

**DESIGNING BLIND SOURCE SEPARATION
ALGORITHMS FOR HIGH ORDER QAM
SIGNALS IN MIMO SYSTEMS**

BY

SYED AWAIS WAHAB SHAH

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

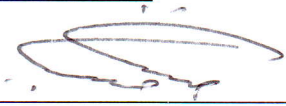
NOVEMBER 2015

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **SYED AWAIS WAHAB SHAH** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**.

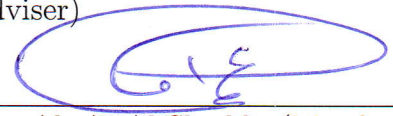
Thesis Committee



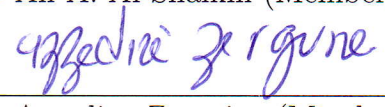
Dr. Tareq Y. Al-Naffouri (Adviser)



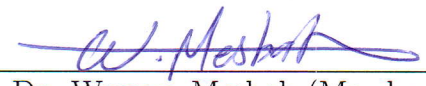
Dr. Karim Abed-Meraim (Co-adviser)



Dr. Ali A. Al-Shaikhi (Member)




Dr. Azzedine Zerguine (Member)



Dr. Wessam Mesbah (Member)



Dr. Ali A. Al-Shaikhi
Department Chairman



Dr. Salam A. Zummo
Dean of Graduate Studies



16/12/15
Date

© Copyright by Syed Awais Wahab Shah 2015
All Rights Reserved

*“My parents taught me to never give up and to always believe that my future
could be whatever I dreamt it to be.”*

Susana Martinez

To My Parents,
for their Endless Love and Prayers.

ACKNOWLEDGEMENTS

I am truly grateful to Allah *subhanahu-wa-ta'ala*, the most Compassionate and the most Merciful, who blessed me with the capability to complete this work. I pray that this work and the growth that I had during my graduation will be used in His cause.

I would like to thank KFUPM for providing me the opportunity to study at this prestigious institute. KFUPM with its highly professional faculty polished my research skills. I also acknowledge the Deanship of Scientific Research (DSR) at KFUPM and King Abdullah University of Science and Technology (KAUST) for partly funding my research work.

I feel really blessed to have the most kind-hearted advisor Dr. Tareq Y. Al-Naffouri. He is the one who developed in me the quality of keep trying and to never give up. From time to time he pushed me hard to achieve my goals and raised my standards. For me, he is a splendid human being and a man with the best management skills. I also thank Dr. Tareq for providing me excellent research facilities at KAUST and for the support that he provided throughout my research work. It is because of him, I visited Technical University of Delft (TU Delft), which was the most exciting and fun part of my life.

I would like to express my sincere gratitude to my co-advisor Dr. Karim Abed-

Meraim for the endless hours of guidance that I received from him. He is the one who with his immense knowledge showed me the magic of mathematical tools and filled me with a desire of conducting good and valuable research. For me, he is a great Muslim and a kind-hearted human being. Without his efforts and guidance, it would not have been possible to complete this work. I also thank him for inviting me to University of Orléans and hosting me at his apartment during my visit there.

I would like to thank my committee members, Dr. Azzedine Zerguine, Dr. Wessam Mesbah and Dr. Ali A. Al-Shaikhi for their suggestions that has helped in sharpening the presentation of this work. I also thank Dr. Wessam Mesbah for helping me in finding a job.

I would like to thank my friends at KFUPM, KAUST, and TU Delft for their love, encouragement and support. Specifically, I thank Anum Ali and Mudassir bhai who made my stay at KAUST and Australia during ICASSP, enjoyable and never forgetting. I also thank Mohammed Tamim, who has always been a nice friend and helped me a lot in dealing with management issues at KFUPM during my stay at KAUST. It would be impossible for me to forget the time I spent with some of my KFUPM fellows during courses, playing games and visiting places.

I am also thankful to my family for their continuous support, prayers and encouragement, which helped a lot in completing this degree. I cannot express in words their contributions in my life.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
LIST OF ABBREVIATIONS	xii
ABSTRACT (ENGLISH)	xiv
ABSTRACT (ARABIC)	xvi
CHAPTER 1 INTRODUCTION	1
1.1 Preliminaries	2
1.1.1 Constant Modulus Signals	4
1.1.2 Multimodulus Signals	5
1.1.3 Batch BSS Algorithms	6
1.1.4 Adaptive BSS Algorithms	6
1.2 Literature Review	6
1.3 Motivation and Research Problem	9
1.4 Notations	10
1.5 Outline of the Thesis	11
CHAPTER 2 SYSTEM MODEL AND BSS PRINCIPLES	12
2.1 System Model	13

2.2	Blind Source Separation (BSS)	15
2.2.1	Assumptions	15
2.2.2	BSS Approach	15
2.2.3	Batch BSS System Model	16
2.2.4	Indeterminacies in BSS and Possible Solution	17
2.2.5	Solution to BSS Problem	19
CHAPTER 3 MULTIMODULUS ALGORITHMS		24
3.1	Review of Givens and hyperbolic rotations	25
3.1.1	Givens Rotations	25
3.1.2	Hyperbolic Rotations	26
3.2	Multimodulus (MM) Cost Function	27
3.3	MM Algorithms Design	27
3.3.1	Motivation for using Real Givens and Hyperbolic Transformations	28
3.3.2	Givens MMA (G-MMA)	30
3.3.3	Hyperbolic G-MMA (HG-MMA)	35
3.4	Simulation Results	42
3.4.1	Experiment 1: Exact vs. Approximate Solution of HG-MMA	44
3.4.2	Experiment 2: Finding Optimum Number of Sweeps . . .	45
3.4.3	Experiment 3: Comparison of Rate of Convergence	46
3.4.4	Experiment 4: Effect of the Number of Samples	46
3.4.5	Experiment 5: Comparison based on SINR	48
3.4.6	Experiment 6: Comparison based on SER	50
3.5	Chapter Conclusions	50
CHAPTER 4 ALPHABET MATCHED ALGORITHMS		54
4.1	Alphabet Matched (AM) Functions	55
4.1.1	Li's AMA	56
4.1.2	Gauss AMA	57
4.1.3	Sinusoidal AMA	58

4.2	AM Algorithms Design	59
4.2.1	Givens AMA (G-AMA)	61
4.2.2	Hyperbolic G-AMA (HG-AMA)	71
4.3	Practical Considerations	81
4.3.1	Numerical Cost	81
4.3.2	Adaptive implementation	82
4.3.3	Complex implementation	82
4.3.4	Performance	83
4.4	Simulation Results	83
4.4.1	Experiment 1: Exact vs. Approximate Solution of G-AMA and HG-AMA	84
4.4.2	Experiment 2: Finding Optimum Number of Sweeps	86
4.4.3	Experiment 3: Comparison of Rate of Convergence	87
4.4.4	Experiment 4: Effect of the Number of Samples	87
4.4.5	Experiment 5: Comparison based on SINR	90
4.4.6	Experiment 6: Comparison based on SER	90
4.5	Chapter Conclusions	92
CHAPTER 5 CONCLUSION AND FUTURE WORK		95
5.1	Conclusions	95
5.2	Future Work	97
APPENDIX A		99
APPENDIX B		101
APPENDIX C		103
REFERENCES		105
VITAE		113

LIST OF TABLES

1.1	Literature Classification	9
1.2	Table of Notations	10
3.1	Givens MMA (G-MMA) Algorithm	35
3.2	Hyperbolic Givens MMA (HG-MMA) Algorithm	43
4.1	Parameters of AM cost functions for square QAM and $\epsilon = 0.001$.	57
4.2	Givens AMA (G-AMA) Algorithm	70
4.3	Hyperbolic Givens AMA (HG-AMA) Algorithm	80
4.4	Numerical complexity comparison of different BSS algorithms . .	81

LIST OF FIGURES

1.1	MIMO wireless communications system.	3
1.2	Constant Modulus (CM) signals	4
1.3	Multimodulus (MM) signals	5
2.1	MIMO wireless communication system model	13
2.2	BSS Block Diagram	23
3.1	Average SINR of exact and approximate solution of HG-MMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_s = 100$ and $N_{Sweeps} = 10$ considering both 16-QAM and 64-QAM.	45
3.2	Average SINR of HG-MMA and G-MMA vs. SNR for different N_{Sweeps} considering $N_t = 5$, $N_r = 7$, $N_s = 150$ and 16-QAM constellation.	46
3.3	Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA vs. N_{Sweeps} for $N_t = 5$, $N_r = 7$ and SNR = 20dB.	47
3.4	Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. the number of samples N_s for $N_t = 5$, $N_r = 7$, SNR = 30dB and $N_{Sweeps} = 10$	49
3.5	Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 10$ and different number of samples N_s	51
3.6	Average SER of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 10$ and different number of samples N_s	52

4.1	The CME term $g(x)$ vs. x for varying n and 64-QAM constellation.	60
4.2	\mathcal{J}_{AMA} vs. θ for random received pre-whitened signal after 5 sweeps of G-MMA with $N_t = 3, N_r = 5, N_s = 300$, SNR = 30dB and normalized 64-QAM constellation	64
4.3	Comparison of exact and approximated Givens AMA cost function for random received pre-whitened signal after 5 sweeps of G-MMA with $N_t = 3, N_r = 5, N_s = 300$, SNR = 30dB and normalized 64-QAM constellation	69
4.4	\mathcal{J}_{AMA} vs. γ for random received pre-whitened signal after $N_{\text{Sweeps}} = 5$ of G-MMA with $N_t = 3, N_r = 5, N_s = 300$, SNR = 30dB and normalized 64-QAM constellation	73
4.5	Comparison of exact and approximated hyperbolic AMA cost function for random received pre-whitened signal after $N_{\text{Sweeps}} = 5$ of G-MMA with $N_t = 3, N_r = 5, N_s = 300$, SNR= 30dB and normalized 64-QAM constellation	78
4.6	Average SINR of exact and approximate solution of HG-AMA and G-AMA vs. SNR for $N_t = 5, N_r = 7, N_{\text{Sweeps}} = 10$	85
4.7	Average SINR of HG-AMA and G-AMA vs. SNR for different N_{Sweeps} considering $N_t = 5, N_r = 7, N_s = 200$ and 64-QAM constellation.	86
4.8	Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. N_{Sweeps} for $N_t = 5, N_r = 7$ and SNR = 30dB.	88
4.9	Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. the number of samples (N_s) for $N_t = 5, N_r = 7$, SNR = 30dB and $N_{\text{Sweeps}} = 8$	89
4.10	Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. SNR for $N_t = 5, N_r = 7, N_{\text{Sweeps}} = 8$ and different number of samples N_s	91

4.11 Average SER of HG-AMA, G-AMA, HG-MMA and G-MMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 8$ and different number of samples N_s 93

LIST OF ABBREVIATIONS

AWGN	Additive White Gaussian Noise
BSS	Blind Source Separation
CM	Constant Modulus
CSI	Channel State Information
EVD	Eigenvalue Decomposition
FSK	Frequency-Shift Keying
LMS	Least Mean Square
LTE	Long Term Evolution
MIMO	Multiple-Input Multiple-Output
MM	Multi-Modulus
MMSE	Minimum Mean Squared Error
NMSE	Normalized Minimum Mean Squared Error
PAM	Pulse Amplitude Modulation
PSK	Phase-Shift Keying
QAM	Quadrature Amplitude Modulation
SINR	Signal to Interference and Noise Ratio

SISO	Single-Input-Single-Output
SNR	Signal to Noise Ratio
SVD	Singular Value Decomposition
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Networks

THESIS ABSTRACT

NAME: Syed Awais Wahab Shah

TITLE OF STUDY: Designing blind source separation algorithms for high order QAM signals in MIMO systems

MAJOR FIELD: Electrical Engineering

DATE OF DEGREE: November 2015

This thesis addresses the problem of blind multiple-input multiple-output deconvolution of a communication system. The main objective is to present efficient blind source separation (BSS) algorithms using as much a priori information as possible to reduce the overhead and thus increasing spectrum efficiency and data throughput. Four new iterative batch blind source separation algorithms are presented dealing with the multimodulus and alphabet matched criteria. For the optimization of these cost functions, iterative methods of unitary Givens and J-unitary hyperbolic rotations are used. Further, we show that the design of algorithm in the complex domain is quite complicated, so a special structure of real separation matrix is suggested and maintained throughout the design. A pre-whitening operation is also utilized to reduce the complexity of design problem. It

is noticed that the designed multimodulus and alphabet matched algorithms using Givens rotations gives satisfactory performance only for large number of samples. However, for small number of samples, the algorithms designed by combining Givens and hyperbolic rotations compensate for the ill-whitening that occurs in this case and thus improves the performance. Two algorithms dealing with the multimodulus criterion are presented for low order QAM signals such as 16-QAM. The other two dealing with the alphabet matched criterion are presented for high order QAM signals such as 64-QAM and 256-QAM. Proposed methods are finally compared with several BSS algorithms in terms of signal-to-interference and noise ratio, symbol error rate and convergence rate. Simulation results show that the proposed methods outperform the contemporary BSS algorithms. Moreover, out of all the currently available batch BSS algorithms and the presented ones, the alphabet matched algorithm designed by combining Givens and hyperbolic rotations is the most efficient one for higher order QAM signals such as 64-QAM and 256-QAM.

CHAPTER 1

INTRODUCTION

The advancement of technology has been pushing the limits of wireless communication system for a long time. The ever increasing requirements of a communication system compelled us to come up with new ideas. High data rate and bandwidth efficiency are the key demands of modern world. Next generation systems expect enhancements in throughput and coverage area along with the consideration of limited power and frequency spectrum. Considering these facts, the signal processing community has been in search for efficient ways that can augment the limitation of bandwidth in order to push the limits of traditional communications. One such method is the blind source separation (BSS), which will be the main topic of discussion throughout this thesis.

This chapter is organized as follows. First, we introduce some basic concepts related to blind techniques mainly blind source separation methodology. Next, we highlight the motivation behind this work and the key contributions. The last section includes used notations and the overview of various chapters of the thesis.

1.1 Preliminaries

The word “blind” is used by the scientific community to refer to the techniques which solely rely on the output observations. In addition to the observation symbols, some statistical information about the source symbols is also utilized for the separation of sources. However, other methods include the use of training sequences i.e., a part of the data packet and its location is already known to the receiver. Such training sequences, usually referred as “pilot symbols” are used to obtain the channel state information (CSI) and thus the source symbols. However, these pilot symbols reduce the bandwidth efficiency, for example in GSM standard, training sequences consume 18% of the total bandwidth [1] and in IEEE 802.11n standard, 4 subcarriers are reserved for pilot symbols which occupies about 7.1% of the available bandwidth [2]. Also, for the transmission of small data packets, blind techniques are desired as they do not require any training and thus reduce the overhead. Blind techniques are also favourable for the systems involving rapidly varying channels, where the number of training sequence transmission must be increased, which may be infeasible in some cases [3]. In short, the key benefit of blind techniques lies in the increase of spectrum efficiency as it does not rely on training sequences.

The blind source separation (BSS) problem differs from the blind equalization by the fact that the system in the former consists of several sources and receivers. Such a system is referred as multiple-input multiple-output (MIMO). Source separation as the name suggests is a method which identifies and separates the source

signals from a mixture. For further clarification, consider a MIMO wireless communication system as shown in Figure 1.1, where a number of sources (e.g., cell phones, laptop and tablet devices) transmit symbols at the same frequency band and in the same time slot. These symbols are passed through a wireless channel and linearly mixed together. The objective of the receiver containing an array of antennas is to identify and separate the symbols from each other. In case of non-blind technique, we can send some symbols at the start to estimate the channel by comparing the symbols at the output with the already known sent symbols (pilots). Once the channel is estimated, we can apply linear inversion to estimate the data symbols sent later on. However, in the case of blind techniques, the objective is to estimate the unknown channel and data symbols without using pilots. Thus, we only rely on the observed signals and utilize some statistical information about the sources.

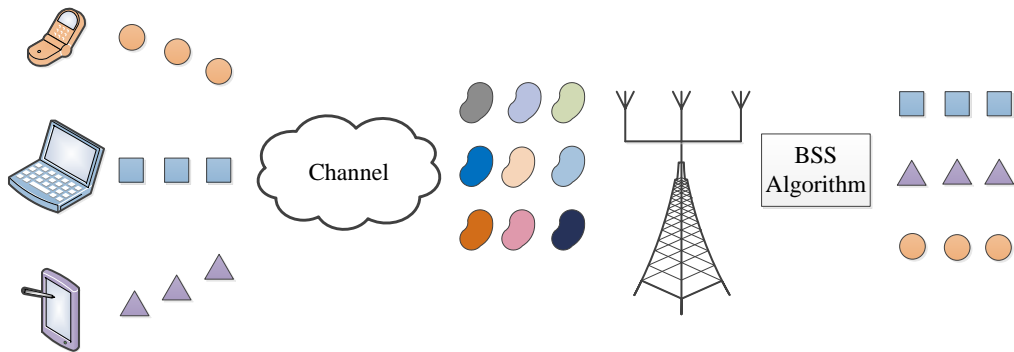


Figure 1.1: MIMO wireless communications system.

In wireless communication systems, the source signals are mostly either constant modulus (CM) or multimodulus (MM). This CM or MM nature of the source

signals is utilized in BSS algorithms, which is discussed below.

1.1.1 Constant Modulus Signals

The phase modulated signals such as phase-shift keying (PSK) and frequency-shift keying (FSK) have a constant amplitude and all of the information is usually stored in either phase or frequency. Such signals are called constant modulus signals and this property of having a constant amplitude is termed as constant modulus property. For simplicity, the amplitude is normalized to 1 for most systems. If we represent different samples of a source having this CM property on a complex plane then all samples will lie on the unit circle as shown in Figure 1.2, where Figure 1.2a shows an example of 8-PSK constellation and 16-PSK is shown in Figure 1.2b. In BSS algorithms, we can test this property for every sample of the output signal. Using this concept, a very well defined cost function is already available in literature, whose optimization results in the estimation of original sources. This will be explained in detail in Chapter 2.

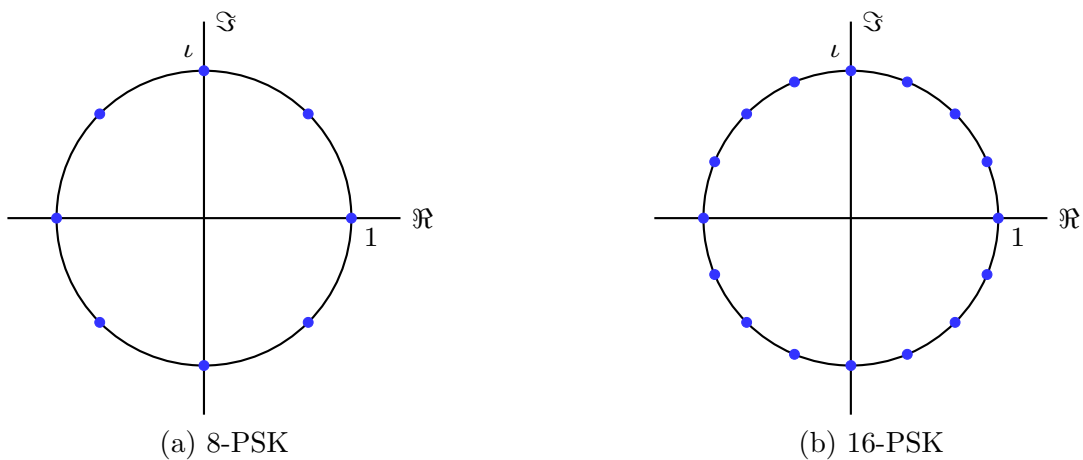


Figure 1.2: Constant Modulus (CM) signals

1.1.2 Multimodulus Signals

Signals having constellation points distributed in several circles of different radii are termed as multimodulus signals. They usually contain information in both amplitude as well as phase such as quadrature amplitude modulation (QAM) signals as shown in Figure 1.3. Figure 1.3a and 1.3b show the constellation points of square 16-QAM and 64-QAM, respectively. One can visualize these constellation points on the radius of circles with different radius. These MM signals are widely used in many modern communication systems such as LTE [4] and WiMAX [5], as they are highly spectral efficient. Considering their properties like information of amplitude as well as phase, a well defined cost function is designed in order to estimate such kind of sources, which will also be explained in detail in chapter 2.

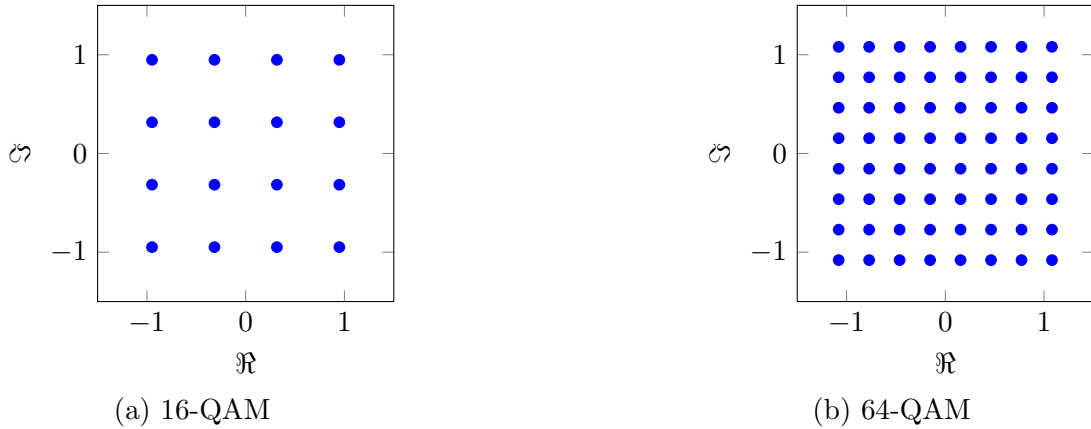


Figure 1.3: Multimodulus (MM) signals

For MIMO systems, there are two types of BSS algorithms on the basis of functioning i.e., the way the algorithm deals with samples of the source signal. The following sections include a brief overview of these types of BSS algorithms.

1.1.3 Batch BSS Algorithms

Algorithms which act on the collection of samples of signal i.e., a whole data packet, are termed as batch BSS algorithms. One has to wait for the arrival of full data packet in order to apply such algorithms. Usually, they are suitable for slowly varying channel conditions i.e., where channel changes after an arrival of at least a single data packet.

1.1.4 Adaptive BSS Algorithms

The output of adaptive algorithms gets updated after every single sample of a signal. In such algorithms, all parameters get updated continuously with the arrival of every sample, thus they are suitable for conditions where channel varies rapidly.

Next, review of BSS algorithms available in literature is presented.

1.2 Literature Review

Blind techniques started with the problem of blind equalization which was first discussed by Allen and Mazo [6] in 1974. In 1975, Sato [7] presented the first adaptive blind equalizer for pulse amplitude modulated (PAM) signals. The original Constant Modulus Algorithm (CMA) designed for CM signals was presented by Treichler and Agee [8] in 1983 for amplitude modulated and frequency modulated signals. This CMA algorithm was able to recover a single source successfully. The work on both amplitude and carrier phase recovery was started by Benveniste and

Goursat [9] in 1984. For MM signals, the design of the MM cost function and thus the Multimodulus Algorithm (MMA) was presented by Oh and Chin [10] in 1995. All of the above mentioned algorithms were presented for SISO systems which is not the focus of this thesis. Please refer to [11, 12] and [13, 14, 15] and references therein to get more details about CM and MM related blind equalization algorithms, respectively.

For MIMO systems operating on CM signals, the first attempt to design BSS algorithms was made by Treichler and Larimore [16] and Gooch and Lundell [17] in 1985 and 1986, respectively. These algorithms became known by the name of multistage CMA and CM array. The CM array is a combination of CMA and least mean square (LMS) algorithm and therefore has a poor performance. The CM array has two stages, first finding a single strongest source signal out of the number of sources and then estimation of the channel vector corresponding to that source using LMS algorithm. Once both the source signal and the channel vector are estimated, they are subtracted from the original data sequence. Then, the resulting signal passes to the next CMA stage to detect second strongest CM signal. This procedure can be repeated until all the source signals have been found. A successful method was proposed by Papadias [18] in 2000 to design a BSS algorithm for MIMO system by using CMA, known as Multiuser Kurtosis (MUK). This adaptive algorithm is able to separate all the sources at once and is also applicable to non-CM signals. It was mainly designed for statistically independent non-Gaussian signals by maximizing the value of the Kurtosis of

the output. For MIMO systems, out of the numerous implementations for CM signals, the algebraic solution named as Analytical Constant Modulus Algorithm (ACMA) is the most famous and widely used algorithm presented by van der Veen and Paulraj [19] in 1996. It provides the exact solution in the noise-free case and is capable of separating all sources in a batch mode using only a few samples. Its design is based on a generalized eigenvalue problem. Later on, van der Veen analyzed the asymptotic properties of ACMA [20] and also presented the adaptive form of this algorithm [21] in 2005. To overcome the drawback of numerical complexity of ACMA, two batch BSS algorithms namely Givens CMA (G-CMA) and Hyperbolic G-CMA (HG-CMA) were presented in 2014 by Ikhlef *et al.* [22, 23]. These algorithms outperform the ACMA. Later on, the adaptive version of G-CMA was presented by Boudjellal *et al.* [24].

For MM signals, an adaptive Multi-Modulus Algorithm (MMA) was presented for MIMO systems by Sansrimahachai *et al.* [25] in 2002, which outperforms the MUK algorithm. This algorithm can be termed as MIMO MMA. It is based on single user MMA and satisfies the orthogonality constraints using Gram-Schmidt orthogonalization procedure to ensure the independence of estimated signals. Seeing the popularity of ACMA, the same analytical approach was used for MM signals and thus an Analytical Multimodulus Algorithm (AMMA) was presented for MIMO systems by Daumont and Le Guennec [26] in 2010. The authors have presented the problem as a generalized eigenvalue problem by performing a batch analytical analysis similar to the ACMA. However, they have not solved the eigen-

value problem for the batch analytical version due to the unavailability of the joint diagonalization method for non-square matrices. So, they have only implemented the adaptive version of AMMA and showed that it outperforms the adaptive ACMA for the case of square QAM. All these algorithms designed for the MM QAM signals are adaptive but do not work well for higher order QAM constellations. Table 1.1 summarizes the literature review and also shows the unresolved research problem that we have targeted in this thesis, which is discussed in the next section.

Table 1.1: Literature Classification

BSS Algorithms	CM	MM
Batch	[19, 22, 23]	open
Adaptive	[16, 17, 18, 21, 24]	[25, 26]

1.3 Motivation and Research Problem

As per our findings, currently no efficient batch BSS algorithm exists dealing with the MM signals especially high order QAM such as 64-QAM. These QAM constellations, due to their good spectral efficiency, are popular in many modern communications systems. Also, many such systems have slowly varying channels or at least the channel changes after a packet arrival e.g., wireless local area network (WLAN). So, there is a need for efficient batch BSS algorithms which are capable of separating the sources in MIMO systems along with the efficient use of frequency spectrum i.e., without pilots. Moreover, once the batch algorithm is designed which is efficient for higher QAM signals, it can be easily converted to

adaptive version which would be suitable for rapidly varying channels.

In summary, the focus of this thesis is to design batch BSS algorithms for MM high order QAM signals used in modern MIMO communication systems. We will compare our algorithms against the state of art batch BSS algorithms such as ACMA [19], G-CMA and HG-CMA [22, 23]. We will make the comparison in terms of signal-to-interference and noise ratio (SINR), symbol error rate (SER) and convergence rate.

1.4 Notations

A list of the notation used in this thesis along with their description is given in Table 1.2.

Table 1.2: Table of Notations

Notation	Description
$(.)^T$	Transpose operation
$(.)^H$	Complex conjugate transpose (or Hermitian) operation
\mathbf{x}	Column vector
x_i	i th entry of the vector
\mathbf{X}	Matrix
x_{ij}	(i, j) -th entry of the matrix
$\hat{\mathbf{x}}$	Estimate of \mathbf{x}
x_R and x_I	Real and imaginary parts of x
\underline{x}	Pre-filtered x
$\hat{\mathbf{X}}$	Matrix \mathbf{X} with all real elements
$E[.]$	Mathematical expectation operator
$ \cdot $	Modulus function
$\ \cdot\ $	The l_2 norm of a vector
\mathbf{I}	Identity matrix

1.5 Outline of the Thesis

This thesis is organized as follows. Chapter 2 includes the system model used throughout the design of algorithms. It also presents the assumptions and principles of BSS algorithms. In Chapter 3, the design of batch BSS multimodulus algorithms is presented. It includes the description of mathematical tools used for the design of algorithms and the detail of derivations. Simulations results are also presented at the end of this chapter. Similarly, Chapter 4 presents the design, derivations and simulation results of alphabet matched algorithms for high order QAM signals. Some practical considerations related to proposed algorithms are also presented in this chapter. The major contributions of this thesis and future work are summarized in Chapter 5.

