



ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
FACULTY OF TECHNOLOGY
DEPARTMENT OF ELECTRICAL AND COMPUTER
ENGINEERING

ADAPTIVE ANTENNA ARRAY ALGORITHMS
AND THEIR IMPACT ON
CODE DIVISION MULTIPLE ACCESS SYSTEMS
(CDMA).

By

Dereje Hadgu Hailu.

ADDIS ABABA

March 2003



ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES
FACULTY OF TECHNOLOGY
DEPARTMENT OF ELECTRICAL AND COMPUTER
ENGINEERING

ADAPTIVE ANTENNA ARRAY ALGORITHMS
AND THEIR IMPACT ON
CODE DIVISION MULTIPLE ACCESS SYSTEMS
(CDMA).

By Dereje Hadgu.

A thesis submitted to the faculty of Graduate studies, Addis Ababa University in partial fulfillment of the requirements for the Degree of Master of Science in Electrical Engineering.

Advisor Dr.Ing.-Mohammed Abdo.

ADDIS ABABA

March 2003

**ADAPTIVE ANTENNA ARRAY ALGORITHMS
AND THEIR IMPACT ON
CODE DIVISION MULTIPLE ACCESS SYSTEMS
(CDMA).**

By

Dereje Hadgu.

Addis Ababa University, School of Graduate Studies, Faculty of Technology,
Department of Electrical and Computer Engineering

Approved by Examination Committee

Name and signature

1. _____

Chairman of the Examination Committee

2. _____

Advisor

3. _____

Internal Examiner

4. _____

External Examiner

ACKNOWLEDGEMENTS

A long journey has come to an end and Thanks to the ALMIGHTY GOD with HIS help all can be done.

I would like to thank my academic advisor, Dr.Ing. - MOHAMMED ABDO for the invaluable Support and encouragement he has provided. I wish to thank all the people who crossed my path of life, who influenced me in ways I realized and in ways I did not. My stay at the university has been technically very satisfying. I would like to thank all my friends who have provided me all kinds of help till the end of the project.

Most of all, I would like to thank my parents and family for their love and encouragement. Specially my mother MULU G/YOHANNES and all my sisters. I could not have completed my degree without their continuous and immeasurable support.

Dereje Hadgu.
March 2003

List of Abbreviations

This is an attempt to summarize a number of abbreviations and expressions used in the Research.

{1,2,3} G	First, second and third Generation Mobile Telecommunications.
3GPP	Third Generation Partnership Project
GPS	Global Positioning System.
WCDMA	Wideband Code Division Multiple Access.
GSM	Global System for Mobile communication.
PSTN	Public Switched Telephone Network.
Em	Electromagnetic Environment.
PCS	Personal Communications Services.
FDMA	Frequency Division Multiple Access.
TDMA	Time Division Multiple Access.
CDMA	Code Division Multiple Access.
SDMA	Space Division Multiple Access.
RF	Radio Frequency.
PDC	Personal Digital Cellular
LMS	Least Mean Square.
DOA	Direction Of Arrival.
CMA	Constant Modulus Algorithm.
AWGN	Additive White Gaussian Noise.
MA	Multiple Access
TACS	Total Access communications System standards
GSM	Global System for Mobile Communication System.
W-CDMA	Wide band CDMA
TD-CDMA	Time Division – CDMA.
IMT-2000	International Mobile Telecommunications 2000
FDD	Frequency Division Duplex.
FH	Frequency Hopping
DS	Direct Sequence
CPS	chips per second
DSSS	Direct Sequence Spread Spectrum.
PN	Pseudo noise.
BW	Band Width.
G	processing gain
W	The spread bandwidth
R	The data rate.
DS-CDMA	Direct Spread-CDMA.
BPSK	Binary Phase Shift Keying.
MS	Mobile Station.
BS	Base Station.

Uplink	Reverse link (i.e. from MS to BS) transmission .
Down link	Forward link(i.e.from BS to MS) transmission.
AOA	Angle-of-Arrival
FIR	Finite Impulse Response.
A/D	Analog to Digital Converter.
SD	Steepest-Descent
MUSIC	Multiple emitter location and signal parameter estimation
ESPRIT	Estimation of signal parameters via rotational invariance
techniques.	
PSK	Phase-shift keying.
FSK	Frequency-shift keying.
SOI	Signal of interest .
CMA	Constant modulus algorithm.
SD-CMA	steepest-descent CMA.
SINR	signal-to-interference-and-noise ratio .
LS-CMA	Least-Squares CMA.
ES-CMA	element-space CMA.
BER	Bit error rate.
DD	Decision directed.
SD-DD	Steepest-descent DD
MT-LSCMA	Multitarget least-squares constant modulus algorithm
MT-DD	Multitarget decision-directed algorithm
LS-DRMTA	Least-squares despread respread multitarget array
LS-DRMTCMA	Least-squares despread respread multitarget constant modulus
algorithm	
GSO	Gram-Schmidt Orthogonalization
DPSK	The differential phase-shift keying
R_b	data bit rate

TABLE OF CONTENTS

Content	Pages
I. Abstract	i
II. Table of Content	ii
III. List of figures	v
IV. List of tables	ix

ABSTRACT

In mobile communications there is a need to increase the channel capacity. The increasing demand for mobile communication services without a corresponding increase in RF spectrum allocation (channel capacity) motivates the need for new techniques to improve spectrum utilization. The CDMA and adaptive antenna array are two approaches that shows real promise for increasing spectrum efficiency. This research focuses on the application of adaptive arrays to the Code Division Multiple Access (CDMA) cellular systems. The adaptive antenna has an intelligent control unit, so the antenna can follow the user, direct the radiation pattern towards the desired user, adapt to varying channel conditions and minimize the interference. Therefore there can be several users in the same channel in the same cell. The driving force of this intelligent control unit are special kinds of algorithms and we are going to investigate the performance of these different adaptive array algorithms in the CDMA systems.

Four each blind adaptive array algorithms are developed, and their performance under different test situations (e. g. AWGN (Additive White Gaussian Noise) channel, and multipath environment) is studied. A MATLAB test bed is created to show their performance on these two test situations and an optimum one can be selected.

TABLE OF CONTENTS

1. Introduction	1
1.1 Cellular – Communications Basics	1
1.2 Objective and Outline of Thesis.....	6
2. Fundamentals of Code Division Multiple Access System	8
2.1 Introduction.....	8
2.2 Cellular Standards.....	8
2.3 Spread Spectrum.....	11
2.4 System Model.....	14
2.5 CDMA benefits	16
3. Fundamentals of Adaptive Antenna Arrays	17
3.1 Introduction.....	17
3.2 Uniformly Spaced Linear Array	18
3.3 Beamforming and Spatial Filtering	23
3.4 Beampattern and Element Spacing.....	27
3.5 Adaptive Arrays	33
4. Overview of Adaptive Beamforming Algorithms	36
4.1 Introduction	36
4.2 Non-blind Adaptive Algorithms.....	36

4.2.1 Wiener Solution.....	36
4.2.2 Method of Steepest-Descent	38
4.2.3 Least-Mean-Squares Algorithm	40
4.3 Blind Adaptive Algorithms.....	41
4.3.1 Algorithms Based on Estimation of DOAs of Received Signals.....	41
4.3.2 Algorithms Based on Property-Restoral Techniques	41
4.3.3 Algorithms Based on Discrete-Alphabet Structure of Digital Signal...51	
4.3.4 Other Blind Beamforming Algorithms	52
4.3.5 Decision Directed Algorithms.....	52
5. Adaptive Beamforming Algorithms Used in the Thesis	54
5.1 Introduction	54
5.2 Multitarget Beamformer	54
5.3 Multitarget Least-Squares Constant Modulus Algorithm	56
5.3.1 Gram-Schmidt Orthogonalization	56
5.3.2 Phase Ambiguity	59
5.3.3 Sorting Procedure.....	59
5.4 Multitarget Decision-Directed Algorithm	63
5.5 Least-Squares Despread Respread Multitarget Array	64

5.5.1 Derivation of LS-DRMTA	64
5.5.2 Advantages of LS-DRMTA	71
5.6 Least-Squares Despread Respread Multitarget Constant Modulus Algorithm	
5.6.1 Derivation of LS-DRMTCMA.....	73
5.6.2 Advantages of LS-DRMTCMA	77
6. Results obtained and Discussion	78
6.1 Introduction	78
6.2 Description of System Parameters	78
6.3 Results in AWGN Channel	79
6.3.1 General Result.....	80
6.3.2 BER Performance for AWGN Channel	85
6.4 BER Performance in Multipath Environment	99
6.5 Conclusion	108
7. Summary and Future work	109
8. Appendix	111
Appendix A. Matlab programs.....	111
Appendix B. Derivation of the P factor.....	125
9. References	129
10. Declaration	132

List of Figures

- 1.1 The Cellular Concept.
- 1.2 Mobile phones make the connection.
- 2.1 Direct Sequence Spreading.
- 2.2 Direct Sequence Spread Spectrum (DSSS) transmitter
- 2.3 DSSS for receiver.
- 2.4 Model for a multirate DS-CDMA.

- 3.1 Illustration of a plane wave incident on a uniformly spaced linear array from direction θ .
- 3.2 A narrowband beamformer forms a linear combination of the sensor outputs.
- 3.3 A wideband beamformer samples the signal in both space and time.
- 3.4 Beampattern for an equal-weight beamformer. In this case, the number of elements is equal to 8, and the element spacing is half of the carrier wavelength.
- 3.5 The same beampattern as shown in Figure 3.4 with polar coordinate plot in azimuth.
- 3.6 A simple narrowband adaptive array.

- 5.1 Illustration of a multitarget LS-CMA adaptive array.
- 5.2 Illustration of sorting procedure in MT-LSCMA for a CDMA system.
- 5.3 Structure of a beamformer using LS-DRMTA.
- 5.4 LS-DRMTA block diagram for user i .
- 5.5 Structure of a beamformer using LS-DRMTCMA.
- 5.6 LS-DRMTCMA block diagram for user i .

- 6.1 Illustration of eight users with DOAs equally spaced between -70° and 90° .
- 6.2 Beampatterns corresponding to different users generated by using LS-DRMTCMA. * denotes user in the system. The signal parameters are shown in Table 6.1.
- 6.3 Beampatterns corresponding to different users generated by using LS-DRMTCMA .
- 6.4 Signal constellation of user 5 before the beamformer processing. The signal parameters are shown in Table 6.1.
- 6.5 Signal constellation of user 5 after the beamformer processing. The signal parameters are shown in Table 6.1.
- 6.6 Original signal waveform of user 5. The signal parameters are shown in Table 6.1.

- 6.7 Corrupted signal waveform of user 5. The signal parameters are shown in Table 5.1.
- 6.8 Reconstructed signal waveform of user 5. The signal parameters are shown in Table 6.1.
- 6.9 Convergence curve for MT-SDDD in port 6 of the beamformer. The signal parameters are shown in Table 6.1.
- 6.10 Convergence curve for LS-DRMTCMA in port 6 of the beamformer. The LS-DRMTCMA is described in equations (5.46)-(5.51). The data block size in each iteration is equal to 60. The signal parameters are shown in Table 6.1.
- 6.11 DOA distribution of all the users for both the non-crowded and crowded cases.
- 6.12 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 8$ dB, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.13 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 8$ dB, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.14 Beampattern of user 5 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.15 Beampattern of user 5 generated by using MT-SDDD. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.16 Beampattern of user 4 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.17 Beampattern of user 4 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.18 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 4$ dB, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

- 6.19 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 4$ dB, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.
- 6.20 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 0 dB, and the time delay between these two paths is $0.5T_c$.
- 6.21 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 6 dB, and the time delay between these two paths is $0.5T_c$.
- 6.22 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 10 dB, and the time delay between these two paths is $0.5T_c$.
- 6.23 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 0 dB, and the time delay between these two paths is $1.5T_c$.
- 6.24 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 6 dB, and the time delay between these two paths is $1.5T_c$.
- 6.25 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 10 dB, and the time delay between these two paths is $1.5T_c$.

- 6.26 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 6 dB, and the time delay between these two paths is $1.5T_c$.
- 6.27 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 10 dB, and the time delay between these two paths is $1.5T_c$.

List of Tables

- 6.1 Signal Parameters of 8 Users Transmitting Signals from Different directions.
- 6.2 Signal Parameters of Multipaths.

CHAPTER 1

INTRODUCTION

1.1 Cellular communication basics.

The concept of cellular or mobile communication was developed in the 1970's at Bell Labs in the U.S. The idea behind the concept was simple: Instead of providing communication services in a centralized fashion through a single high-power transmission / receiver station, they are provided in a distributed fashion via several low power stations.

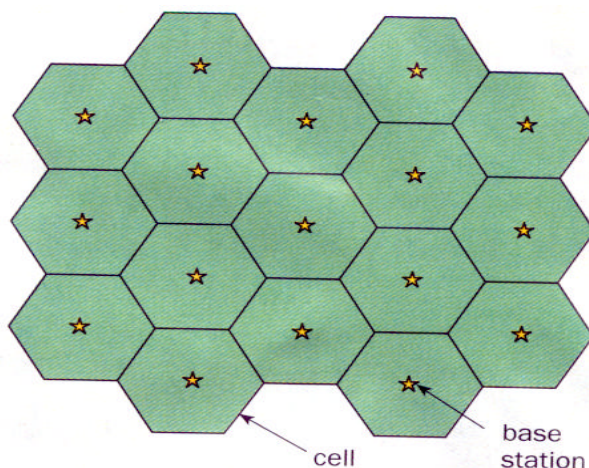


Fig.1.1 The Cellular Concept.

