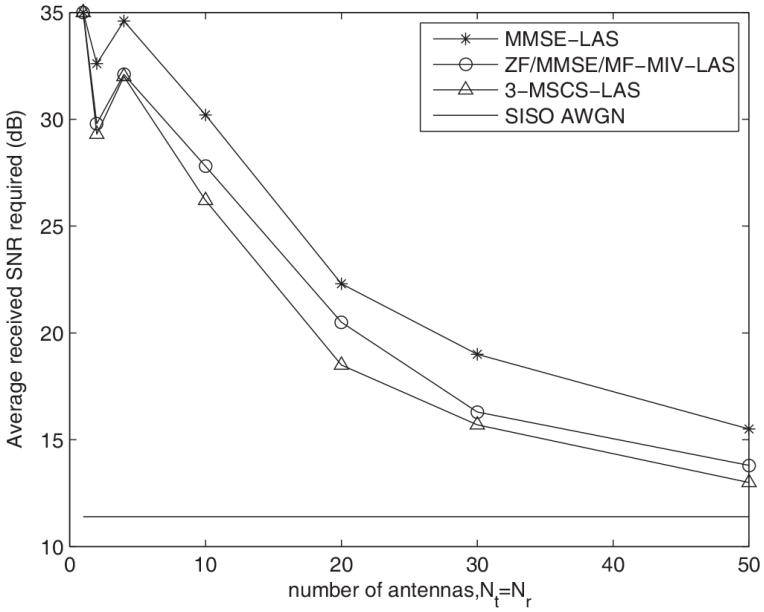

Figure 3.13: Average received SNR required to achieve a target BER of 10^{-4} as a function of number of transmit/receive antennas for QPSK [7].

3.6 Concluding Remarks

In this chapter, decoding techniques are reviewed. It is shown that ML and sphere decoding are not feasible to use in large MIMO systems because of the computational complexity. ZF, MMSE, MF and ZF-SIC are not well scaled to use in large MIMO due to the bad BER performance. On the other hand, LAS algorithm can be used in large MIMO system because it has better performance than linear decoding techniques and ZF-SIC, and also computational complexity of LAS algorithm is lesser than comlexity of ML and sphere decoder.

4. HOPFIELD NEURAL NETWORK

There are numerous examples where artificial neural networks are used in optimization. The application of neural networks in the field of optimization was initiated by Hopfield and Tank [42, 43]. Since then, Hopfield neural network (HNN) has always been the major neural network in solving optimization problems, because of the potential of extremely rapid computation power and speed of neural networks. The main idea of using the HNN in difficult optimization problems is achieved by designing and formulating the cost function of the optimization problem in the form of the energy function of the HNN [44], and then let the HNN dynamically find the solution of the optimization problem. Various optimization problems in different disciplines as mentioned above are indeed formulated similarly. ML problems in image restoration are solved using HNN [45-49].

In this section, general architectures of neural networks and discrete-time Hopfield neural network will be reviewed.

4.1 Architectures of Neural Networks

There are numerous examples where artificial neural networks are used in optimization An artificial neural network is a signal processing system composed of a large number of simple processing elements which are called artificial neurons. They are interconnected by direct links called connections and cooperate to perform parallel distributed processing in order to solve a desired computational task.

The artificial neurons can be modelled as a multi-input nonlinear device with weighted interconnections w_{ji} which is called synaptic weights or strengths. Simplified functional model of an artificial basic neuron cell is given in Figure 4.1.

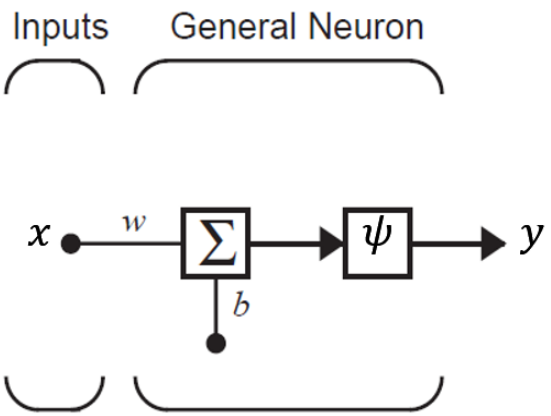

Figure 4.1: Simplified functional model of an artificial basic neuron

The cell body is represented by a nonlinear limiting or threshold function $\psi(u_j)$. Typical nonlinearity functions used in artificial neural networks and corresponding Matlab function is given in Figure 4.2.

Name	Input/Output Relation	Icon	MATLAB Function
Hard Limit	$y = 0 \quad x < 0$ $y = 1 \quad x \geq 0$		hardlim
Symmetrical Hard Limit	$y = -1 \quad x < 0$ $y = +1 \quad x \geq 0$		hardlims
Linear	$y = x$		purelin
Saturating Linear	$y = 0 \quad x < 0$ $y = x \quad 0 \leq x \leq 1$ $y = 1 \quad x > 1$		satlin
Symmetric Saturating Linear	$y = -1 \quad x < -1$ $y = x \quad -1 \leq x \leq 1$ $y = 1 \quad x > 1$		satlins
Log-Sigmoid	$y = \frac{1}{1 + e^{-x}}$		logsig
Hyperbolic Tangent Sigmoid	$y = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$		tansig
Positive Linear	$y = 0 \quad x < 0$ $y = n \quad 0 \leq x$		poslin
Competitive	$y = 1$ neuron with max $y = 0$ all other neurons	C	compet

Figure 4.2: Typical nonlinearity functions used in artificial neural networks.

The output of the artificial neuron can be given as

$$y_j = \psi\left(\sum_{i=1}^{n} w_{ji}x_i + b_j\right) \tag{4.1}$$

where ψ is a limiting function which is also called an activation function, and $b_j \in \Re$ is the external threshold. This threshold is also called as offset or bias. w_{ji} are the synaptic weights or strengths, $x_i, i = 1,2, \dots, n$ are inputs and y_j are the outputs.

4.2 Discrete-time Hopfield Models of Artificial Neural Networks

The application of Hopfield networks to optimization problems is based on the existence of a Lyapunov (energy) function. This function decreases while the network spontaneously evolves. Therefore, the minima of the energy function must coincide to the stable states of the network. In order to solve a particular optimization problem by means of a Hopfield network, the network must be designed so that it possesses a Lyapunov function that matches the target function of the problem. The procedure for optimization with Hopfield networks can be given in the following steps [50].

1) A particular optimization problem is posed.
2) The problem is modelled by a particular neuronal topology.
3) Both the target function and the problem restrictions are converted into mathematical relations among the values of neuron states.
4) The previously defined target function and the standard Lyapunov function of Hopfield networks are matched. The equivalence between corresponding terms of both functions determines the value of each network connection.

Hopfield model can be described by equation given in (4.2-3). A Hopfield network consists of a set n of completely linked neurons without direct recurrences. The activation function of the neurons is the binary threshold function with outputs $\{1, -1\}$. The state of the network consists of the activation states of all neurons. Hence, the state of the network can be given as a binary string as $\{-1, 1\}^{|n|}$.

A hard-limiting quantizer (signum function) is usually used as the nonlinear activation function in the digital implementation of the discrete-time model. The update rule of discrete-time Hopfield networks can be written as

$$x_j^{(k+1)} = sgn\left[\sum_{i=1}^{n} w_{ji} x_i^k + b_j\right] = \begin{cases} 1, & \text{if } u_j^{(k+1)} \geq 0 \\ -1, & \text{otherwise} \end{cases}, k = 0,1,2,\ldots \quad (4.2)$$

The energy function of discrete-time model can be defined as

$$E(x) = -\frac{1}{2}\boldsymbol{x}^T \boldsymbol{W}\boldsymbol{x} - \boldsymbol{x}^T \boldsymbol{b} \quad (4.3)$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T, \boldsymbol{b} = [b_1, b_2, \ldots, b_n]^T, \boldsymbol{W} = \left[w_{ij}\right]_{nxn}$ with $w_{ij} = w_{ji}$, $w_{jj} = 0$. Hopfield proves that energy function given in equation (4.3) is a Lyapunov function of the system, as long as the following conditions are fulfilled: absence of self-connections, and connection symmetry.

Each neuron (node) can take values either $+1$ or -1 in each of the two possible states. The next state of the whole network is computed from the current state by performing the equation given in (4.2). The number of neurons varies from 1 to n.

The discrete-time neural networks operate in different modes depending on the number of neurons changed in each time interval. If the computation is performed in all neurons in the same time, then the network is operating in a fully parallel mod. If the computation is performed at a single neuron in any time interval, then the neural network is operating in a serial mode. All other cases with $1 < N < n$ are defined as parallel modes of operation. The set $\{N\}$ can be chosen randomly or according to some deterministic rule [51].