CHAPTER 2

SYSTEM MODEL AND BSS PRINCIPLES

In this chapter, we study the general framework and principles of BSS algorithms. This includes the basic assumptions and mathematical formulation of the BSS problem. This chapter mainly presents the general idea for solving the BSS problem.

First, we present the system model for a MIMO wireless communication system considered throughout this dissertation. Then, the system parameters are explained and the notations used for these parameters are introduced. Later on, we will review the general assumptions and principles related to the BSS problem. Finally, a widely used cost function for the design of the BSS algorithms is explained.

2.1 System Model

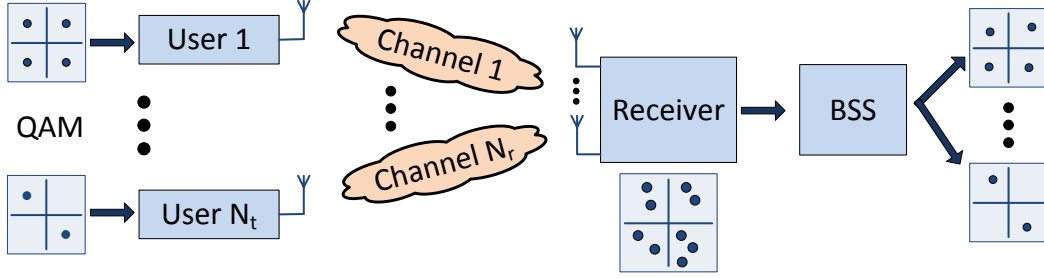


Figure 2.1: MIMO wireless communication system model

Consider Figure 2.1 representing a MIMO wireless communication system. There are N_t independent sources, each having a single antenna element. They transmit symbols at the same frequency band and in the same time slot. Each transmitted source signal $s(i) = s_R(i) + \iota s_I(i)$ is drawn from an L -ary square QAM constellation where $s_R(i), s_I(i) \in \{\pm 1, \pm 3, \dots, \pm(\sqrt{L} - 1)\}$. At instant i , the unknown transmitted source signal $\mathbf{s}(i)$ is passed through a flat fading channel which can be referred as an unknown mixing matrix \mathbf{A} . The receiver is equipped with an array of N_r antennas and therefore the received signal with the added noise can be mathematically represented by a linear system model as

$$\mathbf{y}(i) = \mathbf{A}\mathbf{s}(i) + \mathbf{n}(i) \quad (2.1)$$

where

$$\mathbf{y}(i) = \begin{bmatrix} y_1(i) \\ \vdots \\ y_{N_r}(i) \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1N_t} \\ \vdots & \ddots & \vdots \\ a_{N_r1} & \cdots & a_{N_rN_t} \end{bmatrix},$$

$$\mathbf{s}(i) = \begin{bmatrix} s_1(i) \\ \vdots \\ s_{N_t}(i) \end{bmatrix}, \quad \text{and} \quad \mathbf{n}(i) = \begin{bmatrix} n_1(i) \\ \vdots \\ n_{N_r}(i) \end{bmatrix}$$

Here

- (i) $\mathbf{s}(i) \in \mathbb{C}^{N_t \times 1}$ is the unknown source vector representing symbols transmitted at instant ‘ i ’ from N_t independent sources.
- (ii) $\mathbf{A} \in \mathbb{C}^{N_r \times N_t}$ is the unknown MIMO mixing matrix, whose elements a_{mn} denote the channel path between transmitter n and receiver m . Note that, in this work we use only flat fading channel.
- (iii) $\mathbf{n}(i) \in \mathbb{C}^{N_r \times 1}$ is the additive white noise vector.
- (iv) $\mathbf{y}(i) \in \mathbb{C}^{N_r \times 1}$ is the received vector.

The objective is to recover the transmitted source signals $\mathbf{s}(i)$ without prior knowledge of the channel or without the use of training sequences (pilots). This task is accomplished using a BSS algorithm which only relies on the observation vector $\mathbf{y}(i)$ and also uses some source’s statistical information. However, BSS algorithms suffer from some inherent ambiguities which should be taken into account as discussed next.

2.2 Blind Source Separation (BSS)

This section gives a brief overview of the underlying assumptions and methodology of the BSS algorithms.

2.2.1 Assumptions

Our BSS approach is based on the following assumptions which are generally satisfied in many communication systems.

- (i) The mixing matrix \mathbf{A} is full column rank which implies that $N_r \geq N_t$.
- (ii) The sources are zero mean, independent and identically distributed (i.i.d.) and discrete-valued or more generally sub-Gaussian signals. Sub-gaussian sources are signals with negative valued kurtosis. Note that all communication signals satisfy this condition.
- (iii) The added noise is white Gaussian with zero mean and covariance matrix $\mathbf{R}_n = \mathbb{E}[\mathbf{nn}^H] = \sigma_n^2 \mathbf{I}_{N_r}$, where \mathbf{I}_{N_r} is $(N_r \times N_r)$ identity matrix.

2.2.2 BSS Approach

The aim of BSS algorithms is to estimate the source signals $\mathbf{s}(i)$ blindly (i.e., without using pilots and by only relying on the received data $\mathbf{y}(i)$). This is done using a collection of filtering vectors which are referred as separation vectors and denoted in this thesis as $\mathbf{w}_j \in \mathbb{C}^{N_r \times 1}$. In order to recover N_t source signals, N_t such separation vectors are required where each separation vector when applied

on the received signal $\mathbf{y}(i)$ such that $\mathbf{w}_j^H \mathbf{y}(i) = z_j(i)$ results in an estimate of the source signal $s_j(i)$, so $z_j(i) = \hat{s}_j(i)$. These separation vectors are collected in an $(N_t \times N_r)$ separation matrix $\mathbf{W} = \begin{bmatrix} \mathbf{w}_1 & \dots & \mathbf{w}_{N_t} \end{bmatrix}^H$, which when applied on the received signal $\mathbf{y}(i)$ gives the desired output as

$$\mathbf{z}(i) = \mathbf{W}\mathbf{y}(i) = \mathbf{W}\mathbf{A}\mathbf{s}(i) + \mathbf{W}\mathbf{n}(i) = \mathbf{G}\mathbf{s}(i) + \bar{\mathbf{n}}(i) \quad (2.2)$$

where $\mathbf{z}(i) = [z_1(i), z_2(i), \dots, z_{N_t}(i)]^T$ is the $(N_t \times 1)$ vector of the estimated source signals, $\mathbf{G} = \mathbf{W}\mathbf{A}$ is the $(N_t \times N_t)$ global system matrix and $\bar{\mathbf{n}}(i) \in \mathbb{C}^{N_t \times 1}$ is the filtered noise at the receiver output. Thus, in BSS problem the objective is to find separation matrix \mathbf{W} .

As discussed in Chapter 1, there are two types of BSS algorithms and we will focus only on batch BSS algorithms in this thesis. The basic idea of batch BSS algorithm is already presented in Section 1.1.3. Now, we will present the mathematical system model for these algorithms in the following section.

2.2.3 Batch BSS System Model

In batch BSS algorithms, N_s samples of the data symbols are collected before applying the separation procedure i.e., N_s samples of the $(N_r \times 1)$ received signal $\mathbf{y}(i)$ are collected for $i = \{1, \dots, N_s\}$ in a matrix $\mathbf{Y} \in \mathbb{C}^{N_r \times N_s}$, which corresponds to the transmitted source signal matrix $\mathbf{S} \in \mathbb{C}^{N_t \times N_s}$. So the collection of received

signals and transmitted source signals can be written, respectively, as

$$\begin{aligned}\mathbf{Y} &= \begin{bmatrix} \mathbf{y}(1) & \cdots & \mathbf{y}(N_s) \end{bmatrix} \\ \mathbf{S} &= \begin{bmatrix} \mathbf{s}(1) & \cdots & \mathbf{s}(N_s) \end{bmatrix}\end{aligned}\tag{2.3}$$

We can then relate \mathbf{Y} and \mathbf{S} by a relation similar to (2.1) and relate \mathbf{Z} and \mathbf{Y} by a relation similar to (2.2), i.e.,

$$\begin{aligned}\mathbf{Y} &= \mathbf{AS} + \mathbf{N} \\ \mathbf{Z} &= \mathbf{WY}\end{aligned}\tag{2.4}$$

where the noise matrix $\mathbf{N} \in \mathbb{C}^{N_r \times N_s}$ and the receiver output $\mathbf{Z} \in \mathbb{C}^{N_t \times N_s}$ are defined in a way similar to the definition of \mathbf{S} in (2.3).

2.2.4 Indeterminacies in BSS and Possible Solution

The BSS method allows the recovery of the source signals up to a possible permutation (i.e., ordering of sources is arbitrary) and scaling factor (i.e., arbitrary phase) [23] i.e.,

$$\mathbf{WA} = \mathbf{P}\mathbf{\Lambda}\tag{2.5}$$

where $\mathbf{P} \in \mathbb{C}^{N_t \times N_t}$ is a permutation matrix and $\mathbf{\Lambda} \in \mathbb{C}^{N_t \times N_t}$ is a non-singular diagonal matrix. Thus, (2.5) shows the scale indeterminacies present in BSS problem. These ambiguities can be removed, when required, using a short training sequence and a method explained below [27].

Solution of Permutation Ambiguity

Consider K symbols are inserted in every source data packet \mathbf{S} at its start as pilot symbols and let us denote this known part of data packet by $\mathbf{c}_j \in \mathbb{C}^{1 \times K}$, where $j \in \{1, \dots, N_t\}$ denotes the source number. Similarly, let us collect the first K estimated symbols from the estimated matrix \mathbf{Z} into the $(1 \times K)$ vector \mathbf{x}_k , where $k \in \{1, \dots, N_t\}$ denotes the estimated source number. Now, we define the normalized minimum mean squared error (NMSE) as

$$\text{NMSE}(\mathbf{x}_k, \mathbf{c}_j) = \log_{10} \left[1 - \frac{|\mathbf{x}_k \mathbf{c}_j^H|^2}{\|\mathbf{x}_k\| \|\mathbf{c}_j\|} \right] \quad (2.6)$$

Using the NMSE in (2.6) and pilot symbols \mathbf{c}_j , the permutation ambiguity is resolved by minimizing the following optimization problem [27]

$$\arg \min_{k, j \in \{1, \dots, N_t\}} \text{NMSE}(\mathbf{x}_k, \mathbf{c}_j) \quad (2.7)$$

Solution of Scale Ambiguity

Once the issue of the permutation ambiguity is resolved, the scale ambiguity can be removed by searching for the complex variable α which minimizes the minimum mean squared error (MMSE) between the pilot signal \mathbf{c}_j and the estimated source signal \mathbf{x}_j

$$\text{MMSE} = \mathbb{E} [|\mathbf{c}_j - \alpha \mathbf{x}_j|^2] \quad (2.8)$$

Note that, once the permutation ambiguity is resolved, one can be sure about the ordering of the sources that is why same subscript ‘ j ’ is used for both pilot signal \mathbf{c}_j and estimated signal \mathbf{x}_j .

2.2.5 Solution to BSS Problem

As explained in Section 2.2.2, the main objective of the BSS problem is to find the separation matrix \mathbf{W} . This section explains the procedure used to find this separation matrix. In general, BSS is a two step process including a pre-whitening operation followed by a separation process.

Pre-Whitening Operation

In most of the BSS systems, the number of receivers (sensors) N_r is larger than the number of sources N_t thus \mathbf{A} is a tall matrix. This could result in null-space solutions and in order to avoid it, a pre-whitening operation is used. This pre-whitening operation transforms the mixing matrix \mathbf{A} to an approximately unitary matrix and also reduces the dimension of \mathbf{Y} from $(N_r \times N_s)$ to $(N_t \times N_s)$. Bienvenu and Kopp [28] proved in 1983 that the dominant eigenvectors of the covariance matrix span the signal subspace, this lead to the design of pre-whitening procedure. The following procedure is based on the work of van der Veen [11] presented in 2006.

Let $\mathbf{B} \in \mathbb{C}^{N_t \times N_r}$ denote the pre-whitening matrix which is selected such that the resulting received matrix \mathbf{Y} is white. So, let $\hat{\mathbf{R}}_y = \frac{1}{N_s} \sum_{i=1}^{N_s} \mathbf{y}(i)\mathbf{y}(i)^H = \frac{1}{N_s} \mathbf{Y}\mathbf{Y}^H$ be the noisy covariance matrix of received signals with the following

eigenvalue decomposition (EVD)

$$\hat{\mathbf{R}}_y = \hat{\mathbf{U}}\hat{\Sigma}^2\hat{\mathbf{U}}^H = \begin{bmatrix} \hat{\mathbf{U}}_s & \hat{\mathbf{U}}_n \end{bmatrix} \begin{bmatrix} \hat{\Sigma}_s^2 & \\ & \hat{\Sigma}_n^2 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{U}}_s^H \\ \hat{\mathbf{U}}_n^H \end{bmatrix} \quad (2.9)$$

where $\hat{\mathbf{U}}$ is an $(N_r \times N_r)$ unitary matrix and $\hat{\Sigma}^2$ is an $(N_r \times N_r)$ diagonal matrix containing the singular values of $\mathbf{Y}/\sqrt{N_s}$. The N_t largest eigenvalues are collected into a diagonal matrix $\hat{\Sigma}_s^2$ and the corresponding eigenvectors into $\hat{\mathbf{U}}_s$, which span the signal subspace [28]. Now, the pre-whitening matrix \mathbf{B} is defined as

$$\mathbf{B} = \hat{\Sigma}_s^{-1}\hat{\mathbf{U}}_s^H \quad (2.10)$$

This pre-whitening matrix is selected in this way so that the mixing matrix \mathbf{A} can be transformed into a matrix close to unitary and the resulting received matrix \mathbf{Y} is white i.e., $\hat{\mathbf{R}}_y = \mathbf{I}_{N_t}$. Consider the economy-size singular value decomposition (SVD) of $\mathbf{A} = \mathbf{U}_A\mathbf{\Sigma}_A\mathbf{V}_A^H$, where $\mathbf{U}_A \in \mathbb{C}^{N_r \times N_t}$ is a submatrix of a unitary matrix, $\mathbf{V}_A \in \mathbb{C}^{N_t \times N_t}$ is unitary matrix and $\mathbf{\Sigma}_A \in \mathbb{C}^{N_t \times N_t}$ is diagonal and contains the singular values. Because of the scale indeterminacy inherent to the BSS problem, we can always scale the columns of \mathbf{A} and rows of \mathbf{S} to ensure $\mathbf{S}\mathbf{S}^H = \mathbf{I}_{N_t}$, then for large number of samples $\hat{\mathbf{R}}_y = \mathbf{R}_y$, which is written as

$$\mathbf{R}_y = \mathbf{A}\mathbf{A}^H + \sigma_n^2\mathbf{I}_{N_r} = \begin{bmatrix} \mathbf{U}_A & \mathbf{U}_A^\perp \end{bmatrix} \begin{bmatrix} \mathbf{\Sigma}_A^2 + \sigma_n^2\mathbf{I}_{N_t} & \\ & \sigma_n^2\mathbf{I}_{N_r - N_t} \end{bmatrix} \begin{bmatrix} \mathbf{U}_A^H \\ (\mathbf{U}_A^\perp)^H \end{bmatrix} \quad (2.11)$$

Comparing (2.9) and (2.11), we can identify $\mathbf{U}_s = \mathbf{U}_A$ and $\Sigma_s^2 = \Sigma_A^2 + \sigma_n^2 \mathbf{I}_{N_t}$.

Therefore,

$$\mathbf{B}\mathbf{A} = \Sigma_s^{-1} \mathbf{U}_s^H \mathbf{U}_A \Sigma_A \mathbf{V}_A^H = \Sigma_s^{-1} \Sigma_A \mathbf{V}_A^H = \mathbf{V}^H \quad (2.12)$$

where $\mathbf{V} = \mathbf{A}^H \mathbf{B}^H$ is $(N_t \times N_t)$ unitary if $(\Sigma_A^2 + \sigma_n^2 \mathbf{I}_{N_t})^{-1/2} \Sigma_A = \mathbf{I}_{N_t}$ or a scalar multiple of \mathbf{I}_{N_t} , which is possible if either there is no noise or if \mathbf{A} has orthonormal columns. If this is not the case, then \mathbf{A} is transformed to a matrix which is close to unitary and still better conditioned than original \mathbf{A} . The conditioning of a matrix is defined as the ratio of largest to smallest singular value and matrix is said to be well conditioned if this ratio is close to 1.

The pre-whitening matrix \mathbf{B} found using the covariance of received signals using the above defined method is then applied on the received signal. Consider for simplicity the noise free case here, so the received signal after the pre-whitening operation can be written as

$$\underline{\mathbf{Y}} = \mathbf{B}\mathbf{Y} = \mathbf{B}\mathbf{A}\mathbf{S} = \mathbf{V}^H \mathbf{S} \quad (2.13)$$

After the pre-whitening operation, the problem is reduced to finding a unitary matrix \mathbf{V} , which we will refer to as a filtered separation matrix throughout this thesis. Finding a unitary matrix is a simpler constraint as compared to finding a matrix \mathbf{W} with independent rows i.e., independent separation vectors \mathbf{w}_j , because now we only need to make sure that the rows of \mathbf{V} are orthogonal. Moreover, it is well known in signal processing community that moving to the whitened domain

increases the convergence speed. Next, we will discuss the procedure to find the filtered separation matrix \mathbf{V} .

Source Separation Procedure

A number of separation methods exist in the literature [29, 30], which are based upon the minimization of a contrast/cost function. Various cost functions can be found in the literature [13, 8, 10] depending upon the properties of source signals. Below, we will present a widely used CM cost function and later on include other cost functions as per their use in algorithm design. Once \mathbf{V} is computed, the filtering matrix can be expressed as $\mathbf{W} = \mathbf{VB}$, resulting in output

$$\mathbf{Z} = \mathbf{WY} = \mathbf{VB Y} = \mathbf{VY} = \mathbf{V V}^H \mathbf{S} = \hat{\mathbf{S}} \quad (2.14)$$

CM Cost Function

Many communication systems utilize CM signals (e.g., PSK, FSK, see Section 1.1.1). This CM property of the signal can be tested for every output symbol. Thus, using this information, one proposes to estimate \mathbf{V} by minimizing the CM cost function [8] defined as

$$\mathcal{J}_{\text{CMA}}(\mathbf{V}) = \sum_{j=1}^{N_t} \mathbb{E} \left[(|z_j(i)|^2 - R_j)^2 \right] \quad (2.15)$$

where $R_j = \mathbb{E}[|s_j(i)|^4] / \mathbb{E}[|s_j(i)|^2]$ is the dispersion constant obtained by equating (2.15) to zero and contains the statistical information of the signal. Note that for

simplicity, we assume that $R_j = R \forall j = 1, \dots, N_t$. This cost function is a positive measure of the amount that the squared modulus of the output signal deviates from a constant R . As this function only considers the modulus of the output signal, thus, it is phase blind and the output signal has a phase ambiguity. So, a phase compensator is usually required to recover the phase of the output signal.

Now, one can visualize the structure inside the BSS block of Figure 2.1, which is shown in Figure 2.2.

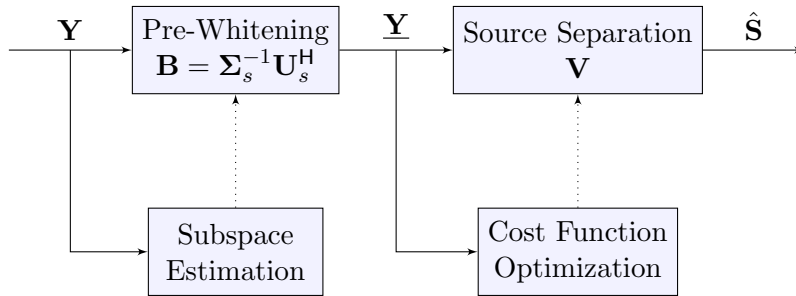


Figure 2.2: BSS Block Diagram

In the coming chapters, we will use the system model and BSS concepts defined here. The pre-whitening operation remains the same, however, we will design new techniques for finding filtered separation matrix \mathbf{V} .

CHAPTER 3

MULTIMODULUS ALGORITHMS

In this chapter, we will present two new iterative batch BSS algorithms mainly targeted for square QAM signals. These algorithms are designed on the basis of multimodulus cost function optimization using Givens and hyperbolic rotations. We will show that these algorithms outperform the contemporary batch BSS algorithms in terms of convergence rate, SINR and SER.

First of all, some of the mathematical tools that can be utilized for the optimization of cost function are presented in this chapter. Mainly, these tools are Givens and hyperbolic rotations because they are easy to use and requires less parameters to be optimized. Then, the MM cost function is reviewed for MIMO systems, which is used here for the design of algorithms. We will discuss some technical difficulties faced during the optimization of MM cost function and how we dealt with them. The major part of this chapter is dedicated for the expla-

nation of the design of MM algorithms using Givens and hyperbolic rotations. Finally, a simulation based comparison is presented to evaluate the performance of the designed algorithms.

3.1 Review of Givens and hyperbolic rotations

3.1.1 Givens Rotations

The unitary Givens rotation matrix $\mathcal{G}_{p,q}(\theta, \alpha)$ is an identity matrix except for the two diagonal elements $\mathcal{G}_{pp}, \mathcal{G}_{qq}$ and two off-diagonal elements $\mathcal{G}_{pq}, \mathcal{G}_{qp}$ in rows and columns ‘ p ’ and ‘ q ’, which are given by

$$\begin{bmatrix} \mathcal{G}_{pp} & \mathcal{G}_{pq} \\ \mathcal{G}_{qp} & \mathcal{G}_{qq} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & e^{\alpha} \sin(\theta) \\ -e^{-\alpha} \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (3.1)$$

where $\theta \in [-\pi/2, \pi/2]$ and $\alpha \in [-\pi/2, \pi/2]$ are the Givens rotation angle parameters. For the case of real transformations, $\alpha = 0$. Givens rotations have the following useful properties:

1. Givens rotations are unitary i.e.,

$$\mathcal{G}\mathcal{G}^H = \mathcal{G}^H\mathcal{G} = \mathbf{I}$$

2. Givens rotations preserve the norm of the vectors i.e., consider two column

vectors \mathbf{a} and \mathbf{b} related through Givens rotation as $\mathbf{b} = \mathcal{G}\mathbf{a}$, then

$$\|\mathbf{b}\|^2 \triangleq \mathbf{b}^H \mathbf{b} = \mathbf{a}^H \mathcal{G}^H \mathcal{G} \mathbf{a} = \mathbf{a}^H \mathbf{a} \triangleq \|\mathbf{a}\|^2$$

Thus, the vectors have same Euclidean norms.

3.1.2 Hyperbolic Rotations

The J-unitary Hyperbolic rotation matrix $\mathcal{H}_{p,q}(\gamma, \beta)$ is an identity matrix, similar to the Givens rotation matrix, except for the four elements \mathcal{H}_{pp} , \mathcal{H}_{qq} , \mathcal{H}_{pq} and \mathcal{H}_{qp} defined by

$$\begin{bmatrix} \mathcal{H}_{pp} & \mathcal{H}_{pq} \\ \mathcal{H}_{qp} & \mathcal{H}_{qq} \end{bmatrix} = \begin{bmatrix} \cosh(\gamma) & e^{\iota\beta} \sinh(\gamma) \\ e^{-\iota\beta} \sinh(\gamma) & \cosh(\gamma) \end{bmatrix} \quad (3.2)$$

where $\gamma \in [-\Gamma, \Gamma]$, $\Gamma > 0$ and $\beta \in [-\pi/2, \pi/2]$. Similar to the Givens rotations, in the real case, $\beta = 0$. The hyperbolic rotation matrix is J-unitary i.e.,

$$\mathcal{H}\mathbf{J}\mathcal{H}^H = \mathcal{H}^H\mathbf{J}\mathcal{H} = \mathbf{J}$$

for some diagonal matrix \mathbf{J} with ± 1 diagonal entries.

For further information related to Givens and hyperbolic rotations, one can refer to [31, 32].

3.2 Multimodulus (MM) Cost Function

For multi-modulus signals (e.g., QAM, see Section 1.1.2), one proposes to estimate the matrix \mathbf{V} by minimizing the MM criterion defined by [10] as

$$\mathcal{J}_{\text{MMA}}(\mathbf{V}) = \sum_{j=1}^{N_t} \mathbb{E} \left[(z_{j,R}^2(i) - R_R)^2 + (z_{j,I}^2(i) - R_I)^2 \right] \quad (3.3)$$

where $R_R = R_I = \mathbb{E}[|s_R(i)|^4]/\mathbb{E}[|s_R(i)|^2]$ are dispersion constants of the real and the imaginary parts, respectively. This cost function is designed in a way that minimization of it can be interpreted as fitting the signal into a square shaped signal. Thus, it contains structural information of the QAM signals and also has an inherent ability to restore the phase of the signal. Moreover, the MM cost function has several advantages over the CM one [13] and leads to: i) faster convergence algorithms [33, 34], ii) carrier phase recovery [35], iii) less undesirable minima [36] and iv) ease in hardware implementation [37].

3.3 MM Algorithms Design

Here, batch BSS algorithms based on the minimization of the MM cost function are presented, as this cost function is more suitable for QAM signals. To guarantee a fast convergence with relatively easy implementation, we propose to decompose the separation matrix \mathbf{V} into a product of elementary rotations, similar to Jacobi-like algorithms [29, 38]. The Jacobi method is generally used for the diagonalization of symmetric matrices. The idea behind it is to use a sequence of

unitary transformations such that $\mathbf{Y} \leftarrow \mathbf{G}^H \mathbf{Y} \mathbf{G}$, where each new \mathbf{Y} is more diagonal than its predecessor. We will use the same idea to find \mathbf{V} using a sequence of Givens and hyperbolic rotations, where the angle parameters are computed according to the MM cost function (3.3). So, instead of zeroing the off diagonal elements (diagonalization), our objective is to minimize the MM cost function.

For large sample sizes $N_s \gg 1$, one can assume that the data whitening using \mathbf{B} is efficient and hence the matrix \mathbf{V} is searched as a unitary one. In this case, \mathbf{V} is decomposed as a product of Givens rotations leading to Givens Multimodulus Algorithm (G-MMA). For moderate or small sample sizes, estimation of the covariance of the received signals as shown in (2.9) is inefficient. This results in ill pre-whitening and one estimates \mathbf{V} as a product of Givens and hyperbolic rotations, which results in Hyperbolic G-MMA (HG-MMA). The latter allows us to deviate from the unitary condition and hence to search \mathbf{V} in a larger space in order to improve the estimation accuracy.

3.3.1 Motivation for using Real Givens and Hyperbolic Transformations

The filtered separation matrix \mathbf{V} is an $(N_t \times N_t)$ matrix, which can be written as

$$\mathbf{V} = \begin{bmatrix} v_{11} & \cdots & v_{1N_t} \\ \vdots & \ddots & \vdots \\ v_{N_t 1} & \cdots & v_{N_t N_t} \end{bmatrix} \quad (3.4)$$

For large number of sources N_t , the difficulty to estimate \mathbf{V} increases. In order to simplify the estimation process, similar to Jacobi-like algorithms, we propose to decompose \mathbf{V} into a product of $N_t(N_t - 1)$ elementary Givens rotations, which can be written as

$$\mathbf{V} = \prod_{N_{Sweeps}} \prod_{1 \leq p, q \leq N_t} \mathcal{G}_{p,q}(\theta, \alpha) \quad (3.5)$$

where N_{Sweeps} denotes the number of iterations until convergence. Considering (3.5) and (3.1), we can notice that each elementary Givens rotation updates a (2×2) sub-matrix of matrix \mathbf{V} . Thus, the filtered separation matrix \mathbf{V} is estimated using an iterative process that utilizes sweeps and it can also be represented as

$$\mathbf{V}^n = \mathcal{G}_{p,q}(\theta, \alpha) \mathbf{V}^{n-1} \quad (3.6)$$

where $n = 1, \dots, N_{Sweeps}$ and $1 \leq p, q \leq N_t$.