The area covered by a mobile-phone network is divided into cell, each of which contains a base station. When a mobile-phone user makes a call he or she sends signals to the nearest base station.

The geographical area that the communication system serves is divided into smaller subareas, called cells. Within each cell, all the communication is carried out via the base station serving that particular cell. When a mobile-phone user wants to make a call or download data from a server, he or she must first connect to the appropriate wireless network. The connection is made using radio links (i.e. em

waves) between the phone or terminal and the nearest basestation operated by that network provider. In turn the basestation is connected to a high capacity fiber-optic “backbone” that links together many other networks. It is via this backbone network that the user can connect to any other conventional phone (PSTN), mobile phone or additional service within the public network.

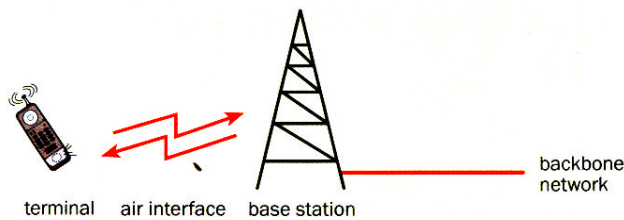


Fig.1.2 Mobile phones make the connection.

The basic idea behind wireless-communication system. The caller connects to the appropriate network by first sending signals to the nearest base station, which is in turn connected to the backbone network.

The most important factor in the success of the cellular concept has been the relative ease with which the total system capacity can be increased. The rapid growth in the number of users of mobile communications means that many operators must find new ways of increasing the capacity of their networks. A higher demand in wireless communications calls for higher system capacities. The capacity of a wireless communication system can be increased by different methods.

These methods include:

1. By directly enlarging the bandwidth of the existing communication channels or by allocating new frequencies to the service in question.
2. By the division of a geographical area into cells.
3. By using Multiple-Access technique.
4. By using Adaptive Antenna.

However one can draw some limitations and drawbacks to the above mentioned methods:

In method one: Since the electromagnetic (em) spectrum is limited, there by making it a valuable resource, and the em environment is increasingly

becoming congested with a proliferation of intentional and unintentional sources of interference, it may not be feasible to increase system capacity by opening new frequency spectrum space for the wireless communications application. The introduction of new frequency bands at 1800 and 1900MHz is an example of allocating frequency to increase capacity. However, compared to 800 and 900MHz systems, mobile communications at 1800 and 1900MHz requires more basestations and greater levels of radiated power. And the problem associated with adding numerous basestations is the cost involved in finding new locations for the antennas and basestation cabinets.

In method two: In a cellular telephone network, where a large geographical coverage is desired and a large number of mobile transceivers must be supported, the region is divided into a large number of cells. This allows the same carrier frequency to be reused at different cells. In principle, the larger the amount of cells in a region, the higher the levels of frequency reuse and hence the higher capacity that can be achieved. This is one of the reasons why macrocell, microcells, and picocells have been proposed for pcs. However, the criterion used for defining co-channel cells (cells that use the same carrier frequency) is that the distance between them is sufficiently large that intercell interference is lower than some acceptable limit, this means that communication signals that are transmitted at the same carrier frequency in different cells are separated by a spatial distance to reduce the level of co-channel interference. For a given base station transmission power level, this puts a limit on the number of geographical area.

In method three: Frequency efficiency can be improved by using multiple access techniques to provide high system capacity. Cellular-communication systems must be able to support a multitude of simultaneous users in the same cell and in neighboring cell. At the same time, the radio-frequency band that represents the communications medium is a very scarce resource that must be managed efficiently. There are two basic features that differentiate the various cellular systems:

(a) The way in which the available frequency resource within a cell are shared among many users.

(b) And the way that the existing radio-frequency band is divided between the different users.

A multiple access method is a definition of how the radio spectrum is divided into channels and how frequencies are allocated to many users of the system. Each system divides the frequency band into orthogonal "subspaces" that are then allocated to different users.

In multiple access, multiple, simultaneous users can be supported. In other words, a large number of users share a common pool of radio channels and any user can gain access to any channel. (A channel can be thought of as merely a portion of the limited radio resource which is temporarily allocated for a specific purpose, such as someone's phone call).

There are three domains in which sharing can take place i.e the most widely used and well known multiple access schemes divide the frequency band into either smaller frequency (FDMA), portion of time (TDMA) or encode the signal (CDMA).

(I) FDMA Frequency Division Multiple Access.

In here the frequency spectrum is divided into segments that are then apportioned among different users,i.e the frequency spectrum is divided into several non-overlapping "subspaces" one or more of which is allocated to each user.

(II) TDMA Time Division Multiple Access.

In here the frequency band is divided in time into different slots. The signals from different users are placed in separate slots, but a caller can only use the frequency band during the time intervals allocated to him or not.

In both schemes, the number of available frequency or time slots limits the number of people who can make calls at the same time.Eventhough,the same time or frequency slot can be used in another distant cell the existence of cochannel interference will limit their performance.

(III) CDMA Code Division Multiple Access.

In CDMA, signals from different callers are transmitted at the same time and in the same frequency band. In this case, the signals can be distinguished from each other by a so called spreading code that is allocated to each user. This spreading code (which is a kind of pseudo-noise) is applied to the signal before it is transmitted. The signals from a particular user can be separated from each other at the receiver by correlating them with the desired spreading code.CDMA employs spreading spectrum modulation (i.e. each user's signal wave form is spread over the entire frequency spectrum by applying the code sequence to the signal).The intended receiver then uses the appropriate code to detect the signal of his or her choice.

In method four: The capacity of cellular systems is mainly determined by their ability to withstand co-channel and adjacent channel interference. An adaptive antenna system can have a direct and positive impact on system capacity. The

use of adaptive antenna which generally consists of arrays of antenna elements together with associated signal processing to control and combine the elements output signals can effectively meet many design changes. Adaptive antenna at the transmitting end, receiving end or both can reduce or eliminate the effects of fading and multiple delay spread.

Conventional basestation antennas in existing operational systems are either omnidirectional or sectorized. There is a waste of resources since the vast majority of transmitted signal power radiates in directions other than toward the desired user. In addition, the signal power radiated throughout the cell area will be experienced as interference by any other user than the desired one. Concurrently, the basestation receives "interference" emanating from the individual users within the system. Adaptive Antennas offer a relief by transmitting the power only to the desired directions or receiving the power only from the desired directions. The signals received by the antenna elements are multiplied with complex weights and then added together in order to provide the spatially filtered signal, i.e. the amplification and the phase shift of the signals from the different antenna elements are determined by a signal processor. This control unit assigns the values for the weights for each channel. The controlling unit can be preprogrammed, but normally it uses appropriate search algorithms or neural networks.

1.2 Objective and Outline of Thesis.

The increasing demand for mobile communication services without a corresponding increase in RF spectrum allocation motivates the need for new techniques to improve spectrum utilization. The existing frequency spectrum or radio frequency band (resource) allocated for mobile wireless communication is very scarce, costly and hence must be managed efficiently.

The most important factor in the success of cellular or mobile concept has been the relative ease with which the total system capacity (i.e. the ability to support multitude of simultaneous users in the same cell or neighboring cells) can be increased. The capacity of cellular systems is mainly determined by their ability to with stand co-channel and adjacent channel interference.

Among the many alternatives taken the Adaptive Antenna and CDMA are two approaches that shows real promise for increasing spectrum efficiency. The adaptive antenna array is capable of automatically forming beam in the directions of the desired signals and steering nulls in the directions of the interfering signals. By using the adaptive antenna in a CDMA system we can reduce the amount of co-channel interference from other users with its own cell and neighboring cell, and therefore increase capacity.

There exist many adaptive algorithms that can be used in the adaptive antenna array. However, for the adaptive array used in CDMA system, where multiple users occupy the same frequency band, the adaptive algorithm should have the ability to separate and extract each users signal simultaneously .It is also preferred that the adaptive antenna can work with out a training signal, i.e the algorithm is blind. Algorithms which can satisfy the above requirements are developed and their performance will be compared with each other under different test conditions. A matlab test bed is created for comarison purpose.

The aim of this Research is then to develop adaptive algorithms for the antenna array used in a CDMA system and compare their performance such that an optimum one can be chosen. And hence the receiving system of a base station antenna is optimized.

This thesis is organized as follows: Chapter 2 gives a brief introduction about code division multiple access (CDMA). Chapter 3 introduces the fundamentals of adaptive antenna arrays, the terminology, and the basic concepts related to the adaptive beam forming. The correspondence between a narrowband beam former and a FIR filter is also introduced. Chapter 4 provides a detailed survey of the adaptive beamforming algorithms. Both non-blind and blind algorithms are described. For the non-blind algorithms, the least mean-square (LMS) is used and for the blind algorithms, (i) the DOA-estimation-based algorithm, (ii) the algorithms based on property-restoral techniques such as the constant modulus algorithm (CMA) and the spectral self-coherence restoral (SCORE) algorithm, the algorithms based on discrete-alphabet structure of digital signals, and other blind algorithms such as the 2D RAKE receiver and (iii) the decision-directed algorithm are presented. Chapter 5 presents four multitarget-type blind adaptive beamformer algorithms used in the project for comparison, the multitarget least-squares constant modulus algorithm (MT-LSCMA), the multitarget decision-directed (MT-DD) algorithm, the least-squares despread respread multitarget array (LS-DRMTA), and the least-squares despread respread multitarget constant modulus algorithm (LS-DRMTCMA). Unlike the MT-LSCMA and the MT-DD algorithm, the other two algorithms utilize the information of the spreading signal of each user in the CDMA system to adapt the weight vector of the array. Furthermore, the LS-DRMTCMA combines the spreading signal and the constant modulus property of the transmitted signal to adapt the weight vector. The derivation and the advantage of the algorithms are also described in this chapter.

Chapter 6 presents the results of different adaptive array algorithms. A clarified comparison of the BER performance of different algorithms in various channel environments (e.g. the AWGN channel, and the multipath environment) is provided in this chapter.

A brief summary and conclusions are provided in Chapter 7 along with some suggestions for future work.

And finally an appendix and references will be given in chapter 8 and 9 respectively.

CHAPTER 2

Fundamentals of Code Division Multiple Access

2.1 Introduction.

One of the most important concepts to any cellular telephone system is that of “multiple access” meaning that multiple, simultaneous users can be supported. In other words, a large number of users share a common pool of radio channels and any user can gain access to any channel (each user is not always assigned to the same channel). A channel can be thought of as merely a portion of the limited radio resource, which is temporarily allocated for a specific purpose, such as someone’s phone call. A multiple access method is a definition of how channels are allocated to the many users of the system.

CDMA is one type of multiple access, which has gained acceptance by cellular radio system operators as an update that will increase both their capacity and service quality. CDMA is a form of spread-spectrum family of digital communication techniques. The core principle of spread spectrum is the use of noise like carrier waves, and as the name implies, bandwidth much wider than that required for simple point-to-point communication at the same data rate. In CDMA system users which are randomly distributed in a cell use the same frequency band and time slot. Each user is assigned with a unique digital code which is known both by the transmitter and receiver. Each user’s signal is spread with this code before it is transmitted and at the receiver the spreading code is removed or despread leaving the desired signal.

2.2 Cellular Standards

In the late 1970’s and early 1980’s consumer wireless communications began to takeoff.

- a. The early mobile phones used first generation (1G) technology which was analog, circuit based, narrowband and suitable only for voice communications.

The traditional first generation analog cellular system standards such as,

- ? Those based on the Advanced Mobile Phone Service (AMPS) and,
- ? Total Access communications System (TACS) standards; use a method of multiple access called FDMA. FDMA channels are

defined by a range of radio frequencies usually expressed in a number of KiloHertz (kHz) out of the radio spectrum.

For example - AMPS systems use 30 kHz “slices” of spectrum for each channel

-TACS channels are 25 kHz wide.

With FDMA, only one subscriber at a time is assigned to a channel. No other conversations can access this channel until the subscriber’s call is finished or until that original call is handed-off to a different channel by the system.

b. Then commercial wireless devices have used second-generation (2G) technology, which is digital, circuit-based, narrow band and suitable for voice and data communications.

A common multiple access method employed in 2G digital cellular system is TDMA. TDMA digital cellular standard includes:

- ? North American Digital Cellular (known by its standard number as IS-54),
- ? Global system for Mobile Communications(GSM),
- ? Personal Digital Cellular (PDC).

TDMA system commonly starts with a slice of spectrum referred to as one “carrier”. Each carrier is then divided into time slots. Only one subscriber at a time is assigned to each time slot or channel. No other conversations can access this channel until the subscribers call is finished or until that original call is handed off to a different channel by the system.

For example - IS-54 systems, designed to coexist with AMPS system, divide the 30 KHz of spectrum into three channels,

- PDC divides the 25 kHz slices of spectrum into three channels,

- GSM systems create eight time-division channels in 200 kHz wide carrier.

c. Third generation (3G) is suitable for voice and advanced data application, including online multimedia and mobile e-commerce.3G promises transmission speeds of up to 2.05Mega bit per second in stationary applications; 380Kbps for slow moving users; and 128kbps for users in vehicles.