Consider a discrete-time neural network operating in the serial mode. At some arbitrary time, suppose that $x_j$is changed to

$$x_j^{(k+1)} = x_j^{(k)} + \Delta x_j \quad (4.4)$$

The resulting change in the energy function can be written as

$$\Delta E(x) = E\left(\boldsymbol{x}^{(k+1)}\right) - E\left(\boldsymbol{x}^{(k)}\right) = -\Delta x_j \frac{1}{2}\left(\sum_{i=1}^{n} w_{ji} x_i^{(k)} + \sum_{p=1}^{n} w_{pi} x_p^{(k)}\right) - w_{jj}\left(\Delta x_j\right)^2 - \Delta x_j b \quad (4.5)$$

Assume that weight matrix $\boldsymbol{W}$ is symmetric, then change in the energy function can be written as

$$\Delta E(x) = -\Delta x_j \left[\sum_{i=1}^{n} w_{ji} x_i^{(k)} + b_j \right] - w_{jj} \left(\Delta x_j \right)^2 \tag{4.6}$$

The sign of Δx_j and the sign of the term in brackets are same. Therefore, the energy is a nonincreasing function. Since the energy $E(x)$ is bounded, the network will always converge to a stable state which corresponds to a local minimum in the energy function $E(x)$.

A state $\boldsymbol{x}(t)$ is defined as stable if and only if

$$\boldsymbol{x}(t) = sgn[\boldsymbol{W}\boldsymbol{x}(t) + \boldsymbol{b}] \tag{4.7}$$

On the other words, there is no change in the state of the network. The network will reach a stable state after at most n^2 time intervals, where n is the number of neurons.

Hopfield neural networks operating procedure can be given briefly as follows. Given initial values of neurons, HNN updates each neuron one by one, by minimizing the energy function given in (4.3), until the network reach a stable state. When network is stable, local minima is found and operating procedure is terminated.

5. A LOW-COMPLEXITY DETECTOR FOR LARGE MIMO SYSTEMS

We present a low complexity algorithm for symbol decoding in large MIMO systems based on Hopfield neural network (HNN) algorithm. Our algorithm achieves near Maximum Likelihood performance for high number of antennas. It has much lesser computational and algorithmic complexity compared to ML decoding and lesser computations for each searching step compared to conventional Likelihood Ascent Search (LAS) algorithm. No conditional evolution, comparison to multiple thresholds and multi-level quantizations are necessary for updating in the search stage. We present simulations confirming that with simpler evolution and lesser computations our algorithm yields comparable results to LAS algorithm in [4-7].

5.1 System Model

We consider a V-BLAST system with n_t transmit antennas and n_r receive antennas($n_t \leq n_r$), where n_t symbols are simultaneously transmitted from n_t transmit antennas. The baseband system model can be written as

$$\boldsymbol{y_c} = \boldsymbol{H_c x_c} + \boldsymbol{n_c} \tag{5.1}$$

where $\boldsymbol{y_c} \in \mathbb{C}^{n_r}$ is the received vector, $\boldsymbol{x_c} \in \mathbb{C}^{n_t}$ is the transmit signal vector. Each element of $\boldsymbol{x_c}$ is an M-QAM or M-PAM symbol. $\boldsymbol{n_c} \in \mathbb{C}^{n_r}$ is the complex white Gaussian noise (AWGN) with zero mean and $E[{n_k}^2] = N_0 = \frac{N_t E_s}{\gamma}$ variance, where n_k is noise sample of kth receive antenna, E_s is the average energy of the transmitted symbols and γ is the average received SNR per receive antenna [21]. $\boldsymbol{H_c} \in \mathbb{C}^{n_r \, x \, n_t}$ is the channel gain matrix whose entries are modelled as i.i.d. Complex Gaussian random variables with zero mean and $E\left[\left(h_{kj}^I\right)^2\right] = E\left[\left(h_{kj}^Q\right)^2\right] = 0.5$, where h_{kj}^I and h_{kj}^Q are the real and imaginary parts of each entries of $\boldsymbol{H_c}$, respectively.

The complex system in (5.1) can be converted into a real-valued system model as in [5],

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{5.2}$$

where

$$\boldsymbol{H} \triangleq \begin{bmatrix} \Re(\boldsymbol{H}) & -\Im(\boldsymbol{H}) \\ \Im(\boldsymbol{H}) & \Re(\boldsymbol{H}) \end{bmatrix} \in \mathbb{R}^{2n_r x 2n_t}, \qquad \boldsymbol{y} \triangleq [\Re(\boldsymbol{y})^T \quad \Im(\boldsymbol{y})^T]^T \in \mathbb{R}^{2n_r},$$

$$\boldsymbol{x} \triangleq [\Re(\boldsymbol{x})^T \quad \Im(\boldsymbol{x})^T]^T \in \mathbb{R}^{2n_t}, \qquad \boldsymbol{n} \triangleq [\Re(\boldsymbol{n})^T \quad \Im(\boldsymbol{n})^T]^T \in \mathbb{R}^{2n_r},$$

Each element of $\boldsymbol{x_c}$is an M-PAM or M-QAM symbol. M-PAM symbols take values from $\{A_k | \, k = 1,2, \dots, M\}$, where $A_k = (2k - 1 - M)$ and M-QAM is two PAMs in quadrature [3]. With real-valued system model, the real part of the complex symbols will be mapped to $[x_1, \dots, x_{N_t}]$ and imaginary part will be mapped into $[x_{N_t+1}, \dots, x_{2N_t}]$. $[x_{N_t+1}, \dots, x_{2N_t}]$ will be zeroes for M-PAM since it has real symbols and imaginary part of these symbols are zeroes.

ML Solution:

$2n_t$-dimensional signal space $\mathbb{S}$ is Cartesian product of $\mathbb{A}_1$ to $\mathbb{A}_{2N_t}$, where $\mathbb{A}_i$ is the signal set of M-PAM or M-QAM. For example, $\mathbb{A}_i = \{-1,1\}$ for 4-QAM modulation. The ML solution vector can be given by,

$$\begin{aligned} \boldsymbol{x}_{ML} &= \arg\min_{\boldsymbol{x} \in \mathbb{S}} \| \, \boldsymbol{y} - \boldsymbol{H}\boldsymbol{x} \, \|^2 \\ &= \arg\min_{\boldsymbol{x} \in \mathbb{S}} \boldsymbol{x}^T \boldsymbol{H}^T \boldsymbol{H}\boldsymbol{x} - 2\boldsymbol{y}^T \boldsymbol{H}\boldsymbol{x} \end{aligned} \tag{5.3}$$

If the entries of $\boldsymbol{x}$ belong to $\sqrt{M}$-PAM or M-QAM with $M > 4$, another representation of these variables is necessary. For instance, in the M-PAM case, ith entry of $\boldsymbol{x}$ is expanded as

$$x_i = z_{i_0} + 2z_{ii_1} + \cdots + 2^N z_{i_N} \tag{5.4}$$

where z'_{i_k}s are binary variables assuming ± 1 values and $N = (\log_2 \sqrt{M}) - 1$. In such a case equations (5.2-3) should be readjusted to handle binary variables.

5.2 The Proposed Detector for Large MIMO Systems

We propose a low-complexity detector based on Hopfield neural network (HNN) algorithms for ML solution in very large MIMO systems. There are two types of discrete Hopfield neural network (DHNN) identified by their operation mode: synchronous (parallel) and asynchronous (serial) mode. We use asynchronous DHNN mode in our algorithm. In asynchronous mode, DHNN updates each element one by one. Given an initial vector $\boldsymbol{x}^{(0)}$, DHNN updates values of neurons until a given stopping criteria is satisfied. Initial vector in our algorithm is either generated randomly or using known detectors, i.e., ZF, MMSE or MF. As a stopping criterion, DHNN updates elements until the network reaches into a stable state.

For the ML solution vector given by **(5.3)**, let us define the transition matrix (or the weight matrix) as $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H}^T\boldsymbol{H})$ and the threshold vector as $\boldsymbol{b} = \boldsymbol{y}^T\boldsymbol{H}$. $\mathbb{Z}(.)$ is diagonal zeroing operator. Zhuo et al. proposed a DHNN solution for Four-Coloring MAP problem by rearranging the weight matrix to zero diagonal [52]. Since the optimization is discrete with variables assuming ±1 values, the original optimization problem is unaffected by this operation. In DHNN, rearranging the diagonal of weight matrix will not affect the solution set which is demonstrated below. Let $\boldsymbol{\Delta} = \boldsymbol{H}^T\boldsymbol{H} - \mathbb{Z}(\boldsymbol{H}^T\boldsymbol{H})$, then

$$\begin{aligned}\frac{1}{2}\boldsymbol{x}^{(n)^T}(\boldsymbol{W}+\boldsymbol{\Delta})\boldsymbol{x}^{(n)} - \boldsymbol{b}^T\boldsymbol{x}^{(n)} &= \frac{1}{2}\boldsymbol{x}^{(n)^T}\boldsymbol{W}\boldsymbol{x}^{(n)} - \boldsymbol{b}^T\boldsymbol{x}^{(n)} + \sum_{i=1}^{N_t}\Delta_{ii}x_i^2 \\ &= \tfrac{1}{2}\boldsymbol{x}^{(n)^T}\boldsymbol{W}\boldsymbol{x}^{(n)} - \boldsymbol{b}^T\boldsymbol{x}^{(n)} + \textstyle\sum_{i=1}^{N_t}\Delta_{ii}\end{aligned} \tag{5.5}$$

$\sum_{i=1}^{N_t}\Delta_{ii}$ is a constant and does not contribute to the solution set.