In order to minimize the MM criterion given in (3.3), we only need to find optimal Givens rotation parameters (θ) and (α) to compute the desired \mathbf{V} . Consider a unitary transformation $\mathbf{Z} = \mathcal{G}_{p,q} \mathbf{Y}$, which according to (3.1) only changes rows ' $j = p$ ' and ' $j = q$ ' of \mathbf{Y} such that

$$\begin{aligned} z_{ji} &= \underline{y}_{ji} \quad \text{for } j \neq p, q \\ z_{pi} &= \cos(\theta) \underline{y}_{pi} + e^{\alpha} \sin(\theta) \underline{y}_{qi} \\ z_{qi} &= -e^{-\alpha} \sin(\theta) \underline{y}_{pi} + \cos(\theta) \underline{y}_{qi} \end{aligned} \quad (3.7)$$

As a result of the above mentioned transformation, (3.3) can be re-written as:

$$\begin{aligned} \mathcal{J}_{\text{MMA}}(\mathbf{G}_{pq}) = & \sum_{i=1}^{N_s} \left[(z_{pi,R}^2 - R_R)^2 + (z_{qi,R}^2 - R_R)^2 + (z_{pi,I}^2 - R_I)^2 + (z_{qi,I}^2 - R_I)^2 \right] \\ & + \sum_{i=1}^{N_s} \sum_{\substack{j=1 \\ j \neq p,q}}^{N_t} \left[(z_{ji,R}^2 - R_R)^2 + (z_{ji,I}^2 - R_I)^2 \right] \end{aligned} \quad (3.8)$$

where we can omit the constant terms of \mathbf{Z} independent of (θ, α) i.e., \mathbf{z}_{ji} for $j \neq p, q$. After some manipulations, the four terms of z_{pi} and z_{qi} involved in the cost function (3.8), results in a function of seven different non linear terms involving parameters (θ, α) given in Appendix A. Therefore, further analytical simplification and thus the solution of (3.8) is quite complicated. Similar is the case with hyperbolic rotations. These difficulties motivated us to come up with a different solution explained next.

Remark: Until now, we were working in the complex domain (i.e., all matrices \mathbf{Y} , \mathbf{V} etc. were complex) and in order to deal with the above mentioned challenges, we will now work in real domain. We have presented this work in [39, 40].

3.3.2 Givens MMA (G-MMA)

To deal with previously mentioned difficulties, we consider here a version of G-MMA using real matrices. Hence, the pre-whitened complex received signal matrix $\underline{\mathbf{Y}}$ is first converted into a real matrix $\hat{\underline{\mathbf{Y}}}$ containing real and imaginary parts in separate rows as defined in (3.10). Moreover, a special structure of the matrix \mathbf{V} is introduced and maintained while applying the rotations. The transformed real

received signal (neglecting noise) and output signal can now be written as

$$\underline{\mathbf{Y}} = \mathbf{V}^\top \underline{\mathbf{S}}, \quad \underline{\mathbf{Z}} = \mathbf{V} \underline{\mathbf{Y}} \quad (3.9)$$

where

$$\underline{\mathbf{Y}} = \begin{bmatrix} \underline{\mathbf{Y}}_R \\ \underline{\mathbf{Y}}_I \end{bmatrix} (2N_t \times N_s), \quad \mathbf{V} = \begin{bmatrix} \mathbf{V}_R & -\mathbf{V}_I \\ \mathbf{V}_I & \mathbf{V}_R \end{bmatrix} (2N_t \times 2N_t) \quad (3.10)$$

Similarly, $\underline{\mathbf{S}}$ and $\underline{\mathbf{Z}}$ are now $(2N_t \times N_s)$ real matrices, which can be represented in a way similar to the definition of $\underline{\mathbf{Y}}$ in (3.10). In order to find the required matrix \mathbf{V} , considering Lemma 1 of [30], the following sequence of real Givens rotations are used as a counterpart of (3.5)

$$\mathbf{V} = \prod_{N_{Sweeps}} \prod_{\substack{1 \leq p, q \leq N_t \\ p \neq q}} \mathcal{G}_{p,q}(\theta) \mathcal{G}_{p+N_t, q+N_t}(\theta) \mathcal{G}_{p, q+N_t}(\dot{\theta}) \mathcal{G}_{q, p+N_t}(\dot{\theta}) \prod_{1 \leq p \leq N_t} \mathcal{G}_{p, p+N_t}(\ddot{\theta}) \quad (3.11)$$

Remark: $\underline{\mathbf{Y}}$ obtained using the sequence of real Givens rotations shown in (3.11) corresponds to same complex \mathbf{V} given in (3.5).

The rotations $\mathcal{G}_{p,q}(\theta)$ and $\mathcal{G}_{p+N_t, q+N_t}(\theta)$ are applied successively using the same angle parameter (θ) . Similarly, the rotations $\mathcal{G}_{p, q+N_t}(\dot{\theta})$ and $\mathcal{G}_{q, p+N_t}(\dot{\theta})$ are applied with another angle parameter $(\dot{\theta})$. Note that, these rotations are applied in this way in order to preserve the structure of \mathbf{V} given in (3.10) [30]. The rotation $\mathcal{G}_{p, p+N_t}(\ddot{\theta})$ is applied to deal with the phase shift introduced by the diagonal entries of the mixing matrix \mathbf{A} .

We only need to find rotation angle parameters (θ) , $(\dot{\theta})$ and $(\ddot{\theta})$ in order to minimize the MM criterion (3.3), using above explained iterative method. Later on, we will express the MM cost function in terms of the angle parameter (θ) which is computed such that $\mathcal{J}_{\text{MMA}}(\theta)$ is minimized. Now, consider a unitary transformation $\underline{\dot{\mathbf{Z}}} = \mathcal{G}_{p,q}\underline{\dot{\mathbf{Y}}}$, which according to (3.1) only changes the rows ‘ p ’ and ‘ q ’ of $\underline{\dot{\mathbf{Y}}}$ such as

$$\begin{aligned} \dot{z}_{ji} &= \dot{y}_{ji} \quad \text{for } j \neq p, q \\ \dot{z}_{pi} &= \cos(\theta)\dot{y}_{pi} + \sin(\theta)\dot{y}_{qi} \\ \dot{z}_{qi} &= -\sin(\theta)\dot{y}_{pi} + \cos(\theta)\dot{y}_{qi} \end{aligned} \tag{3.12}$$

Similarly, the rotation $\mathcal{G}_{p+N_t, q+N_t}$ with the same angle parameter (θ) modifies the rows ‘ $p + N_t$ ’ and ‘ $q + N_t$ ’ in a similar way as shown in (3.12). Note that for simplicity, we keep the notation $\underline{\dot{\mathbf{Y}}}$ unchanged even though the matrix is modified after each rotation. Now, (3.3) can be re-written in terms of the Givens angle parameter (θ) (omitting the terms of $\underline{\dot{\mathbf{Z}}}$ that are independent of (θ) and assuming for simplicity that $R_R = R_I = R$)

$$\mathcal{J}_{\text{MMA}}(\theta) = \sum_{i=1}^{N_s} \left[(\dot{z}_{pi}^2 - R)^2 + (\dot{z}_{qi}^2 - R)^2 + (\dot{z}_{p+N_t, i}^2 - R)^2 + (\dot{z}_{q+N_t, i}^2 - R)^2 \right] \tag{3.13}$$

Let’s express (3.13) in a more compact form. Starting from (3.12) and using

following double angle identities

$$\begin{aligned}
\cos^2(\theta) &= \frac{1}{2}(1 + \cos(2\theta)) \\
\sin^2(\theta) &= \frac{1}{2}(1 - \cos(2\theta)) \\
\sin(2\theta) &= 2 \sin(\theta) \cos(\theta)
\end{aligned} \tag{3.14}$$

we can write

$$\begin{aligned}
z_{pi}^2 &= \mathbf{t}_i^\top \mathbf{v} + \frac{1}{2} \left(\underline{y}_{pi}^2 + \underline{y}_{qi}^2 \right) \\
z_{qi}^2 &= -\mathbf{t}_i^\top \mathbf{v} + \frac{1}{2} \left(\underline{y}_{pi}^2 + \underline{y}_{qi}^2 \right)
\end{aligned} \tag{3.15}$$

where

$$\begin{aligned}
\mathbf{v} &= \begin{bmatrix} \cos(2\theta) & \sin(2\theta) \end{bmatrix}^\top \\
\mathbf{t}_i &= \begin{bmatrix} \frac{1}{2}(\underline{y}_{pi}^2 - \underline{y}_{qi}^2) & \underline{y}_{pi}\underline{y}_{qi} \end{bmatrix}^\top
\end{aligned} \tag{3.16}$$

This allows to express the first two terms in (3.13) as

$$\left(z_{pi}^2 - R \right)^2 + \left(z_{qi}^2 - R \right)^2 = 2\mathbf{v}^\top \mathbf{t}_i \mathbf{t}_i^\top \mathbf{v} + 2 \left(\frac{\underline{y}_{pi}^2 + \underline{y}_{qi}^2}{2} - R \right)^2 \tag{3.17}$$

which is a sum of a quadratic form and a constant term independent of (θ) .

Similarly, the terms $z_{p+N_t,i}^2$ and $z_{q+N_t,i}^2$ can be obtained by replacing the indices ‘ p ’ and ‘ q ’ with ‘ $p + N_t$ ’ and ‘ $q + N_t$ ’, respectively in (3.15). Thus, the last two terms in (3.13) can be written as

$$\begin{aligned}
\left(z_{p+N_t,i}^2 - R \right)^2 + \left(z_{q+N_t,i}^2 - R \right)^2 &= 2\mathbf{v}^\top \mathbf{t}_i \mathbf{t}_i^\top \mathbf{v} \\
&+ 2 \left(\frac{\underline{y}_{p+N_t,i}^2 + \underline{y}_{q+N_t,i}^2}{2} - R \right)^2
\end{aligned} \tag{3.18}$$

where $\hat{\mathbf{t}}_i = \left[\frac{1}{2}(\underline{\dot{y}}_{p+N_t,i}^2 - \underline{\dot{y}}_{q+N_t,i}^2) \quad \underline{\dot{y}}_{p+N_t,i} \underline{\dot{y}}_{q+N_t,i} \right]^\top$. Combining (3.17) and (3.18), we can express $\mathcal{J}(\theta)$ as a quadratic form (up to a constant term that is irrelevant in determining the optimum value of θ)

$$\mathcal{J}_{\text{MMA}}(\theta) = \mathbf{v}^\top \sum_{i=1}^{N_s} [\mathbf{t}_i \mathbf{t}_i^\top + \hat{\mathbf{t}}_i \hat{\mathbf{t}}_i^\top] \mathbf{v} = \mathbf{v}^\top \mathbf{T} \mathbf{v} \quad (3.19)$$

which is a (2×2) eigenvalue problem and can be solved explicitly (see Appendix B). Thus, the solution $\mathbf{v}^\circ = \begin{bmatrix} v_1^\circ & v_2^\circ \end{bmatrix}^\top$ that minimizes (3.19) is given by the unit norm eigenvector of \mathbf{T} corresponding to the smallest eigenvalue and using (3.16), we can write

$$\cos(\theta) = \sqrt{\frac{1 + v_1^\circ}{2}} \quad \text{and} \quad \sin(\theta) = \frac{v_2^\circ}{\sqrt{2(1 + v_1^\circ)}} \quad (3.20)$$

Using (3.20), the computation of $\mathcal{G}_{p,q}$ and $\mathcal{G}_{p+N_t,q+N_t}$ follows directly from (3.1). The remaining Givens rotations $\mathcal{G}_{p,q+N_t}(\dot{\theta})$ and $\mathcal{G}_{q,p+N_t}(\dot{\theta})$ can be found similarly and applied successively on $\underline{\dot{\mathbf{Y}}}$ to compute the filtered separation matrix $\dot{\mathbf{V}}$ according to (3.11). The Givens rotation $\mathcal{G}_{p,p+N_t}(\ddot{\theta})$ for ‘ $p = q$ ’ can be found similarly by following the above explained method. By replacing ‘ q ’ with ‘ $p + N_t$ ’ in (3.16) and (3.17), the cost function (3.3) (with the constant terms omitted) can be written as

$$\mathcal{J}_{\text{MMA}}(\ddot{\theta}) = \mathbf{v}^\top \sum_{i=1}^{N_s} [\mathbf{t}_i \mathbf{t}_i^\top] \mathbf{v} = \mathbf{v}^\top \dot{\mathbf{T}} \mathbf{v} \quad (3.21)$$

Hence, the solution \mathbf{v}° is the least unit norm eigenvector of $\dot{\mathbf{T}}$ and $\mathcal{G}_{p,p+N_t}(\ddot{\theta})$ is

Table 3.1: Givens MMA (G-MMA) Algorithm

Initialization: $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$	
1. Pre-whitening: $\underline{\mathbf{Y}} = \mathbf{B}\mathbf{Y}$ using (2.10)	$\mathcal{O}(N_s N_r^2)$
2. Construct real matrix $\underline{\hat{\mathbf{Y}}}$ using (3.10)	
3. Givens Rotations:	$(20N_s N_t^2) + \mathcal{O}(N_s N_t)/\text{Sweep}$
for $n = 1 : N_{\text{Sweeps}}$ do	
for $p = 1 : N_t$ do	
for $q = p : N_t$ do	
if $p = q$ then	
a) Compute $\mathcal{G}_{p,p+N_t}$ using (3.21), (3.20) and (3.1) for $\dot{\theta}$	$(6N_s)$
b) $\underline{\hat{\mathbf{Y}}} = \mathcal{G}_{p,p+N_t} \underline{\hat{\mathbf{Y}}}$	$(4N_s)$
c) $\hat{\mathbf{V}} = \mathcal{G}_{p,p+N_t} \hat{\mathbf{V}}$	
else	
d) Compute $\mathcal{G}_{p,q}$ & $\mathcal{G}_{p+N_t,q+N_t}$ using (3.19), (3.20) and (3.1) for same (θ)	$(12N_s)$
e) $\underline{\hat{\mathbf{Y}}} = \mathcal{G}_{p,q} \mathcal{G}_{p+N_t,q+N_t} \underline{\hat{\mathbf{Y}}}$	$(8N_s)$
f) $\hat{\mathbf{V}} = \mathcal{G}_{p,q} \mathcal{G}_{p+N_t,q+N_t} \hat{\mathbf{V}}$	
repeat (d to f) for $(p, q + N_t)$ & $(q, p + N_t)$ using same $(\dot{\theta})$	$(20N_s)$
end if	
end for	
end for	
end for	
4. Construct complex matrix \mathbf{W} similar to \mathbf{V} using (2.14) and (3.10)	
5. Estimated Sources: $\hat{\mathbf{S}} = \mathbf{W}\mathbf{Y}$	

computed using (3.20) and (3.1). Matrix $\hat{\mathbf{V}}$ is initialized as $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$ and the overall algorithm is summarized in Table 3.1.

3.3.3 Hyperbolic G-MMA (HG-MMA)

For a small number of samples N_s , the pre-whitening operation is not effective and thus the transformed channel matrix \mathbf{A} may be far from a unitary matrix. In this case, the performance of G-MMA decreases and thus the J-unitary real hyperbolic rotations are applied alternatively along with the Givens rotations to overcome this limitation. This results in an algorithm named Hyperbolic Givens

MMA (HG-MMA). So, now the matrix $\hat{\mathbf{V}}$ can be decomposed into a product of elementary hyperbolic rotations, Givens rotations, and normalization transformation as follows

$$\hat{\mathbf{V}} = \prod_{N_{Sweeps}} \prod_{\substack{1 \leq p, q \leq N_t \\ p \neq q}} \mathbf{\Gamma}_{p,q} \mathbf{\Gamma}_{p+N_t, q+N_t} \mathbf{\Gamma}_{p, q+N_t} \mathbf{\Gamma}_{q, p+N_t} \prod_{1 \leq p \leq N_t} \mathbf{G}_{p, p+N_t}(\theta) \quad (3.22)$$

where

$$\mathbf{\Gamma}_{p,q} = \mathcal{N}_{p,q} \mathbf{G}_{p,q} \mathcal{H}_{p,q}$$

Here $\mathcal{N}_{p,q}$, $\mathbf{G}_{p,q}$ and $\mathcal{H}_{p,q}$ refer to the normalization, Givens and hyperbolic transformations, respectively. Similar to the Givens rotations, the hyperbolic rotations $\mathcal{H}_{p,q}$ and $\mathcal{H}_{p+N_t, q+N_t}$ are applied using the same parameter (γ) while $\mathcal{H}_{p, q+N_t}$ and $\mathcal{H}_{q, p+N_t}$ are applied using another same but opposite parameter ($\dot{\gamma}$) and ($-\dot{\gamma}$), respectively. We will consider dispersion parameters R_R and R_I be equal to 1 and use $\mathcal{N}_{p,q}$ for normalization which is a diagonal matrix with diagonal elements equal to one except for the two elements $\mathcal{N}_{pp} = \lambda_p$ and $\mathcal{N}_{qq} = \lambda_q$. Below we give a brief of finding the hyperbolic and the normalization transformation parameters to minimize the MM criterion (3.3).

Calculating the Hyperbolic and Givens rotations

Similar to the Givens rotations, let us consider one hyperbolic transformation $\mathbf{Z} = \mathcal{H}_{p,q} \hat{\mathbf{Y}}$, which according to (3.2) only changes the rows ‘ p ’ and ‘ q ’ of $\hat{\mathbf{Y}}$ such

as

$$\begin{aligned}
z_{ji} &= \underline{y}_{ji} \quad \text{for } j \neq p, q \\
z_{pi} &= \cosh(\gamma)\underline{y}_{pi} + \sinh(\gamma)\underline{y}_{qi} \\
z_{qi} &= \sinh(\gamma)\underline{y}_{pi} + \cosh(\gamma)\underline{y}_{qi}
\end{aligned} \tag{3.23}$$

Similarly, the rotation $\mathcal{H}_{p+N_t, q+N_t}$ uses the same parameter (γ) and modifies the rows ' $p + N_t$ ' and ' $q + N_t$ ' in a similar way as in (3.23).

Now let's see how we can represent the MM criterion (3.13) for the hyperbolic parameters. To this end, we set the dispersion parameters to 1. Moreover, as derived for the Givens rotations, using following hyperbolic double angle identities

$$\begin{aligned}
\cosh^2(\gamma) &= \frac{1}{2}(\cosh(2\gamma) + 1) \\
\sinh^2(\gamma) &= \frac{1}{2}(\cosh(2\gamma) - 1) \\
\sinh(2\gamma) &= 2 \sinh(\gamma) \cosh(\gamma)
\end{aligned} \tag{3.24}$$

we can show that

$$\begin{aligned}
z_{pi}^2 &= \mathbf{r}_i^\top \mathbf{u} + \frac{1}{2} \left(\underline{y}_{pi}^2 - \underline{y}_{qi}^2 \right) \\
z_{qi}^2 &= \mathbf{r}_i^\top \mathbf{u} - \frac{1}{2} \left(\underline{y}_{pi}^2 - \underline{y}_{qi}^2 \right)
\end{aligned} \tag{3.25}$$

where

$$\begin{aligned}
\mathbf{u} &= \left[\cosh(2\gamma) \quad \sinh(2\gamma) \right]^\top \\
\mathbf{r}_i &= \left[\frac{1}{2}(\underline{y}_{pi}^2 + \underline{y}_{qi}^2) \quad \underline{y}_{pi}\underline{y}_{qi} \right]^\top
\end{aligned} \tag{3.26}$$

Similar expressions can be derived for $z_{p+N_t, i}^2$ and $z_{q+N_t, i}^2$. Substituting these ex-

pressions in (3.13) and omitting the terms that are independent of (γ) yields

$$\begin{aligned}\mathcal{J}_{\text{MMA}}(\gamma) &= \mathbf{u}^\top \left[\sum_{i=1}^{N_s} \mathbf{r}_i \mathbf{r}_i^\top + \dot{\mathbf{r}}_i \dot{\mathbf{r}}_i^\top \right] \mathbf{u} - 2\mathbf{u}^\top \left[\sum_{i=1}^{N_s} \mathbf{r}_i + \dot{\mathbf{r}}_i \right] \\ &= \mathbf{u}^\top \mathbf{R} \mathbf{u} - 2\mathbf{u}^\top \mathbf{r}\end{aligned}\quad (3.27)$$

where $\dot{\mathbf{r}}_i = \left[\frac{1}{2}(\dot{y}_{p+N_t,i}^2 + \dot{y}_{q+N_t,i}^2) \quad \dot{y}_{p+N_t,i} \dot{y}_{q+N_t,i} \right]^\top$. The optimization problem in (3.27) can be solved using either Lagrange multiplier method (exact solution) or by taking linear approximation of hyperbolic sine and cosine around zero (approximate solution). Both methods are discussed below.

A) Exact Solution

Using the Lagrange multiplier method, the optimization problem in (3.27) can be written as

$$\min_{\mathbf{u}} \mathcal{F}(\mathbf{u}) = \mathbf{u}^\top \mathbf{R} \mathbf{u} - 2\mathbf{r}^\top \mathbf{u} \quad \text{s.t.} \quad \mathbf{u}^\top \mathbf{J}_2 \mathbf{u} = 1 \quad (3.28)$$

where $\mathbf{J}_2 = \text{diag}([1, -1])$, such a constraint is equivalent to $\cosh^2(2\gamma) - \sinh^2(2\gamma) =$

1. The Lagrangian of (3.28) can be written as

$$\mathcal{L}(\mathbf{u}, \lambda) = \mathbf{u}^\top \mathbf{R} \mathbf{u} - 2\mathbf{r}^\top \mathbf{u} + \lambda (\mathbf{u}^\top \mathbf{J}_2 \mathbf{u} - 1) \quad (3.29)$$

The solution of this Lagrangian is given by

$$\mathbf{u} = (\mathbf{R} + \lambda \mathbf{J}_2)^{-1} \mathbf{r} \quad (3.30)$$

where λ is the solution of the 4-th order polynomial equation (see Appendix C).

Using (3.30) in constraint equation results in

$$\mathbf{r}^\top (\mathbf{R} + \lambda \mathbf{J}_2)^{-1} \mathbf{J}_2 (\mathbf{R} + \lambda \mathbf{J}_2)^{-1} \mathbf{r} = 1 \quad (3.31)$$

Of the four roots of (3.31), we use the real value of λ that results in the minimum value of $\mathcal{L}(\mathbf{u}, \lambda)$. We then solve for $\mathbf{u}^\circ = [u_1^\circ, u_2^\circ]^\top$ from (3.30) and solve for the hyperbolic sine and cosine of (γ) as

$$\cosh(\gamma) = \sqrt{\frac{1 + u_1^\circ}{2}} \quad \text{and} \quad \sinh(\gamma) = \frac{u_2^\circ}{\sqrt{2(1 + u_1^\circ)}} \quad (3.32)$$

which allows us to construct the hyperbolic rotations $\mathcal{H}_{p,q}$ and $\mathcal{H}_{p+N_t, q+N_t}$ defined in (3.2).

For the remaining hyperbolic rotations $\mathcal{H}_{p,q+N_t}$ and $\mathcal{H}_{q,p+N_t}$, the optimization problem in (3.27) is conducted for another hyperbolic parameter $(\dot{\gamma})$ using

$$\mathbf{r}_i = \begin{bmatrix} \frac{1}{2}(\dot{y}_{pi}^2 + \dot{y}_{q+N_t, i}^2) & \dot{y}_{pi} \dot{y}_{q+N_t, i} \end{bmatrix}^\top$$

$$\dot{\mathbf{r}}_i = \begin{bmatrix} \frac{1}{2}(\dot{y}_{qi}^2 + \dot{y}_{p+N_t, i}^2) & -\dot{y}_{qi} \dot{y}_{p+N_t, i} \end{bmatrix}^\top$$

Then, the modified optimization problem is minimized using the same method as explained above. This provides the solution $\mathbf{u}^\circ = [\dot{u}_1^\circ, \dot{u}_2^\circ]^\top$ and we have

$$\cosh(\dot{\gamma}) = \sqrt{\frac{1 + \dot{u}_1^\circ}{2}} \quad \text{and} \quad \sinh(\dot{\gamma}) = \frac{\dot{u}_2^\circ}{\sqrt{2(1 + \dot{u}_1^\circ)}} \quad (3.33)$$

The computation of the hyperbolic rotations $\mathcal{H}_{p,q+N_t}(\dot{\gamma})$ and $\mathcal{H}_{q,p+N_t}(-\dot{\gamma})$ follows

directly from (3.33) and (3.2). Note that these rotations are applied using same but opposite hyperbolic angle parameter ($\hat{\gamma}$). It is equivalent to saying that the hyperbolic rotations $\mathcal{H}_{p,q+N_t}$ and $\mathcal{H}_{q,p+N_t}$ have same diagonal and opposite off-diagonal elements.

B) Approximate solution

In this approach, we will consider the linear approximation of hyperbolic sine and cosine around zero given by

$$\sinh(2\gamma) \approx 2 \sinh(\gamma) \quad \text{and} \quad \cosh(2\gamma) \approx \cosh(\gamma) \quad (3.34)$$

Now, let us define the elements of symmetric matrix \mathbf{R} and vector \mathbf{r} used in (3.27)

as

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \quad \text{and} \quad \mathbf{r} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \quad (3.35)$$

Using (3.26), (3.35) and neglecting the terms independent of (γ), the cost function (3.27) can be re-written as

$$\mathcal{J}_{\text{MMA}}(\gamma) = \cosh(4\gamma) \frac{r_{11} + r_{22}}{2} + \sinh(4\gamma)r_{12} - 2 \cosh(2\gamma)r_1 - 2 \sinh(2\gamma)r_2 \quad (3.36)$$

Setting the derivative of (3.36) w.r.t (γ) to zero and using (3.34), we obtain

$$\sinh(2\gamma) (r_{11} + r_{22} - r_1) - \cosh(2\gamma) (r_2 - r_{12}) = 0 \quad (3.37)$$

and thus the solution (γ) is

$$\gamma = \frac{1}{2} \operatorname{arctanh} \left(\frac{r_2 - r_{12}}{r_{11} + r_{22} - r_1} \right) \quad (3.38)$$

In a similar way, the hyperbolic rotation parameter ($\dot{\gamma}$) can be found using appropriate \mathbf{R} and \mathbf{r} as explained in section 3.3.3A. The hyperbolic rotations are computed using (3.38) and (3.2) and applied accordingly as explained in section 3.3.3A.

After applying the hyperbolic rotations, Givens rotations are applied in a similar way as explained in section 3.3.2 and then normalization rotations are applied as explained below.

Calculating the normalization transformations

The normalization transformation is applied in order to compensate for the dispersion parameters R_R and R_I . Let's consider that we have transformed only one row ' p ' of the pre-whitened complex received matrix $\underline{\mathbf{Y}}$, which corresponds to the transformation of rows ' p ' and ' $p + N_t$ ' for pre-whitened real received matrix $\underline{\dot{\mathbf{Y}}}$. In this case, the normalization transformation \mathcal{N}_p is an identity matrix except for the two diagonal elements $\mathcal{N}_{pp} = \lambda_p$ and $\mathcal{N}_{p+N_t, p+N_t} = \lambda_p$ and the MM cost function (3.3) (with the constant terms omitted) can be written as

$$\mathcal{J}_{\text{MMA}}(\lambda_p) = \sum_{i=1}^{N_s} \left[\left(\left(\lambda_p \underline{\dot{y}}_{pi} \right)^2 - 1 \right)^2 + \left(\left(\lambda_p \underline{\dot{y}}_{p+N_t, i} \right)^2 - 1 \right)^2 \right] \quad (3.39)$$

Taking the derivative of (3.39) w.r.t (λ_p) and setting the result to zero gives optimal normalization parameter as

$$\lambda_p = \sqrt{\frac{\sum_{i=1}^{N_s} \underline{y}_{pi}^2 + \underline{y}_{p+N_t,i}^2}{\sum_{i=1}^K \underline{y}_{pi}^4 + \underline{y}_{p+N_t,i}^4}} \quad (3.40)$$

The normalization matrices $\mathcal{N}_{p,q}$ and $\mathcal{N}_{p+N_t,q+N_t}$ are calculated using (3.40). They are identity matrices except for the diagonal elements $\mathcal{N}_{pp} = \mathcal{N}_{p+N_t,p+N_t} = \lambda_p$ and $\mathcal{N}_{qq} = \mathcal{N}_{q+N_t,q+N_t} = \lambda_q$, where λ_q is obtained by replacing ‘ p ’ with ‘ q ’ in (3.40). The remaining normalization matrices can be found similarly.