3G technology comprises three primary standards: These are;

- ? W-CDMA (wide band CDMA),
- ? CDMA2000,
- ? TD-CDMA.

Each standard is based on and provides an upgrade path for at least one of today's primary wireless interfaces: TDMA, GSM, and CDMA.

The international Telecommunication Union has adapted International Mobile Telecommunications 2000 (IMT-2000) to formally standardize the already developed 3G wireless flavors to let them offer a consistent set of services through out the world, and to provide a roadmap for upgrade.

With CDMA, unique digital codes, rather than separate RF frequencies or channels, are used to differentiate subscribers. The codes are shared by both the mobile station (cellular phone) and the basestation, and are called "pseudo-random code sequences". All users share the same range of radio spectrum.

The goals of multiple access communications system, meaning cellular and pcs are;

- Near-wire line quality voice service,
- Near-universal geographical coverage,
- Low equipment cost, both subscribers and fixed plant,
- Minimum number of fixed radio lines.

Spectrum for mobile users is normally allocated in frequency division duplex (FDD). Spectral allocation for wireless communication is in the 800-900MHz. Regulatory agencies have allocated limited bandwidth to these services (i.e. to wireless communication), so there must be a system designed to utilize spectrum efficiently. Cellular operators have 25MHz each, split between the two directions of communication i.e. in transmit and receive system.

Several hundred channels are available with in the spectrum allocation. One channel of one basestation is used for each conversation. When a subscriber moved between cells, over the air messaging is used to transfer control from the old cell to the new cell. This transfer of control is termed as hand-off or handover.

2.3 SPREAD SPECTRUM

CDMA is a “spread spectrum” technology, which means that it spreads the information contained in a particular signal of interest over much greater bandwidth than the original signal. The goal of spread spectrum is a substantial increase in bandwidth of an information-bearing signal, far beyond that needed for basic communication.

The core principle of spread spectrum is the use of noise-like carrier waves and as the name, bandwidths much wider than that required for simple point-to-point of communication at the same data rate.

A CDMA call starts with a standard rate of 9600bps. This is then spread to a transmitted rate of about 1.23 mega bit per second. Spreading means that digital codes are applied to the data bits associated with users in a cell. These data bits are transmitted along with the signals of all other users in that cell. When the signal is received, the codes are removed from the desired signal, separating the users and returning the call to a rate of 9600bps.

Spread spectrum signals are demodulated in two steps: First the spectrum spreading modulation is removed, and second the signal is demodulated.

The process of despreading a signal is called correlation. The spread spectrum signal is despread when the proper synchronization between the transmitter and receiver is achieved. Synchronization is the most difficult aspect of the receiver.

Spread spectrum systems generally fall into one of two categories:

- (I) Frequency Hopping (FH) or
- (II) Direct Sequence (DS).

In both cases synchronization of transmitter and receiver is required. Both forms can be regarded as using a pseudo-random carrier, but they create that carrier in different ways.

(I) Frequency Hopping (FH)

Is typically accomplished by rapid switching of fast-selecting frequency synthesizers in a pseudo-random pattern. In frequency hopping systems, the carrier frequency of the transmitter abruptly changes (or hops) in accordance with a pseudo-random code sequence. The order of frequencies selected by the transmitter is dictated by the code sequence. The receiver tracks these changes and produces a constant IF signal.

(II) Direct Sequence (DS).

CDMA uses a form of direct sequence spread spectrum. DS is one of most widely known and utilized spread spectrum system and is easy to implement. A narrowband carrier is modulated by a code sequence. The carrier phase of the transmitted signal is abruptly changed in accordance with this code sequence. The code sequence is generated by a pseudo random generator that has a fixed length. After a given number of bits the code sequence repeats itself exactly. The spread of the code sequence is called the chipping rate, measured in chips per second (CPS). For direct sequence, the amount of spreading is dependent upon the ratio of chips per bit of information. At the receiver, the information is recovered by multiplying the signal with a locally generated replica of the code sequence.

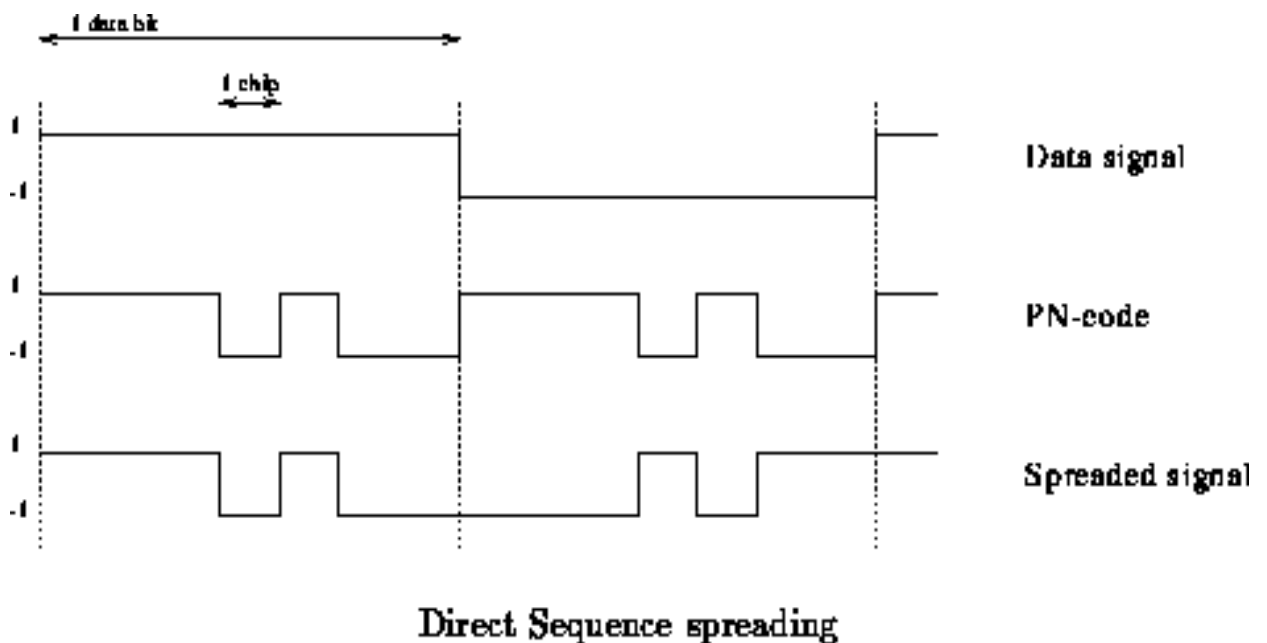


Fig 2.1 Direct Sequence Spreading

Direct sequence is, in essence, multiplication of a more conventional communication waveform by a pseudo noise (PN) \pm binary sequences in the transmitter. A second multiplication by a replica of the same \pm sequences in the receiver recovers the original signal.

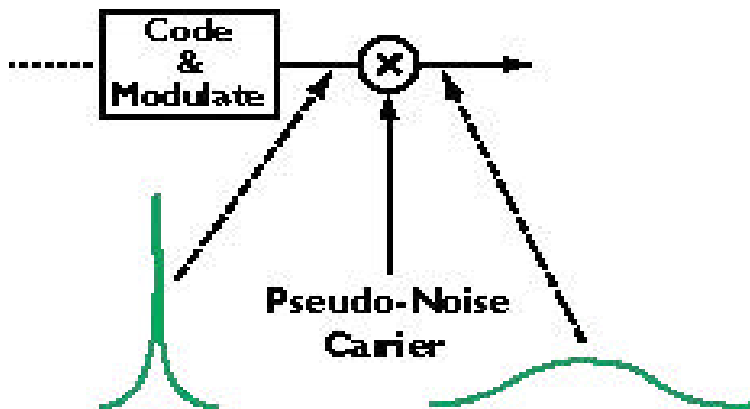


Fig 2.2 Direct Sequence Spread Spectrum (DSSS) transmitter.

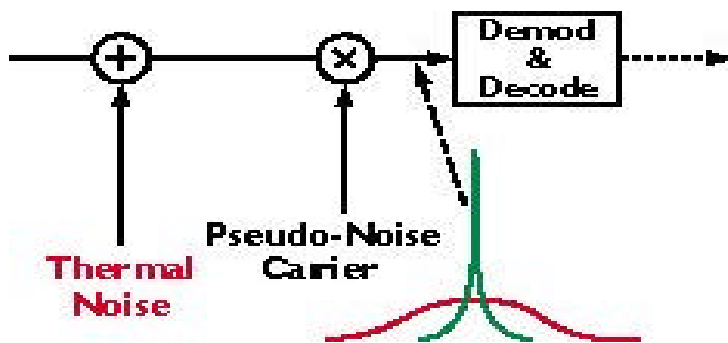


Fig 2.3 DSSS for receiver.

The noise and interference, being uncorrelated with the PN sequence, become noise-like and increase in BW when they reach the detector. The signal-to-interference ratio can be enhanced by narrow band filtering that rejects most of the interference power. It is often said that the SNR is enhanced by the so-called processing gain $G=W/R$, where W is the spread bandwidth and R is the data rate.

In IS-95 CDMA $G=W/R=10\log (W/R)=10\log(1.2288\text{mhz}/9600\text{hz})=21\text{db}$ for 9600bps data rate.

The actual binary PN spreading sequence can be approximated by an ideal Bernoulli (coin-tossing) sequence with equiprobable outcomes. That is $p(0)=p(1)=1/2$, with all trials independent. Mapping the (0,1) sequence to a

(+1, -1) discrete modulation sequence $\{a_n\}$, the autocorrelation of the latter is a kronecker delta function, that is;

$$R_a[m,n]=E[a_m a_n]=\begin{cases} 1 & \text{if } m=n \\ 0 & \text{if } m \neq n \end{cases}$$

The actual sequences have off-time corrections of the order of $1/N$, where N is the sequence. This approximation is well justified in practice because the random relative RF phases of the interferes tend to remove the small bias that the approximation might otherwise introduce.

2.4 SYSTEM MODEL

The system model for a multirate DS-CDMA system is shown below. For simplicity, a coherent BPSK modulation is assumed and only the reverse link (i.e. uplink MS to BS) transmission in a single cell is considered.

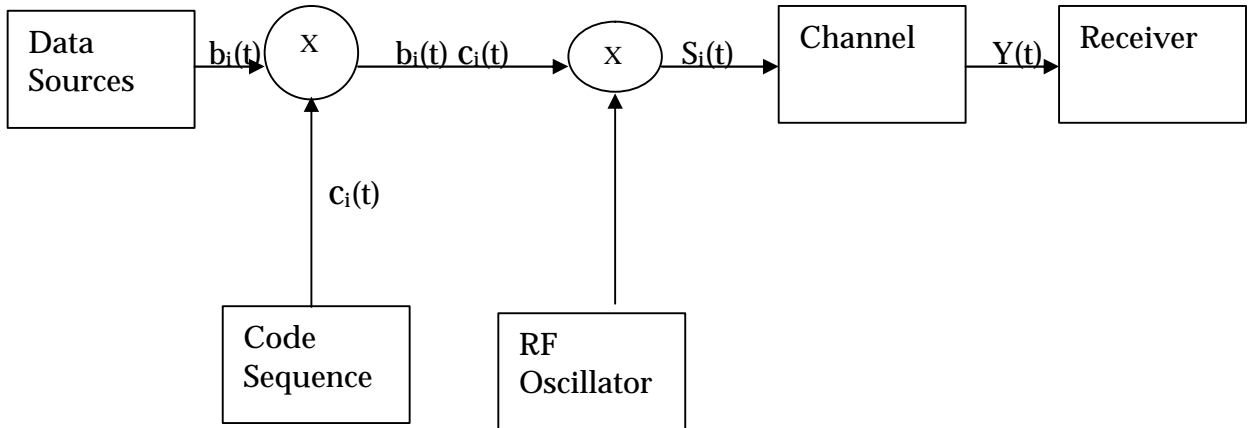


Fig 2.4 Model for a multirate DS-CDMA

Assume that the spread bandwidth for each connection is fixed at w (Hz) and there are a total of P users in the system with the ability of transmitting any one of S source bandwidths $B_o = B_1, \dots = B_{s-1}$ (Hz) corresponding to S different types of multimedia traffic.

For DS-CDMA system with P users, the signal transmitted by the i th user can be expressed as;

$$S_i(t) = \sqrt{2P_i} b_i(t) c_i(t) \cos(W_c t + \phi_i), \quad i = 1, 2, \dots, p \quad (2.1)$$

The complex envelope of the signal transmitted is:

$$S_i(t) = \sqrt{2P_i} b_i(t) c_i(t) \exp\{j\theta_i\}, \quad (2.2)$$

where P_i , $b_i(t)$, $c_i(t)$ and θ_i are the power, the data signal, the spreading signal (PN sequence), and the random phase of the i^{th} signal respectively.

The data signal $b_i(t)$ is given by:

$$b_i(t) = \sum_{n} b_{in} P_{T_b}(t - nT_b), \quad (2.3)$$

Where $b_{in} \in \{-1, +1\}$ is the n^{th} data bit of the i^{th} user, and P_{T_b} is a unit rectangular pulse of duration T_b . T_b is the bit period of the CDMA signal.

The spreading signal $c_i(t)$ is given by:

$$c_i(t) = \sum_m c_{im} P_{T_c}(t - mT_c), \quad (2.4)$$

Where $c_{im} \in \{-1, +1\}$ is the m^{th} chip of i^{th} user, and P_{T_c} is a unit rectangular pulse of duration T_c . T_c is the chip period of the CDMA signal.