Assuming binary variables for the entries of $\boldsymbol{x}$, at the nth iteration, the ML cost function is given by

$$q\left(\boldsymbol{x}^{(n)}\right) = \frac{1}{2}\boldsymbol{x}^{(n)^T}\boldsymbol{W}\boldsymbol{x}^{(n)} - \boldsymbol{b}^T\boldsymbol{x}^{(n)} \tag{5.6}$$

where $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H}^T\boldsymbol{H})$, $\boldsymbol{b} = \boldsymbol{y}^T\boldsymbol{H}$.

Discrete HNN Update Rule for ith symbol and $(n+1)$st iteration (with a slight modification) can be written as,

$$x_i^{(n+1)} = -sgn\{\boldsymbol{W}_i \boldsymbol{x}^n - b_i\} \tag{5.7}$$

where $\boldsymbol{W_i}$ is ith row of $\boldsymbol{W}$, b_i is ith element of $\boldsymbol{b}$. $\boldsymbol{x} \in \{-1,1\}^{\boldsymbol{n_t}}$ and $W_{i,i} = 0$, $W_{i,j} = W_{j,i}$ (symmetry). $sgn(.)$ is a signum function (or bipolar binary function) defined by

$$sgn(y) = \begin{cases} 1, \text{if } y \geq 0 \\ -1, \text{otherwise} \end{cases} \tag{5.8}$$

Expressing the cost function at the $(n+1)$st iteration, we obtain

$$\begin{aligned} q\left(\boldsymbol{x}^{(n+1)}\right) &= q\left(\boldsymbol{x}^{(n)}\right) + \Delta\boldsymbol{x}^{(n)^T} \nabla \mathrm{q}\left(\mathbf{x}^{(\mathrm{n})}\right) \\ &= q\left(\boldsymbol{x}^{(n)}\right) + \left(\boldsymbol{x}_i^{(n+1)} - \boldsymbol{x}_i^{(n)}\right)\left(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i\right) \\ &= q\left(\boldsymbol{x}^{(n)}\right) + \left(-sgn\left(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i\right) - x_i^{(n)}\right)\left(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i\right) \\ &= q\left(\boldsymbol{x}^{(n)}\right) - \left|\left(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i\right)\right| - x_i^{(n)}(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i) \end{aligned} \tag{5.9}$$

where $\nabla \mathrm{q}(\mathbf{x}) = \boldsymbol{W}\boldsymbol{x} - \boldsymbol{b}$ and $\Delta\boldsymbol{x}^{(n)} = \boldsymbol{x}^{(n+1)} - \boldsymbol{x}^{(n)}$. The quadratic terms do not exist due to zero diagonal. Since the additional term to $q\left(\boldsymbol{x}^{(n)}\right)$ in (5.9) is either $-2\left|\left(\boldsymbol{W}_i \boldsymbol{x}^{(n)} - b_i\right)\right|$ or 0, the cost function is non-increasing. Through the iterations, the equations of (5.7) would settle in one of the local minima of the cost function given in (5.6). The algorithm of the proposed detector with initial vector generated by known decoding algorithm is given below:

1) Input: $\boldsymbol{H}, \boldsymbol{y}$. Obtain $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H}^T\boldsymbol{H})$, and $\boldsymbol{b} = \boldsymbol{y}^T\boldsymbol{H}$.
2) Initialize: Compute initial vector $\boldsymbol{x}^{(0)}$ using known decoding algorithm, i.e., ZF, MF, or MMSE.
3) Update all symbols sequentially, i.e., $i = 1, 2, \ldots, 2n_t$ for M-QAM modulation and $i = 1,2, \ldots, n_t$ for M-PAM modulation, using equation (5.7).
4) Compare $\boldsymbol{x}^{(n+1)}$ with $\boldsymbol{x}^{(n)}$. If there is no change with $(n+1)$st update, solution is found. Terminate the algorithm.

The algorithm of the proposed detector with initial vector generated randomly is given below:

1) Input: $\boldsymbol{H}, \boldsymbol{y}$. Obtain $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H}^T\boldsymbol{H})$, and $\boldsymbol{b} = \boldsymbol{y}^T\boldsymbol{H}$.
2) Initialize: Generate K different initial vector $\boldsymbol{s_k}|k = 1,2, \ldots K$ randomly.
3) For K parallel update, search all symbols sequentially, i.e., $i = 1, 2, \ldots, 2n_t$ for M-QAM modulation and $i = 1,2, \ldots, n_t$ for M-PAM modulation, using equation (5.7).

5) For all parallel update, Compare $\boldsymbol{s_k^{(n+1)}}$ with $\boldsymbol{s_k^{(n)}}$. If there is no change with $(n+1)$st update, solution is found. Terminate the algorithm.
4) Select the vector $\hat{\boldsymbol{s}}$ from K different vector with the minimum metric, i.e.,

$$\hat{\mathrm{s}} = \underset{k=1,2,\ldots,K}{\arg\min} \parallel \boldsymbol{y} - \boldsymbol{H}\boldsymbol{s_k} \parallel^2 \tag{5.10}$$

Another approach is generating initial solution vector by iterative method which is given below.

$$x_i^{(n+1)} = \frac{-\boldsymbol{Q_{1,i}}\boldsymbol{x^{(n)}} - \boldsymbol{Q_{2,i}}\boldsymbol{x^{(n)}} + b_i}{diag(\boldsymbol{H^T H})_i}, i = 1,2,\ldots 2n_t \tag{5.11}$$

where $\boldsymbol{Q_1}$ and $\boldsymbol{Q_2}$ is the lower and upper triangular matrix of $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H^T H})$, respectively, $\boldsymbol{b} = \boldsymbol{y^T H}$, and $diag(\boldsymbol{H^T H})_i$ shows the *i*th diagonal element of the $\boldsymbol{H^T H}$. $\boldsymbol{Q_{1,i}}$ and $\boldsymbol{Q_{2,i}}$ shows the *i*th row of these matrices, b_i and x_i show the *i*th element of $\boldsymbol{b}$ and $\boldsymbol{x}$, respectively.

The search algorithm starts with a random initial vector $\boldsymbol{x^{(0)}}$. Then, update each element of $\boldsymbol{x}$, until the solution vector reaches a stable state.

It is not guaranteed whether the iteration reaches a stable state. Hence, for each $2n_t$ iteration, algorithm checks whether the cost function decreases. If the iterative search algorithm reaches a stable state or the cost function does not decrease for the next iteration, the algorithm terminates and a solution is found. Then, the proposed detector use this solution as an initial vector and searching algorithm is run. The algorithm using this method can be summarized as

1) Generate a random initial vector $\boldsymbol{x^{(0)}}$.
2) Calculate $\boldsymbol{W} = \mathbb{Z}(\boldsymbol{H^T H})$, $\boldsymbol{Q_1}$, $\boldsymbol{Q_2}$, $\boldsymbol{b} = \boldsymbol{y^T H}$, and $diag(\boldsymbol{H^T H})$.
3) Run search algorithm given in (5.11).
4) Compare $\boldsymbol{x}^{(n+1)}$ with $\boldsymbol{x}^{(n)}$. If there is no change with $(n+1)$st update, terminate the algorithm and go step 6.
5) Calculate the cost function for each $2n_t$ iteration. If cost function does not decreases, terminate the algorithm and go step 6.
6) Use this solution vector as an initial solution for the proposed detector.

7) Update all symbols sequentially, i.e., $i = 1, 2, \dots, 2n_t$ for M-QAM modulation and $i = 1,2, \dots, n_t$ for M-PAM modulation, using equation (5.7).
8) Compare $\boldsymbol{x}^{(n+1)}$ with $\boldsymbol{x}^{(n)}$. If there is no change with $(n+1)$st update, a solution is found. Terminate the algorithm

When initial vector is generated by ZF and MMSE detectors, algorithm needs to deal with matrix inversion which is not necessary anymore for the iterative method.

For high order modulation, i.e., M-QAM or $\sqrt{M}$-PAM with $M > 4$, the algorithm should be rearranged. $\boldsymbol{x}$ takes values from

$$\mathbb{A} = \left\{a_q = (2q - 1 - \sqrt{M}), q = 1,2, \dots, \sqrt{M}\right\} \tag{5.12}$$

The value of each entry of $\boldsymbol{x}$ can be written as a linear combination of its bits as

$$x_i = \sum_{j=0}^{p-1} 2^j z_i^{(j)}, i = 1,2, \dots, 2n_t \tag{5.13}$$

where $p = log_2(\sqrt{M})$, and $z_i^{(j)} \in \{\pm 1\}$.

$\boldsymbol{z} \in \{\pm 1\}^{2pn_t}$ can be defined as

$$\boldsymbol{z} = \left[z_1^{(0)}, z_2^{(0)}, \dots, z_{2n_t}^{(0)}, z_1^{(1)}, z_2^{(1)}, \dots, z_{2n_t}^{(1)}, \dots, z_1^{(p-1)}, z_2^{(p-1)}, \dots, z_{2n_t}^{(p-1)}\right] \tag{5.14}$$

Let us define

$$\boldsymbol{Q} = \boldsymbol{H^T H}, \widehat{\boldsymbol{b}} = \boldsymbol{y^T H} \tag{5.15}$$

Energy function of Hopfield network can be written as

$$q(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{z^T}(\boldsymbol{e} \otimes \boldsymbol{e^T} \otimes \boldsymbol{Q})\boldsymbol{z} - \left(\boldsymbol{e^T} \otimes \widehat{\boldsymbol{b}}^T\right)\boldsymbol{z} \tag{5.16}$$

where $e = [2^0\ 2^1 \dots 2^{p-1}]$, $\otimes$ denotes Kronecker product.