During simulations, we have observed that the normalization rotation is not necessary at each step and can be performed only once per sweep. In this case the normalization matrix \mathcal{N} is a diagonal matrix of dimension $2N_t$ with diagonal elements $\mathcal{N}_{pp} = \mathcal{N}_{p+N_t,p+N_t} = \lambda_p$ given as in (3.40) where $1 \leq p \leq N_t$. HG-MMA is presented in Table 3.2.

3.4 Simulation Results

In order to evaluate the performance of the proposed algorithms, simulation results are presented in this section. The system model is implemented on MATLAB software (from MathWorks Inc., Natick, MA). Due to lack of any batch BSS algorithm dealing with the MM criterion, we have shown a comparison with contemporary batch BSS algorithms such as ACMA, G-CMA and HG-CMA. As a performance

Table 3.2: Hyperbolic Givens MMA (HG-MMA) Algorithm

<p>Initialization: $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$</p> <p>Subspace projection or approximate pre-whitening using (2.10) if $N_r > N_t$ $\mathcal{O}(N_s N_t^2)$</p> <p>1. Create real matrix $\underline{\mathbf{Y}}$ using (3.10)</p> <p>2. Hyperbolic, Givens & Normalization Rotations: $(40N_s N_t^2) + \mathcal{O}(N_s N_t)$</p> <p>for $n = 1 : N_{Sweeps}$ do</p> <p> for $p = 1 : N_t$ do</p> <p> for $q = p : N_t$ do</p> <p> if $p = q$ then</p> <p> a) Apply Givens rotation using (a to c) of Table 3.1 (10N_s)</p> <p> else</p> <p> b) Compute $\mathcal{H}_{p,q}$ & $\mathcal{H}_{p+N_t,q+N_t}$ using (3.32) and (3.2) for (γ) (12N_s)</p> <p> c) $\underline{\mathbf{Y}} = \mathcal{H}_{p,q} \mathcal{H}_{p+N_t,q+N_t} \underline{\mathbf{Y}}$ (8N_s)</p> <p> d) $\hat{\mathbf{V}} = \mathcal{H}_{p,q} \mathcal{H}_{p+N_t,q+N_t} \hat{\mathbf{V}}$</p> <p> e) Apply Givens rotation using (d to f) of Table 3.1 (20N_s)</p> <p> repeat steps (b to e) for $(p, q+N_t)$ & $(q, p+N_t)$ using $(\dot{\theta}, \dot{\gamma})$ & $(\dot{\theta}, -\dot{\gamma})$, respectively (40N_s)</p> <p> end if</p> <p> end for</p> <p> end for</p> <p> f) Compute \mathcal{N} using (3.40) (6N_sN_t)</p> <p> g) $\underline{\mathbf{Y}} = \mathcal{N} \underline{\mathbf{Y}}$ (2N_sN_t)</p> <p> h) $\hat{\mathbf{V}} = \mathcal{N} \hat{\mathbf{V}}$</p> <p> end for</p> <p>4. Construct complex matrix \mathbf{W} similar to \mathbf{V} using (2.14) and (3.10)</p> <p>5. Estimated Sources: $\hat{\mathbf{S}} = \mathbf{W}\mathbf{Y}$</p>
--

measure, SINR, convergence rate and SER are used where SINR is defined as

$$\text{SINR} = \frac{1}{N_t} \sum_{j=1}^{N_t} \text{SINR}_j \quad (3.41)$$

with

$$\text{SINR}_j = \frac{|g_{jj}\mathbf{s}_j|^2/N_s}{\sum_{l,l \neq j} |g_{jl}\mathbf{s}_l|^2/N_s + \mathbf{w}_j \mathbf{R}_n \mathbf{w}_j^H} \quad (3.42)$$

where SINR_j is the signal to interference and noise ratio at the j th output with

$g_{ij} = \mathbf{w}_i \mathbf{a}_j$, where \mathbf{w}_i and \mathbf{a}_j are the i th row vector and j th column vector of

separation matrix \mathbf{W} and mixing matrix \mathbf{A} , respectively. $\mathbf{R}_n = \mathbb{E}[\mathbf{nn}^H] = \sigma_n^2 \mathbf{I}_{N_r}$ is the noise covariance matrix and \mathbf{s}_j is the $(1 \times N_s)$ source signal vector at j th input.

We consider a MIMO system consisting of 5 transmitters and 7 receivers ($N_t = 5, N_r = 7$) with the data model given in Section 2.1. Every uncoded data symbol transmitted by each source is drawn from 16-QAM and 64-QAM constellations. The resulting signals are then passed through a channel matrix \mathbf{A} , generated randomly at each Monte Carlo run with controlled conditioning and with i.i.d complex Gaussian variable entries of zero mean and unity variance. The noise variance is adjusted according to specified signal to noise ratio (SNR). Further, sources, noise and channel have the same properties as specified in Section 2.2.1. The results are averaged over 1000 Monte Carlo runs.

3.4.1 Experiment 1: Exact vs. Approximate Solution of HG-MMA

In this experiment, we compare the performance of exact and approximate solutions of HG-MMA in terms of SINR vs. SNR for 16-QAM and 64-QAM constellations. The number of sweeps N_{Sweeps} and samples N_s are set equal to 10 and 100, respectively. From Figure 3.1, we notice that both the exact and approximate solutions have the same performance for the considered constellations. Therefore, in the following simulations for the HG-MMA, we will use the approximate solution, as it is cheaper and easier to implement.

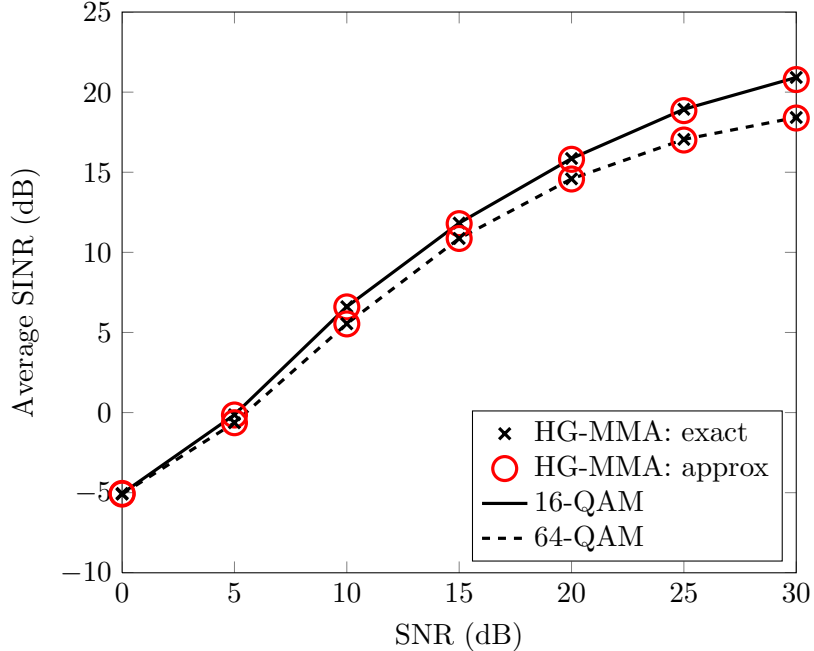


Figure 3.1: Average SINR of exact and approximate solution of HG-MMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_s = 100$ and $N_{Sweeps} = 10$ considering both 16-QAM and 64-QAM.

3.4.2 Experiment 2: Finding Optimum Number of Sweeps

Here, we examine the effect of the number of sweeps N_{Sweeps} on the performance of the G-MMA and HG-MMA. Figure 3.2 compares the SINR vs. SNR for different number of sweeps. In this simulation, $N_s = 150$ symbols are drawn from 16-QAM constellation. We notice that the performance of proposed algorithms increases with the number of sweeps and remains almost unchanged after 5 sweeps. So, in the following simulations we will fix the number of sweeps to 10. Moreover, it can be seen that for a small number of sweeps, the G-MMA performs better than the HG-MMA but after 5 sweeps HG-MMA takes the lead.

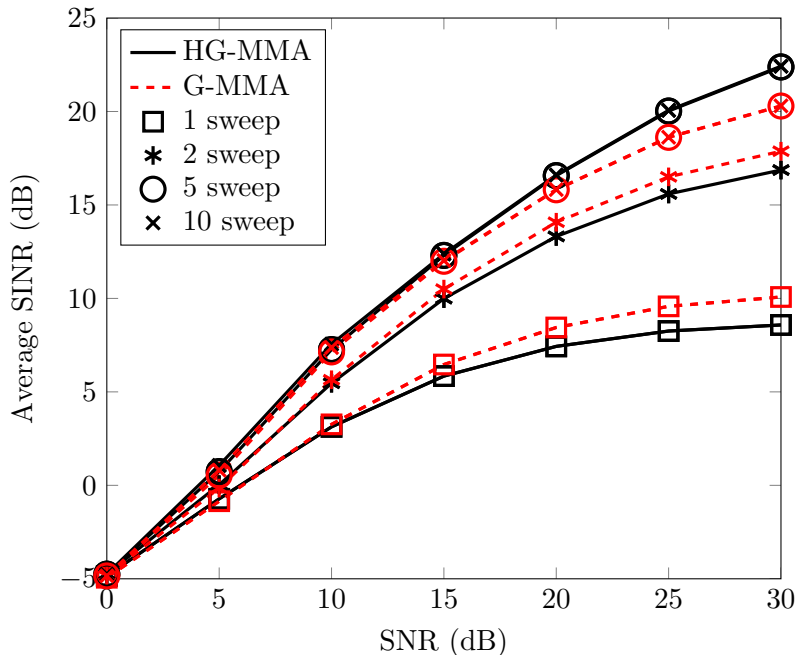


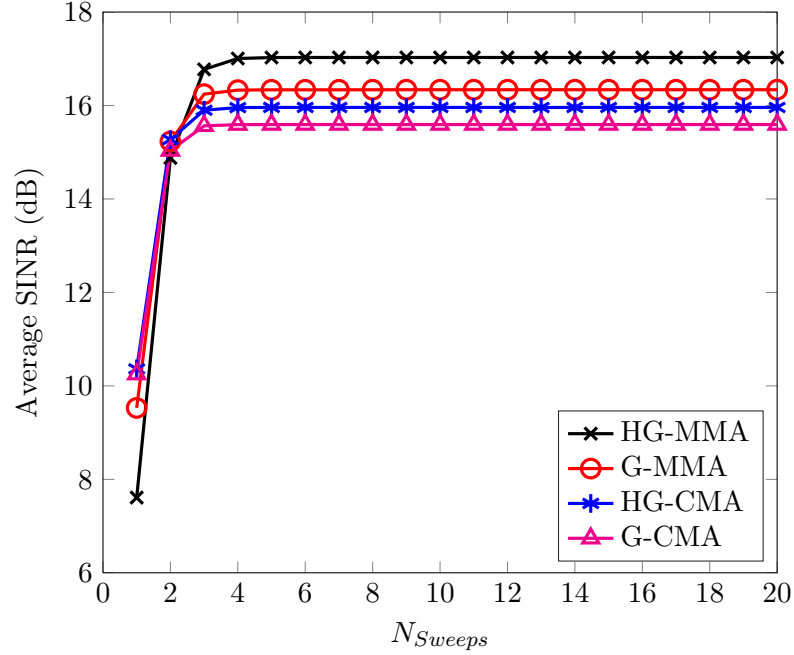
Figure 3.2: Average SINR of HG-MMA and G-MMA vs. SNR for different N_{Sweeps} considering $N_t = 5$, $N_r = 7$, $N_s = 150$ and 16-QAM constellation.

3.4.3 Experiment 3: Comparison of Rate of Convergence

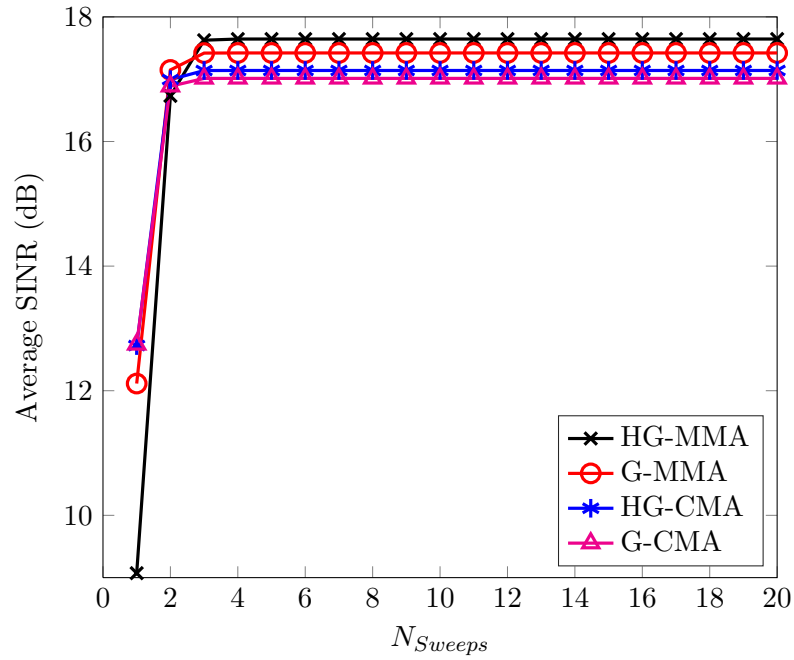
In Figure 3.3, we have compared the convergence rate of the proposed and benchmarked algorithms which are iterative like G-CMA and HG-CMA. The SNR is fixed at 20 dB and N_s is selected as 200 and 700 for 16-QAM and 64-QAM, respectively. It can be noticed that all the algorithms converge in 5 sweeps. However, the performance of the proposed algorithms HG-MMA and G-MMA is better than those of the HG-CMA and G-CMA.

3.4.4 Experiment 4: Effect of the Number of Samples

Figure 3.4a and 3.4b, show the SINR performance of our proposed algorithms with the aforementioned algorithms vs. the number of samples N_s for 16-QAM and



(a) 16-QAM, $N_s = 200$



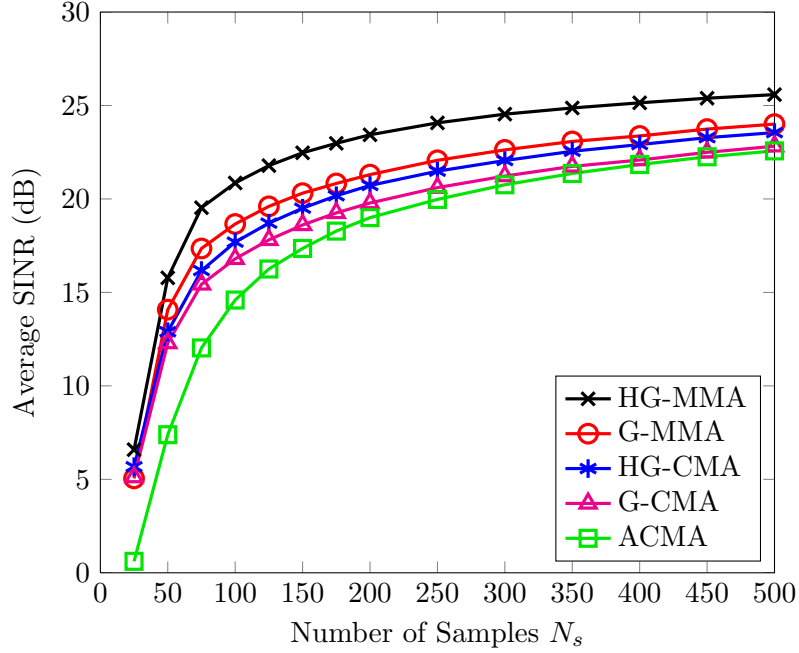
(b) 64-QAM, $N_s = 700$

Figure 3.3: Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA vs. N_{Sweeps} for $N_t = 5$, $N_r = 7$ and SNR = 20dB.

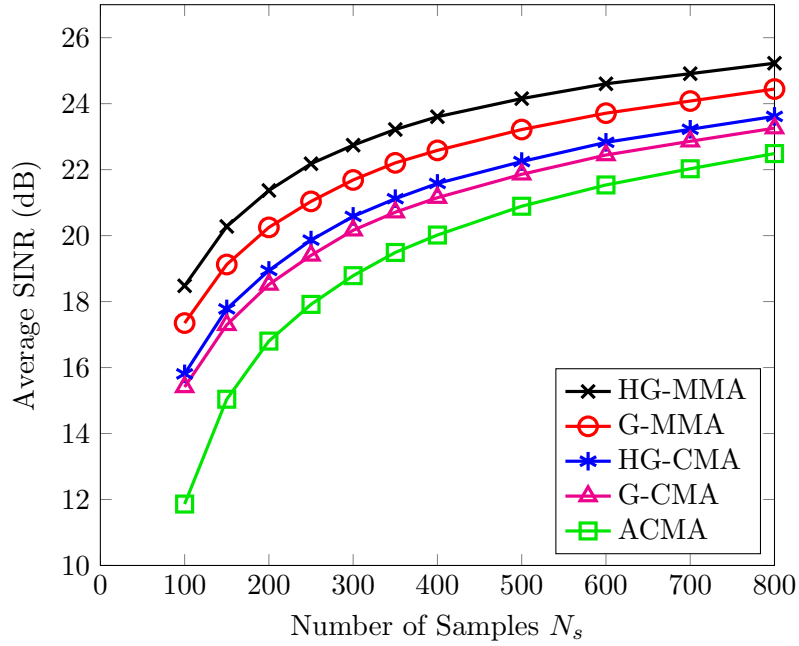
64-QAM constellations, respectively. The SNR is fixed at 30dB for both figures. It can be noticed that, as expected, the larger the number of samples the better the performance of proposed as well as other algorithms. The reason is that for the large number of samples, whitening is effective and thus mixing matrix \mathbf{A} can be inverted more accurately using \mathbf{W} . It can be seen that the HG-MMA takes the lead among all other algorithms, which is more significant for the higher QAM constellation (i.e., 64-QAM). For large number of samples and low order constellations, the performance of ACMA is quite close to G-CMA and HG-CMA, however, our proposed algorithms still performs better than all of them.

3.4.5 Experiment 5: Comparison based on SINR

Figure 3.5a compares the SINR performance of the proposed and benchmarked algorithms as a function of SNR. In this figure, two different number of samples ($N_s = 50$ and $N_s = 200$) are considered for 16-QAM constellation. As noticed previously, all algorithms perform better for large number of samples. Also, the difference between the performance of the HG-MMA and G-MMA increases with the increase in number of samples. The G-MMA cannot perform well for small number of samples because of the ineffective pre-whitening operation. For all algorithms, SINR is proportional to SNR. The highest SINR is obtained with the proposed HG-MMA algorithm, followed by the G-MMA and HG-CMA, then by the G-CMA and the lowest SINR is obtained with the ACMA algorithm. It is very clear from the figure that the ACMA with small number of samples is not



(a) 16-QAM



(b) 64-QAM

Figure 3.4: Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. the number of samples N_s for $N_t = 5$, $N_r = 7$, SNR = 30dB and $N_{Sweeps} = 10$.

suitable for the QAM constellation.

In Figure 3.5b, we consider the case of 64-QAM constellation with two different

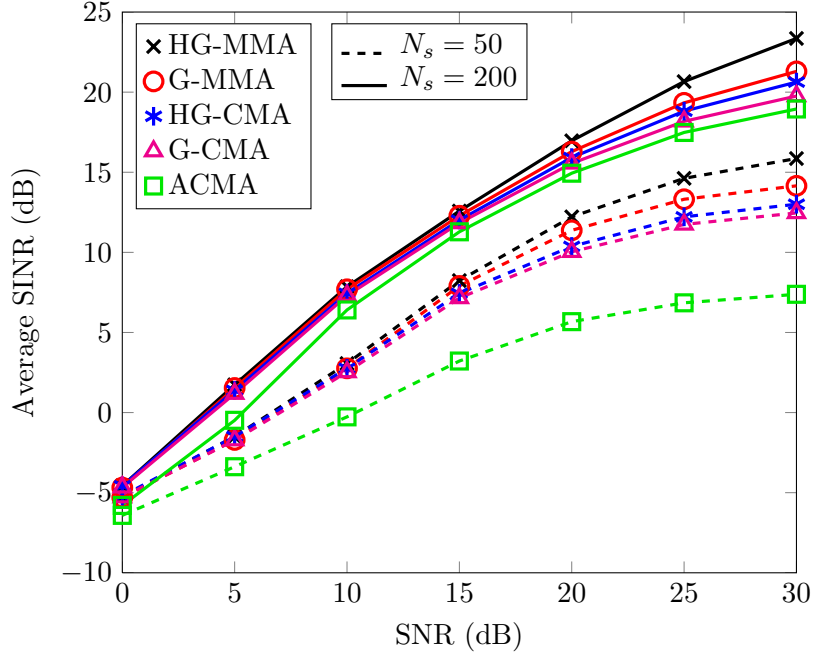
number of samples ($N_s = 150$ and $N_s = 700$). It is noticed that the performance of proposed algorithms is significantly better than other algorithms even for small number of samples N_s . For an SNR lower than 10dB and large number of samples, the performance of all algorithms is nearly the same but for 15dB and above, the proposed algorithms perform better than the rest of the algorithms.

3.4.6 Experiment 6: Comparison based on SER

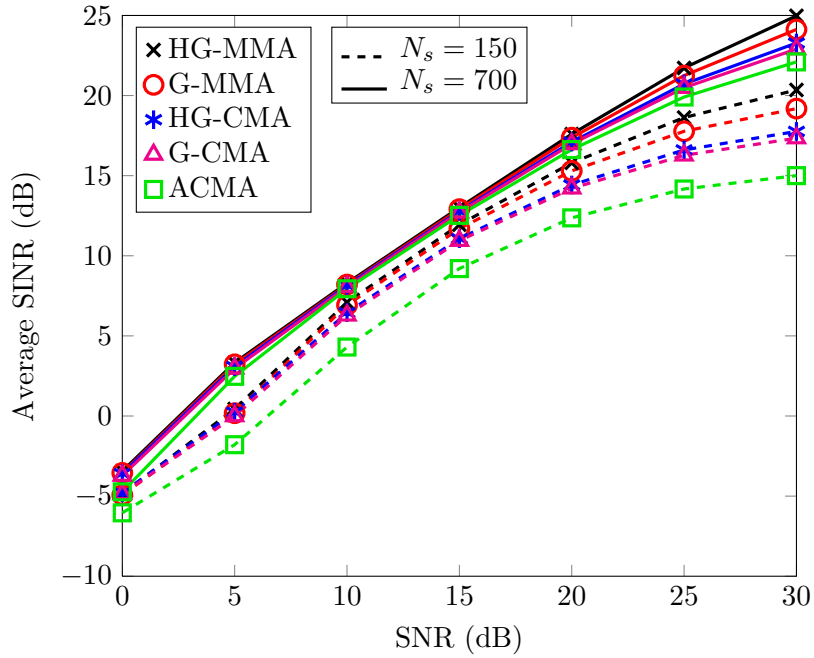
Figure 3.6a and 3.6b depict the SER of proposed and benchmarked algorithms vs. SNR for the case of 16-QAM and 64-QAM constellations, respectively. In both figures, different number of samples are considered i.e., for 16-QAM ($N_s = 50$ and $N_s = 200$) and for 64-QAM ($N_s = 150$ and $N_s = 700$). As noticed previously, the performance of the HG-MMA is significantly better than all the other algorithms. Comparison of Figure 3.6a and 3.6b show that in the case of lower QAM (such as 16-QAM) for small number of samples, the performance of proposed algorithms is nearly the same, however, for higher constellations (such as 64-QAM), HG-MMA performs better than G-MMA. Similar to other figures, same pattern of performance is observed i.e., the HG-MMA takes the lead followed by the G-MMA, HG-CMA, then by the G-CMA and ACMA.

3.5 Chapter Conclusions

In this chapter, we have reviewed Givens and hyperbolic rotations, which can be used for the diagonalization of matrices. However, here we utilized them for the

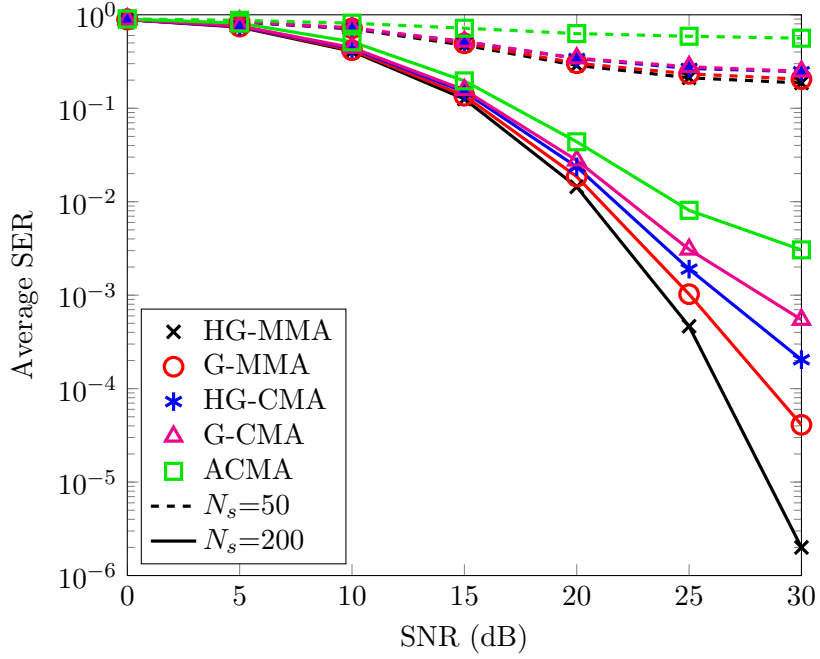


(a) 16-QAM, $N_s = 50$ and $N_s = 200$

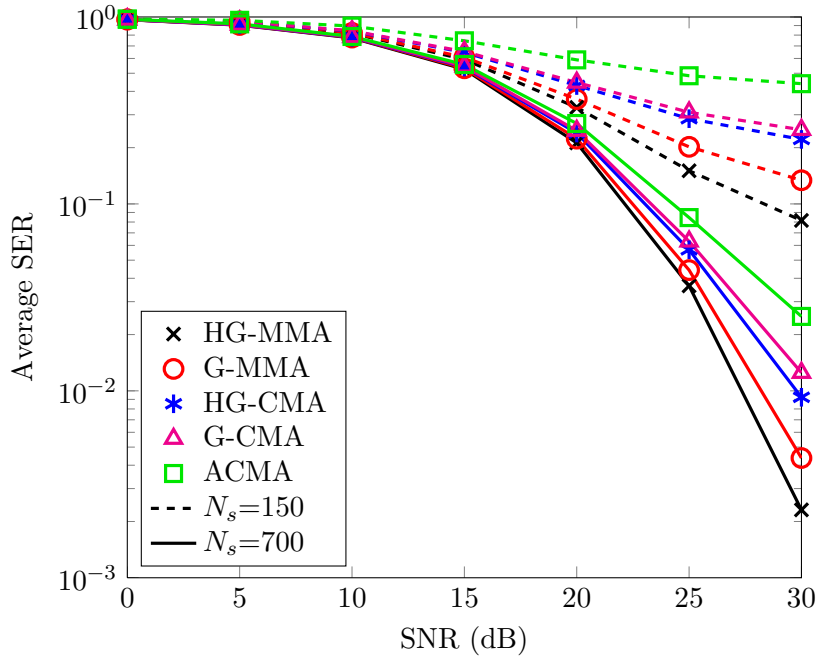


(b) 64-QAM, $N_s = 150$ and $N_s = 700$

Figure 3.5: Average SINR of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 10$ and different number of samples N_s .