The ratio of the bit period (T_b) to the chip period (T_c) is called the processing gain, $N_c = T_b/T_c$, systems are usually designed to have high processing gain where $N_c \gg 1$ or $T_b \gg T_c$.

Assuming we have perfect power control in the system the output of port or user i in the beamformer (i.e. the received signal at the basestation) is:

$$Y(t) = \sum_{i=1}^p S_i(t - \tau_i) \alpha_i e^{-\gamma_i/2} d_i^{-\alpha_i/2} + n(t), \quad (2.5)$$

where $n(t)$ is the additive white Gaussian noise, with two sided spectral density $\sigma_n^2/2$, $d_i^{-\alpha_i/2}$ is due to the pathloss with α_i being the propagation exponent dependent on the environment and d_i is the distance between the basestation and mobile station. τ_i is the propagation delay of the i^{th} user, and α_i is due to the multipath signals arriving at the Mobile Station (MS), which is modeled as Rayleigh fading with variance $\sigma_{\alpha_i}^2 = 2\sigma_{\alpha_i}^2$, the factor $e^{-\gamma_i/2}$ represent the lognormal shadowing and γ_i is a Gaussian random variable with zero mean and variance $\sigma_{\gamma_i}^2$.

CDMA receivers separate communication channels by means of a pseudo-random modulation that is applied and removed in the digital

domain and in the final stage of the encoding of the radio link from the basestation to the mobile, CDMA adds a special, "pseudo-random code" to the signal that repeats itself after a finite amount of time. Basestation in the system distinguish themselves from each other by transmitting different portions of the code at a given time. In other words, the basestations transmit time-offset versions of the same pseudo-random code. In order to assure that the time offset used remains unique from each other, CDMA stations must remain synchronized to a common time reference.

The Global Positioning System (GPS) provides this precise common time references. GPS is a satellite based radio navigation system capable of providing a practical and affordable means of determining continuous position, velocity, and time to an unlimited number of users.

CDMA is altering the face of cellular communication by;

- ? Dramatically improving the telephone traffic capacity,
- ? Dramatically improving the voice quality and eliminating the audible effects of multipath fading,
- ? Reducing the incidence of dropped calls due to handoff failures,
- ? Providing reliable transport mechanism for data communications, such as facsimile and internet traffic,
- ? Reducing the number of sites needed to support any given amount of traffic, and simplifying site selection,
- ? Reducing deployment and operating costs because fewer sites are needed ,
- ? Reducing average transmitted power,
- ? Reducing interference to other electronic devices,
- ? Simplified system planning through the use of the same frequency in every sector of every cell,
- ? Enhanced privacy, since the CDMA signal is spread over a larger bandwidth it is hard to be effected by interference and loss of information, because of wide bandwidth of a spread spectrum signal it is difficult to jam with, to interfere with, and to identify. This is in contrast to technologies using a narrowband width of frequencies,
- ? Bandwidth on demand.

Chapter 3

Fundamentals of Adaptive Antenna Arrays

3.1 Introduction

An antenna array consists of a set of antenna elements that are spatially distributed at known locations with reference to a common fixed point [4]. By changing the phase and amplitude of the exciting currents in each of the antenna elements, it is possible to electronically scan the main beam and/or place nulls in any direction.

The antenna elements can be arranged in various geometries, with linear, circular and planar arrays being very common. In the case of a linear array, the centers of the elements of the array are aligned along a straight line. If the spacing between the array elements is equal, it is called a uniformly spaced linear array. A circular array is one in which the centers of the array elements lie on a circle. In the case of a planar array, the centers of the array elements lie on a single plane. Both the linear array and circular array are special cases of the planar array. Arrays whose element locations conform to a given non-planar surface are called conformal arrays.

The radiation pattern of an array is determined by the radiation pattern of the individual elements, their orientation and relative positions in space, and the amplitude and phase of the feeding currents. If each element of the array is an isotropic point source, then the radiation pattern of the array will depend solely on the geometry and feeding current of the array, and the radiation pattern so obtained is called the array factor. If each of the elements of the array are similar but non-isotropic, by the principle of pattern multiplication, the radiation pattern can be computed as a product of the array factor and the individual element pattern [5].

3.2 Uniformly Spaced Linear Array

Consider an M-element uniformly spaced linear array which is illustrated in Figure 3.1. In Figure 3.1, the array elements are equally spaced by a distance d , and a plane wave arrives at the array from a direction θ off the array broadside. The angle θ is called the direction-of-arrival (DOA) or angle-of-arrival (AOA) of the received signal, and is measured clockwise from the broadside of the array. The received signal at the first element may be expressed as:

$$\bar{X}_1(t) = u(t) \cos(2\pi f_c t + \phi(t)) \quad (3.1)$$

where f_c is the carrier frequency of the modulated signal, $\phi(t)$ is the information carrying component, $u(t)$ is the amplitude of the signal, and ϕ is a random phase. It is convenient to use the complex envelope representation of $\bar{X}_1(t)$ which is given by

$$X_1(t) = u(t) \exp[j(\phi(t) + 2\pi f_c t)] \quad (3.2)$$

The received signal at the first element $\bar{X}_1(t)$ and its complex envelope $X_1(t)$ may be related by

$$\bar{X}_1(t) = \text{Re}[X_1(t) \exp\{-j(2\pi f_c t)\}] \quad (3.3)$$

where $\text{Re}[\cdot]$ stands for the real part of $[\cdot]$. Now taking the first element in the array as the reference point, if the signals have originated far away from the array, and these plane waves advance through a non-dispersive medium that only introduces propagation delays, the output of any other array element can be represented by a time-advanced or time-delayed version of the signal at the first element. From Figure 3.1, we see that the plane wavefront at the first element should propagate through a distance $d \sin \theta$ to arrive at the second element. The time delay due to this additional propagation distance is given by

$$\tau = \frac{d \sin \theta}{c} \quad (3.4)$$

where c is the velocity of light. Now, the received signal of the second element may be expressed as

$$\bar{X}_2(t) = \bar{X}_1(t - \tau) = u(t - \tau) \cos(2\pi f_c (t - \tau) + \phi(t - \tau)) \quad (3.5)$$

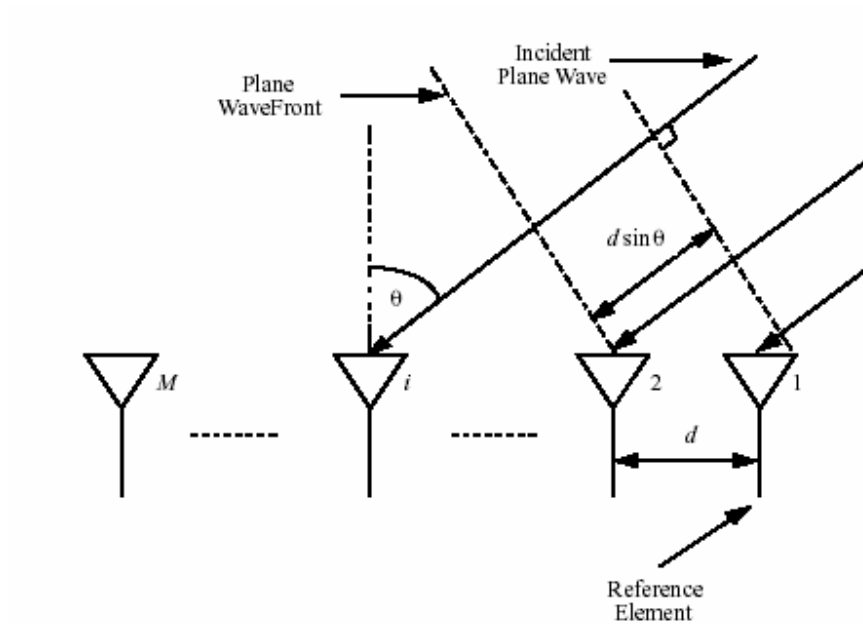


Figure 3.1: Illustration of a plane wave incident on a uniformly spaced linear array from direction θ .

If the carrier frequency f_c is large compared to the bandwidth of the impinging signal, then the modulating signal may be treated as quasi-static during time intervals of order $1/f_c$ and in that case equation (3.5) reduces to

$$\overline{X}_2(t) \approx u(t) \cos(2\pi f_c t - 2\pi f_c \tau(t)) \quad (3.6)$$

The complex envelope of $\overline{X}_2(t)$ is therefore given by

$$X_2(t) \approx u(t) \exp\{j(2\pi f_c t - \tau(t))\} \\ \approx X_1(t) \exp\{j(2\pi f_c \tau)\} \quad (3.7)$$

From equation (3.7) we see that the effect of the time delay on the signal can now be represented by a phase shift term $\exp\{-j(2\pi f_c \tau)\}$. Substituting equation (3.4) into (3.7),

We have

$$X_2(t) = X_1(t) \exp\left\{j\left(2\pi f_c \frac{d \sin \theta}{c}\right)\right\} \\ = X_1(t) \exp\left\{j\left(\frac{2\pi}{\lambda} d \sin \theta\right)\right\} \quad (3.8)$$

where λ is the wavelength of the carrier. In equation (3.8), we have used the relation between λ and f_c , that is, $f_c = c/\lambda$.

Similarly, for element i , the complex envelope of the received signal may be expressed as

$$X_i(t) = X_1(t) \exp\left\{j\left(\frac{2\pi}{\lambda}(i-1)d \sin \theta\right)\right\}, \quad i = 1, 2, \dots, M \quad (3.9)$$

Let

$$X(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_m(t) \end{bmatrix} \quad (3.10)$$

$$a(\theta) = \begin{bmatrix} 1 \\ e^{j\frac{2\pi}{\lambda} d \sin \theta} \\ \vdots \\ e^{j\frac{2\pi}{\lambda} (m-1)d \sin \theta} \end{bmatrix} \quad (3.11)$$

then equation (3.9) may be expressed in vector form as

$$X(t) = \mathbf{a}(\theta) X_1(t) \quad (3.12)$$

The vector $\mathbf{x}(t)$ is often referred to as the *array input data vector* or the *illumination vector*, and $\mathbf{a}(\theta)$ is called the *steering vector*. The steering vector is also called direction vector, array vector, array response vector, array manifold vector, DOA vector, or aperture vector. In this case, the steering vector is only a function of the angle-of-arrival (AOA). In general, however, the steering vector is also a function of the individual element response, the array geometry, and signal frequency. The collection of steering vectors for all angles and frequencies is referred to as the array manifold. Though for many simple arrays such as the uniformly spaced linear array discussed above, the array manifold can be computed analytically, in practice, the array manifold is measured as point source responses of the array at various angles and frequencies. This process of obtaining the array manifold is called array calibration.

In the above discussion, the bandwidth of the impinging signal expressed in equation (3.9) is assumed to be much smaller than the reciprocal of the propagation time across the array. Any signal satisfying this condition is referred to as narrowband; otherwise it is referred to as wideband. In most of the discussion that follows, the signal is assumed to be narrowband unless specified otherwise.

We could extend the above simple case to a more general case. Suppose there are q narrowband signals $s_1(t), \dots, s_q(t)$, all centered around a known frequency, say f_c , impinging on the array with a DOA θ_i , $i=1, 2, \dots, q$. These signals may be uncorrelated, as for the signals coming from different users, or can be fully correlated as happens in multipath propagation, where each path is a scaled and time-delayed version of the original transmitted signal, or can be partially correlated due to the noise corruption. The received signal at the array is a superposition of all the impinging signals and noise. Therefore, the input data vector may be expressed as

$$X(t) = \sum_{i=1}^q \mathbf{a}(\theta_i) s_i(t) + n(t) \quad (3.13)$$

$$\mathbf{a}(\theta) = \begin{bmatrix} 1 \\ e^{j\frac{2\pi}{\lambda}d \sin \theta} \\ \vdots \\ e^{j\frac{2\pi}{\lambda}(m-1)d \sin \theta} \end{bmatrix} \quad (3.14)$$

and $\mathbf{n}(t)$ denotes the $M \times 1$ vector of the noise at the array elements. In matrix notation, equation (3.13) becomes

$$\mathbf{X}(t) = \mathbf{A}(\theta) \mathbf{s}(t) + \mathbf{n}(t) \quad (3.15)$$

Where $\mathbf{A}(\theta)$ is the $M \times q$ matrix of the steering vectors

$$\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_q)] \quad (3.16)$$

and

$$\mathbf{s}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_q(t) \end{bmatrix} \quad (3.17)$$

Equation (3.15) represents the most commonly used narrowband input data model.

In equation (3.15), if the data vector $\mathbf{X}(t)$ is sampled K times, at t_1, \dots, t_K , the sampled data may be expressed as

$$\mathbf{X} = \mathbf{A}(\theta) \mathbf{S} + \mathbf{N} \quad (3.18)$$

where \mathbf{X} and \mathbf{N} are the $M \times K$ matrices containing K snapshots of the input data vector and noise vector, respectively,

$$\mathbf{X} = [\mathbf{x}(t_1), \dots, \mathbf{x}(t_K)] = [\mathbf{x}(1), \dots, \mathbf{x}(K)] \quad (3.19)$$

$$N = [n(t_1), \dots, n(t_K)] = [n(1), \dots, n(K)] \quad (3.20)$$

and S is the $q \times K$ matrix containing K snapshots of the narrowband signals

$$S = [s(t_1), \dots, s(t_K)] = [s(1), \dots, s(K)] \quad (3.21)$$

In equation (3.19), (3.20), and (3.21), we have replaced the time index t_i with $i = 1, \dots, K$, for notational simplicity.