Let us define

$$\boldsymbol{W} = \mathbb{Z}(\boldsymbol{e} \otimes \boldsymbol{e^T} \otimes \boldsymbol{Q}), \boldsymbol{b} = \left(\boldsymbol{e^T} \otimes \widehat{\boldsymbol{b}}^{\boldsymbol{T}}\right) \tag{5.17}$$

Then, energy function of Hopfield network can be written as

$$q\left(\boldsymbol{z}^{(n)}\right) = \frac{1}{2}\boldsymbol{z}^{(n)^T}\boldsymbol{W}\boldsymbol{z}^{(n)} - \boldsymbol{b}^T\boldsymbol{z}^{(n)} \tag{5.18}$$

The update rule of the proposed algorithm is given below.

$$z_i^{(n+1)} = -sgn\{\boldsymbol{W}_i\boldsymbol{z}^n - b_i\}, i = 1,2,\dots,2pn_t \tag{5.19}$$

For M=16, i.e., 16-QAM or 4-PAM modulation,

$$\boldsymbol{e} = [1\ 2], \boldsymbol{W} = \mathbb{Z}\left\{\begin{bmatrix} \boldsymbol{Q} & 2\boldsymbol{Q} \\ 2\boldsymbol{Q} & 4\boldsymbol{Q} \end{bmatrix}\right\}, \boldsymbol{b} = \left[\widehat{\boldsymbol{b}}^T\ 2\widehat{\boldsymbol{b}}^T\right]^T \tag{5.20}$$

Furthermore, another approach for high order modulation can be given below. Equation 5.13 can be written as

$$x_i = \sum_{j=0}^{p-1} z_i^{(j)}, i = 1,2,\dots,2n_t \tag{5.21}$$

where $p = \sqrt{M}$. Equation 5.14-19 are also valid for this method and $\boldsymbol{e} = [1\ 1\ \dots\ 1]$ is (p-1) dimensional vector.

For M=16, i.e., 16-QAM or 4-PAM modulation,

$$\boldsymbol{e} = [1\ 1\ 1], \boldsymbol{W} = \mathbb{Z}\left\{\begin{bmatrix} \boldsymbol{Q} & \boldsymbol{Q} & \boldsymbol{Q} \\ \boldsymbol{Q} & \boldsymbol{Q} & \boldsymbol{Q} \\ \boldsymbol{Q} & \boldsymbol{Q} & \boldsymbol{Q} \end{bmatrix}\right\}, \boldsymbol{b} = \left[\widehat{\boldsymbol{b}}^T\ \widehat{\boldsymbol{b}}^T\ \widehat{\boldsymbol{b}}^T\right]^T \tag{5.22}$$

This approach introduces redundancy compared to the former method because symbols can be written many different ways. For example, 1 can be written as linear combination of $\{-1,1,1\}, \{1,-1,1\}$, or $\{1,1,-1\}$. Hence, this method is expected to perform better than the former method. Simulation results is given in next section.

The theorem given below shows that the proposed detector and one-symbol update LAS with $x_i \in \{\pm 1\}$ are the same algorithms.

Theorem:

One-symbol update LAS algorithm with $x_i \in \{\pm 1\}$ is the same as the proposed detector given in 5.7.

Proof:

Update rule of the LAS algorithm for one-symbol search is given below.

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)} + \tilde{l}_{i,opt}^{(n)} sgn(z_i^{(n)})\boldsymbol{e_i} \tag{5.23}$$

$$\boldsymbol{z}^{(n+1)} = \boldsymbol{z}^{(n)} - \tilde{l}_{i,opt}^{(n)} sgn(z_i^{(n)})\boldsymbol{w_i} \tag{5.24}$$

where $\boldsymbol{w}_i$ is ith column of $\boldsymbol{G} = \boldsymbol{H^T H}$. If $x_i \in \{\pm 1\}$, then $\tilde{l}_{i,opt}^{(n)} \in \{0,2\}$. $\boldsymbol{z}^{(n)} = \boldsymbol{H}^T\left(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}^{(n)}\right) = -\boldsymbol{g}^{(n)}$ where $\boldsymbol{g}^{(n)}$ is the gradient vector of the likelihood function. Hence, equation 5.23 can be written as

$$x_i^{(n+1)} = x_i^{(n)} - \tilde{l}_{i,opt}^{(n)} sgn(g_i^{(n)}) \tag{5.25}$$

If $g_i^{(n)} > 0$ and $x_i^{(n)} = 1$, $\tilde{l}_{i,opt}^{(n)} = 2, x_i^{(n+1)} = -1$

If $g_i^{(n)} > 0$ and $x_i^{(n)} = -1$, $\tilde{l}_{i,opt}^{(n)} = 0, x_i^{(n+1)} = -1$

If $g_i^{(n)} < 0$ and $x_i^{(n)} = 1$, $\tilde{l}_{i,opt}^{(n)} = 0, x_i^{(n+1)} = 1$

If $g_i^{(n)} < 0$ and $x_i^{(n)} = -1$, $\tilde{l}_{i,opt}^{(n)} = 2, x_i^{(n+1)} = 1$. Therefore, the value of $x_i^{(n+1)}$ depends only the sign of $g_i^{(n)}$. Equation 5.25 can be written as

$$x_i^{(n+1)} = -sgn(g_i^{(n)}) \tag{5.26}$$

where $g_i^{(n)}$ can be given as $g_i^{(n)} = \boldsymbol{W}_i \boldsymbol{x}^n - b_i$, and equation 5.26 is the same as equation 5.7. ∎

5.3 The Computational Complexity of the Proposed Decoding Algorithm

The complexity of the proposed detector comprises three main components.

i) Computation of the initial vector $\boldsymbol{x}^{(0)}$: Using known detector, the per-symbol complexity is $(n_r), O(n_t n_r), and\ O(n_t n_r)$ for MF, ZF, and MMSE, respectively. Generating initial vector randomly instead of using MMSE, or ZF, we get rid of calculation of matrix inversion and calculation cost of these detectors.

ii) Computation of $\boldsymbol{W}$: $\boldsymbol{H}^T\boldsymbol{H}$ can be computed in $O(n_t n_r)$ per-symbol complexity. The computational cost is reduced by diagonal zeroing operator because we do not require to compute diagonal elements of $\boldsymbol{W}$.

iii) The search operation: Due to the simulation results given in Figure (5.1), (5.2), and (5.3), the average number of steps required to reach a fixed point is linear in n_t, i.e., constant per-bit complexity where the constant depends on SNR, n_t, n_r, and the initial vector. The complexity of search operation using random initial vector is about K times the complexity of the proposed detector using known detector (ZF, MMSE, and MF). In spite of the increased complexity, the performance of the detector using random initial vector is better. Because of this, it is acceptable for small K.

The overall average per-bit complexity of the proposed detector is $O(n_t n_r)$, which is dominated by initial vector generation and computation of $\boldsymbol{W}$. Generating initial vector randomly, the calculation of matrix inversion is no longer required.

Contrary to the LAS algorithm, the proposed detector does not update the gradient vector given in (3.50) in each step, and does not require to compare it with the threshold calculated in (3.48). The computational complexity of the proposed detector is the same with the complexity of LAS detector and better than the complexity of ZF-SIC which is $O(n_t^2 n_r)$.

From Figure 5.1 and Figure 5.2**,** it can be seen that the average number of steps grows with increasing number of antennas. For 15 dB SNR, the number of steps of the proposed detector given in Figure 5.1. is larger than the number of steps of LAS algorithm given in Figure 5 of [6]. For 10 dB SNR, the number of steps of the proposed detector is close to the number of steps of LAS detector, and, for 5 dB SNR, the number of steps of the proposed detector is lower.

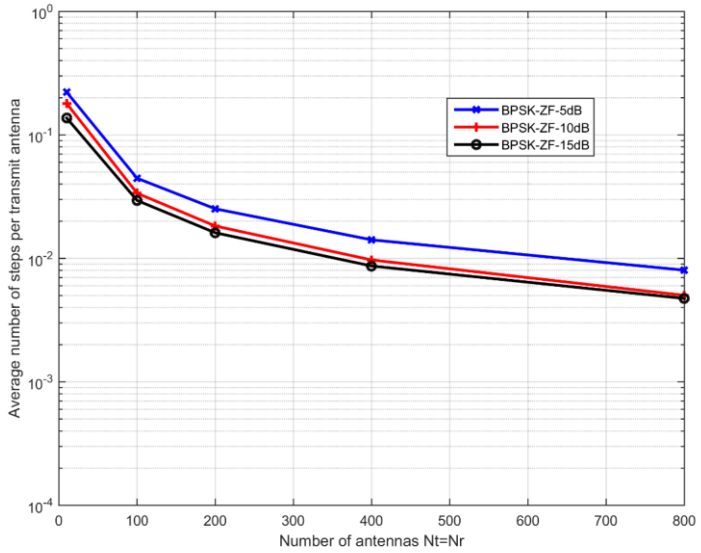

Figure 5.1: The complexity of the proposed detector using ZF detector and BPSK modulation in terms of average number of steps per transmit antenna till fixed point reached.