(a) 16-QAM, $N_s = 50$ and $N_s = 200$



(b) 64-QAM, $N_s = 150$ and $N_s = 700$

Figure 3.6: Average SER of HG-MMA, G-MMA, HG-CMA, G-CMA and ACMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 10$ and different number of samples N_s .

minimization of MM cost functions. It is shown that an MM cost function is suitable for QAM modulations and has number of advantages over the CM criterion. Two new iterative batch BSS algorithms named G-MMA and HG-MMA were presented. The proposed algorithms are designed using a pre-whitening operation to reduce the complexity of design problem, followed by a recursive separation method of unitary Givens and J-unitary hyperbolic rotations to minimize the MM criterion. The difficulties faced while dealing with complex matrices are also detailed. Thus, instead of using complex matrices, a real transformation is considered where a special structure of the separation matrix in the whitened domain is suggested and maintained throughout all transformations.

The proposed algorithms are mainly designed for the blind deconvolution of MIMO systems involving QAM signals. Simulation results demonstrate their favorable performance as compared to contemporary batch BSS algorithms. It is noticed that the G-MMA is cheaper and more suitable for large number of samples but in the case of small number of samples the HG-MMA should be used. For higher constellations, the algorithm's performance deteriorates, especially for small and moderate sample sizes. In such cases, we will consider in Chapter 4 the combined criteria using the MMA cost function together with alphabet matching ones [41, 42].

CHAPTER 4

ALPHABET MATCHED ALGORITHMS

Multimodulus criterion is suitable for QAM but as can be observed from Chapter 3, the algorithms based on MM criterion do not work well for high order QAM. Such modulations are used in many modern communication systems such as LTE [4] and WiMAX [5], which require high data rates. For these modulations, MMA leads to a considerable amount of residual errors and does not ensure low SER. This affects the maximum achievable data rate and quality of service (QoS).

This chapter includes the review of cost functions which are more suitable for high order QAM, known as alphabet matched (AM) functions. The method of optimization i.e., sequence of Givens and hyperbolic rotations is used to minimize the AM cost function, in order to design alphabet matched algorithms (AMA). At the end, some practical considerations of algorithms and simulation results are presented, which shows that the newly designed AMA algorithms outperform the

rest of the batch BSS algorithms in terms of convergence rate, SINR and SER.

4.1 Alphabet Matched (AM) Functions

Considering the fact that the transmitted signal takes discrete values from finite alphabets and every alphabet member is equidistant from neighbouring members, some of the key properties that are desired in cost function for high order square QAM signals are:

1. It should not favor or penalize alphabet members over others, thus it should have uniform behavior.
2. It should be locally symmetric around each alphabet point.
3. It should place the highest penalty at the maximum deviation i.e., the midpoint between two alphabet points and should not place any penalty for zero errors i.e., at the alphabet point.

One can visualize a square QAM constellation and can observe that these properties incorporate the information of constellation in a proper manner. Properties 1 and 2 consider the fact that the alphabet points are equidistant. Moreover, one can conclude that properties 1 and 2 serve to shape the cost function in a way that should not be biased towards any specific alphabet point. Property 3 keeps track of the amount of error. Thus, these properties should be taken into account while designing a cost function for higher QAM constellations.

A number of efforts have been made to incorporate information of the signal constellation into the cost function, which results in a variety of functions. Here, we will include some of the widely used functions.

4.1.1 Li's AMA

The very first cost function considering alphabet matched technique was presented by Li [43] in 1995 for multilevel signals. Its behaviour is thoroughly studied for 8-PAM and 4-QAM signals by Li *et al.* [44] in 1997. The presented cost function is based on the idea to match the output signal with one of the constellation points and can be written as

$$\mathcal{J}_{\text{Li}}(\mathbf{V}) = \sum_{j=1}^{N_t} \mathbb{E} \left[\prod_{l=1}^L |z_j(i) - c(l)|^2 \right] \quad (4.1)$$

where $c(l) = c_R(l) + \iota c_I(l)$ with $l = 1, \dots, L$ are the constellation points of L -QAM and $c_R, c_I \in \{\pm 1d, \pm 3d, \dots, \pm(\sqrt{L} - 1)d\}$.

One can notice that for $z_j(i) = c(l)$, the product in (4.1) equals zero. It shows that Li's function is designed to give a minimum value of 0 at the constellation points. However, it is observed that this cost function does not satisfy the uniformity and symmetry properties i.e., Property 1 and 2. Another drawback of Li's AMA cost function is studied in [45] that it requires an extremely good initialization for satisfactory convergence specifically for high order constellations. Moreover, this function is expensive in terms of flops because the number of computations depends on the number of constellation points, which increases with the

order of QAM.

4.1.2 Gauss AMA

To overcome the drawbacks of Li's AMA, another cost function which is a complement of the sum of Gaussian functions centered at the constellation points was presented by Barbarossa *et al.* [45] in 1997, which can be written as

$$\mathcal{J}_{\text{Gauss}}(\mathbf{V}) = \sum_{j=1}^{N_t} \mathbb{E} \left[1 - \sum_{l=1}^L e^{\frac{-|z_j^{(i)} - c(l)|^2}{2\sigma^2}} \right] \quad (4.2)$$

where $\sigma \leq \frac{2d}{\sqrt{-\ln(\epsilon)}}$ controls the width of the nulls, $2d$ is the minimum distance between constellation points and $\epsilon = 0.001$ is a small number close to zero. The relationship for the width of nulls σ can be found by satisfying the inequality $e^{-|c(k) - c(l)|^2 / 2\sigma^2} \approx 0, \forall k \neq l$. The values of these parameters for the case of square QAM constellations are listed in Table 4.1, where the width of nulls σ is computed considering $\epsilon = 0.001$.

Table 4.1: Parameters of AM cost functions for square QAM and $\epsilon = 0.001$

Constellation	Minimum distance $2d$	Width of nulls σ
4-QAM	1.4142	0.5381
16-QAM	0.6325	0.2406
64-QAM	0.3086	0.1174
256-QAM	0.1534	0.0584

This cost function satisfies all properties listed above. In terms of computational complexity, it is still expensive as the number of computations equal to the number of constellation points. Thus, this results in high cost with the increase

in order of constellation.

4.1.3 Sinusoidal AMA

For high order square QAM constellations, He *et al.* [46] in 2001 suggested a constellation matched error (CME) term given by

$$g(x) = 1 - \sin^{2n}\left(x\frac{\pi}{2d}\right) \quad (4.3)$$

where n is an integer number. This CME term is an even powered sinusoid. It is designed to give minimum value of 0 at the real or imaginary parts of the constellation points and maximum value of 1 at the center points in between two consecutive alphabet points as shown in Figure 4.1a for un-normalized 64-QAM. Also, similar pattern is obtained with normalized constellation because CME term adjusts itself according to the minimum distance between the alphabet points. Moreover, in Figure 4.1b and 4.1c, we have studied the effect of increasing n in the CME term for normalized 64-QAM. Comparing them, one can notice that this term becomes sharper at the constellation points with the increase in n .

The designed CME term is thoroughly studied and tested for the blind equalization of QAM signals by Amin *et al.* [47, 48] using the cost function

$$\mathcal{J}_{\text{AMA}}(\mathbf{V}) = \sum_{j=1}^{N_t} \mathbb{E} [g(z_{j,R}(i)) + g(z_{j,I}(i))] \quad (4.4)$$

where $g(x)$ is the CME term defined in (4.3). The authors have shown that the

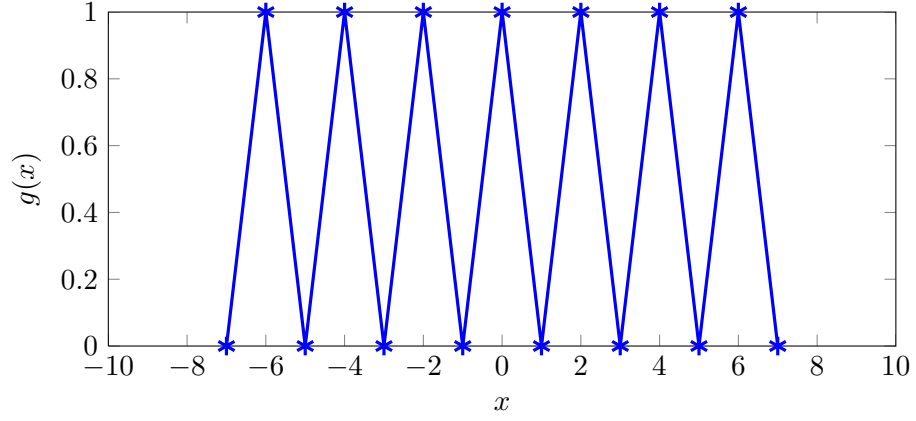
sinusoidal AMA cost function has good local convergence properties and requires a good initialization. Thus, it should be used along with CMA, which is considered for having good initial values of the AMA cost function and hence guaranteeing the overall algorithm convergence.

4.2 AM Algorithms Design

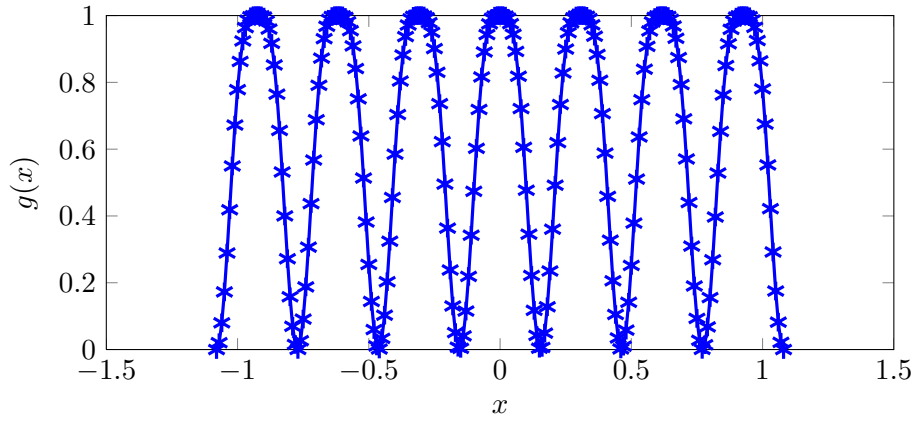
For the design of AM algorithms, the sinusoidal AMA cost function is selected because of the following reasons:

1. It satisfies all three properties presented in Section 4.1, which are sufficient conditions to shape the cost function for high order square QAM signals.
2. It is the simplest among all other AM cost functions and computationally less expensive. Note that, unlike Gauss or Li's AM cost functions, the computation in this one is independent of the number of constellation points.
3. It deals with the real and imaginary parts of the output signal, separately. Thus, it is relatively easier to optimize using real Givens and hyperbolic rotations. Moreover, for initialization, instead of CMA we will use MMA algorithm presented in Chapter 3. So, there is no need to transform the matrices from real to complex, while switching algorithm from MMA to AMA.

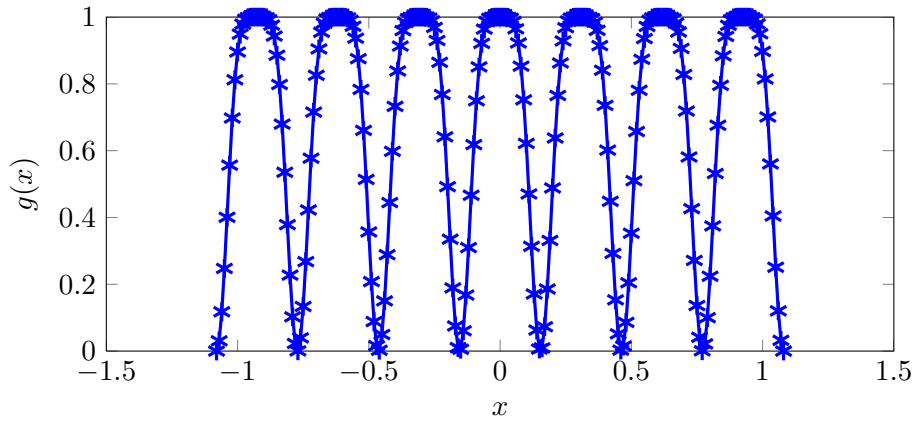
The combination of MMA and AMA is not new and recently used by Labeled *et al.* [49] in 2012 for the problem of blind equalization. In this work, G-MMA is



(a) $n = 1$, un-normalized QAM



(b) $n = 2$, normalized QAM



(c) $n = 3$, normalized QAM

Figure 4.1: The CME term $g(x)$ vs. x for varying n and 64-QAM constellation.

used as an initialization followed by optimization of sinusoidal AM cost function (with $n = 1$ for CME term) using Givens and hyperbolic rotations, which results in algorithms G-AMA and HG-AMA. As per observations from the rate of convergence for G-MMA shown in Section 3.4.3, G-MMA converges in $N_{Sweeps} = 5$ for our considered case. Thus, the following algorithms are switched from G-MMA to AMA after 5 sweeps. **Remark:** One can choose the number of sweeps in general as the one corresponding to an almost flat variation of the cost function.

4.2.1 Givens AMA (G-AMA)

After using G-MMA for the algorithm initialization, the required matrix $\hat{\mathbf{V}}$ is updated iteratively until convergence using following Givens rotations

$$\hat{\mathbf{V}}^n = \mathcal{G}_{p,q+N_t}(\dot{\theta})\mathcal{G}_{q,p+N_t}(\dot{\theta})\mathcal{G}_{p,q}(\theta)\mathcal{G}_{p+N_t,q+N_t}(\theta)\hat{\mathbf{V}}^{n-1} \quad (4.5)$$

where $n = n_0, \dots, N_{Sweeps}$, N_{Sweeps} is the number of iterations of G-AMA until convergence and n_0 is the number of iterations of G-MMA for initialization.

Similar to the case of G-MMA, the rotations $\mathcal{G}_{p,q}(\theta)$ and $\mathcal{G}_{p+N_t,q+N_t}(\theta)$ are applied successively using the same angle parameter (θ) . Also, the rotations $\mathcal{G}_{p,q+N_t}(\dot{\theta})$ and $\mathcal{G}_{q,p+N_t}(\dot{\theta})$ are applied with another angle parameter $(\dot{\theta})$. Note that, these rotations are applied in this way in order to preserve the structure of $\hat{\mathbf{V}}$ given in (3.10).

We only need to find rotation angle parameters (θ) and $(\dot{\theta})$ in order to minimize the AM criterion (4.4), using the above explained iterative method. Later

on, we will express the AM cost function in terms of the angle parameter (θ) which is computed such that $\mathcal{J}_{\text{AMA}}(\theta)$ is minimized. Now, consider a unitary transformation $\underline{\mathbf{Z}} = \mathcal{G}_{p,q}\underline{\mathbf{Y}}$, which according to the definition of Givens rotations in (3.1) only changes rows ‘ p ’ and ‘ q ’ of $\underline{\mathbf{Y}}$ such as

$$\begin{aligned} \dot{z}_{ji} &= \dot{y}_{ji} \quad \text{for } j \neq p, q \\ \dot{z}_{pi} &= \cos(\theta)\dot{y}_{pi} + \sin(\theta)\dot{y}_{qi} \\ \dot{z}_{qi} &= -\sin(\theta)\dot{y}_{pi} + \cos(\theta)\dot{y}_{qi} \end{aligned} \tag{4.6}$$

Similarly, the rotation $\mathcal{G}_{p+N_t, q+N_t}$ with the same angle parameter (θ) modifies rows ‘ $p + N_t$ ’ and ‘ $q + N_t$ ’ in a similar way as shown in (4.6). Note that for simplicity, we keep the notation $\underline{\mathbf{Y}}$ unchanged even though the matrix is modified after each rotation. Now, the AMA cost function in (4.4) can be re-written in terms of the Givens angle parameter (θ) (omitting the terms of $\underline{\mathbf{Z}}$ that are independent of (θ))

$$\mathcal{J}_{\text{AMA}}(\theta) = \sum_{i=1}^{N_s} [g(\dot{z}_{pi}) + g(\dot{z}_{qi}) + g(\dot{z}_{p+N_t, i}) + g(\dot{z}_{q+N_t, i})] \tag{4.7}$$

where the four terms in (4.7) can be defined using (4.6) and (4.3) with $n = 1$ as

$$\begin{aligned} g(\dot{z}_{pi}) &= 1 - \sin^2 \left\{ \left(\cos(\theta)\dot{y}_{pi} + \sin(\theta)\dot{y}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{qi}) &= 1 - \sin^2 \left\{ \left(-\sin(\theta)\dot{y}_{pi} + \cos(\theta)\dot{y}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{p+N_t, i}) &= 1 - \sin^2 \left\{ \left(\cos(\theta)\dot{y}_{p+N_t, i} + \sin(\theta)\dot{y}_{q+N_t, i} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{q+N_t, i}) &= 1 - \sin^2 \left\{ \left(-\sin(\theta)\dot{y}_{p+N_t, i} + \cos(\theta)\dot{y}_{q+N_t, i} \right) \left(\frac{\pi}{2d} \right) \right\} \end{aligned} \tag{4.8}$$

It can be noticed that the above mentioned problem in (4.7) is a bounded non-linear optimization problem and can be stated as

$$\min_{\theta} \mathcal{J}_{\text{AMA}} \quad \text{s.t. } \theta \in [-\pi/4, \pi/4] \quad (4.9)$$

The optimization problem in (4.9) can be solved either by using MATLAB optimization toolbox that can be termed as ‘exact solution’ or by using Taylor series approximation of trigonometric functions around zero, which will be referred to as ‘approximate solution’. This approximation can be justified using Figure 4.2, which plots the values of AMA cost function \mathcal{J}_{AMA} in (4.7) vs. θ for some random received pre-whitened signal $\underline{\mathbf{Y}}$ after 5 sweeps of G-MMA with $N_t = 3, N_r = 5, N_s = 300, \text{SNR} = 30\text{dB}$ and normalized 64-QAM constellation. It can be noticed that the optimum θ° is very close to zero. Thus in the following section, it is showed that for certain range of θ close to zero, the approximation exactly fits the original values of the AMA cost function. Further, it can be noticed that this function is periodic with a period of $\pi/2$, which justifies the bounds $[-\pi/4, \pi/4]$ in (4.9). This periodicity is because of the periodic nature of the trigonometric terms appearing in cost function as shown in (4.8). **Remark:** During simulations, we found that for low SNR values (i.e., $\text{SNR} < 15\text{dB}$ for normalized 64-QAM constellation), the optimum θ is far from zero.

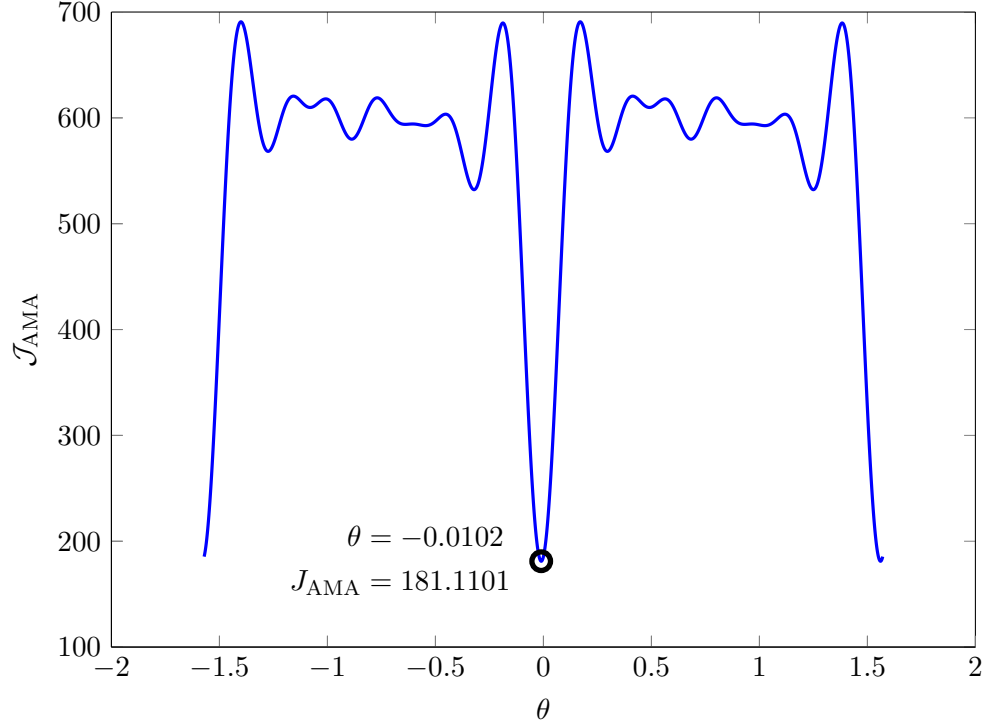


Figure 4.2: \mathcal{J}_{AMA} vs. θ for random received pre-whitened signal after 5 sweeps of G-MMA with $N_t = 3, N_r = 5, N_s = 300$, SNR = 30dB and normalized 64-QAM constellation

A) Exact Solution

There are number of MATLAB optimization toolbox that can be used to find the local minimum. The concerned optimization problem in (4.9) is bounded and non-linear. So, the optimization toolbox is selected accordingly which considers upper and lower bounds, non-linearity and also takes an initial starting point as input to find the minimum value of the cost function close to that point. In this scenario, the most suitable option is to use MATLAB optimization toolbox function named as ‘**fminsearchbnd**’ which is a non-linear optimization toolbox satisfying the above mentioned criteria. This MATLAB optimization toolbox is developed by John DErrico in 2005 and available at MATLAB Central File Exchange, where

the currently available version updated on 06 Feb., 2012 is used here. **Remark:** Our objective here is just to compute an ‘exact’ solution of (4.9), which can be obtained by a linesearch algorithm as well.

Now, an objective function is defined according to (4.7) and (4.8). Also, for the initialization of G-AMA, values of matrix $\underline{\dot{Y}}$ corresponds to the one obtained after 5 sweeps of G-MMA. This objective function is passed to the optimization toolbox ‘**fminsearchbnd**’ along with $\theta_0 = 0.001$ as a starting point and bounds $\theta \in [-\pi/4, \pi/4]$, in order to find optimum θ° for the minimization of (4.9). Once optimum θ° is found, Givens rotation matrices $\mathcal{G}_{p,q}(\theta^\circ)$ and $\mathcal{G}_{p+N_t, q+N_t}(\theta^\circ)$ are computed using (3.1) and applied to update \dot{V} according to (4.5). The remaining Givens rotations $\mathcal{G}_{p,q+N_t}(\dot{\theta})$ and $\mathcal{G}_{q,p+N_t}(\dot{\theta})$ can be found similarly by replacing subscripts accordingly in (4.7) and (4.8) and then computing optimum $\dot{\theta}^\circ$. Then, the separation matrix \dot{V} is updated again according to (4.5). This process is repeated until convergence.

B) Approximate Solution

As observed from Figure 4.2, the optimum θ° is close to zero, thus the Taylor series approximation of trigonometric functions around zero can be applied. Here, we will consider the approximation up to 4th order using following approximate identities

$$\sin(\theta) \approx \theta - \frac{\theta^3}{6}, \quad \cos(\theta) \approx 1 - \frac{\theta^2}{2} + \frac{\theta^4}{24} \quad (4.10)$$

Let's consider the first term of (4.7), which is given in (4.8) as

$$g(\underline{z}_{pi}) = 1 - \sin^2 \left\{ \left(\cos(\theta) \underline{y}_{pi} + \sin(\theta) \underline{y}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \quad (4.11)$$

Now, using the trigonometric approximation given in (4.10) to 'cos(θ)' and 'sin(θ)' in the argument of outer 'sin' in (4.11) and expanding the terms results in

$$g(\underline{z}_{pi}) \approx 1 - \sin^2 \left\{ \left(24 \underline{y}_{pi} + 24 \underline{y}_{qi} \theta - 12 \underline{y}_{pi} \theta^2 - 4 \underline{y}_{qi} \theta^3 + \underline{y}_{pi} \theta^4 \right) \left(\frac{\pi}{12d} \right) \right\} \quad (4.12)$$

Now, again the same approximation given in (4.10) is applied leading to

$$g(\underline{z}_{pi}) \approx \frac{1}{48d^4} c_4^{pi} \theta^4 + \frac{1}{12d^3} c_3^{pi} \theta^3 + \frac{1}{4d^2} c_2^{pi} \theta^2 - \frac{1}{2d} c_1^{pi} \theta + \frac{1}{2} c_0^{pi} \quad (4.13)$$

where

$$\begin{aligned}
c_4^{pi} &= 4\pi^2 d^2 \underline{y}_{qi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) + \pi^4 \underline{y}_{qi}^4 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) - 3\pi^2 d^2 \underline{y}_{pi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
&\quad - \pi d^3 \underline{y}_{pi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) - 6\pi^3 d \underline{y}_{pi} \underline{y}_{qi}^2 \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_3^{pi} &= \pi d^2 \underline{y}_{qi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) + \pi^3 \underline{y}_{qi}^3 \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
&\quad + 3\pi^2 d \underline{y}_{pi} \underline{y}_{qi} \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_2^{pi} &= \pi d \underline{y}_{pi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) - \pi^2 \underline{y}_{qi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_1^{pi} &= \pi \underline{y}_{qi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_0^{pi} &= 1 + \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right)
\end{aligned} \tag{4.14}$$

Using the same method, the 2nd term $g(\underline{z}_{qi})$ of (4.7) can be approximated and re-written as

$$g(\underline{z}_{qi}) \approx \frac{1}{48d^4} c_4^{qi} \theta^4 - \frac{1}{12d^3} c_3^{qi} \theta^3 + \frac{1}{4d^2} c_2^{qi} \theta^2 + \frac{1}{2d} c_1^{qi} \theta + \frac{1}{2} c_0^{qi} \tag{4.15}$$

where all the coefficients are obtained by replacing ‘ p ’ with ‘ q ’ and ‘ q ’ with ‘ p ’ in (4.14). The 3rd term $g(\underline{z}_{p+N_t,i})$ of (4.7) has the same approximation as given in (4.13), where the coefficients are obtained by replacing ‘ p ’ with ‘ $p + N_t$ ’ and ‘ q ’ with ‘ $q + N_t$ ’ in (4.14). The last term $g(\underline{z}_{q+N_t,i})$ of (4.7) is approximated as (4.15), where the coefficients are obtained by replacing ‘ p ’ with ‘ $q + N_t$ ’ and ‘ q ’ with ‘ $p + N_t$ ’ in (4.14).

Now, using (4.13) and (4.15) in cost function (4.7) results in 4th order polynomial equation

$$\mathcal{J}_{\text{AMA}}(\theta) \approx \frac{1}{48d^4}C_4\theta^4 + \frac{1}{12d^3}C_3\theta^3 + \frac{1}{4d^2}C_2\theta^2 + \frac{1}{2d}C_1\theta + \frac{1}{2}C_0 \quad (4.16)$$

where the coefficients in (4.16) are obtained by applying summation over all the coefficients in (4.13) and (4.15) as shown below

$$\begin{aligned} C_l &= \sum_{i=1}^{N_s} \left(c_l^{pi} + c_l^{qi} + c_l^{p+N_t,i} + c_l^{q+N_t,i} \right) \\ C_3 &= \sum_{i=1}^{N_s} \left(c_3^{pi} - c_3^{qi} + c_3^{p+N_t,i} - c_3^{q+N_t,i} \right) \\ C_1 &= \sum_{i=1}^{N_s} \left(-c_1^{pi} + c_1^{qi} - c_1^{p+N_t,i} + c_1^{q+N_t,i} \right) \end{aligned} \quad (4.17)$$

where $l \in \{0, 2, 4\}$.

Taking the gradient of (4.16) with respect to θ , we get

$$\frac{\partial \mathcal{J}_{\text{AMA}}(\theta)}{\partial \theta} \approx \frac{1}{12d^4}C_4\theta^3 + \frac{1}{4d^3}C_3\theta^2 + \frac{1}{2d^2}C_2\theta + \frac{1}{2d}C_1 \quad (4.18)$$

where the coefficients are the same as defined in (4.17). Equation (4.18) is a simple 3rd order polynomial equation and its solution is obtained by equating it to zero. Out of the three possible solutions, the optimum θ° is selected which results in minimum value of $\mathcal{J}_{\text{AMA}}(\theta)$ in (4.7).

Now, to justify that the approximation in (4.16) is good enough and will result in same minimum of the original optimization problem in (4.7), we have compared

the original cost function and approximated one for a certain range of θ around zero in Figure 4.3.