With the data model created above, most array processing problems may be categorized as follows. Given the sampled data X in a wireless system, determine:

1. The number of signals q
2. The DOAs $\theta_1, \dots, \theta_q$
3. The signal waveforms $s(1), \dots, s(K)$.

We shall refer to (1) as the detection problem, to (2) as the localization problem, and to (3) as the **beamforming** problem, which is the focus of this research.

3.3 Beamforming and Spatial Filtering

Beamforming is one type of processing used to form beams to simultaneously receive a signal radiating from a specific location and attenuate signals from other locations [7]. Systems designed to receive spatially propagating signals often encounter the presence of interference signals. If the desired signal and interference occupy the same frequency band, unless the signals are uncorrelated, e. g., CDMA signals, then temporal filtering often cannot be used to separate signal from interference. However, the desired and interfering signals usually originate from different spatial locations. This spatial separation can be exploited to separate signal from interference using a spatial filter at the receiver. Implementing a temporal filter requires processing of data collected over a temporal aperture. Similarly, implementing a spatial filter requires processing of data collected over a spatial aperture.

A beamformer is a processor used in conjunction with an array of sensors (i. e., antenna elements in an adaptive array) to provide a versatile form of spatial filtering. The sensor array collects spatial samples of propagating wave fields, which are processed by the beamformer. Typically a beamformer linearly combines the spatially sampled time series from each sensor to obtain a scalar

output time series in the same manner that an FIR filter linearly combines temporally sampled data. There are two types of beamformers, narrowband beamformer and wideband beamformer. A narrowband beamformer is shown in Figure 3.2.

In Figure 3.2, the output at time k , $y(k)$, is given by a linear combination of the data at the M sensors at time k :

$$y(k) = \sum_{i=1}^M w_i^* x_i(k) \quad (3.26)$$

Where $*$ denotes complex conjugate. Since we are now using the complex envelope representation of the received signal, both $x_i(t)$ and w_i are complex. The weight w_i is called the complex weight. The beamformer shown in Figure 3.2 is typically used for processing narrowband signals. In the following discussion, each sensor is assumed to have all necessary receiver electronics and A/D converters if beamforming is performed digitally.

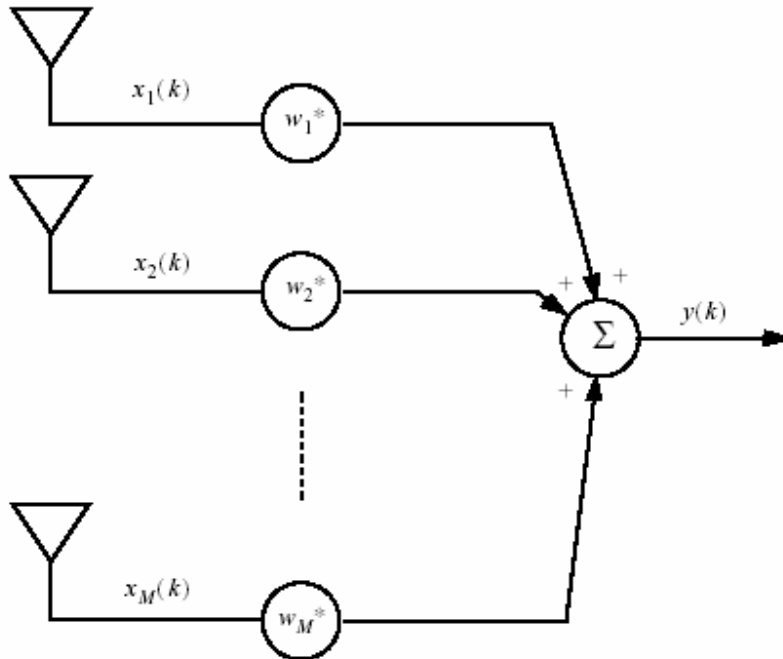


Figure 3.2: A narrowband beamformer forms a linear combination of the sensor outputs.

Equation 3.26 may also be written in vector form as:

$$\mathbf{y}(\mathbf{k}) = \mathbf{W}^H \mathbf{X}(\mathbf{k}) \quad (3.27)$$

where

$$\mathbf{W} = \begin{bmatrix} w_1 & \dots & w_m \\ \vdots & \ddots & \vdots \\ w_1 & \dots & w_m \end{bmatrix} \quad (3.28)$$

and H denotes the Hermitian (complex conjugate) transpose. The vector w is called the complex weight vector.

Different from a narrowband beamformer, a wideband beamformer samples the propagating wave field in both space and time and is often used when signals of significant frequency extent (broadband) are of interest [7]. A wideband beamformer is shown in Figure 3.3. The output in this case may be expressed as

$$y(k) = \sum_{i=1}^M \sum_{l=0}^{K-1} w_{i,l} x_i(k-l) \quad (3.29)$$

where $K - 1$ is the number of delays in each of the M sensor channels. Let

$$\mathbf{w} = [w_{1,0}, \dots, w_{1,K-1}, \dots, w_{M,0}, \dots, w_{M,K-1}]^T \quad (3.30)$$

and

$$\mathbf{X}(\mathbf{k}) = [X_1(k), \dots, X_1(k-K+1), \dots, X_M(k), \dots, X_M(k-K+1)]^T \quad (3.31)$$

where T denotes the conventional transpose, equation (3.29) may also be expressed in vector form as in equation (3.27). In this case, both w and $X(k)$ are $MK \times 1$ column vectors. Comparing Figure 3.2 with Figure 3.3, we see that a wideband beamformer is more complex than narrowband beamformer. Since both types of beamformers may share the same data model, we will concentrate on the narrowband beamformer in the following discussion.

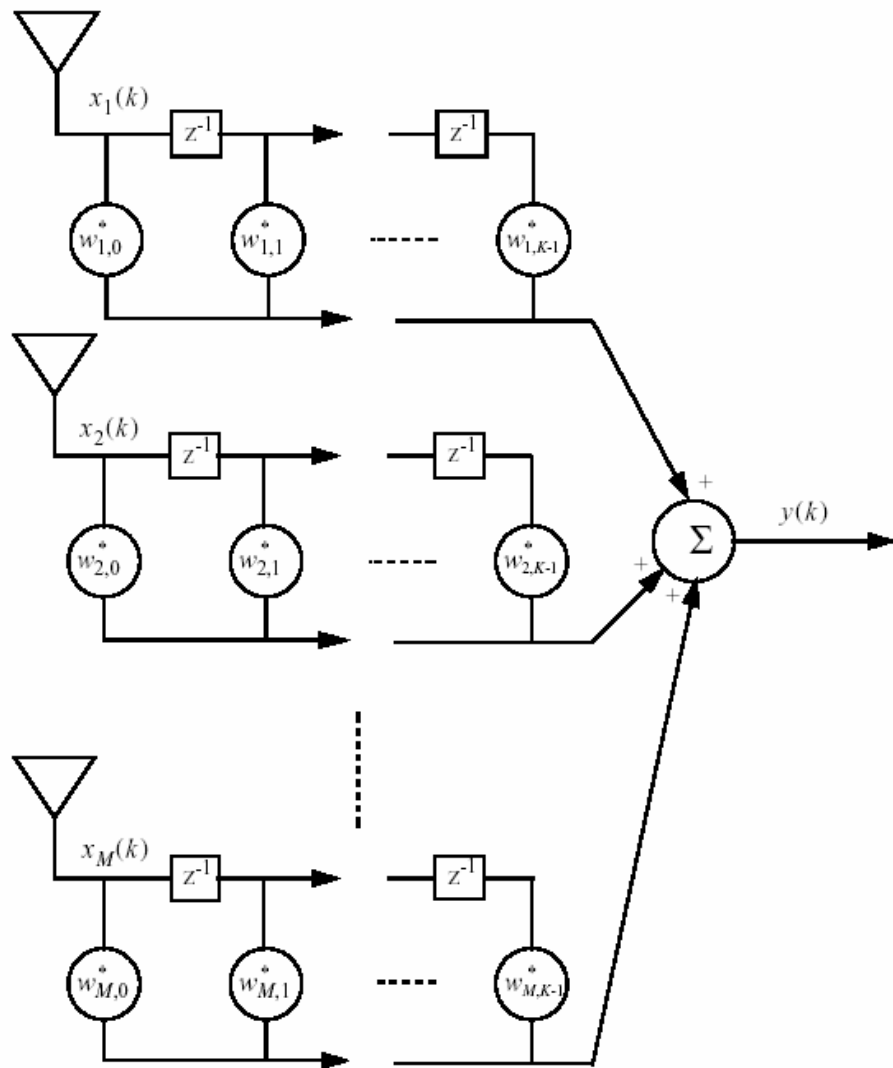


Figure 3.3: A wideband beamformer samples the signal in both space and time.

3.4 Beampattern and Element Spacing

The beampattern and element spacing of an antenna array may be viewed as the counterpart of the magnitude response of a FIR filter and the sampling period of a discrete time signal in the time domain, respectively. To illustrate this point, we may compare the harmonic retrieval problem in the time domain with the beamforming problem in the space domain [6]. Consider a signal $X(t)$ composed of q complex sinusoids with unknown parameters embedded in additive noise:

$$x(t) = \sum_{i=1}^q a_i \exp\{j(2\pi f_i t + \phi_i)\} + n(t), \quad (3.32)$$

where f_i , a_i , and ϕ_i are the frequency, amplitude, and phase, respectively, of the i th sinusoid. Suppose that the signal is sampled with a sampling period T_s unrelated to the frequency of the unknown sinusoid, and let $x(l)$ denote the signal at time instant lT_s , we have

$$x(l) = \sum_{i=1}^q a_i \exp\{j(2\pi f_i(lT_s) + \phi_i)\} + n(lT_s), \quad (3.33)$$

Suppose the sampled signal is fed into an FIR filter with $M - 1$ delay units, which is shown in Figure 3.3, to perform filtering. At time instant lT_s , the filter input and the $M - 1$ outputs of the delay units may be expressed as

$$x(l) = \sum_{i=1}^q a(f_i) s_i(l) + n(l) \quad (3.34)$$

where

$$x(l) = [x(l), x(l - 1), \dots, x(l - M + 1)]^T,$$

$$n(l) = [n(lT_s), \dots, n(l - M + 1)T_s]^T$$

$$a(f_i) = \begin{bmatrix} 1 \\ e^{j2\pi f_i T_s} \\ \vdots \\ e^{j2\pi (M-1)T_s f_i} \end{bmatrix} \quad (3.35)$$

and

$$s_i(l) = a_i \exp\{j(2\pi f_i (lT_s) + \phi_i)\} \quad (3.36)$$

Comparing equation (3.34), (3.35) with equation (3.13), (3.14), we see that for a narrowband uniform linear array (ULA), there is a correspondence between the normalized element spacing, d/λ and the sampling period, T_s , in the FIR filter, also the sine of the DOA $\theta_i, \sin(\theta_i)$, can be related to the temporal frequency f_i of the FIR filter input [7].

Since there is a mapping between the ULA and the FIR filter, a theorem applied to the FIR filter in the time domain may also be applied to the uniform linear array in space domain. In time domain, the Nyquist sampling theorem [8] stated that for a bandlimited signal with highest frequency f , the signal is uniquely determined by its discrete time samples if the sampling rate is equal to or greater than $2f$. If the sampling rate is less than $2f$, aliasing will occur. In the space domain, the sampling rate corresponds to the inverse of the normalized element spacing, and the highest frequency is corresponding to 1 (since $\sin(\theta_i)$ is always less than 1). From the Nyquist sampling theorem, to avoid spatial aliasing, we should have

$$\frac{1}{d/\lambda} \geq 2 \quad (3.37)$$

$$i.e. \quad \frac{\lambda}{d} \geq 2 \quad (3.38)$$

Therefore, the element spacing of an array should always be less than or equal to half of the carrier wavelength. However, the element spacing cannot be made arbitrarily small since two closely spaced antenna elements will exhibit mutual coupling effects. It is difficult to generalize these effects since they depend

heavily on the type of antenna element and the array geometry. However, the mutual coupling between two elements typically tends to increase as the distance between elements is reduced [5]. Thus the spacing between elements must be large enough to avoid significant mutual coupling. In practical linear arrays, the element spacing is often kept near a half wavelength so that the spatial aliasing is avoided and the mutual coupling effect is minimized.

The frequency response of an FIR filter with tap weights w_i , $i = 1, \dots, M$ and a sampling period T_s is given by

$$H(e^{j2\pi f T_s}) = \sum_{i=1}^M w_i e^{j2\pi f T_s (i-1)} \quad (3.39)$$

where $H(e^{j2\pi f T_s})$ represents the response of the filter to a complex sinusoid of frequency f . For the harmonic retrieval problem, if we want to extract the signal with frequency f_i , we need to find a set of complex weights such that the frequency response of the filter has a higher gain at f_i and lower gains (or ideally, nulls) at other frequencies. For the beamforming problem, since f and T_s are corresponding to θ and d/λ , respectively, we can replace f and T_s in equation (3.39) with θ and d/λ , respectively, to get the **beamformer response**,

$$g(\theta) = \sum_{i=1}^M w_i \exp\left\{j \frac{2\pi}{\lambda} (i-1) d \sin \theta\right\} \quad (3.40)$$

where $g(\theta)$ represents the response of the array to a signal with DOA equal to θ . So if there are several signals coming from different directions, and we want to extract the signal with direction θ_i , we need to find a set of weights such that the array response has a higher gain at direction θ_i and lower gains (or ideally, nulls) at other directions.