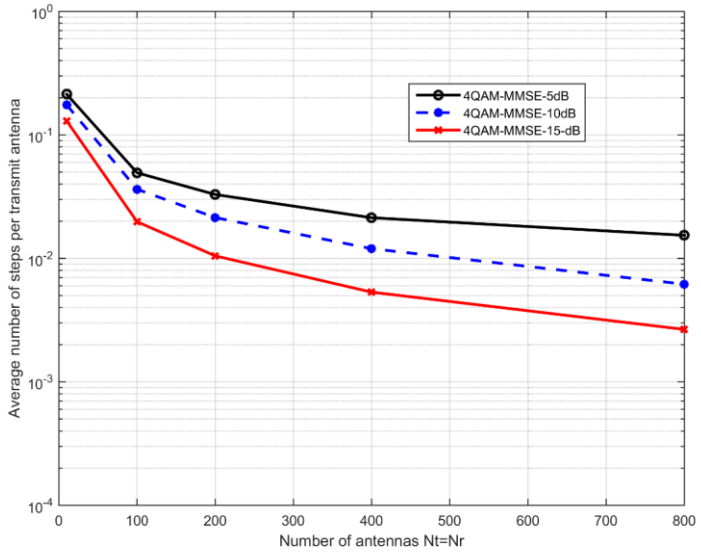

Figure 5.2: The complexity of the proposed detector using MMSE detector and 4QAM modulation in terms of average number of steps per transmit antenna till fixed point reached.

Figure 5.3**.** shows that the average number of steps of the proposed detector using random initial vector is larger than number of steps of the proposed detector with initial vector generated by known detectors. However, as can be seen in Figure 6.9, the BER performance of the former method is better. Furthermore, using random initial vectors, there is no necessity to compute an inverse matrix.

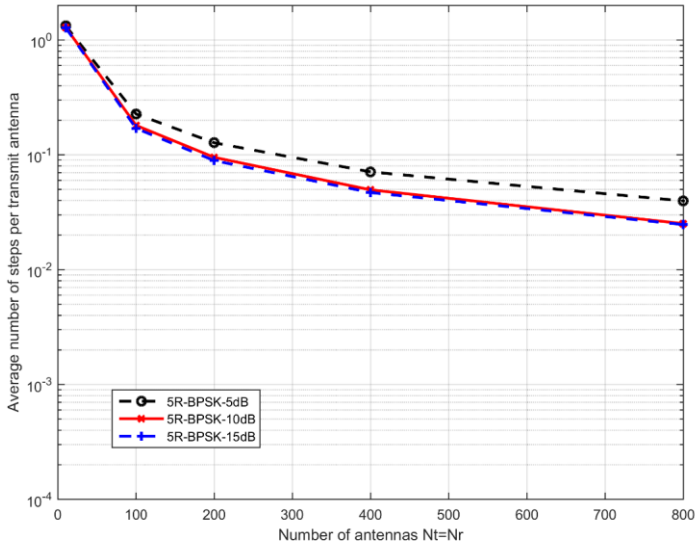

Figure 5.3: The complexity of the proposed detector using random initial vector and BPSK modulation in terms of average number of steps per transmit antenna till fixed point reached.

6. EXPERIMENTAL RESULTS

We simulate the BER performance of our algorithm as a function of average received SNR. In Rayleigh fading channel, we transmit symbols using BPSK and 4QAM modulation techniques. We generate an initial vector using Zero-Forcing (ZF), MMSE equalizer or randomly. Since ML performance in large MIMO cannot be simulated because of its exponential complexity, unfaded single-input single output (SISO) AWGN performance is included as a lower bound on ML performance which is tight at high SNRs and very large MIMO.

Figure 6.1 shows the BER performance of the proposed detector for BPSK modulation. We use ZF equalizer to get the initial vector. Using BPSK modulation, we get better BER performance. Our algorithm achieves near-ML performance when we use $200x200$ or higher number of transmit antennas. These results are comparable to the LAS results given in Figure 3 of [6].

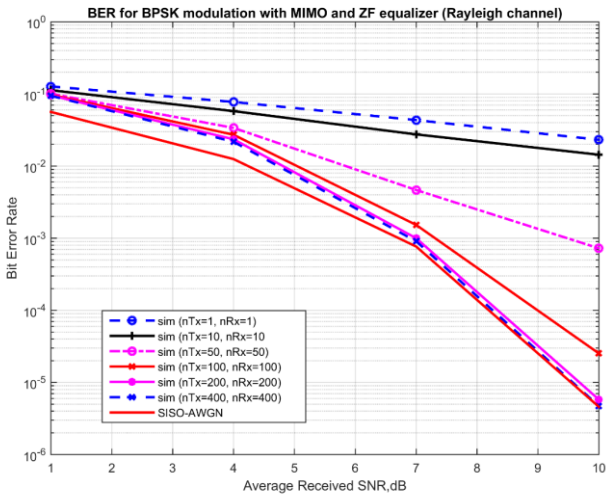

Figure 6.1: The BER performance of the proposed detector for BPSK with ZF initial filter.

In Figure 6.2, it can be seen that the BER performance of the proposed decoding algorithm is very close to performance of SISO AWGN when more than 100 antennas are used for 7dB SNR. For 10 dB SNR, the BER performance increase dramatically with increasing number of antennas. Beyond the 200 antennas, performance is very close to SISO AWGN performance.

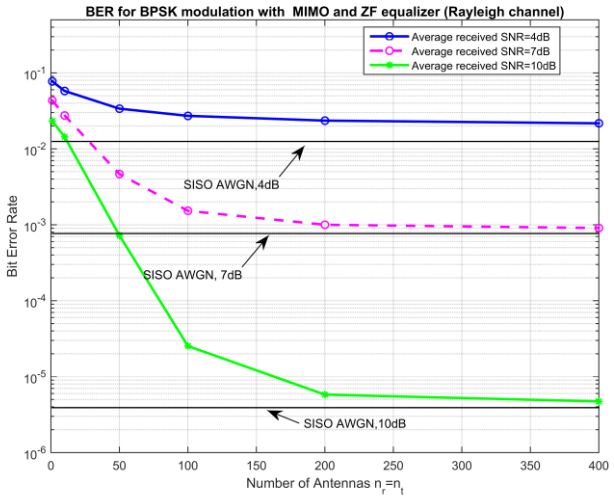

Figure 6.2: The BER performance of the proposed detector with BPSK modulation terms of number of antennas, (ZF).

In Figure 6.3, the BER performance comparison of ZF-LAS and the proposed decoding algorithm is given. The initial solution vector is generated by ZF decoding method and BPSK modulation is used. Simulation results show that the performance of the proposed detector and ZF-LAS is almost same.

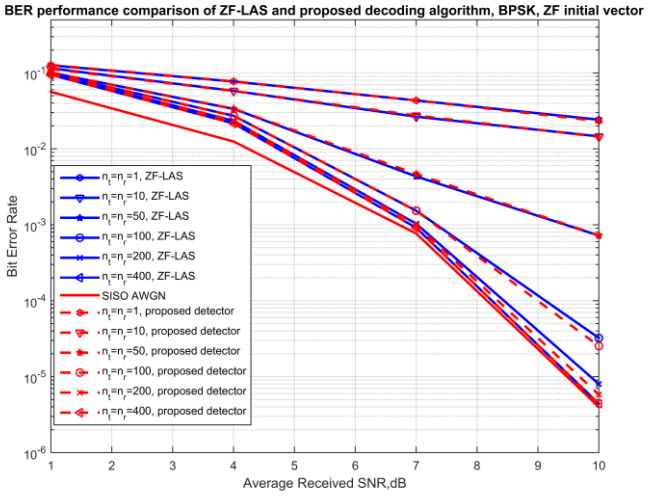

Figure 6.3: BER performance comparison of ZF-LAS and proposed decoding algorithm.

In Figure 6.4, the BER performance of the proposed detector with MMSE equalizer generating the initial solution is given. BPSK modulation is used. The simulation shows that the performance improves with increasing SNR and number of antennas. Beyond 7dB SNR, the BER performance of the proposed detector converges to SISO AWGN performance when more than 200 antennas are used at the transmitter and the receiver.

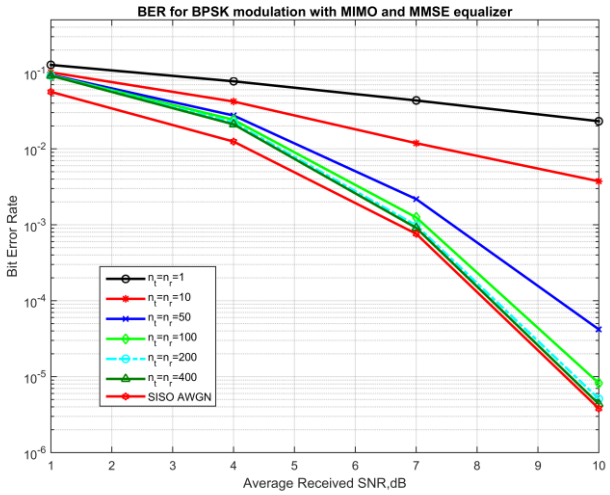

Figure 6.4: BER performance of the proposed decoding algorithm with MMSE initial vector, BPSK.