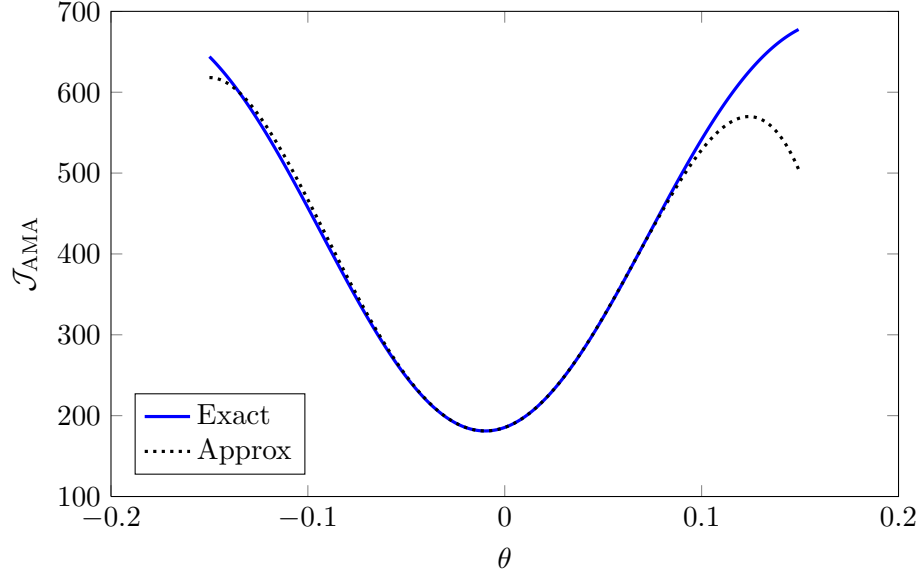


Figure 4.3: Comparison of exact and approximated Givens AMA cost function for random received pre-whitened signal after 5 sweeps of G-MMA with $N_t = 3$, $N_r = 5$, $N_s = 300$, SNR = 30dB and normalized 64-QAM constellation

The remaining Givens rotations $\mathcal{G}_{p,q+N_t}(\dot{\theta})$ and $\mathcal{G}_{q,p+N_t}(\dot{\theta})$ can be found similarly by replacing subscripts accordingly and computing optimum $\dot{\theta}^\circ$. Then, the rotations are applied successively on $\underline{\dot{\mathbf{Y}}}$.

In summary, pre-filtered separation matrix $\dot{\mathbf{V}}$ is initialized as identity matrix i.e., $\dot{\mathbf{V}} = \mathbf{I}_{2N_t}$. Then, G-MMA is applied for $N_{Sweeps} = 5$ followed by the update of the matrix $\dot{\mathbf{V}}$ according to (4.5) by applying Givens rotations on modified $\underline{\dot{\mathbf{Y}}}$ using the above explained method, until convergence. The overall algorithm is summarized in Table 4.2.

Table 4.2: Givens AMA (G-AMA) Algorithm

<p>Initialization: $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$</p> <ol style="list-style-type: none"> 1. Pre-whitening: $\underline{\mathbf{Y}} = \mathbf{B}\mathbf{Y}$ using (2.10) 2. Construct real matrix $\hat{\underline{\mathbf{Y}}}$ using (3.10) 3. Givens Rotations: <ol style="list-style-type: none"> for $n = 1 : N_{Sweeps}$ do if $n \leq 5$ then <ol style="list-style-type: none"> a) Apply G-MMA as given in Table 3.1 else <ol style="list-style-type: none"> for $p = 1 : N_t - 1$ do for $q = p + 1 : N_t$ do <ol style="list-style-type: none"> b) Find optimum (θ°) using roots of (4.18) which gives minimum value of (4.7) c) Compute $\mathcal{G}_{p,q}$ & $\mathcal{G}_{p+N_t,q+N_t}$ using (3.1) for same (θ°) d) $\hat{\underline{\mathbf{Y}}} = \mathcal{G}_{p,q} \mathcal{G}_{p+N_t,q+N_t} \hat{\underline{\mathbf{Y}}}$ e) $\hat{\mathbf{V}} = \mathcal{G}_{p,q} \mathcal{G}_{p+N_t,q+N_t} \hat{\mathbf{V}}$ repeat (b to e) for $(p, q + N_t)$ & $(q, p + N_t)$ using same ($\hat{\theta}^\circ$) end for end for end if end for 4. Construct complex matrix \mathbf{W} similar to \mathbf{V} using (2.14) and (3.10) 5. Estimated Sources: $\hat{\mathbf{S}} = \mathbf{W}\mathbf{Y}$
--

4.2.2 Hyperbolic G-AMA (HG-AMA)

Similar to the case of G-MMA, the performance of G-AMA is not satisfactory for small number of samples N_s . In this case, for which \mathbf{A} is far from unitary matrix, J-unitary real hyperbolic rotations are applied alternatively along with the Givens rotations to overcome the limitation of ill-whitening. This results in algorithm HG-AMA, which is explained below.

For HG-AMA, first of all G-MMA is used for initialization. Then, matrix \mathbf{V} is updated iteratively until convergence using following hyperbolic and Givens rotations

$$\begin{aligned}\hat{\mathbf{V}}^n &= \mathbf{\Gamma}_{p,q+N_t} \mathbf{\Gamma}_{q,p+N_t} \mathbf{\Gamma}_{p,q} \mathbf{\Gamma}_{p+N_t,q+N_t} \hat{\mathbf{V}}^{n-1} \\ \mathbf{\Gamma}_{p,q} &= \mathbf{G}_{p,q} \mathbf{H}_{p,q}\end{aligned}\tag{4.19}$$

where $\mathbf{G}_{p,q}$ and $\mathbf{H}_{p,q}$ refer to the Givens and hyperbolic transformations, respectively. The hyperbolic rotations $\mathbf{H}_{p,q}$ and $\mathbf{H}_{p+N_t,q+N_t}$ are applied using the same parameter (γ), while $\mathbf{H}_{p,q+N_t}$ and $\mathbf{H}_{q,p+N_t}$ are applied using another same but opposite parameter ($\dot{\gamma}$) and ($-\dot{\gamma}$), respectively. Note that the rotations are designed in this way to preserve the structure of matrix $\hat{\mathbf{V}}$ in (3.10). Below, we give a brief of finding the hyperbolic rotation parameters to minimize the sinusoidal AMA criterion in (4.4).

Similar to the case of Givens rotations, let us consider a unitary transformation $\hat{\mathbf{Z}} = \mathbf{H}_{p,q} \hat{\mathbf{Y}}$, which according to the definition of hyperbolic rotations in (3.2) only

changes rows ‘ p ’ and ‘ q ’ of $\underline{\mathbf{Y}}$ such as

$$\begin{aligned}
\dot{z}_{ji} &= \underline{\dot{y}}_{ji} \quad \text{for } j \neq p, q \\
\dot{z}_{pi} &= \cosh(\gamma)\underline{\dot{y}}_{pi} + \sinh(\gamma)\underline{\dot{y}}_{qi} \\
\dot{z}_{qi} &= \sinh(\gamma)\underline{\dot{y}}_{pi} + \cosh(\gamma)\underline{\dot{y}}_{qi}
\end{aligned} \tag{4.20}$$

Similarly, the rotation $\mathcal{H}_{p+N_t, q+N_t}$ with the same parameter (γ) modifies rows ‘ $p+N_t$ ’ and ‘ $q+N_t$ ’ in the same way as shown in (4.20). Thus, AMA cost function can be re-written in terms of hyperbolic rotations parameter (γ) (omitting the terms of $\underline{\mathbf{Z}}$ independent of (γ)) as

$$\mathcal{J}_{\text{AMA}}(\gamma) = \sum_{i=1}^{N_s} [g(\dot{z}_{pi}) + g(\dot{z}_{qi}) + g(\dot{z}_{p+N_t, i}) + g(\dot{z}_{q+N_t, i})] \tag{4.21}$$

where the four terms in (4.21) can be defined using (4.20) and (4.3) as

$$\begin{aligned}
g(\dot{z}_{pi}) &= 1 - \sin^2 \left\{ \left(\cosh(\gamma)\underline{\dot{y}}_{pi} + \sinh(\gamma)\underline{\dot{y}}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \\
g(\dot{z}_{qi}) &= 1 - \sin^2 \left\{ \left(\sinh(\gamma)\underline{\dot{y}}_{pi} + \cosh(\gamma)\underline{\dot{y}}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \\
g(\dot{z}_{p+N_t, i}) &= 1 - \sin^2 \left\{ \left(\cosh(\gamma)\underline{\dot{y}}_{p+N_t, i} + \sinh(\gamma)\underline{\dot{y}}_{q+N_t, i} \right) \left(\frac{\pi}{2d} \right) \right\} \\
g(\dot{z}_{q+N_t, i}) &= 1 - \sin^2 \left\{ \left(\sinh(\gamma)\underline{\dot{y}}_{p+N_t, i} + \cosh(\gamma)\underline{\dot{y}}_{q+N_t, i} \right) \left(\frac{\pi}{2d} \right) \right\}
\end{aligned} \tag{4.22}$$

Figure 4.4 shows the values of AMA cost function \mathcal{J}_{AMA} in (4.21) vs. (γ) for some random received pre-whitened signal $\underline{\mathbf{Y}}$ after 5 sweeps of G-MMA with $N_t = 3$, $N_r = 5$, $N_s = 300$, SNR = 30dB and normalized 64-QAM constellation. It can be noticed that optimum (γ°) is very close to zero. Thus, we can apply Taylor

series approximation of hyperbolic and trigonometric functions around zero, in order to find the solution of the optimization problem in (4.21). Moreover, it can be noticed that this function is not periodic, thus the optimization problem is unbounded.

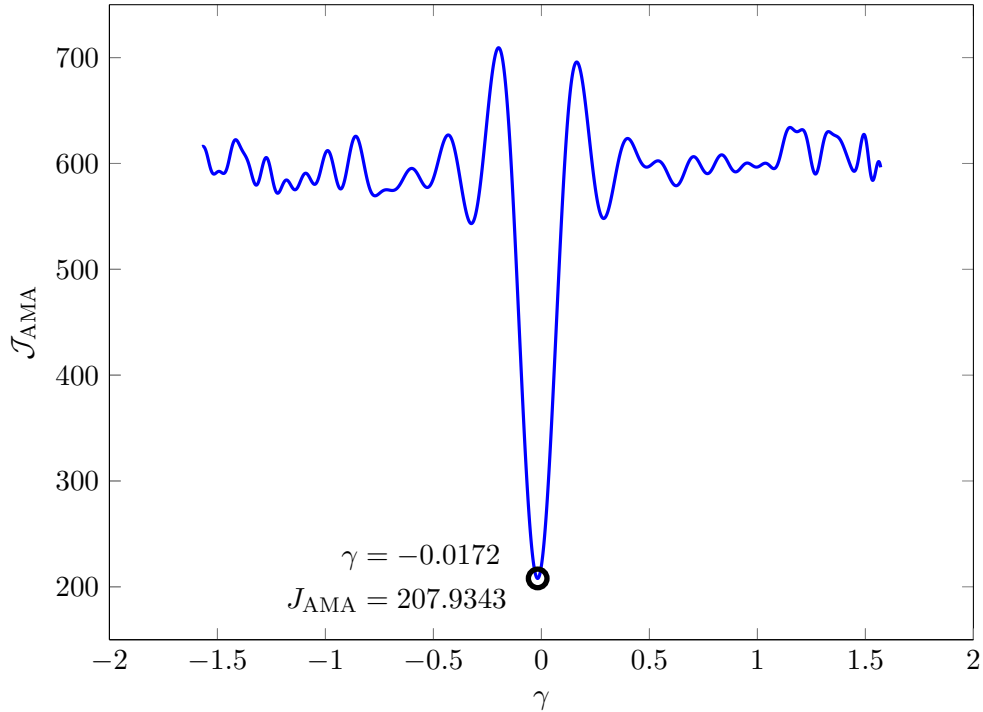


Figure 4.4: \mathcal{J}_{AMA} vs. γ for random received pre-whitened signal after $N_{\text{Sweeps}} = 5$ of G-MMA with $N_t = 3$, $N_r = 5$, $N_s = 300$, SNR = 30dB and normalized 64-QAM constellation

In the following sections, two possible ways are detailed to solve the optimization problem in (4.21). One of it is named ‘exact solution’ and uses the MATLAB optimization toolbox ‘**fminsearch**’, while the other one utilizes the Taylor series approximation as mentioned above, thus referred as ‘approximate solution’.

A) Exact Solution

The optimization problem in (4.21) is unbounded and non-linear. So, the optimization toolbox is selected accordingly which considers non-linearity and also takes an initial starting point as input to find the minimum value of the cost function close to that point. In this scenario, the most suitable option is to use MATLAB optimization toolbox function named as ‘**fminsearch**’ which is a non-linear optimization toolbox and also satisfies the above mentioned criteria.

An objective function is defined according to (4.21) and (4.22). Then, it is passed to the toolbox ‘**fminsearch**’ with $\gamma_0 = 0.001$ as starting point. The toolbox returns the optimum hyperbolic rotation parameter (γ°), which minimizes (4.21). Using (γ°) and (3.2), the hyperbolic rotation matrices $\mathcal{H}_{p,q}(\gamma^\circ)$ and $\mathcal{H}_{p+N_t,q+N_t}(\gamma^\circ)$ are computed and applied to update $\hat{\mathbf{V}}$ according to (4.19).

For the remaining hyperbolic rotations $\mathcal{H}_{p,q+N_t}(\dot{\gamma})$ and $\mathcal{H}_{q,p+N_t}(-\dot{\gamma})$, the sinusoidal AMA cost function can be re-written as (omitting constant terms of $\hat{\mathbf{Z}}$)

$$\mathcal{J}_{\text{AMA}}(\dot{\gamma}) = \sum_{i=1}^{N_s} [g(\dot{z}_{pi}) + g(\dot{z}_{q+N_t,i}) + g(\dot{z}_{qi}) + g(\dot{z}_{p+N_t,i})] \quad (4.23)$$

with

$$\begin{aligned} g(\dot{z}_{pi}) &= 1 - \sin^2 \left\{ \left(\cosh(\dot{\gamma}) \underline{y}_{pi} + \sinh(\dot{\gamma}) \underline{y}_{q+N_t,i} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{q+N_t,i}) &= 1 - \sin^2 \left\{ \left(\sinh(\dot{\gamma}) \underline{y}_{pi} + \cosh(\dot{\gamma}) \underline{y}_{q+N_t,i} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{qi}) &= 1 - \sin^2 \left\{ \left(\cosh(-\dot{\gamma}) \underline{y}_{qi} + \sinh(-\dot{\gamma}) \underline{y}_{p+N_t,i} \right) \left(\frac{\pi}{2d} \right) \right\} \\ g(\dot{z}_{p+N_t,i}) &= 1 - \sin^2 \left\{ \left(\sinh(-\dot{\gamma}) \underline{y}_{qi} + \cosh(-\dot{\gamma}) \underline{y}_{p+N_t,i} \right) \left(\frac{\pi}{2d} \right) \right\} \end{aligned} \quad (4.24)$$

Now, another objective function is defined using (4.23) and (4.24). The optimum ($\dot{\gamma}^\circ$) and thus the rotation matrices $\mathcal{H}_{p,q+N_t}$ and $\mathcal{H}_{q,p+N_t}$ are computed using the above explained method and applied successively on $\underline{\dot{Y}}$ to compute the separation matrix $\underline{\dot{V}}$ according to (4.19). The process is repeated until convergence.

B) Approximate Solution

In order to find the approximate solution, we will use the Taylor series approximation of trigonometric angles given in (4.10) and hyperbolic angles around zero up to 4th order, which can be written as

$$\sinh(\gamma) \approx \gamma + \frac{\gamma^3}{6}, \quad \cosh(\gamma) \approx 1 + \frac{\gamma^2}{2} + \frac{\gamma^4}{24} \quad (4.25)$$

Let's consider the first term of (4.21), which is given in (4.22) as

$$g(\dot{z}_{pi}) = 1 - \sin^2 \left\{ \left(\cosh(\gamma) \underline{\dot{y}}_{pi} + \sinh(\gamma) \underline{\dot{y}}_{qi} \right) \left(\frac{\pi}{2d} \right) \right\} \quad (4.26)$$

Now, applying the hyperbolic angle approximation given in (4.25) to 'cosh(γ)' and 'sinh(γ)' in the argument of 'sin' in (4.26) and expanding the terms, we get

$$g(\dot{z}_{pi}) \approx 1 - \sin^2 \left\{ \left(24 \underline{\dot{y}}_{pi} + 24 \underline{\dot{y}}_{qi} \gamma + 12 \underline{\dot{y}}_{pi} \gamma^2 + 4 \underline{\dot{y}}_{qi} \gamma^3 + \underline{\dot{y}}_{pi} \gamma^4 \right) \left(\frac{\pi}{12d} \right) \right\} \quad (4.27)$$

Finally, the trigonometric approximation given in (4.10) is used leading to

$$g(\dot{z}_{pi}) \approx \frac{1}{48d^4} c_4^{pi} \gamma^4 + \frac{1}{12d^3} c_3^{pi} \gamma^3 - \frac{1}{4d^2} c_2^{pi} \gamma^2 - \frac{1}{2d} c_1^{pi} \gamma + \frac{1}{2} c_0^{pi} \quad (4.28)$$

where

$$\begin{aligned}
c_4^{pi} &= \pi^4 \underline{y}_{qi}^4 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) + 6\pi^3 d \underline{y}_{pi} \underline{y}_{qi}^2 \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) - 4\pi^2 d^2 \underline{y}_{qi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
&\quad - 3\pi^2 d^2 \underline{y}_{pi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) - \pi d^3 \underline{y}_{pi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_3^{pi} &= \pi^3 \underline{y}_{qi}^3 \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) - \pi d^2 \underline{y}_{qi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
&\quad - 3\pi^2 d \underline{y}_{pi} \underline{y}_{qi} \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_2^{pi} &= \pi^2 \underline{y}_{qi}^2 \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right) + \pi d \underline{y}_{pi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_1^{pi} &= \pi \underline{y}_{qi} \sin\left(\frac{\pi \underline{y}_{pi}}{d}\right) \\
c_0^{pi} &= 1 + \cos\left(\frac{\pi \underline{y}_{pi}}{d}\right)
\end{aligned} \tag{4.29}$$

Using the same method, the other terms $g(\underline{z}_{qi})$, $g(\underline{z}_{p+N_t,i})$ and $g(\underline{z}_{q+N_t,i})$ of (4.21) can be approximated as (4.28), where the coefficients are obtained by replacing ‘ p ’ with ‘ q ’ and ‘ q ’ with ‘ p ’ for $g(\underline{z}_{qi})$, ‘ p ’ with ‘ $p + N_t$ ’ and ‘ q ’ with ‘ $q + N_t$ ’ for $g(\underline{z}_{p+N_t,i})$ and ‘ p ’ with ‘ $q + N_t$ ’ and ‘ q ’ with ‘ $p + N_t$ ’ for $g(\underline{z}_{q+N_t,i})$ in (4.29).

Now, we have the full approximation of our optimization problem in (4.21), which is just a 4th order polynomial equation and can be written as

$$\mathcal{J}_{\text{AMA}}(\gamma) \approx \frac{1}{48d^4} C_4 \gamma^4 + \frac{1}{12d^3} C_3 \gamma^3 - \frac{1}{4d^2} C_2 \gamma^2 - \frac{1}{2d} C_1 \gamma^1 + \frac{1}{2} C_0 \tag{4.30}$$

where the coefficients in (4.30) are obtained by applying summation over all the

coefficients in (4.28) as shown below

$$C_l = \sum_{i=1}^{N_s} \left(c_l^{pi} + c_l^{qi} + c_l^{p+N_t,i} + c_l^{q+N_t,i} \right) \quad (4.31)$$

where $l \in \{0, \dots, 4\}$.

Taking the gradient with respect to (γ) of AMA cost function in (4.30), we get

$$\frac{\partial \mathcal{J}_{\text{AMA}}(\gamma)}{\partial \gamma} \approx \frac{1}{12d^4} C_4 \gamma^3 + \frac{1}{4d^3} C_3 \gamma^2 - \frac{1}{2d^2} C_2 \gamma - \frac{1}{2d} C_1 \quad (4.32)$$

Now, equating (4.32) to zero results in three possible solutions. Out of them, the optimum (γ°) is selected which results in minimum value of $\mathcal{J}_{\text{AMA}}(\gamma)$ in (4.21).

Figure 4.5 shows the comparison of original cost function and approximated one for a certain range of (γ) around zero. It can be noticed that for this range of values of (γ) , the approximation in (4.30) gives the same result as of original optimization problem in (4.21).

The remaining hyperbolic rotations $\mathcal{H}_{p,q+N_t}(\dot{\gamma})$ and $\mathcal{H}_{q,p+N_t}(-\dot{\gamma})$ are applied using another hyperbolic angle parameter $(\dot{\gamma})$. The optimization problem for $(\dot{\gamma})$ is given in (4.23). The first two terms $g(\dot{z}_{pi})$ and $g(\dot{z}_{q+N_t,i})$ of cost function (4.23) have the same approximation as given in (4.28) with the replacement of (γ) by $(\dot{\gamma})$. The coefficients are obtained by replacing ‘ q ’ with ‘ $q+N_t$ ’ in (4.29) for $g(\dot{z}_{pi})$. Whereas, the coefficients for $g(\dot{z}_{q+N_t,i})$ are obtained by replacing ‘ p ’ with ‘ $q+N_t$ ’ and ‘ q ’ with ‘ p ’ in (4.29). Using the above explained method, the third term $g(\dot{z}_{qi})$

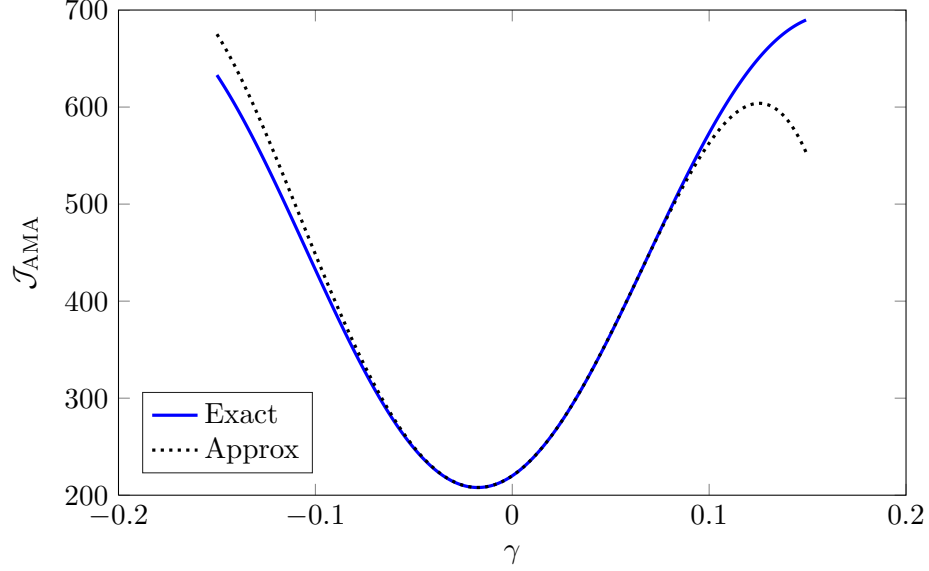


Figure 4.5: Comparison of exact and approximated hyperbolic AMA cost function for random received pre-whitened signal after $N_{Sweeps} = 5$ of G-MMA with $N_t = 3$, $N_r = 5$, $N_s = 300$, SNR= 30dB and normalized 64-QAM constellation

of the cost function in (4.23) can be approximated as

$$g(z_{qi}) \approx \frac{1}{48d^4}c_4^{qi}\dot{\gamma}^4 - \frac{1}{12d^3}c_3^{qi}\dot{\gamma}^3 - \frac{1}{4d^2}c_2^{qi}\dot{\gamma}^2 + \frac{1}{2d}c_1^{qi}\dot{\gamma} + \frac{1}{2}c_0^{qi} \quad (4.33)$$

where the coefficients are obtained by replacing ‘ p ’ with ‘ q ’ and ‘ q ’ with ‘ $p + N_t$ ’ in (4.29). The last term $g(z_{p+N_t,i})$ has the same approximation as given in (4.33), where the coefficients are obtained by replacing ‘ p ’ with ‘ $p + N_t$ ’ in (4.29). The full form of approximated cost function in (4.23) can be written as

$$\mathcal{J}_{AMA}(\dot{\gamma}) \approx \frac{1}{48d^4}C_4\dot{\gamma}^4 + \frac{1}{12d^3}C_3\dot{\gamma}^3 - \frac{1}{4d^2}C_2\dot{\gamma}^2 + \frac{1}{2d}C_1\dot{\gamma}^1 + \frac{1}{2}C_0 \quad (4.34)$$

where the coefficients in (4.34) are obtained by applying summation over all the

coefficients in (4.28) and (4.33) as shown below

$$\begin{aligned}
C_l &= \sum_{i=1}^{N_s} \left(c_l^{pi} + c_l^{q+N_t,i} + c_l^{qi} + c_l^{p+N_t,i} \right) \\
C_3 &= \sum_{i=1}^{N_s} \left(c_3^{pi} + c_3^{q+N_t,i} - c_3^{qi} - c_3^{p+N_t,i} \right) \\
C_1 &= \sum_{i=1}^{N_s} \left(-c_1^{pi} - c_1^{q+N_t,i} + c_1^{qi} + c_1^{p+N_t,i} \right)
\end{aligned} \tag{4.35}$$

where $l \in \{0, 2, 4\}$.

The final solution is obtained by taking the gradient and following the same procedure as explained before, where the gradient of (4.34) is written as

$$\frac{\partial \mathcal{J}_{\text{AMA}}(\dot{\gamma})}{\partial \dot{\gamma}} \approx \frac{1}{12d^4} C_4 \dot{\gamma}^3 + \frac{1}{4d^3} C_3 \dot{\gamma}^2 - \frac{1}{2d^2} C_2 \dot{\gamma} + \frac{1}{2d} C_1 \tag{4.36}$$

Once we obtain the solution $(\dot{\gamma}^\circ)$, the hyperbolic rotation matrices $\mathcal{H}_{p,q+N_t}(\dot{\gamma}^\circ)$ and $\mathcal{H}_{q,p+N_t}(-\dot{\gamma}^\circ)$ are computed using (3.2). The separation matrix $\hat{\mathbf{V}}$ is then updated according to (4.19).

In summary, pre-filtered separation matrix $\hat{\mathbf{V}}$ is initialized as identity matrix i.e., $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$. Then, G-MMA is applied for 5 sweeps followed by the update of the matrix $\hat{\mathbf{V}}$ according to (4.19) by applying Givens and hyperbolic rotations successively on modified $\hat{\mathbf{Y}}$ using the above explained method, until convergence. The overall algorithm is summarized in Table 4.3.

Table 4.3: Hyperbolic Givens AMA (HG-AMA) Algorithm

<p>Initialization: $\hat{\mathbf{V}} = \mathbf{I}_{2N_t}$ Subspace projection or approximate pre-whitening using (2.10) if $N_r > N_t$ 1. Construct real matrix $\hat{\mathbf{Y}}$ using (3.10) 2. Hyperbolic & Givens Rotations: for $n = 1 : N_{Sweeps}$ do if $n \leq 5$ then a) Apply G-MMA as given in Table 3.1 else for $p = 1 : N_t - 1$ do for $q = p + 1 : N_t$ do b) Find optimum (γ°) using roots of (4.32) which gives minimum value of (4.21) c) Compute $\mathcal{H}_{p,q}$ & $\mathcal{H}_{p+N_t,q+N_t}$ using (3.2) for same (γ°) d) $\hat{\mathbf{Y}} = \mathcal{H}_{p,q} \mathcal{H}_{p+N_t,q+N_t} \hat{\mathbf{Y}}$ e) $\hat{\mathbf{V}} = \mathcal{H}_{p,q} \mathcal{H}_{p+N_t,q+N_t} \hat{\mathbf{V}}$ f) Apply Givens rotations using (b to e) of Table 4.2 repeat steps (b to f) for $(p, q + N_t)$ & $(q, p + N_t)$ using $(\theta^\circ, \dot{\gamma}^\circ)$ & $(\theta^\circ, -\dot{\gamma}^\circ)$, respectively end for end for end if end for 3. Construct complex matrix \mathbf{W} similar to \mathbf{V} using (2.14) and (3.10) 4. Estimated Sources: $\hat{\mathbf{S}} = \mathbf{W}\mathbf{Y}$</p>

4.3 Practical Considerations

We provide here some comments to get more insight into the proposed algorithms.