The array response $g(\theta)$ may also be expressed in vector form as

$$g(\theta) = \mathbf{W}^H \mathbf{a}(\theta) \quad (3.41)$$

where \mathbf{w} and $\mathbf{a}(\theta)$ are defined in equation (2.28) and (2.14), respectively. The beamformer response may also be viewed as the ratio of the beamformer output to the signal at the reference element when a single plane wave is incident on the array.

The beampattern is defined as the magnitude of $g(\theta)$ [7] and is given by

$$G(\theta) = |g(\theta)| \quad (3.42)$$

Using $G(\theta)$, we may define the normalized beamformer response,

$$g_n(\theta) = \frac{g(\theta)}{\max\{G(\theta)\}} \quad (3.43)$$

where $g_n(\theta)$ is also known as the normalized radiation pattern or array factor of the array.

The spatial discrimination capability of a beamformer depends on the size of the spatial aperture of the array; as the aperture increases, discrimination improves. The absolute aperture size is not important, rather its size in wavelengths is the critical parameter. To illustrate this point, let us consider a uniform linear array with equal weight for each element. From equation (3.40), we get the beamformer response

$$g(\theta) = \sum_{i=1}^M e^{j \frac{2\pi}{\lambda} (i-1) d \sin \theta}$$

$$= \frac{1 - e^{j \frac{2\pi}{\lambda} M d \sin \theta}}{1 - e^{j \frac{2\pi}{\lambda} d \sin \theta}} \quad (3.44)$$

$$= \frac{\sin\left(\frac{M d}{\lambda} \sin \theta\right)}{\sin\left(\frac{d}{\lambda} \sin \theta\right)} e^{j \frac{(M-1)d}{\lambda} \sin \theta} \quad (3.45)$$

The beampattern of this equal-weight beamformer is shown in Figure 3.4. The polar coordinate plot of the beampattern is also shown in Figure 3.5. In Figure 3.4, the normalized beampattern gain is expressed in dB. From equation (3.45), we see that the null-to-null beamwidth, θ_{BW} , of the array is determined by

$$\theta_{BW} = \frac{Md}{H} \sin^{-1} \left(\frac{\lambda}{Md} \right) \quad (3.46)$$

The solution of equation (3.46) may be expressed as

$$\theta_H = \arcsin \left(\frac{\lambda}{Md} \right) \quad (3.47)$$

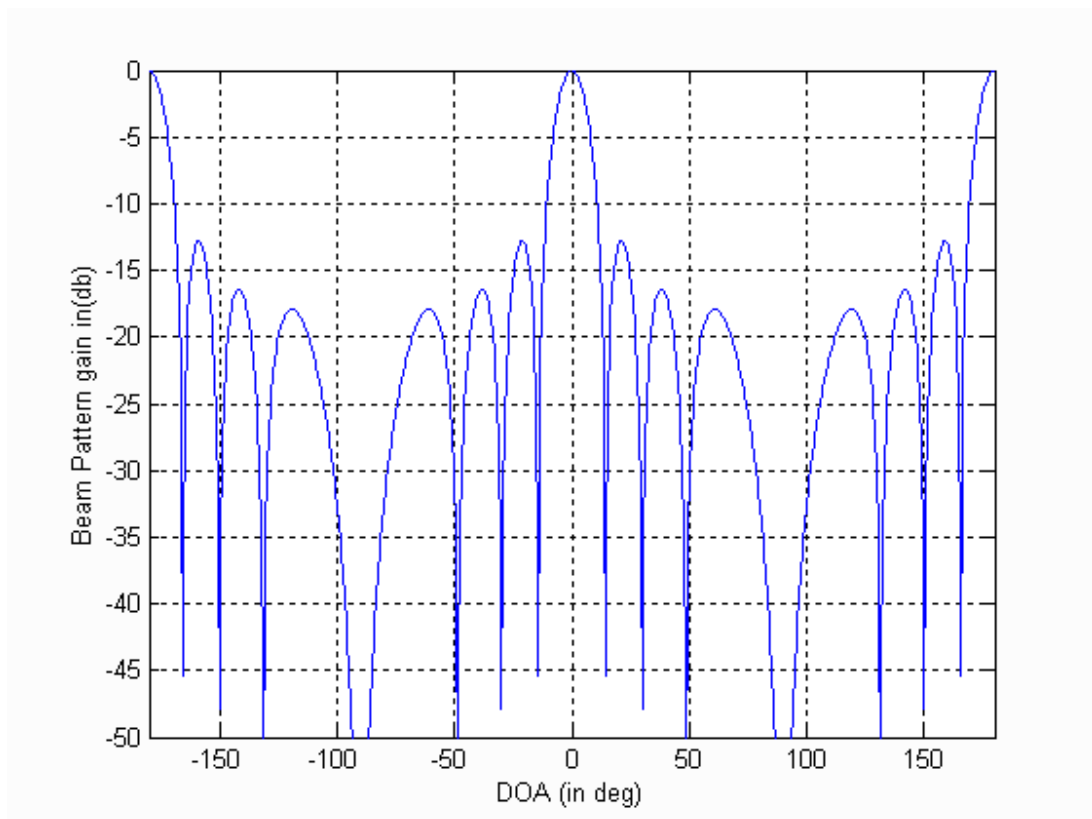


Figure 3.4: Beampattern for an equal-weight beamformer. In this case, the number of elements is equal to 8, and the element spacing is half of the carrier wavelength.

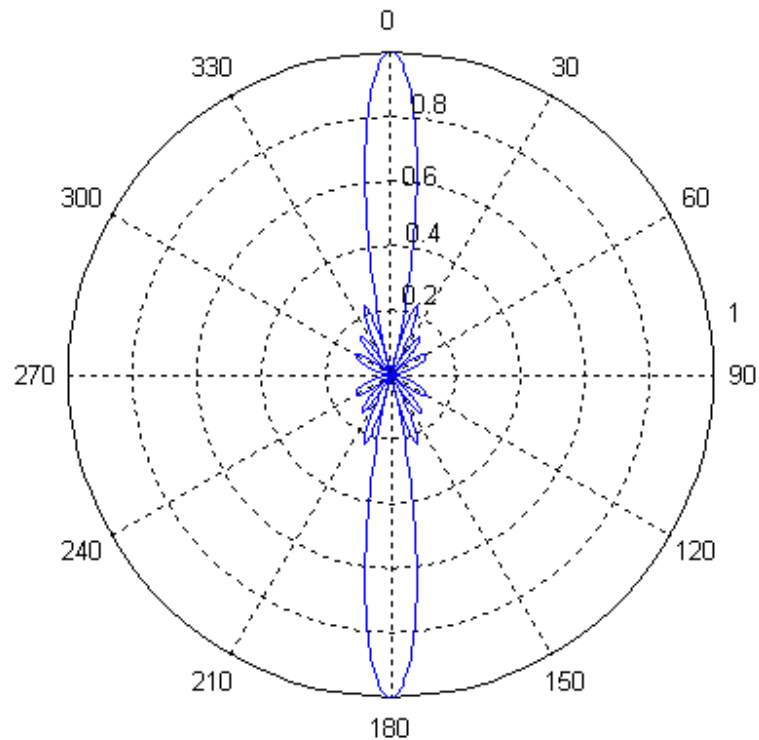


Figure 3.5: The same beampattern as shown in Figure 3.4 with polar coordinate plot in azimuth.

where θ_H is the peak-to-null beamwidth, or half of the null-to-null beamwidth. The null-to-null beamwidth is then obtained by

$$\theta_{BW} = 2\theta_H = 2 \arcsin \left(\frac{\lambda}{Md} \right) \quad (3.48)$$

From equation (3.48), we see that the beamwidth, which is the width of the main lobe of the beampattern, is inversely proportional to the term Md/λ .

Therefore, if the aperture size in wavelengths is large, the beamwidth of the array will be small, and the beamformer will have a high spatial discrimination capability. For a general beamformer with unequal weights w_{iue} ; $i = 1, 2, \dots, M$, the weight w_{iue} may be viewed as the product of the equal weight series and a periodic weight series:

$$w_{iue} = w_{ie} \bar{w}_{iue}, \quad i = 1, 2, \dots, M \quad (3.49)$$

where

$$w_{ie} = \begin{cases} 1 & i = 1, 2, \dots, M \\ 0 & \text{otherwise} \end{cases} \quad (3.50)$$

and

$$\bar{w}_{(i \neq k)ue} = w_{iue}, \quad i = 1, 2, \dots, M \text{ and } k = 1, 2, \dots, M \quad (3.51)$$

Since in the time domain, the frequency response of a periodic time series contains a delta function, similarly, in the space domain, the beampattern corresponding to the weight series \bar{w}_{iue} will have a high angle resolution. The resulting beampattern corresponding to the unequal weight w_{iue} thus can be viewed as the convolution of the equal-weight beampattern with a beampattern having high angle resolution, and the beamwidth (or equivalently, the aperture size in wavelength) of the equal-weight beamformer determines the spatial discrimination capability of a general beamformer. So, in addition to minimizing the mutual coupling effect between the array elements, maximizing the aperture size is another reason to keep the element spacing of the array near a half wavelength.

From Figure 3.5, we see that the beampattern of the array is symmetrical about the axis connecting the array elements. This symmetry of the beampattern is inherent in the uniform linear array. From equation (3.40), we see that $g(\theta) = g(\pi - \theta)$, thus the beampattern of the uniform linear array is symmetrical about the endfire of the array. Arrays with nonlinear spacing or with other geometry, such as the circular array, do not exhibit this kind of symmetry.

3.5 Adaptive Arrays

In a mobile communication system, the mobile is generally moving; therefore the DOAs of the received signals in the base station are time-varying. Also, due to the time-varying wireless channel between the mobile and the base station, and the existence of the cochannel interference, multipath, and noise, the parameters of each impinging signal are varied with time. For a beamformer with constant weights, the resulting beampattern cannot track these time-varying factors. However, an adaptive array [9] may change its patterns automatically in response to the signal environment. An adaptive array is an antenna system that can modify its beampattern or other parameters, by means of internal feedback

control while the antenna system is operating. Adaptive arrays are also known as adaptive beamformers, or adaptive antennas. A simple narrowband adaptive array is shown in Figure 3.6.

In Figure 3.6, the complex weights W_1, \dots, W_M are adjusted by the adaptive control processor. The method used by the adaptive control processor to change the weights is called the adaptive algorithm. Most adaptive algorithms are derived by first creating a performance criterion, and then generating a set of iterative equations to adjust the weights such that the performance criterion is met. Some of the most frequently used performance criteria include minimum mean squared error (MSE), maximum signal-to-interference-and noise ratio (SINR), maximum likelihood (ML), minimum noise variance, minimum output power, maximum gain, etc. [9]. These criteria are often expressed as cost functions which are typically inversely associated with the quality of the signal at the array output. As the weights are iteratively adjusted, the cost function becomes smaller and smaller. When the cost function is minimized, the performance criterion is met and the algorithm is said to have converged. For one adaptive array, there may exist several adaptive algorithms that could be used to adjust the weight vector. The choice of one algorithm over another is determined by various factors [10]:

1. Rate of convergence. This is defined as the number of iterations required for the algorithm, in response to stationary input, to converge to the optimum solution. A fast rate of convergence allows the algorithm to adapt rapidly to a stationary environment of unknown statistics.
2. Tracking. When an adaptive algorithm operates in a nonstationary environment, the algorithm is required to track statistical variations in the environment.
3. Robustness. In one context, robustness refers to the ability of the algorithm to operate satisfactorily with ill-conditioned input data. The term robustness is also used in the context of numerical behavior.
4. Computational requirements. Here the issues of concern include (a) the number of operations (i.e. multiplications, divisions, and additions/subtractions) required to make one complete iteration of the algorithm, (b) the size of memory locations required to store the data and the program, and (c) the investment required to program the algorithm on a computer or a DSP processor.

Since there exists a mapping between the narrowband beamformer and the FIR filter, most of the adaptive algorithms used by the adaptive filter [10] may be applied to the adaptive beamformer. However, some of the adaptive

beamforming algorithms also have their unique aspects that an adaptive filter algorithm does not possess. A thorough survey of adaptive beamforming algorithms is given in Chapter 4.