Figure 6.5 shows the BER performance comparison of the proposed detector with MMSE initial vector and ZF initial vector.

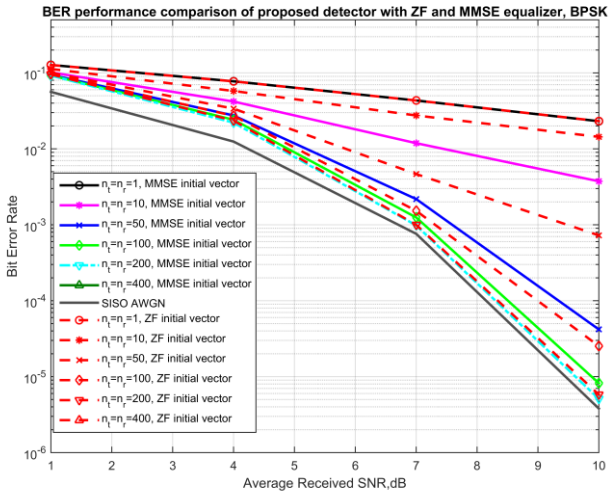

Figure 6.5: BER performance comparison of proposed detector with ZF and MMSE initial vector, BPSK.

It can be seen that, when the initial vector is generated by MMSE, the BER performance of the proposed detector improves. Using MMSE initial vector, the proposed detector yields the best result. However, using high number of antennas, i.e., greater than 400, there is no difference between MMSE or ZF.

Figure 6.6 shows the BER performance of the proposed detector with BPSK modulation. Following the procedure in [7], we randomly generate 5 initial vectors, to run 5 parallel iterations yielding 5 separate outputs $\boldsymbol{s_k}|k = 1,2,..,5$ at the end.

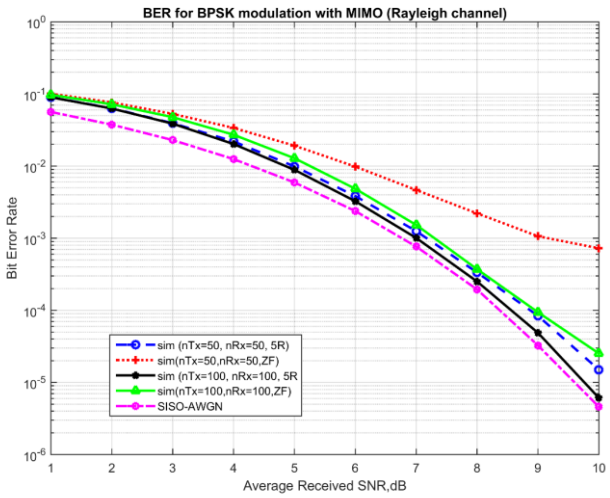

Figure 6.6: The BER performance of the proposed detector for BPSK modulation with random initial vector.

Our simulations show that the BER performance of our algorithm with random initial vectors has better performance than ZF, and MMSE. The computational cost increases linearly with the number of initial vectors. However, with randomly generated initial vectors, there is no need any inverse matrix calculations. It can be noticed from Figure 6.6, beyond 6 dB, the BER performance of our algorithm and the ML performance (approximated as SISO AWGN) are almost the same.

The Energy function in equation (4.5) shows that diagonal elements of the weight matrix affect to the performance of the network. Hopfield neural networks are designed with symmetric weight matrix and diagonal elements of the weight matrix are zero in order to guarantee the stability of the network. If diagonal elements are greater than zero, the energy function will decrease further but there is no guarantee that Hopfield network reaches to a stable state.

In order to find a local minimum of the energy function in an annealing schedule, the Hopfield network is rearranged. The update algorithm is first run without setting the diagonal elements of weight matrix zero. Then, diagonal elements are set to zero gradually. The Hopfield networks reach a stable state after at most n^2 update, where n is the number of neurons used in the network. Hence, after n^2 time interval, the algorithm is run by setting all diagonal elements zero in order to guarantee the stability of the network.

Figure 6.7 shows that, the algorithm with full diagonal achieves better performance than the algorithm with zero diagonal. It can be seen that performance improvement is greater for lower antenna configurations.

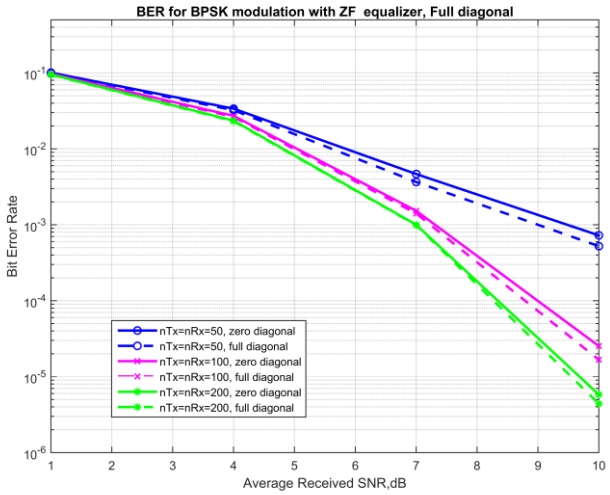

Figure 6.7: The BER performance of the proposed detector for BPSK modulation with ZF initial vector, Full Diagonal.

Figure 6.8 shows the BER performance for BPSK modulation. We generate initial vector using iterative method given in (5.11). The BER performance of this method can be compared to the method using initial vector generated by ZF. Both methods give same results. Generating initial vector by iterative method, there is no need for any inverse matrix calculation.

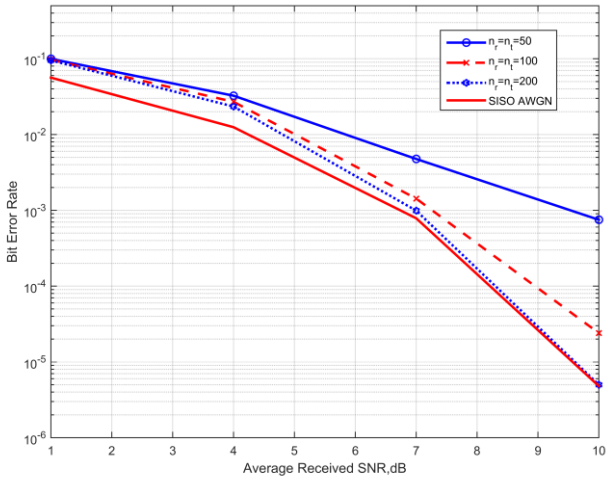

Figure 6.8: The BER performance of the proposed detector with BPSK modulation, initial vector generated by iterative method.

Figure 6.9 shows the BER performance for 4QAM modulation. We generate initial vector using MMSE.

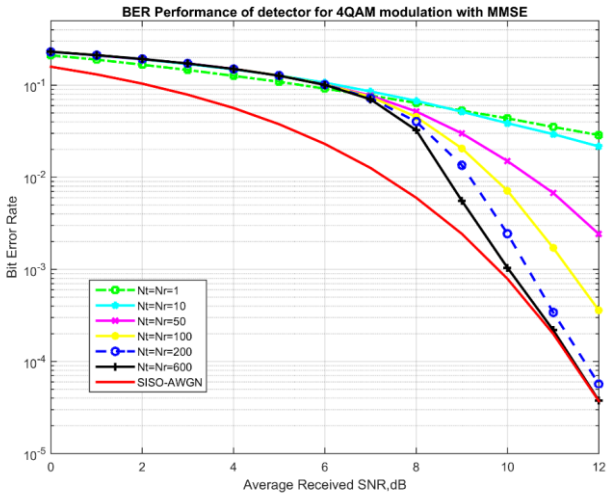

Figure 6.9: The BER performance of the proposed detector for 4QAM modulation with MMSE initial filter.

The BER performance of the proposed algorithm increases when the number of antennas (variables) increase. It can be noticed from Figure 6.9 that, as the number of antennas (variables) increase, the BER performance improves. After 200 antennas, the performance improvement is not significant, and the BER curve approaches SISO curve. These results are comparable to the LAS results given in Figure 2 of [4].

Figure 6.10 shows that the BER performance of the proposed detector improves when the number of antennas used in MIMO system increases.

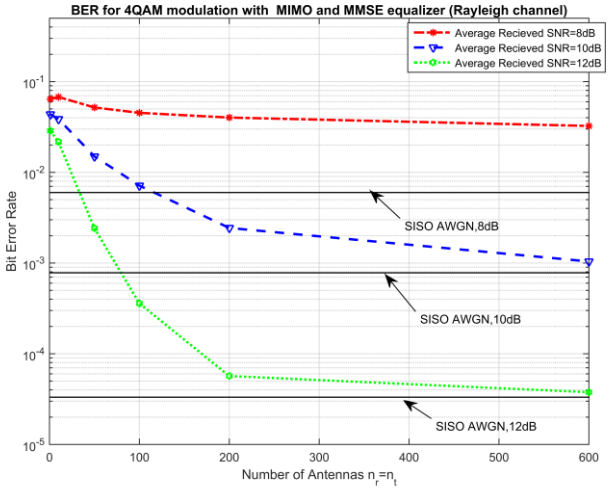

Figure 6.10: The BER performance of the proposed detector with 4QAM modulation in terms of number of antennas, MMSE.