4.3.1 Numerical Cost

Taking into account the structure of the rotation matrices, the numerical cost of the proposed algorithms are compared with other CMA-like BSS algorithms in terms of the number of flops per sweep in Table 4.4. Note that a flop corresponds to a real multiplication and a real addition. As can be seen from Table 4.4, the

Table 4.4: Numerical complexity comparison of different BSS algorithms

BSS Algorithm	Complexity Order
HG-AMA	$140N_sN_t^2 + \mathcal{O}(N_sN_t)$
G-AMA	$70N_sN_t^2 + \mathcal{O}(N_sN_t)$
HG-MMA	$40N_sN_t^2 + \mathcal{O}(N_sN_t)$
G-MMA	$20N_sN_t^2 + \mathcal{O}(N_sN_t)$
HG-CMA	$30N_sN_t^2 + \mathcal{O}(N_sN_t)$
G-CMA	$15N_sN_t^2 + \mathcal{O}(N_sN_t)$
ACMA	$\mathcal{O}(N_sN_t^4)$

proposed algorithms are much cheaper than ACMA and of the same cost order with G-CMA and HG-CMA. Moreover, the proposed algorithms have very fast convergence (typically less than 10 sweeps) as shown next in simulation experiments. Also, it can be noticed that HG-AMA is more expensive but still in terms of performance it is much better than all the other algorithms as can be observed from the simulations results presented next.

All considered BSS algorithms use a pre-whitening operation which costs $\mathcal{O}(N_sN_r^2)$ flops. The numerical cost of G-MMA and HG-MMA in Table 4.4 has

to be multiplied by the number of sweeps to obtain the overall cost.

4.3.2 Adaptive implementation

The numerical cost of the designed iterative batch algorithms increases linearly with the sample size N_s . Furthermore, in real life environments, systems are time varying and hence the separation matrix \mathbf{W} has to be re-estimated or updated along the time axis. Thus, for slowly time varying systems, this update can be obtained by using adaptive estimation methods. Utilizing a sliding window technique as in [23], one can achieve such source separation in an adaptive manner with a numerical cost proportional to $\mathcal{O}(\dot{N}_s N_t^2)$ where \dot{N}_s is the window size (instead of total sample size N_s).

4.3.3 Complex implementation

As shown in section 3.3.1, the real matrix representation has been introduced to overcome the difficulties encountered for the optimization of parameters of complex Givens and hyperbolic rotations. However, we can observe that the obtained results can be cast into complex matrix forms using the following straightforward relations:

$$\begin{aligned}
\mathcal{G}_{p,q}(\theta) \mathcal{G}_{p+N_t,q+N_t}(\theta) \underline{\dot{\mathbf{Y}}} &\iff \mathcal{G}_{p,q}(\theta, 0) \underline{\mathbf{Y}} \\
\mathcal{H}_{p,q}(\gamma) \mathcal{H}_{p+N_t,q+N_t}(\gamma) \underline{\dot{\mathbf{Y}}} &\iff \mathcal{H}_{p,q}(\gamma, 0) \underline{\mathbf{Y}} \\
\mathcal{G}_{p,q+N_t}(\dot{\theta}) \mathcal{G}_{q,p+N_t}(\dot{\theta}) \underline{\dot{\mathbf{Y}}} &\iff \mathcal{G}_{p,q}(\theta, -\frac{\pi}{2}) \underline{\mathbf{Y}} \\
\mathcal{H}_{p,q+N_t}(\dot{\gamma}) \mathcal{H}_{q,p+N_t}(\dot{\gamma}) \underline{\dot{\mathbf{Y}}} &\iff \mathcal{H}_{p,q}(\gamma, -\frac{\pi}{2}) \underline{\mathbf{Y}}
\end{aligned} \tag{4.37}$$

where all the matrices on left side of (4.37) are real and the right ones are complex.

Note that: Somehow, we have replaced the two degrees of freedom of complex rotations $\mathcal{G}_{p,q}(\theta, \alpha)$ (resp. $\mathcal{H}_{p,q}(\gamma, \beta)$) by the two free parameters θ and $\dot{\theta}$ (resp. γ and $\dot{\gamma}$). This way we have avoided the complex non-linear optimization problem discussed in section 3.3.1.

4.3.4 Performance

The main advantage of the proposed algorithms resides in their fast convergence in terms of the number of sweeps (typically less than 10 sweeps are needed for convergence) and also in terms of sample size (typically $N_s = \mathcal{O}(10N_t)$ is sufficient for the algorithm's convergence). Comparatively, the ACMA method requires $N_s = \mathcal{O}(10N_t^2)$ samples for its convergence and standard CMA-like methods need even more samples to converge to their steady state.

4.4 Simulation Results

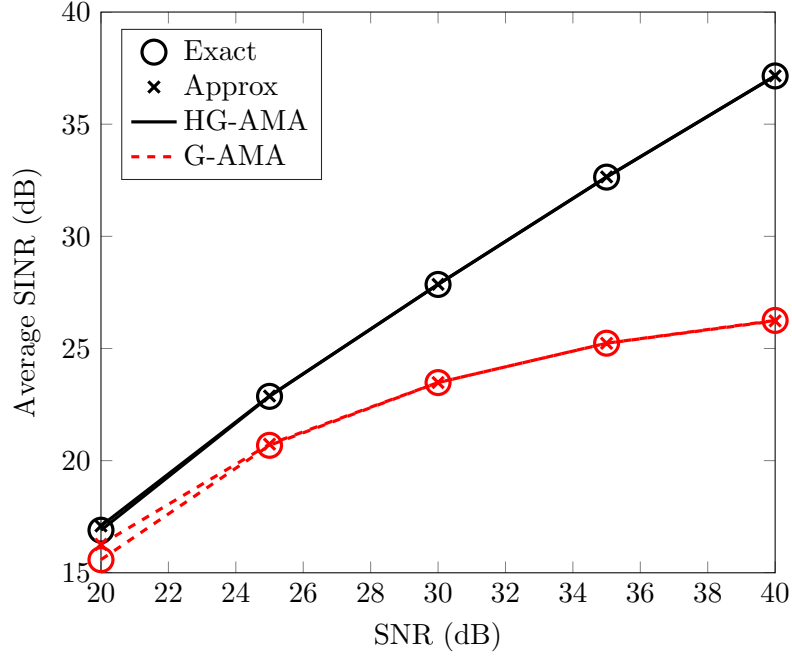
In order to evaluate the performance of the proposed algorithms, simulation results are presented in this section. Here, we have shown a comparison with our batch BSS algorithms presented in Chapter 3 i.e., G-MMA and HG-MMA, which deals with the MM criterion and performs better than contemporary batch BSS algorithms such as ACMA, G-CMA and HG-CMA. As a performance measure, SINR, convergence rate and SER are used, where SINR is defined in (3.41).

We consider a MIMO system consisting of 5 transmitters and 7 receivers ($N_t =$

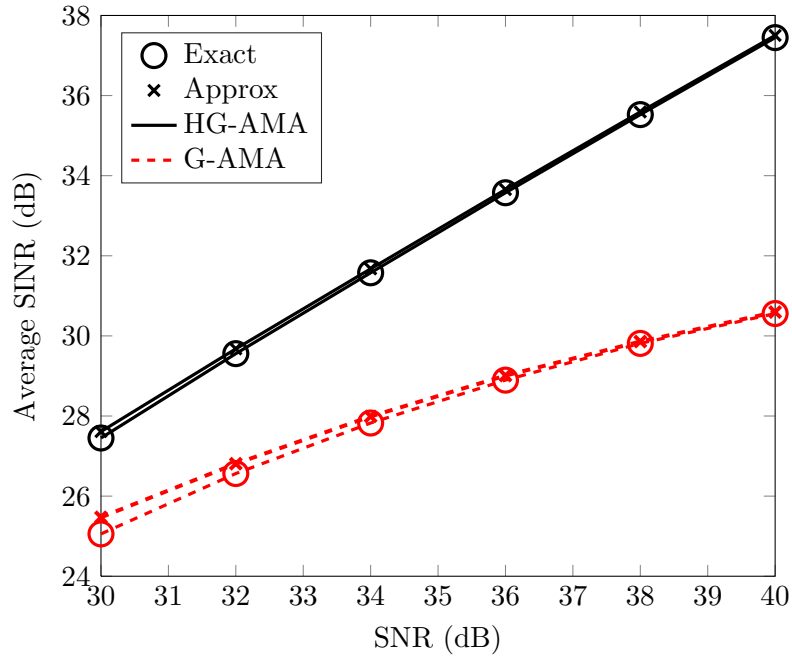
5, $N_r = 7$) with the data model given in Section 2.1. Every uncoded data symbol transmitted by each source are drawn from 64-QAM and 256-QAM constellations. The resulting signals are then passed through a channel matrix \mathbf{A} , generated randomly at each Monte Carlo run with controlled conditioning and with i.i.d complex Gaussian variable entries of zero mean and unity variance. The noise variance is adjusted according to specified signal to noise ratio (SNR). Further, sources, noise and channel have the same properties as specified in Section 2.2.1. The results are averaged over 1000 Monte Carlo runs.

4.4.1 Experiment 1: Exact vs. Approximate Solution of G-AMA and HG-AMA

In this experiment, we compare the performance of exact and approximate solutions presented for G-AMA and HG-AMA in terms of SINR vs. SNR. Figure 4.6a and 4.6b shows the plots for 64-QAM and 256-QAM constellations, respectively. The number of sweeps N_{Sweeps} is fixed at 10, where we used 5 sweeps of G-MMA followed by 5 sweeps of AMAs. The number of samples N_s is selected as 200 and 500 for 64-QAM and 256-QAM, respectively. From Figure 4.6, we notice that both the exact and approximate solutions have the same performance for the considered constellations. Therefore, in the following simulations for the G-AMA and HG-AMA, we will use the approximate solution, as it is cheaper and easier to implement.



(a) 64-QAM, $N_s = 200$



(b) 256-QAM, $N_s = 500$

Figure 4.6: Average SINR of exact and approximate solution of HG-AMA and G-AMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 10$.

4.4.2 Experiment 2: Finding Optimum Number of Sweeps

Here, we examine the effect of the number of sweeps N_{Sweeps} on the performance of the G-AMA and HG-AMA. Figure 4.7 compares the SINR vs. SNR for different number of sweeps. In this simulation, $N_s = 200$ symbols are drawn from 64-QAM constellation. We notice that the performance of proposed algorithms increases with the number of sweeps and remains almost unchanged after 8 sweeps (5 G-MMA + 3 AMA sweeps). So, in the following simulations we will fix the number of sweeps to 8.

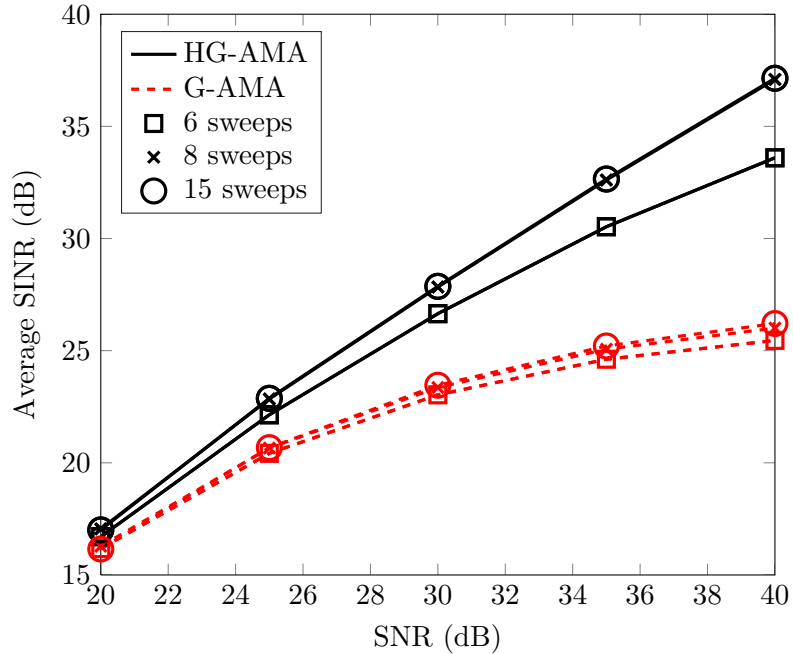


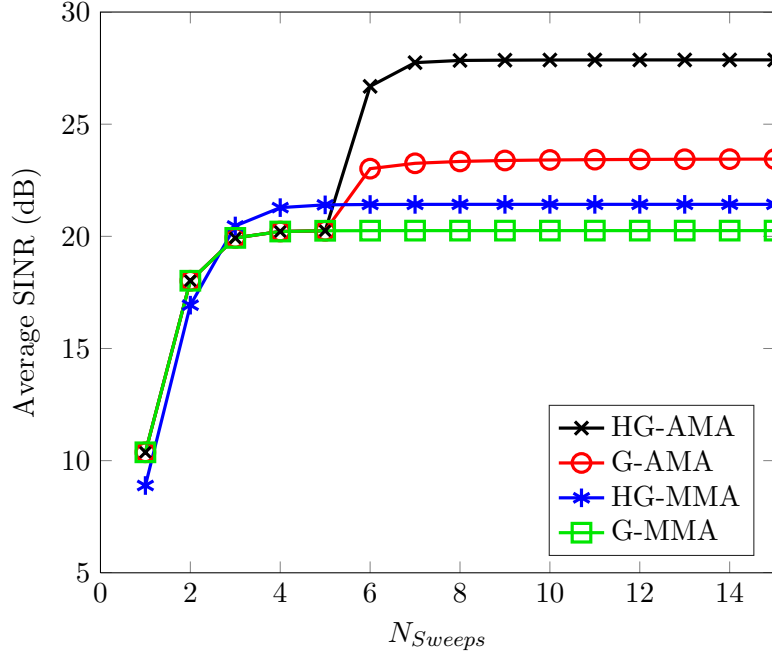
Figure 4.7: Average SINR of HG-AMA and G-AMA vs. SNR for different N_{Sweeps} considering $N_t = 5$, $N_r = 7$, $N_s = 200$ and 64-QAM constellation.

4.4.3 Experiment 3: Comparison of Rate of Convergence

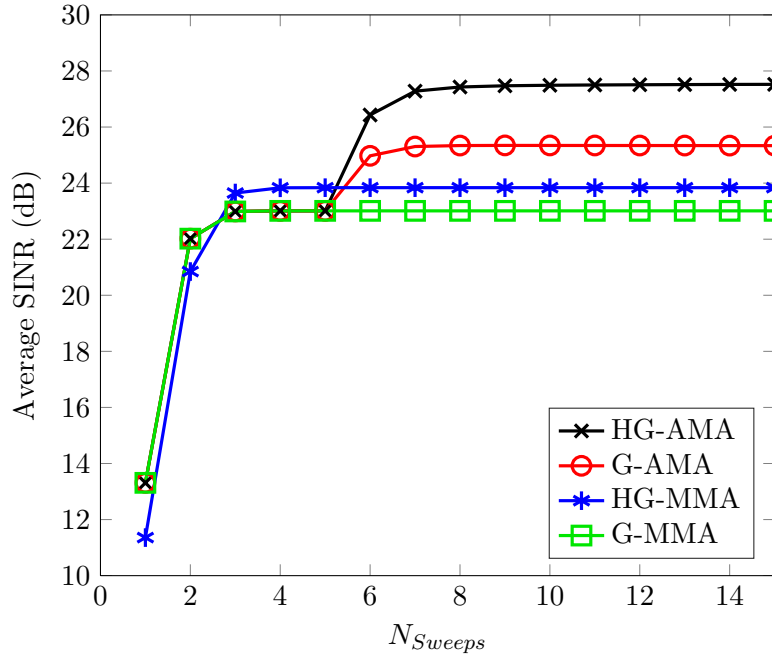
In Figure 4.8, we have compared the convergence rate of the proposed algorithms with G-MMA and HG-MMA. Note that all of them are iterative algorithms. The SNR is fixed at 30 dB and N_s is selected as 200 and 500 for 64-QAM and 256-QAM, respectively. It can be noticed that for the considered case, G-MMA and HG-MMA converge in 5 sweeps, while G-AMA and HG-AMA converges in 8 sweeps. Even though the proposed algorithms G-AMA and HG-AMA require 3 extra sweeps, the performance is much better than the HG-MMA and G-MMA.

4.4.4 Experiment 4: Effect of the Number of Samples

Figure 4.9a and 4.9b, show the SINR performance of our proposed algorithms vs. the number of samples N_s for 64-QAM and 256-QAM constellations, respectively. The SNR and the total number of sweeps N_{Sweeps} are fixed at 30 dB, and 8, respectively. It can be noticed that as expected, the larger the number of samples the better the performance of proposed algorithms. However, we observe a threshold point after which the gain is not significant as the SINR will be essentially limited by the SNR value. It can be seen that the performance of AM algorithms is better than MM algorithms. Also, HG-AMA takes the lead among all other algorithms. HG-AMA reaches a maximum SINR of 28 dB in $N_s = 250$ samples for 64-QAM and $N_s = 600$ samples for 256-QAM.

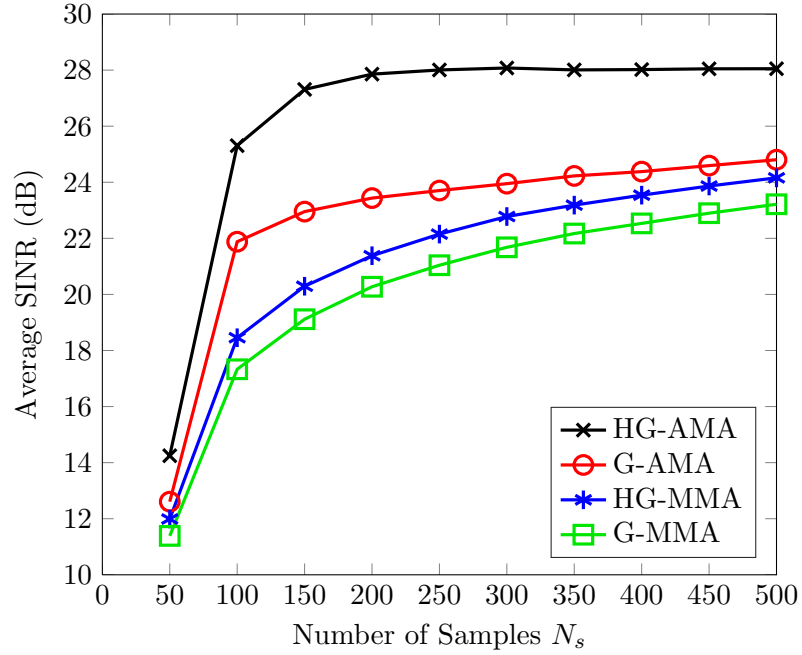


(a) 64-QAM, $N_s = 200$

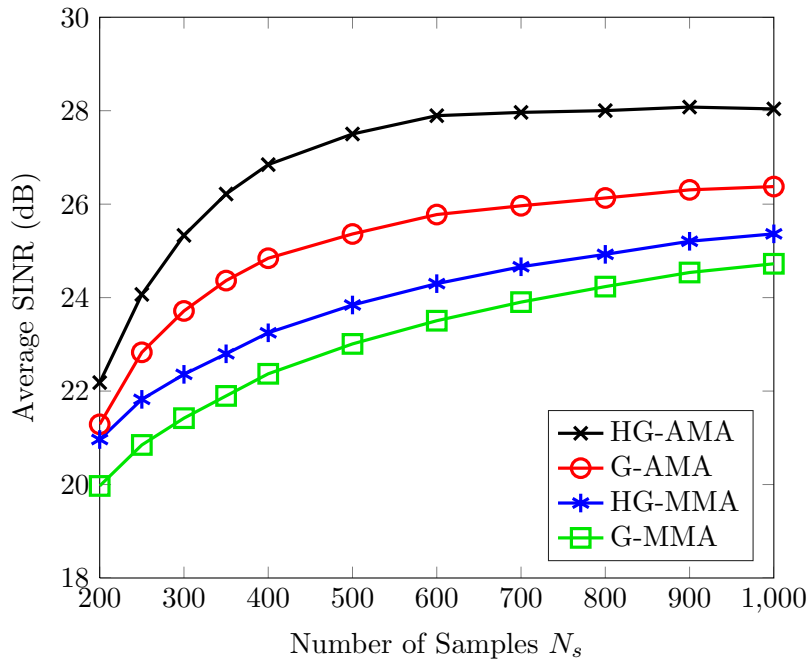


(b) 256-QAM, $N_s = 500$

Figure 4.8: Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. N_{Sweeps} for $N_t = 5$, $N_r = 7$ and SNR = 30dB.



(a) 64-QAM



(b) 256-QAM

Figure 4.9: Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. the number of samples (N_s) for $N_t = 5$, $N_r = 7$, SNR = 30dB and $N_{Sweeps} = 8$.

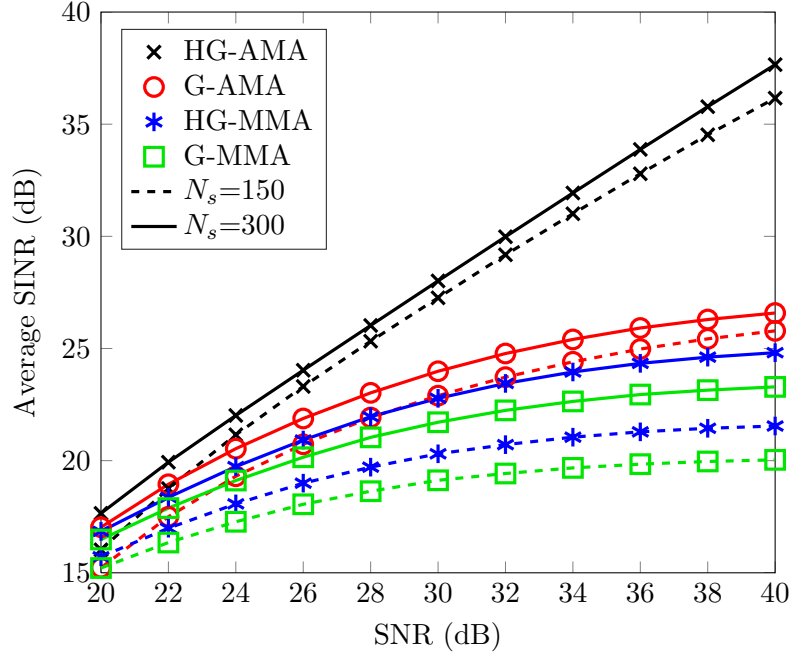
4.4.5 Experiment 5: Comparison based on SINR

Figure 4.10a compares the SINR performance of the AM and MM algorithms as a function of SNR. In this figure, two different number of samples ($N_s = 150$ and $N_s = 300$) are considered for 64-QAM constellation. As noticed previously, all algorithms perform better for large number of samples. Also, the difference between the performance of the HG-AMA and HG-MMA increases with the increase in number of samples. The G-AMA cannot perform well for small number of samples because of the ineffective pre-whitening operation. It is very clear from the figure that the MM algorithms are not suitable for this constellation. For an SNR higher than 22dB, the performance of HG-AMA for small number of samples i.e., $N_s = 150$ is better than G-AMA even with large number of samples i.e., $N_s = 300$.

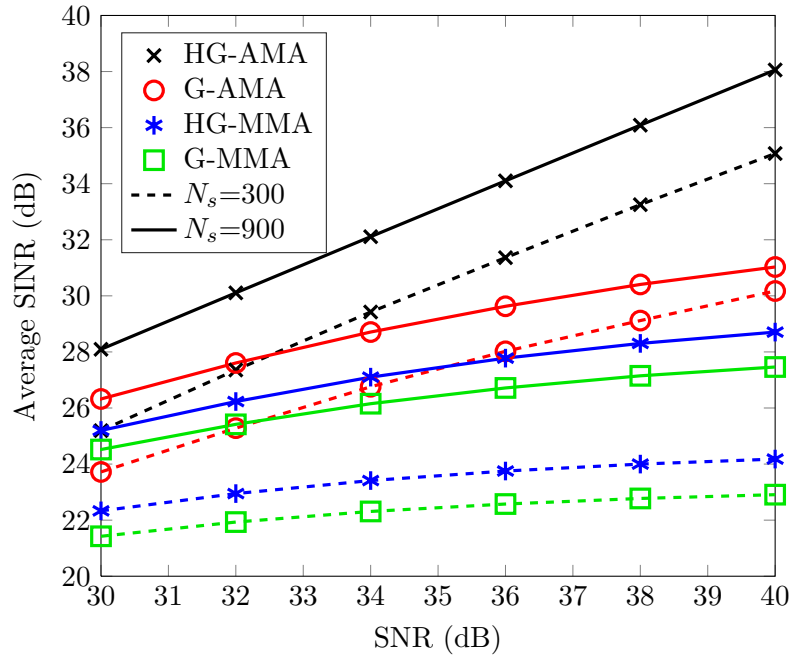
In Figure 4.10b, we consider the case of 256-QAM constellation with two different number of samples ($N_s = 300$ and $N_s = 900$). It is noticed that the performance of proposed algorithms is significantly better than other algorithms even for small number of samples N_s . For an SNR higher than 32dB, the performance of HG-AMA for small number of samples i.e., $N_s = 300$ is better than G-AMA even with large number of samples i.e., $N_s = 900$.

4.4.6 Experiment 6: Comparison based on SER

Figure 4.11a and 4.11b depict the SER of AM and MM algorithms vs. SNR for the case of 64-QAM and 256-QAM constellations, respectively. In both figures,



(a) 64-QAM, $N_s = 150$ and $N_s = 300$



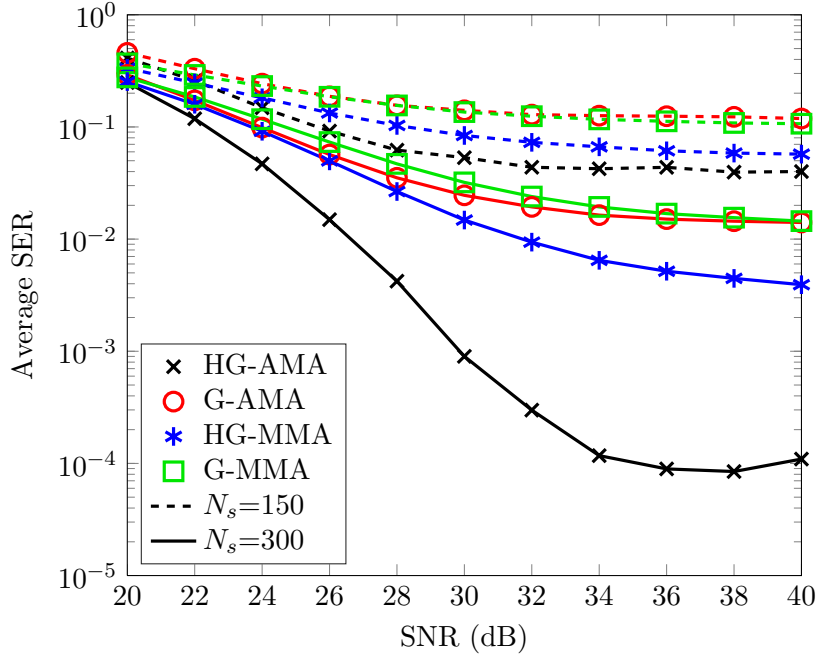
(b) 256-QAM, $N_s = 300$ and $N_s = 900$

Figure 4.10: Average SINR of HG-AMA, G-AMA, HG-MMA and G-MMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 8$ and different number of samples N_s .