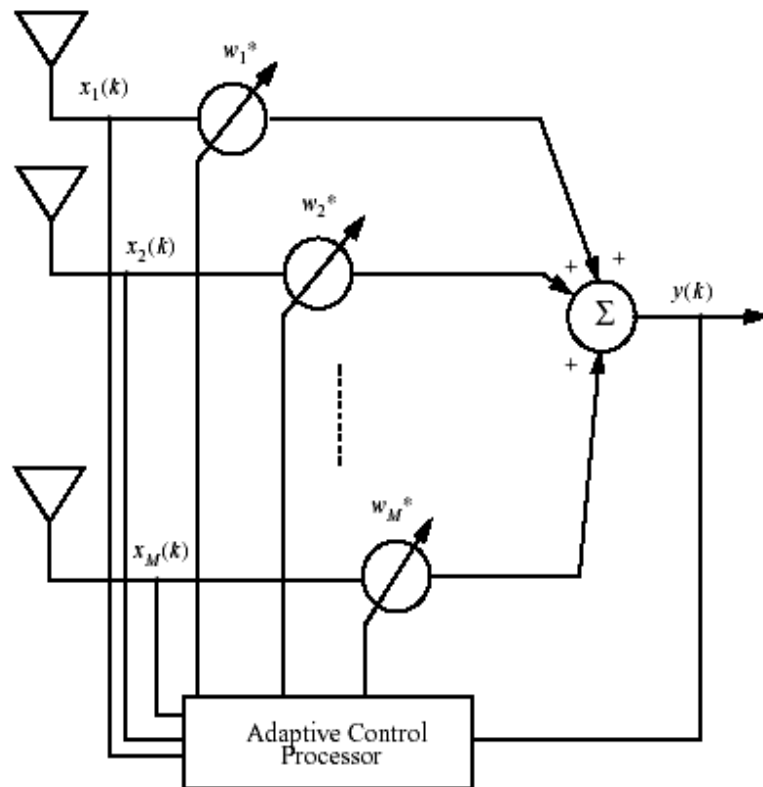


Figure 3.6 A simple narrowband adaptive array.

Chapter 4

Overview of Adaptive Beamforming Algorithms

4.1 Introduction

This chapter provides a thorough survey of adaptive beamforming algorithms. Most of these algorithms may be categorized into two classes according to whether a training signal is used or not. One class of these algorithms is the non-blind adaptive algorithm in which a training signal is used to adjust the array weight vector. Another technique is to use a blind adaptive algorithm, which does not require a training signal.

Since the non-blind algorithms use a training signal, during the training period, data cannot be sent over the radio channel. This reduces the spectral efficiency of the system. Therefore, the blind algorithms are of more research interest. The research here focuses primarily on blind beamforming algorithms.

4.2 Non-blind Adaptive Algorithms

In a non-blind adaptive algorithm, a training signal, $d(t)$, which is known, to both the transmitter and receiver, is sent from the transmitter to the receiver during the training period. The beamformer in the receiver uses the information of the training signal to compute the optimal weight vector, \mathbf{w}_{opt} . After the training period, data is sent and the beamformer uses the weight vector computed previously to process the received signal. If the radio channel and the interference characteristics remain constant from one training period until the next, the weight vector \mathbf{w}_{opt} will contain the information of the channel and the interference, and their effect on the received signal will be compensated in the output of the array.

4.2.1 Wiener Solution

Most of the non-blind algorithms try to minimize the mean-squared error between the desired signal $d(t)$ and the array output $y(t)$. Let $y(k)$ and $d(k)$ denote the sampled signal of $y(t)$ and $d(t)$ at time instant t_k , respectively. Then the error signal is given by [10]

$$e(k) = d(k) - y(k) \tag{4.1}$$

and the mean-squared error is defined by

$$J = E [e(k)^2] \quad (4.2)$$

where $E [\]$ denotes the ensemble expectation operator. Substituting equation (4.1) and (3.27) into equation (4.2), we have

$$\begin{aligned} J &= E [|d(k) - y(k)|^2] \\ &= E [\{d(k) - y(k)\} \{d(k) - y(k)\}^*] \\ &= E [\{d(k) - \mathbf{W}^H \mathbf{x}(k)\} \{d(k) - \mathbf{W}^H \mathbf{x}(k)\}^*] \\ &= E [|d(k)|^2 - d(k) \mathbf{x}^H(k) \mathbf{w} - \mathbf{W}^H \mathbf{x}(k) d^*(k) + \mathbf{W}^H \mathbf{x}(k) \mathbf{X}^H(k) \mathbf{w}] \\ &= E [|d(k)|^2 - \mathbf{P}^H \mathbf{W} - \mathbf{W}^H \mathbf{P} + \mathbf{W}^H \mathbf{R} \mathbf{W}] \end{aligned} \quad (4.3)$$

where

$$\mathbf{R} = E [\mathbf{x}(k) \mathbf{x}^H(k)] \quad (4.4)$$

and

$$\mathbf{P} = E [\mathbf{x}(k) d^*(k)] \quad (4.5)$$

In equation (4.3), \mathbf{R} is the $M \times M$ correlation matrix of the input data vector $\mathbf{X}(k)$, and \mathbf{P} is the $M \times 1$ cross-correlation vector between the input data vector and the desired signal $d(k)$.

The gradient vector of J , $\nabla (J)$, is defined by

$$\nabla (J) = 2 \frac{\nabla J}{\nabla \mathbf{W}} \quad (4.6)$$

Where $\frac{\nabla J}{\nabla \mathbf{W}}$, denotes the conjugate derivative with respect to the complex vector \mathbf{w} (see Appendix B for more details on differentiation with respect to a complex vector). When the mean-squared error J is minimized, the gradient vector will be equal to an $M \times 1$ null vector.

$$\nabla (J)_{\mathbf{W}_{opt}} = \mathbf{0} \quad (4.7)$$

Substituting equation (4.3) into equation (4.7), we have

$$-2\mathbf{p} + 2\mathbf{R}\mathbf{w}_{opt} = \mathbf{0} \quad (4.8)$$

Or equivalently

$$\mathbf{R}\mathbf{w}_{opt} = \mathbf{p} \quad (4.9)$$

Equation (4.9) is called the Wiener-Hopf equation [10]. Premultiplying both sides of equation (4.9) by \mathbf{R}^{-1} , the inverse of the correlation matrix, we obtain

$$\mathbf{W}_{opt} = \mathbf{R}^{-1}\mathbf{p} \quad (4.10)$$

The optimum weight vector \mathbf{w}_{opt} in equation (4.10) is called the Wiener solution. From equation (4.10), we see that the computation of the optimum weight vector \mathbf{w}_{opt} requires knowledge of two quantities: (1) the correlation matrix \mathbf{R} of the input data vector $\mathbf{x}(k)$, and (2) the cross-correlation vector \mathbf{p} between the input data vector $\mathbf{x}(k)$ and the desired signal $d(k)$.

4.2.2 Method of Steepest-Descent

Although the Wiener-Hopf equation may be solved directly by calculating the product of the inverse of the correlation matrix \mathbf{R} and the cross-correlation vector \mathbf{p} , nevertheless, this procedure presents serious computational difficulties since calculating the inverse of the correlation matrix results in a high computational complexity. An alternative procedure is to use the method of steepest-descent [10]. To find the optimum weight vector \mathbf{w}_{opt} by the steepest-descent method we proceed as follows:

1. Begin with an initial value $\mathbf{w}(0)$ for the weight vector, which is chosen arbitrarily. Typically, $\mathbf{w}(0)$ is set equal to a column vector of an $M \times M$ identity matrix.
2. Using this initial or present guess, compute the gradient vector $\nabla(J)$ at time k (i. e., the k th iteration).
3. Compute the next guess at the weight vector by making a change in the initial or present guess in a direction opposite to that of the gradient vector.
4. Go back to step 2 and repeat the process.

It is intuitively reasonable that successive corrections to the weight vector in the direction of the negative of the gradient vector should eventually lead to the minimum mean-squared error J_{min} , at which the weight vector assumes its optimum value \mathbf{w}_{opt} .

Let $\mathbf{w}(k)$ denote the value of the weight vector at time k . According to the method of steepest-descent, the update value of the weight vector at time $k + 1$ is computed by using the simple recursive relation

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{1}{2} \mu \nabla J(\mathbf{w}(k)), \quad (4.11)$$

where μ is a positive real-valued constant. The factor $1/2$ is used merely for convenience.

From equation (4.8) we have

$$\nabla J = -2\mathbf{P} + 2\mathbf{R}\mathbf{w}(k) \quad (4.12)$$

Substituting equation (4.12) into (4.11), we obtain

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu [\mathbf{P} - \mathbf{R}\mathbf{w}(k)] \quad k = 0, 1, 2, \dots \quad (4.13)$$

Using equation (4.4), (4.5), (4.1), and (4.27), the gradient vector in equation (4.12) may be written in another form

$$\begin{aligned} \nabla J &= -2E[\mathbf{x}(k)d^*(k) - \mathbf{x}(k)\mathbf{x}^H(k)\mathbf{w}(k)] \\ &= -2E[\mathbf{x}(k)\{d(k) - y(k)^*\}] \\ &= -2E[\mathbf{x}(k)\mathbf{e}^*(k)] \end{aligned} \quad (4.14)$$

Also, equation (4.11) can be expressed as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu E[\mathbf{x}(k)\mathbf{e}^*(k)] \quad (4.15)$$

We observe that the parameter μ controls the size of the incremental correction applied to the weight vector as we proceed from one iteration cycle to the next. We therefore refer to μ as the step-size parameter or weighting constant. Equations (4.13) and (4.15) describe the mathematical formulation of the steepest-descent method.

4.2.3 Least-Mean-Squares Algorithm

If it were possible to make exact measurements of the gradient vector $\nabla(J)$ at each iteration, and if the step-size parameter μ is suitably chosen, then the weight vector computed by using the steepest-descent method would indeed converge to the optimum Wiener solution. In reality, however, exact measurements of the gradient vector are not possible since this would require prior knowledge of both the correlation matrix R of the input data vector and the cross-correlation vector P between the input data vector and the desired signal.

Consequently, the gradient vector must be estimated from the available data. In other words, the weight vector is updated in accordance with an algorithm that adapts to the incoming data. One such algorithm is the least-mean-squares (LMS) algorithm [10][11]. A significant feature of the LMS algorithm is its simplicity; it does not require measurements of the pertinent correlation functions, nor does it require matrix inversion. To develop an estimate of the gradient vector $\nabla(J)$, the most obvious strategy is to substitute the expected value in equation (4.14) with the instantaneous estimate,

$$\hat{\nabla}(J(k)) \approx -2\mathbf{x}(k)e^*(k) \quad (4.16)$$

Substituting this instantaneous estimate of the gradient vector into equation (4.11), we have

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{x}(k)e^*(k) \quad (4.17)$$

Now, we can describe the **LMS** algorithm by the following three equations

$$y(k) = \mathbf{w}^H(k)\mathbf{x}(k) \quad (4.18)$$

$$e(k) = d(k) - y(k) \quad (4.19)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\mathbf{x}(k)e^*(k) \quad (4.20)$$

The LMS algorithm is a member of a family of stochastic gradient algorithms since the instantaneous estimate of the gradient vector is a random vector that depends on the input data vector $\mathbf{x}(k)$. The LMS algorithm requires about $2M$ complex multiplications per iteration, where M is the number of weights (elements) used in the adaptive array. The response of the LMS algorithm is determined by three principal factors: (1) the step-size parameter, (2) the number

of weights, and (3) the eigen-value of the correlation matrix of the input data vector.

4.3 Blind Adaptive Algorithms

Blind adaptive algorithms do not require a training sequence. Instead, they exploit some known properties of the desired received signal. Most of the blind algorithms may be categorized into the following three classes or combinations of them:

- ? Algorithms based on estimation of the DOAs of the received signals.
- ? Algorithms based on property-restoral techniques.
- ? Algorithms based on the discrete-alphabet structure of digital signals.

4.3.1 Algorithms Based on Estimation of DOAs of Received Signals

In the DOA-estimation-based algorithms, the DOAs of the received signals are first determined by using prior knowledge of the array response, i. e., the array manifold or special array structure. The high-resolution techniques for DOA estimation include MUSIC [14] and ESPRIT [15]. After the DOAs are estimated, an optimum beamformer is then constructed from the corresponding array response to extract the desired signals from interference and noise [16]. The performance of this technique strongly depends on the reliability of the prior spatial information, e. g. the array manifold. In many situations of practical interest, this information is not available. Even if this information is available, the cost is very expensive and the information may be inaccurate. The computational complexity of these algorithms is also very high. Another disadvantage of this technique is that the number of DOAs that the algorithm can estimate is limited by the number of array elements. In a wireless communication system, especially in a code division multiple access (CDMA) system, the number of users in a radio channel may be greater than the number of array elements, and if the multipaths of each user's signal are taken into account, the total number of signals impinging on the array will far exceed the number of array elements, and the DOA estimation algorithm in this case will fail. Furthermore, this approach does not exploit the known temporal structure of the incoming signals.

4.3.2 Algorithms Based on Property-Restoral Techniques

Generally, most digital communication signals possess some kinds of properties such as the constant modulus property or the spectral self-coherence property.

Due to the interference, noise, and the time-varying channel in a communication system, these properties may be corrupted when the signal is received at the receiver. The adaptive array in the receiver tries to restore these properties using a property-restoration-based algorithm, and hopes that by restoring these properties, the output of the array is a reconstructed version of the transmitted signal.

Constant Modulus Algorithm

Some communication signals such as phase-shift keying (PSK), frequency-shift keying (FSK), and analog FM signals have a constant envelope. This constant envelope may be distorted when the signal is transmitted through the channel. The constant modulus algorithm (CMA) adjusts the weight vector of the adaptive array to minimize the variation of the envelope at the output of the array. After the algorithm converges, the array can steer a beam in the direction of the signal of interest (SOI), and nulls in the directions of the interference.