For 8dB average received SNR, the BER performance of the proposed detector with 600x600 MIMO converges the BER performance of SISO AWGN. Furthermore, for 12dB average received SNR, when more than 200 antennas are used in MIMO system, the BER performance of the proposed decoding algorithm converges to performance of SISO AWGN.

Figure 6.11 shows the BER performance of the proposed detector with 16-QAM modulation. MMSE filter is used to generate initial vector. The value of each entry of $\boldsymbol{x}$ is written as a linear combination of its bits using Equation 5.13. In Figure 6.11, it can be seen that BER performance of the proposed detector is improved with increasing number of antennas. To perform same target BER, the proposed detector with 16-QAM need more SNR than BPSK and 4-QAM.

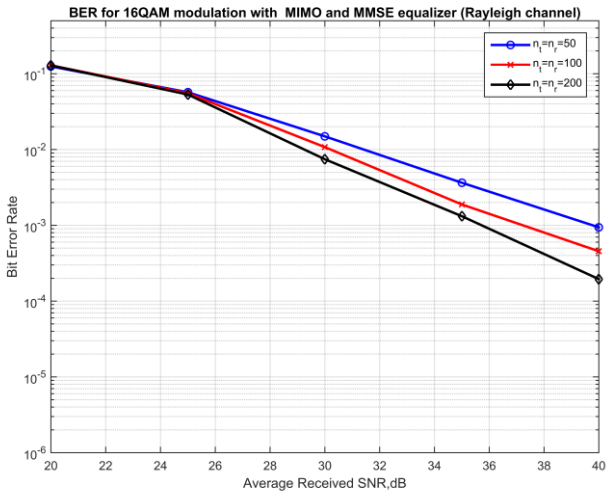

Figure 6.11: BER performance of the proposed detector with 16-QAM modulation using Equation 5.13.

Figure 6.12 show the BER performance of the proposed detector with 16-QAM. In order to generate initial vector, MMSE filter is used. Each entry of $\boldsymbol{x}$ is written as a linear combination of its bits using Equation 5.21. With increasing number of antennas, BER performance of the proposed detector is improved. In Figure 6.13, BER performance comparison between the proposed detector using Equation 5.13 and Equation 5.21 is given. It can be seen that the latter algorithm achieve better BER performance than the former algorithm.

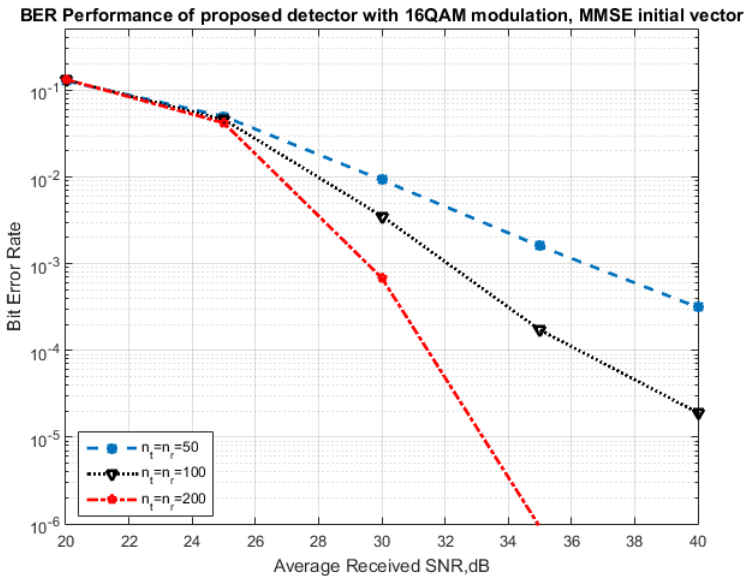

Figure 6.12: The BER performance of the proposed detector with 16QAM, using Equation 5.21.

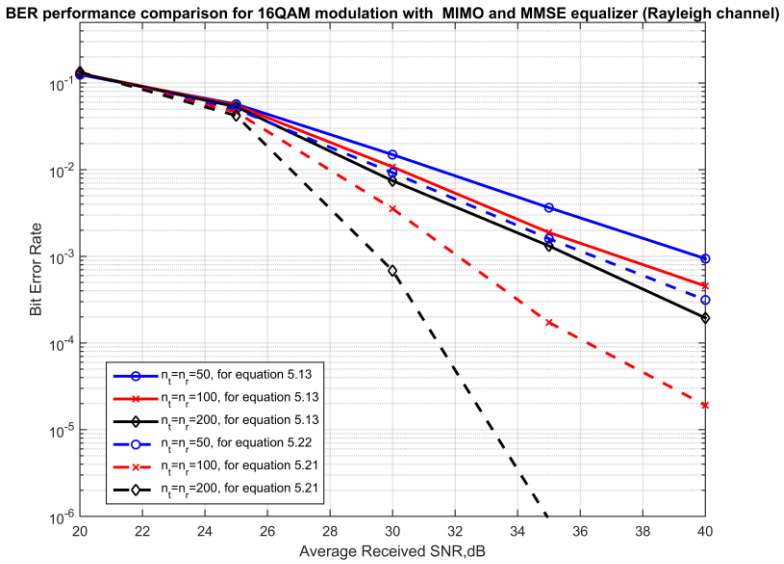

Figure 6.13: The BER performance comparison between the proposed detector using Equation 5.13 and 5.21.

Figure 6.14 shows average received SNR required to achieve a target BER of 10^{-4} in MIMO for 16-QAM proposed detector with MMSE initial vector. When the number of antennas increases, the BER performance of the proposed detector is improved. It can be seen that, the BER performance approach to SISO AWGN performance, when 1000 or higher number of antennas are used. These results are comparable to the BER the LAS results given in Figure 4 of [4].

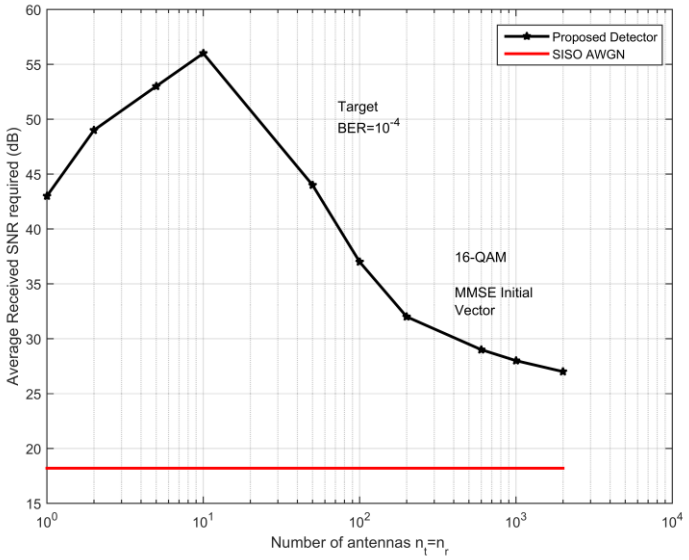

Figure 6.14: Average Received SNR required to achieve a target BER 10^{-4} in terms of number of antennas.

Consequently, it can be seen that the BER performance of the proposed detector improves when number of antennas used in MIMO system increases and it is converges to SISO AWGN performance when large number of antennas are used. For high order modulations, larger number of antennas should be used at transmitter and receiver in order to converge to SISO AWGN BER performance. The computational complexity of the proposed detector is lesser than the computational complexity of ML. It has lower algorithmic and computational complexity compared to LAS algorithm.

7. CONCLUSION

In this thesis, the benefit of using MIMO systems are investigated and it is shown that, in order to achieve full potential of MIMO systems, very large number of antennas should be used. In this system, which is called large MIMO, decoding is one of the challenging problems. Decoding algorithms should achieve good BER performance and should have low computational complexity. There are various decoding methods in the literature. The optimum method in term of minimizing the cost function is ML decoding algorithm. Due to the computational complexity of this algorithm, it is not feasible to use ML decoding in large MIMO systems. On the other hand, low complexity decoding algorithms, i.e., ZF, MF, MMSE, are not well scaled to use in large MIMO due to their poor BER performance. There are some decoding algorithms in the literature which have low complexity and high BER performance. LAS algorithm is one of these decoding methods. The LAS has lower computational complexity than ML decoding algorithm and its BER performance converges to performance of ML when large number of antennas are used.

We show that ML decoding of M-QAM modulated symbols can be formulated as a binary optimization problem and may be solved via discrete-time Hopfield neural networks. We define a new algorithm whose BER performance converges to SISO AWGN performance when large number of antennas are used in the transmitter and the receiver. The proposed decoding algorithm has lower computational complexity than ML decoding method. Additionally, there is no necessity to conditional evolution, comparison to multiple thresholds and multi-level quantizations for updating in the each search stage.

We demonstrate through simulations that in very large MIMO, where ML solution is impossible due to its exponential complexity, very accurate ML solutions can be obtained using discrete HNNs. The BER performance of the proposed detector improves when number of antennas used in MIMO system increases and it is converges to SISO AWGN performance when large number of antennas are used. Our

simulation results are comparable to LAS algorithm with lower algorithmic and computational complexity.

Moreover, our findings can be extended to realizations by continuous-time HNNs which may lead to real-time decoding in very large MIMO systems. We will pursue that in future research.