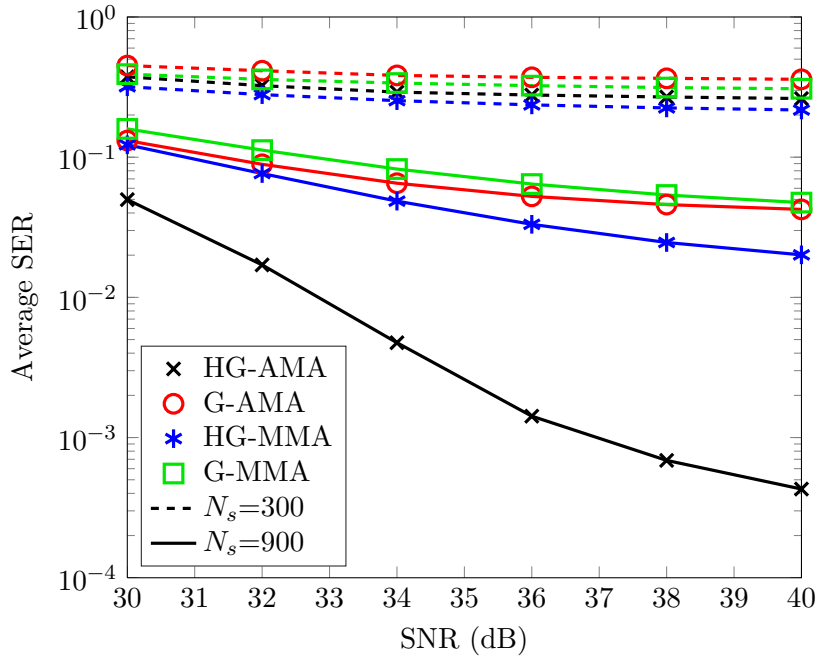
different number of samples are considered i.e., for 64-QAM ($N_s = 150$ and $N_s = 300$) and for 256-QAM ($N_s = 300$ and $N_s = 900$). As noticed previously, the performance of the HG-AMA is significantly better than all the other algorithms. Similar to other figures, same pattern of performance is observed i.e., the HG-AMA takes the lead followed by the G-AMA, HG-MMA and then by the G-MMA. Comparison of Figure 4.11a and 4.11b show that for small number of samples, the performance of all algorithms is nearly the same. However, for large number of samples HG-AMA performs better than every other algorithm. In fact, we can say that HG-AMA is the only algorithm which works very well for higher QAM constellation.

4.5 Chapter Conclusions

As per our conclusions from Chapter 3, the MM criterion is not suitable for higher QAM signals. Thus, we have reviewed AM criterion which incorporates information about the higher QAM constellations in a better way. It is known that the AM criterion has good local convergence properties and should be initialized with CMA/MMA. However, in our design we have used our algorithm G-MMA as an initialization then the algorithm is switched to AM criterion minimization. Thus, two new iterative batch BSS algorithms are presented namely the G-AMA and HG-AMA. The proposed algorithms are designed using a pre-whitening operation to reduce the complexity of optimization problem and then initialized with G-MMA, followed by a recursive separation method of unitary Givens and J-unitary



(a) 64-QAM, $N_s = 150$ and $N_s = 300$



(b) 256-QAM, $N_s = 300$ and $N_s = 900$

Figure 4.11: Average SER of HG-AMA, G-AMA, HG-MMA and G-MMA vs. SNR for $N_t = 5$, $N_r = 7$, $N_{Sweeps} = 8$ and different number of samples N_s .

hyperbolic rotations to minimize the AM criterion. Similar to G-MMA and HG-MMA, we have used real rotation matrices for the design of AM algorithms.

For the minimization of AM criterion using Givens and hyperbolic rotations, two possible solutions are presented. One of them uses MATLAB optimization toolbox ‘fminsearchbnd’ and ‘fminsearch’, which is named as ‘exact solution’. The other solution utilizes the trigonometric approximations around zero as we have observed that the optimum parameters of rotation matrices are very close to zero. Thus, this solution is named as ‘approximate solution’, which involves solving a simple 3^{rd} order polynomial equation.

The proposed algorithms are mainly designed for the blind deconvolution of MIMO systems involving higher QAM signals. Simulation results demonstrate their favorable performance as compared to G-MMA and HG-MMA. Thus, we can say that the newly designed algorithms G-AMA and HG-AMA performs better than all other contemporary batch BSS algorithms as well as G-MMA and HG-MMA.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusions

In this thesis, fundamental problems with the physical layer for MIMO systems are addressed. The targeted problems include channel estimation and blind deconvolution. Mainly, the problem focussed here is to design algorithms for higher QAM signals without using pilot symbols.

In the start, basic concepts related to blind source separation problem are presented. The literature review shows that there is a need of good batch BSS algorithms mainly for QAM signals, because such signals are highly used in modern communication systems. After the motivation behind this work, the model for MIMO communication system is presented. The underlying assumptions and general methodology of BSS algorithms are studied.

Next, the unitary Givens and J -unitary hyperbolic rotations are reviewed for the optimization of cost functions. The multimodulus (MM) criterion which is suitable for QAM signals is minimized using iterative Givens and hyperbolic rotations. During the design of algorithms, it has been found that the complex rotations results in complicated optimization problem, which is not easy to solve. Moreover, as MM criterion deals with the real and imaginary parts of signal separately, thus it is showed that using real Givens and hyperbolic rotations is more convenient than complex ones. Using real Givens and hyperbolic rotations, two iterative batch BSS algorithms are presented dealing with the MM criterion and thus named as G-MMA and HG-MMA. A MATLAB based simulation setup showed that the designed algorithms perform better than contemporary batch BSS algorithms for QAM signals. Also, these algorithms are less expensive and has better convergence rate.

It is observed during simulations that the designed and contemporary batch BSS algorithms do not provide satisfactory performance for higher QAM signals such as 64-QAM. Thus, two new algorithms are presented dealing with the alphabet matched (AM) criterion. For the design of these algorithms, the same optimization method of iterative Givens and hyperbolic rotations are used and thus named as G-AMA and HG-AMA. During the design, it has been found that the optimization is quite complicated involving several non-linear terms. Thus, an approximate solution using trigonometric approximations is presented, which is compared with the exact solution obtained using MATLAB optimization tool-

box. The comparison showed that the optimum optimization parameters are very close to zero. Thus, our approximation is valid and results in the same solution. These algorithms are compared with G-MMA and HG-MMA in terms of SINR, convergence rate, and SER. The comparison showed that G-AMA and specially HG-AMA is the most suitable algorithm for higher QAM signals such as 64-QAM and 256-QAM. HG-AMA is capable of blindly and efficiently separating a number of higher QAM signals in a MIMO communication system.

In summary, the proposed algorithms are mainly designed for the blind deconvolution of MIMO systems involving higher QAM signals. Simulation results demonstrate their favorable performance as compared to contemporary batch BSS algorithms.

5.2 Future Work

Following are the suggested topics for future work related to the work presented in this thesis.

1. This thesis deals with the batch BSS algorithms where the channel is assumed to be constant for a data packet consisting of some small number of samples. However, a number of communication system models includes mobility where the channel varies nearly at every symbol. Thus, adaptive algorithms are more suitable for these kind of scenarios. The presented algorithms can be modified for such time-varying channels using the concepts of adaptive BSS methods.

2. The performance of proposed algorithms are evaluated using simulation results only. Thus, one can use actual channel data to verify the effectiveness of proposed algorithms.
3. Similar optimization method can be used to design algorithms for non-square multimodulus signals for which a number of cost functions are available in literature.
4. While going through the literature, it has been found that an analytical method dealing with the MM criterion is derived but not optimized due to unavailability of joint diagonalization method for non-square matrices. However, the joint diagonalization technique using Givens and hyperbolic rotations is valid also for non-square matrices. Thus, it can be used to design a batch analytical BSS algorithm for MM and AM criterion.

APPENDIX A

In order to separate the real and imaginary parts of z_{pi} and z_{qi} given in (3.7), following equalities can be used

$$z_{pi,R} = \frac{z_{pi} + z_{pi}^H}{2} \quad z_{pi,I} = \frac{z_{pi} - z_{pi}^H}{2l} \quad (\text{A.1})$$

Using double angle identities, (A.1) and (3.7), we can write

$$z_{pi,R}^2 = \mathbf{g}_j^T \mathbf{c} + g_8 \quad (\text{A.2})$$

where

$$\mathbf{c} = \begin{bmatrix} \cos(2\theta) & \cos(2\theta) \cos(2\alpha) & \cos(2\theta) \sin(2\alpha) & \sin(2\theta) \cos(\alpha) \\ & \sin(2\theta) \sin(\alpha) & \cos(2\alpha) & \sin(2\alpha) \end{bmatrix}^T \quad (\text{A.3})$$

$$\mathbf{g}_j = \begin{bmatrix} g_1 \cdots g_7 \end{bmatrix}$$

with

$$\begin{aligned}
g_1 &= \frac{y_{pj,R}^2}{2} - \frac{y_{qj,R}^2}{4} - \frac{y_{qj,I}^2}{4} & g_2 &= \frac{y_{qj,I}^2}{4} - \frac{y_{qj,R}^2}{4} & g_3 &= \frac{y_{qj,R}y_{qj,I}}{2} \\
g_4 &= \frac{y_{pj,R}y_{qj,R}}{2} & g_5 &= -\frac{y_{pj,R}y_{qj,I}}{2} & g_6 &= \frac{y_{qj,R}^2}{4} - \frac{y_{qj,I}^2}{4} \\
g_7 &= -\frac{y_{qj,R}y_{qj,I}}{2} & g_8 &= \frac{y_{pj,R}^2}{2} + \frac{y_{qj,R}^2}{4} + \frac{y_{qj,I}^2}{4}
\end{aligned}$$

Remaining three terms can be found similarly having different elements of \mathbf{g}_j .

APPENDIX B

We will show here the solution of eigenvalue problem of the form

$$\mathcal{J} = \mathbf{v}^T \mathbf{T} \mathbf{v} \tag{B.1}$$

with the constraint $\|\mathbf{v}\|^2 = 1$, where \mathbf{v} is a column vector and \mathbf{T} is a square matrix. So, using Lagrange multiplier method, we can represent it as

$$\mathcal{L}(\mathbf{v}, \lambda) = \mathbf{v}^T \mathbf{T} \mathbf{v} - \lambda(\mathbf{v}^T \mathbf{v} - 1) \tag{B.2}$$

Taking the partial derivative of (B.2) with respect to \mathbf{v} and setting it to zero results in

$$\mathbf{v}^T \mathbf{T} = \lambda \mathbf{v}^T \tag{B.3}$$

which shows the \mathbf{v} is an eigenvector of matrix \mathbf{T} corresponding to the eigenvalue λ and must have a unit norm. Now, using (B.3) in (B.1) provides

$$\mathcal{J} = \lambda \mathbf{v}^T \mathbf{v} = \lambda \|\mathbf{v}\|^2 = \lambda \tag{B.4}$$

Thus, in order to minimize cost function \mathcal{J} , one should use minimum λ . This shows that the optimum \mathbf{v}° which minimizes \mathcal{J} in (B.1) is the unit norm eigenvector of matrix \mathbf{T} corresponding to the smallest eigenvalue.

APPENDIX C

In order to show that (3.31) given in subsection 3.3.3 is a 4-th order polynomial equation, let us consider the 2×2 matrices \mathbf{U} and $\mathbf{\Lambda} = \text{diag} \left(\begin{bmatrix} \lambda_1 & \lambda_2 \end{bmatrix} \right)$ be the generalized eigenvectors and eigenvalues matrices of the matrix pair $(\mathbf{R}, \mathbf{J}_2)$, i.e.,

$$\mathbf{R} = \mathbf{J}_2 \mathbf{U} \mathbf{\Lambda} \mathbf{U}^{-1} \quad (\text{C.1})$$

and hence

$$(\mathbf{R} + \lambda \mathbf{J}_2)^{-1} = \mathbf{U} (\mathbf{\Lambda} + \lambda \mathbf{I}_2)^{-1} \mathbf{U}^{-1} \mathbf{J}_2 \quad (\text{C.2})$$

Using (C.2) in (3.31), we get

$$\mathbf{r}^\top \mathbf{U} (\mathbf{\Lambda} + \lambda \mathbf{I}_2)^{-2} \mathbf{U}^{-1} \mathbf{J}_2 \mathbf{r} = \mathbf{a}^\top (\mathbf{\Lambda} + \lambda \mathbf{I}_2)^{-2} \mathbf{b} = 1 \quad (\text{C.3})$$

where

$$\begin{aligned} \mathbf{a}^\top &= \mathbf{r}^\top \mathbf{U} = \begin{bmatrix} a_1 & a_2 \end{bmatrix} \\ \mathbf{b} &= \mathbf{U}^{-1} \mathbf{J}_2 \mathbf{r} = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^\top \\ (\mathbf{\Lambda} + \lambda \mathbf{I}_2)^{-2} &= \text{diag} \left[(\lambda + \lambda_1)^{-2} \quad (\lambda + \lambda_2)^{-2} \right] \end{aligned}$$

So, (C.3) can be re-written as

$$\sum_{i=1}^2 \frac{a_i b_i}{(\lambda + \lambda_i)^2} = 1 \quad (\text{C.4})$$

which is equivalent to

$$\prod_{i=1}^2 (\lambda + \lambda_i)^2 - (a_1 b_1 (\lambda + \lambda_2)^2 + a_2 b_2 (\lambda + \lambda_1)^2) \quad (\text{C.5})$$

which is a 4-th order polynomial equation of the form

$$P_4(\lambda) = c_0 \lambda^4 + c_1 \lambda^3 + c_2 \lambda^2 + c_3 \lambda + c_4 = 0$$

with coefficients given as

$$c_0 = 1$$

$$c_1 = 2(\lambda_1 + \lambda_2)$$

$$c_2 = \sum_{i=1}^2 \lambda_i^2 + 4\lambda_1 \lambda_2 - \mathbf{a}^T \mathbf{b}$$

$$c_3 = 2 \sum_{i=1}^2 \left\{ (\lambda_i - a_i b_i) \sum_{j=1, j \neq i}^2 \lambda_j \right\}$$

$$c_4 = \lambda_1^2 \lambda_2^2 - \sum_{i=1}^2 a_i b_i \prod_{j=1, j \neq i}^2 \lambda_j^2$$

REFERENCES

- [1] M. Rahnema, “Overview of the GSM system and protocol architecture,” *IEEE Commun. Mag.*, vol. 31, no. 4, pp. 92–100, April 1993.
- [2] T. Y. Al-Naffouri, A. A. Dahman, M. S. Sohail, W. Xu, and B. Hassibi, “Low-complexity blind equalization for OFDM systems with general constellations,” *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6395–6407, 2012.
- [3] V. Kavitha and V. Sharma, “Comparison of training, blind and semi blind equalizers in MIMO fading systems using capacity as measure,” in *Proc. IEEE ICASSP*, vol. 3. IEEE, 2005, pp. iii/589–iii/592.
- [4] H. Lüders, A. Minwegen, B. Eschbach, and P. Vary, “An improvement of UMTS LTE 64QAM performance,” *European Trans. on Telecomm.*, vol. 21, no. 5, pp. 443–448, 2010.
- [5] M. Khairy, “A novel frequency offset estimation technique for mobile WiMAX,” *European Trans. on Telecomm.*, vol. 22, no. 1, pp. 45–50, 2011.

- [6] J. Allen and J. Mazo, "A decision-free equalization scheme for minimum-phase channels," *IEEE Trans. on Commun.*, vol. 22, no. 10, pp. 1732–1733, Oct 1974.
- [7] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Trans. on Commun.*, vol. 23, no. 6, pp. 679–682, Jun 1975.
- [8] J. Treichler and B. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. 31, no. 2, pp. 459–472, Apr 1983.
- [9] A. Benveniste and M. Goursat, "Blind equalizers," *IEEE Trans. on Commun.*, vol. 32, no. 8, pp. 871–883, Aug 1984.
- [10] K. Oh and Y. Chin, "Modified constant modulus algorithm: blind equalization and carrier phase recovery algorithm," in *Proc. IEEE ICC*, vol. 1, Jun. 1995, pp. 498–502.
- [11] A. J. van der Veen and A. Leshem, "Constant modulus beamforming," in *Robust Adaptive Beamforming*, (J. Li and P. Stoica, eds.). John Wiley & Sons, Inc., ch. 6, pp. 299–351, 2005.
- [12] S. Abrar, A. Ali, A. Zerguine, and A. Nandi, "Tracking performance of two constant modulus equalizers," *IEEE Commun. Lett.*, vol. 17, no. 5, pp. 830–834, 2013.

- [13] S. Abrar and A. K. Nandi, “Blind equalization of square-QAM signals: a multimodulus approach,” *IEEE Trans. Commun.*, vol. 58, no. 6, pp. 1674–1685, June 2010.
- [14] A. W. Azim, S. Abrar, A. Zerguine, and A. K. Nandi, “Performance analysis of a family of adaptive blind equalization algorithms for square-QAM,” *Digital Signal Processing*, vol. 48, pp. 163 – 177, 2016.
- [15] S. Abrar, A. Zerguine, and A. K. Nandi, “Blind adaptive carrier phase recovery for QAM systems,” *Digital Signal Processing*, 2015.
- [16] J. Treichler and M. Larimore, “New processing techniques based on the constant modulus adaptive algorithm,” *IEEE Trans. Acoust., Speech and Signal Process.*, vol. 33, no. 2, pp. 420–431, Apr 1985.
- [17] R. Gooch and J. Lundell, “The CM array: An adaptive beamformer for constant modulus signals,” in *Proc. IEEE ICASSP*, vol. 11, Apr 1986, pp. 2523–2526.
- [18] C. B. Papadias, “Globally convergent blind source separation based on a multiuser kurtosis maximization criterion,” *IEEE Trans. Signal Process.*, vol. 48, no. 12, pp. 3508–3519, Dec 2000.
- [19] A. J. van der Veen and A. Paulraj, “An analytical constant modulus algorithm,” *IEEE Trans. Signal Process.*, vol. 44, no. 5, pp. 1136–1155, May 1996.

- [20] A. J. van der Veen, “Asymptotic properties of the algebraic constant modulus algorithm,” *IEEE Trans. Signal Process.*, vol. 49, no. 8, pp. 1796–1807, Aug 2001.
- [21] —, “An adaptive version of the algebraic constant modulus algorithm [blind source separation applications],” in *Proc. IEEE ICASSP*, vol. 4, March 2005, pp. iv/873–iv/876.
- [22] A. Ikhlef, K. Abed-Meraim, and D. Le Guennec, “On the constant modulus criterion: A new algorithm,” in *Proc. IEEE ICC*, May 2010, pp. 1–5.
- [23] A. Ikhlef, R. Iferroujene, A. Boudjellal, K. Abed-Meraim, and A. Belouchrani, “Constant modulus algorithms using hyperbolic Givens rotations,” *Signal Process.*, vol. 104, pp. 412 – 423, 2014.
- [24] A. Boudjellal, K. Abed-Meraim, A. Belouchrani, and P. Ravier, “Adaptive constant modulus algorithm based on complex givens rotations,” in *IEEE Workshop on Stat. Signal Process.*, June 2014, pp. 165–168.
- [25] P. Sansrimahachai, D. B. Ward, and A. G. Constantinides, “Blind source separation for BLAST,” in *Proc. IEEE DSP*, vol. 1, 2002, pp. 139–142.
- [26] S. Daumont and D. Le Guennec, “An analytical multimodulus algorithm for blind demodulation in a time-varying MIMO channel context,” *International journal of digital multimedia broadcasting*, 2010.
- [27] Y. Fadlallah, A. Aissa-El-Bey, K. Abed-Meraim, K. Amis, and R. Pyndiah, “Semi-blind source separation in a multi-user transmission system with inter-

- ference alignment,” *IEEE Wireless Commun. Lett.*, vol. 2, no. 5, pp. 551–554, October 2013.
- [28] G. Bienvenu and L. Kopp, “Optimality of high resolution array processing using the eigensystem approach,” *IEEE Trans. Acoust., Speech and Signal Process.*, vol. 31, no. 5, pp. 1235–1248, 1983.
- [29] J.-F. Cardoso and A. Souloumiac, “Jacobi angles for simultaneous diagonalization,” *SIAM Journal on Mat Anal. Appl.*, vol. 17, no. 1, pp. 161–164, Jan. 1996.
- [30] A. Mesloub, K. Abed-Meraim, and A. Belouchrani, “A new algorithm for complex non-orthogonal joint diagonalization based on Shear and Givens rotations,” *IEEE Trans. Signal Process.*, vol. 62, no. 8, pp. 1913–1925, April 2014.
- [31] A. H. Sayed, *Adaptive filters*. John Wiley & Sons, 2011.
- [32] —, *Fundamentals of adaptive filtering*. John Wiley & Sons, 2003.
- [33] G. Picchi and G. Prati, “Blind equalization and carrier recovery using a ”stop-and-go” decision-directed algorithm,” *IEEE Trans. Commun.*, vol. 35, no. 9, pp. 877–887, Sep 1987.
- [34] J. Yang, J.-J. Werner, and G. Dumont, “The multimodulus blind equalization and its generalized algorithms,” *IEEE J. Sel. Areas Commun.*, vol. 20, no. 5, pp. 997–1015, 2002.

- [35] J.-T. Yuan and K.-D. Tsai, “Analysis of the multimodulus blind equalization algorithm in QAM communication systems,” *IEEE Trans. Commun.*, vol. 53, no. 9, pp. 1427–1431, Sep 2005.
- [36] X.-L. Li and W.-J. Zeng, “Performance analysis and adaptive Newton algorithms of multimodulus blind equalization criterion,” *Signal Process.*, vol. 89, no. 11, pp. 2263–2273, 2009.
- [37] M. Mizuno and J. Okello, “A high throughput pipelined architecture for blind adaptive equalizer with minimum latency,” *IEICE Trans. Fund. Elect. Commun. Comp. Sci.*, vol. 86, no. 8, pp. 2011–2019, 2003.
- [38] G. H. Golub and C. F. Van Loan, *Matrix Computations*, ser. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 1996.
- [39] S. A. W. Shah, K. Abed-Meraim, and T. Y. Al-Naffouri, “Multi-modulus algorithms using hyperbolic and Givens rotations for blind deconvolution of MIMO systems,” in *Proc. IEEE ICASSP*, Apr. 2015, pp. 2155–2159.
- [40] S. A. W. Shah, K. Abed-Meraim, and T. Y. Al-Naffouri, “Multi-modulus algorithms using hyperbolic and Givens rotations for MIMO deconvolution,” *ArXiv e-prints arXiv:1506.06650 [cs.IT]*, Jun. 2015.
- [41] L. He, M. G. Amin, C. Reed Jr., and R. C. Malkemes, “A hybrid adaptive blind equalization algorithm for QAM signals in wireless communications,” *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 2058–2069, 2004.

- [42] A. Beasley and A. Cole-Rhodes, "Blind adaptive equalization for QAM signals using an alphabet-matched algorithm," in *Proc. IEEE GLOBECOM*, 2006, pp. 1–5.
- [43] T.-H. Li, "Blind deconvolution of linear systems with multilevel nonstationary inputs," *Ann. Stat.*, vol. 23, no. 2, pp. 690–704, 04 1995.
- [44] T.-H. Li and K. Mbarek, "A blind equalizer for nonstationary discrete-valued signals," *IEEE Trans. Signal Process.*, vol. 45, no. 1, pp. 247–254, Jan 1997.
- [45] S. Barbarossa and A. Scaglione, "Blind equalization using cost function matched to the signal constellation," in *Proc. Asilomar Conf. Signals Syst. Comput.*, vol. 1, Nov 1997, pp. 550–554 vol.1.
- [46] L. He, M. G. Amin, and C. Reed Jr., "Adaptive equalization techniques for indoor dynamic wireless communication channels," in *Proc. SPIE*, vol. 4395, Apr. 2001, pp. 28–38.
- [47] M. Amin, L. He, C. Reed Jr., and R. Malkemes, "A modified constant modulus algorithm for adaptive channel equalization for qam signals," in *Proc. IEEE Signal Process. Workshop on Stat. Signal Process.* IEEE, 2001, pp. 563–566.
- [48] L. He, M. Amin, C. Reed Jr., and R. Malkemes, "A hybrid adaptive blind equalization algorithm for QAM signals in wireless communications," *IEEE Trans. Signal Process.*, vol. 52, no. 7, pp. 2058–2069, July 2004.

- [49] A. Labeled, T. Chonavel, A. Aïssa-El-Bey, and A. Belouchrani, “Min-norm based alphabet-matching algorithm for adaptive blind equalisation of high-order QAM signals,” *Trans. on Emerging Telecomm. Tech.*, vol. 24, no. 6, pp. 552–556, 2013.

VITAE

Syed Awais Wahab Shah

Date of Birth: 05-March-1990
Nationality: Pakistani
E-mail: awaiswahab100@gmail.com
LinkedIn: pk.linkedin.com/in/syedawais

EDUCATION

Master of Science in Electrical Engineering (Sep. 2013 – Dec. 2015)

Institution KFUPM, Dhahran, Saudi Arabia
Dissertation Designing Blind Source Separation algorithms for MIMO Systems
Advisors Dr. Tareq Y. Al-Naffouri and Prof. Karim Abed-Meraim
Courses Stochastic Processes, Digital Communication I, Digital Signal Processing I, Adaptive Filtering/Application, Image Processing, Design of Digital Systems, Computer Networks
CGPA 3.857/4

Bachelors of Science in Electrical Engineering (Feb. 2008 – Jan. 2012)

Institution COMSATS Institute of Information Technology (CIIT), Islamabad, Pakistan
Majors Telecommunications
Awards **2 Gold Medals (Campus Gold Medal and Chancellor Gold Medal)**
CGPA 3.88/4 (RANK - 1st)

ACADEMIC EXPERIENCE

Visiting Research Scholar (Jun – Jul, 2014) , (Jan – May, 2015) & (Oct – Dec, 2015)

Institution King Abdullah University of Science and Technology (KAUST), Saudi Arabia
Supervisor Dr. Tareq Y. Al-Naffouri
Description Designed multi-modulus algorithms using Givens and hyperbolic rotations for square QAM signals in MIMO systems

Visiting Research Scholar (Jun. 2015 – Aug. 2015)

Institution Delft University of Technology (TU Delft), Netherlands
Supervisor Prof. Geert Leus
Description Worked on the topic of Spatial Modulation and Compressed Sensing

Lab Training (Spring 2014)

Institution King Fahd University of Petroleum and Minerals (KFUPM), Saudi Arabia
Course Communications Engineering I

PROFESSIONAL EXPERIENCE

Operations Engineer

(Aug. 2012 – Aug. 2013)

Company	Nayatel (Pvt.) Ltd./Micronet Broadband, Islamabad, Pakistan
Department	CORE Technical Assistance Center (CORE-TAC)

PUBLICATIONS

Journal	<ol style="list-style-type: none">1. S. A. W. Shah, K. Abed-Meraim, T. Y. Al-Naffouri, “A New Analytical Multi-modulus Algorithm for blind demodulation of MIMO systems”, under preparation2. S. A. W. Shah, K. Abed-Meraim, T. Y. Al-Naffouri, “Alphabet Matched Algorithms using Hyperbolic and Givens rotations for MIMO deconvolution”, under preparation3. S. A. W. Shah, K. Abed-Meraim, T. Y. Al-Naffouri, “Multi-modulus algorithms using Hyperbolic and Givens rotations for MIMO deconvolution”, submitted to IEEE Trans. Signal Process, June 2015.
Conference	S. A. W. Shah , K. Abed-Meraim, T. Y. Al-Naffouri, “Multi-Modulus algorithms using Hyperbolic and Givens rotations for blind deconvolution of MIMO systems”, in <i>Proc. IEEE ICASSP, April 2015</i> .
Conference Posters	S. A. W. Shah , K. Abed-Meraim, T. Y. Al-Naffouri, “Multi-Modulus algorithms for blind deconvolution of MIMO systems”, in King Abdullah University of Science and Technology – National Science Foundation Conference, 2015.

SKILLS

Languages	Urdu and English (IELTS overall band score: 6.5 (2012))
Management Skills	<ul style="list-style-type: none">• Organizer of Inter-University event CEPEX 2011• Decoration - Team Lead of Uni. event VIRSA (Reviving the Culture)
Computer Skills	Microsoft Office (Word, Excel, PowerPoint, Outlook, Visio) and LaTeX
Programming Languages	C++, MATLAB, Verilog and JOOMLA (Web Development)
Technical Skills	HFSS (Antenna Designing), WinAVR Studio (Microcontroller), Proteus (Circuit Designing), Xilinx (FPGA), AutoCAD (Hardware Design)
Interests	Traveling, Swimming and Football

AWARDS & HONORS

- **Fully Funded Scholarship** for postgraduate studies at KFUPM, Saudi Arabia
- **Campus Gold Medal** (1st position in EE dept. CIIT, Islamabad, Pakistan)
- **Chancellor Gold Medal** (1st position in EE dept. among all campuses of CIIT, Pakistan)