REFERENCES

[1] L. Azzam and E. Ayanoglu, "Reduced Complexity Sphere Decoding for Square QAM via a New Lattice Representation," in *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, 2007, pp. 4242-4246.

[2] S. Verdu, *Multiuser Detection*: Cambridge University Press, 1998.

[3] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*: Cambridge University Press, 2005.

[4] A. C. S. Mohammed, and B. Rajan. "Asymptotic analysis of the performance of LAS algorithm for large MIMO detection",online arXiv,. *vol. 806*.

[5] S. K. Mohammed, A. Chockalingam, and B. S. Rajan, "A Low-Complexity Near-ML Performance Achieving Algorithm for Large MIMO Detection," *2008 IEEE International Symposium on Information Theory Proceedings, Vols 1-6,* pp. 2012-2016, 2008.

[6] K. V. Vardhan, S. K. Mohammed, A. Chockalingam, and B. S. Rajan, "A low-complexity detector for large MIMO systems and multicarrier CDMA systems," *IEEE Journal on Selected Areas in Communications,* vol. 26, pp. 473-485, Apr 2008.

[7] P. Li and R. D. Murch, "Multiple Output Selection-LAS Algorithm in Large MIMO Systems," *IEEE Communications Letters,* vol. 14, pp. 399-401, May 2010.

[8] L. Heng Siong and B. Venkatesh, "An efficient local search heuristics for asynchronous multiuser detection," *Communications Letters, IEEE,* vol. 7, pp. 299-301, 2003.

[9] T. Peng Hui and L. K. Rasmussen, "Multiuser detection in CDMA - a comparison of relaxations, exact, and heuristic search methods," *Wireless Communications, IEEE Transactions on,* vol. 3, pp. 1802-1809, 2004.

[10] T. Peng Hui and L. K. Rasmussen, "A reactive tabu search heuristic for multiuser detection in CDMA," in *Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on*, 2002, p. 472.

[11] Z. Hui, L. Hang, and W. Wenbo, "Tabu Search Detection for MIMO Systems," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, 2007, pp. 1-5.

[12] F. Glover, "Tabu search-Part I," *ORSA J. Computing,* vol. 1, pp. 190-206, Summer 1989.

[13] F. Glover, "Tabu Search-Part II," *ORSA J. Computing,* vol. 2, pp. 4-32, Winter 1990.

[14] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Cont. Syst. Mag.,* vol. 29, pp. 82-100, Dec. 2009.

[15] Y. Bar-Shalom and T. E. Fortmann, "Tracking and Data Association," *San Diego, CA: Academic,* 1988.

[16] Y. Bar-Shalom and X. R. Li, "Estimation and Tracking: Principles, Techniques and Software," *Dedham, MA: Artech House,* 1993.

[17] K. Sengar, N. Rani, A. Singhal, D. Sharma, S. Verma, and T. Singh, "Study and Capacity Evaluation of SISO, MISO and MIMO RF Wireless Communication Systems," *International Journal of Engineering Trends and Technology (IJETT),* vol. 9, March 2014.

[18] A. Chockalingam and B. S. Rajan, *Large MIMO Systems*: Cambridge University Press, 2014.

[19] T. Brown, P. Kyritsi, and E. De Carvalho, *Practical Guide to MIMO Radio Channel*. Hoboken, NJ, USA: John Wiley & Sons, 2012.

[20] Z. Lizhong and D. N. C. Tse, "Diversity and multiplexing: a fundamental tradeoff in multiple-antenna channels," *Information Theory, IEEE Transactions on,* vol. 49, pp. 1073-1096, 2003.

[21] H. Jafarkhani, *Space-Time Coding: Theory and Practice*: Cambridge University Press, 2005.

[22] R. Heath, J. Paulraj, and A. Paulraj, "Switching between multiplexing and diversity based on constellation distance," in *Allerton Conf. Communication, Control and Computing*, Oct. 2000.

[23] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal, The,* vol. 27, pp. 379-423, 1948.

[24] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Trans. Telecommun.,* vol. 10, pp. 585-595, November 1999.

[25] G. J. Foschini and M. J. Gans, "On Limits of Wireless Communications in a Fading Environment when Using Multiple Antennas," *Wireless Personal Communications,* vol. 6, pp. 311-335, 1998/03/01 1998.

[26] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Technical Journal,* vol. 1, pp. 41-59, 1996.

[27] A. J. Goldsmith, "The capacity of downlink fading channels with variable rate and power," *Vehicular Technology, IEEE Transactions on,* vol. 46, pp. 569-580, 1997.

[28] A. J. Goldsmith and C. Soon-Ghee, "Variable-rate variable-power MQAM for fading channels," *Communications, IEEE Transactions on,* vol. 45, pp. 1218-1230, 1997.

[29] A. J. Goldsmith and P. P. Varaiya, "Capacity of fading channels with channel side information," *Information Theory, IEEE Transactions on,* vol. 43, pp. 1986-1992, 1997.

[30] A. J. Goldsmith and C. Soon-Ghee, "Adaptive coded modulation for fading channels," *Communications, IEEE Transactions on,* vol. 46, pp. 595-602, 1998.

[31] T. K. Y. Lo, "Maximum ratio transmission," *Communications, IEEE Transactions on,* vol. 47, pp. 1458-1461, 1999.

[32] T. L. Marzetta, G. Caire, M. Debbah, I. Chih-Lin, and S. K. Mohammed, "Special issue on Massive MIMO," *Communications and Networks, Journal of,* vol. 15, pp. 333-337, 2013.

[33] T. L. Marzetta, "Multi-cellular wireless with base stations employing unlimited numbers of antennas," *in Proc. UCSD Inf. Theory Applicat. Workshop,* Feb. 2010.

[34] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas " *IEEE Trans. Wireless Commun.,* vol. 9, pp. 3590-3600, Nov. 2010.

[35] K. Kyung Jun, C. Kyung Jun, L. Seong Ro, and K. Kwang Soon, "Multi-user massive MIMO for next-generation WLAN systems," *Electronics Letters,* vol. 51, pp. 792-794, 2015.
[36] S. Yang and L. Hanzo, "Fifty Years of MIMO Detection: The Road to Large-Scale MIMOs," *Communications Surveys & Tutorials, IEEE,* vol. PP, pp. 1-1, 2015.
[37] H. Q.Ngo, E.G.Larsson, and T. L.Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.,* vol. 61, pp. 1436-1449, Apr. 2013.
[38] A. M. Tulino and S. Verdu, "Random Matrix Theory and Wireless Communications," *Now Publishers,* 2004.
[39] T. Kailath, H. Vikalo, and B. Hassibi, ""MIMO receive algorithms," in Space-Time Wireless Systems: From Array Processing to MIMO Communications, H. Bolcskei, D. Gesbert, C. B. Papadias, and A. jan van der veen, Eds. Cambridge, UK: Cambridge University Press, 2006, ch. 15."
[40] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *Signal Processing, IEEE Transactions on,* vol. 53, pp. 2806-2818, 2005.
[41] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *Information Theory, IEEE Transactions on,* vol. 45, pp. 1639-1642, 1999.
[42] J. J. Hopfield and D. W. Tank, ""Neural computation of decisions in optimization problems," Biol. Cybern., vol. 52, no. 3, pp. 141-146, ," Jul. 1985.
[43] J. J. Hopfield, ""Neural networks and physical systems with emergent collective computational abilities" in Proc. Nat. Acad. Sci., vol. 79, no.8, pp. 2554-2558," 1982.
[44] M. T. Hagan, H. B. Demuth, and M. Beale, "Neural Network Design, Boston, MA, USA: PWS Publishing Company," 1996.
[45] Y. T. Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins, "Image-Restoration Using a Neural Network," *IEEE Transactions on Acoustics Speech and Signal Processing,* vol. 36, pp. 1141-1151, Jul 1988.
[46] J. K. Paik and A. K. Katsaggelos, "Image restoration using a modified Hopfield network," *IEEE Transactions on Image Processing,* vol. 1, pp. 49-63, Jan 1992.
[47] Y. D. Wu, Y. Sun, H. Y. Zhang, and S. X. Sun, "Variational PDE based image restoration using neural network," *IET Image Processing,* vol. 1, pp. 85-93, Mar 2007.
[48] A. Cichochi and R. Unbehauen, "Neural networks for optimization and signal processing," *John Wiley & Sons, New York,* 1993.
[49] R. Rojas, "Neural networks-a systematic introduction," *Springer,* 1996.
[50] G. Joya, "Contribución de las redes neuronales artificales de alto orden al diseño de sistemas autónomos.," *Doctoral thesis. Ed. SPICUM,* 1997.
[51] J. Bruck, "On the convergence properties of the Hopfield model," *Proc. IEEE (Special Issue on neural networks analysis, techniques and applicatons, Eds. C. Lau and B. Widrow,* pp. 1579-1585, Oct. 1990.
[52] X. J. Zhuo and X. S. Zhang, "Hopfield-type neural network for solving four-coloring map problems," *OR Transactions,* vol. 3, pp. 35-43, 1999.

CURRICULUM VITAE

Name Surname	**:** Ergin ASLAN
Place and Date of Birth	**:** İzmir-24.06.1988
E-Mail	**:** aslanerg@itu.edu.tr

EDUCATION

B.Sc.	**:** Mersin University-2010