The CMA tries to minimize the cost function

$$J(k) = E [|y(k)|^p - 1]^q \quad (4.27)$$

The convergence of the algorithm depends on the coefficients p and q in equation (4.27). Usually, the cost function J with $p = 1, q = 2$, or $p = 2, q = 2$ is used. Here we use J with $p = 1, q = 2$. With the cost function of this 1-2 form, the CMA minimizes the function

$$J(k) = E [|y(k)| - 1]^2 \quad (4.28)$$

The gradient vector is given by

$$\nabla (J(k)) = 2 \frac{\nabla J(k)}{\nabla w^*(k)}$$

$$\begin{aligned}
& 2E\left\{ |y(k)| \frac{|y(k)|}{w^*(k)} \right\} \\
& 2E\left\{ |y(k)| \frac{\left| \{y(k)y^*(k)\}^{\frac{1}{2}} \right|}{w^*(k)} \right\} \\
& 2E\left\{ |y(k)| \frac{\left| \{w^H(k)x(k)x^H(k)w(k)\}^{\frac{1}{2}} \right|}{w^*(k)} \right\} \\
& E\left\{ |y(k)| \{w^H(k)x(k)x^H(k)w(k)\}^{\frac{1}{2}} \frac{\left| \{w^H(k)x(k)x^H(k)w(k)\} \right|}{w^*(k)} \right\} \\
& E\left\{ |y(k)| \frac{1}{|y(k)|} x(k)x^H(k)w(k) \right\} \\
& E\left\{ \left(1 - \frac{1}{|y(k)|}\right) x(k)y(k)^* \right\} \\
& E\left\{ x(k)y(k) \frac{y(k)^*}{|y(k)|} \right\} \tag{4.29}
\end{aligned}$$

Ignoring the expectation operator in equation (4.29), the instantaneous estimate of the gradient vector can be written as

$$\hat{J}(k) = x(k)y(k) \frac{y(k)^*}{|y(k)|} \tag{4.30}$$

Using the method of steepest-descent, and replacing the gradient vector with its instantaneous estimate, we can update the weight vector by

$$w(k+1) = w(k) - \mu \hat{\nabla} J(k) \quad (4.31)$$

$$\hat{\nabla} J(k) = x(k) y(k) \frac{y(k)^*}{|y(k)|}$$

where μ is the step-size parameter. Now, we can describe the steepest-descent CMA (SD-CMA) by the following three equations

$$y(k) = w^H(k)x(k) \quad (4.32)$$

$$e(k) = y(k) \frac{y(k)}{|y(k)|} \quad (4.33)$$

$$w(k+1) = w(k) + \mu x(k) e^*(k) \quad (4.34)$$

From equation (4.33) we see that when the output of the array has a unity magnitude, i. e., $|y(k)| = 1$, the error signal becomes zero. Comparing the above three equations with equation (4.18), (4.19), and (4.20), we see that the CMA is very similar to the LMS algorithm, and the term $\frac{y(k)}{|y(k)|}$ in CMA plays the same role as the desired signal $d(t)$ in the LMS algorithm. However, the reference signal $d(t)$ must be sent from the transmitter to the receiver and must be known for both the transmitter and receiver if the LMS algorithm is used. The CMA algorithm does not require a reference signal to generate the error signal at the receiver. Several properties of the constant modulus algorithm are discussed in [17]. One of the properties is the phase roll or phase ambiguity. In CMA, if a weight vector w generates an array output with constant modulus, so does its phase shifted version,

$$w_r = e^{j\theta} w \quad (4.35)$$

In other words, the phase of the array output, $y(k)$, is indeterminate, which is in contrast to the LMS algorithm. This is reasonable, since CMA uses only the

amplitude information of the array output in the control of the system and the phase information is not utilized. The convergence behavior of CMA is discussed in [20][21][22][23]. It was found that the constant modulus cost function will in general have two classes of stationary solutions in the single signal-of-interest (SOI) environment: signal-capture solutions where the signal is captured with maximum attainable signal-to-interference-and-noise ratio (SINR), and noise-capture solutions where the desired signal is nulled by the array. The noise-capture solutions must be avoided if the array is being used to extract the signal of interest.

Least-Squares CMA The constant modulus algorithm was first used by Gooch [24] in the beamforming problem. After that, many CMA-type algorithms have been proposed for use in adaptive arrays. In [25], B. G. Agee developed the least-squares constant modulus algorithm (LS-CMA) by using the extension of the method of nonlinear least-squares (Gauss's method) [26]. The extension of Gauss's method states that if a cost function can be expressed in the form

$$F(w) = \sum_{k=1}^K |g_k(w)|^2 \quad (4.36)$$

$$= \|g(w)\|_2^2$$

where

$$g(w) = [g_1(w), g_2(w), \dots, g_K(w)]^T \quad (4.37)$$

then the cost function has a partial Taylor-series expansion with sum-of-squares form

$$F(w) \approx F(w_0) + \frac{1}{2} \left\| g(w_0) + D^H(w_0)(w - w_0) \right\|_2^2 \quad (4.38)$$

where w_0 is an offset vector, and

$$D(w) = [g_1'(w), g_2'(w), \dots, g_K'(w)] \quad (4.39)$$

The gradient vector of $F(w)$ with respect to w is given by

$$\begin{aligned}
\frac{\partial F(w)}{\partial w} &= 2 \frac{\partial F(w)}{\partial w} \\
&= 2 \frac{\partial \left(\|g(w)\|_2^2 + g^H(w) D^H(w) g(w) \right)}{\partial w} \\
&= 2 \frac{\partial \left(\|g(w)\|_2^2 + g^H(w) D^H(w) g(w) + D^H(w) D(w) \right)}{\partial w} \\
&= 2 \left(D(w) g(w) + D(w) D^H(w) \right)
\end{aligned} \tag{4.40}$$

Setting $\frac{\partial F(w)}{\partial w}$ equal to zero, we can find the offset vector which minimizes $F(w)$,

$$D(w) D^H(w) D(w) g(w) = 0 \tag{4.41}$$

Therefore, the weight vector can be updated by

$$w(l+1) = w(l) + \eta D^H(w(l)) D(w(l)) g(w(l)) \tag{4.42}$$

where l denotes the iteration number. The LS-CMA is derived by applying equation (4.42) to the constant modulus cost function,

$$\begin{aligned}
F(w) &= \sum_{k=1}^K \left| |y(k)| - 1 \right|^2 \\
&= \sum_{k=1}^K \left| |w^H x(k)| - 1 \right|^2
\end{aligned} \tag{4.43}$$

Comparing equation (4.43) with (4.36), we see that in this case,

$$\begin{aligned}
g_k(w) &= |y(k)| - 1 \\
&= |w^H x(k)| - 1
\end{aligned} \tag{4.44}$$

Substituting equation (4.44) into (4.37), we obtain

$$g(w) = \begin{bmatrix} |y(1)| - 1 \\ |y(2)| - 1 \\ \vdots \\ |y(k)| - 1 \end{bmatrix} \quad (4.45)$$

The gradient vector of $\mathbf{g}_k(w)$ is given by

$$\frac{\partial g_k(w)}{\partial w^*} = x(k) \frac{y^*(k)}{|y(k)|} \quad (4.46)$$

Substituting equation (4.46) into (4.39), $D(w)$ can be expressed as

$$D(w) = \begin{bmatrix} (g_1(w)) \\ (g_2(w)), \dots \\ (g_k(w)) \end{bmatrix} = \begin{bmatrix} x(1) \frac{y^*(1)}{|y(1)|} \\ x(2) \frac{y^*(2)}{|y(2)|} \\ \vdots \\ x(k) \frac{y^*(k)}{|y(k)|} \end{bmatrix} \quad (4.47)$$

$$= XY_{CM}$$

where

$$X = [x(1), x(2), \dots, x(K)] \quad (4.48)$$

and

$$Y_{CM} \begin{bmatrix} \frac{y^*(1)}{|y(1)|} & 0 & \dots & 0 \\ 0 & \frac{y^*(2)}{|y(2)|} & & \\ \vdots & \vdots & & \vdots \\ 0 & \dots & 0 & \frac{y^*(k)}{|y(k)|} \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ \vdots & & & \vdots \end{bmatrix} \quad (4.49)$$

Using equation (4.47) and (4.45), we have

$$D(w)D^H(w) = XY_{CM}Y_{CM}^H X^H = XX^H \quad (4.50)$$

and

$$D(w)g(w) = XY_{CM} \begin{bmatrix} |y(K)| \\ |y(K)| \\ \vdots \\ |y(K)| \end{bmatrix}$$

$$\begin{bmatrix}
 y(1) & \frac{y^*(1)}{|y(1)|} \\
 y(2) & \frac{y^*(2)}{|y(2)|} \\
 \vdots & \vdots \\
 y(K) & \frac{y^*(K)}{|y(K)|}
 \end{bmatrix}$$

$$\mathbf{X}(\mathbf{y}; \mathbf{r})^* \tag{4.51}$$

where

$$\mathbf{y} = [y(1), y(2), \dots, y(K)]^T \tag{4.52}$$

$$\mathbf{r} = \left[\frac{y(1)}{|y(1)|}, \frac{y(2)}{|y(2)|}, \dots, \frac{y(K)}{|y(K)|} \right]^T \tag{4.53}$$

The vector \mathbf{y} and \mathbf{r} are called the output data vector and complex-limited output data vector, respectively. Substituting equation (4.50) and (4.51) into equation (4.42), we obtain

$$\begin{aligned}
 W(l+1) &= W(l) \mathbf{X} \mathbf{X}^H \mathbf{X}(\mathbf{y}(l); \mathbf{r}(l))^* \\
 &= W(l) \mathbf{X} \mathbf{X}^H \mathbf{X} \mathbf{X}^H W(l) \mathbf{X} \mathbf{X}^H \mathbf{X}(\mathbf{y}(l); \mathbf{r}(l))^* \\
 &= \mathbf{X} \mathbf{X}^H \mathbf{X}(\mathbf{y}(l); \mathbf{r}(l))^*
 \end{aligned} \tag{4.54}$$

where $\mathbf{y}(l)$ and $\mathbf{r}(l)$ are the output data vector and complex-limited output data vector corresponding to the weight vector $w(l)$ in the l th iteration, respectively.

The LS-CMA can be implemented either statically or dynamically. The static LS-CMA repeatedly uses one data block X , which contains K snapshots of the input data vectors, in the updating of the weight vector w . In the static LS-CMA, after a new weight vector $w(l+1)$ is calculated using equation (4.54), this new weight vector is used with the input data block X , which was also used in the last iteration, to generate the new output data vector $y(l+1)$ and the complex-limited output data vector $r(l+1)$. The new complex-limited output data vector is then substituted into equation (4.54) to generate a new weight vector. In dynamic LS-CMA, however, different input data blocks are used during the updating of the weight vector. Let $X(l)$ denote the input data block used in the l th iteration. $X(l)$ can be expressed, as where L is the number of iterations required for the algorithm to converge.

$$X(l) = [X(1:k), X(2:k), \dots, X((l-1)k)]^T \quad l = 0, 1, \dots, L \quad (4.55)$$

Using $X(l)$, we can describe the dynamic LS-CMA by the following equations

$$y(l) = [W^H(l)X(l)]^T \\ = [y(1:k), y(2:k), \dots, y((l-1)k)]^T \quad (4.56)$$

$$r(l) = \left[\frac{y(1:k)}{y(1:k)}, \frac{y(2:k)}{y(2:k)}, \dots, \frac{y((l-1)k)}{y(1:k)} \right]^T \quad (4.57)$$

$$w(l+1) = [X(l)X^H(l)]^{-1} X(l)r(l) \quad (4.58)$$

From the above equations we see that while the SD-CMA updates the weight vector on a sample-by-sample basis, the dynamic LS-CMA adjusts the weight vector on a block-by-block basis. Since the LS-CMA can exploit the information in a data block, it converges faster than the SD-CMA algorithm.

In [27], B. G. Agee applied the dynamic LS-CMA to a multitarget problem and developed the multitarget least-squares constant modulus algorithm (MT-LSCMA). The challenge of the multitarget problem is to separate and extract the signals if all the signals in the communication system have the same constant modulus property. We will discuss the MT-LSCMA in greater detail in Chapter 5.

All the beamformer adaptation algorithms discussed to date are performed in data space (element space). The adaptation process can also be performed in beam space. In [35], Chiba et al. proposed the beam-space CMA (BS-CMA). In the BS-CMA, a multibeam is formed by means of a fast Fourier transform (FFT) and only the beams whose output power exceeds a threshold level are selected. The CMA is then used on the selected beam outputs to perform the adaptation process. It was reported in [35] that the BS-CMA has an advantage of carrying out the capture of the desired signal and the elimination of the interference effectively with degrees of freedom corresponding to the number of impinging signals. Hence, this algorithm is effective for applications to adaptive arrays with a relatively large number of elements (more than 10) used in mobile satellite communications. It was also reported in [35] that the BS-CMA converges faster than the conventional element-space CMA (ES-CMA) and can increase the dynamic range of the received signal power.

4.3.3 Algorithms Based on Discrete-Alphabet Structure of Digital Signal

This type of algorithm exploits the discrete-alphabet property of digitally modulated signals to provide unique signal estimates. Similar to ILSP-CMA, this type of algorithm tries to minimize the cost function

$$J(A, s) = \|X - AS\|_F^2$$

where the elements of \mathbf{S} are constrained to a finite alphabet. In [43], Talwar et al. proposed three algorithms: the iterative least squares with projection (ILSP) algorithm, the iterative least squares with enumeration (ILSE) algorithm and the recursive least squares with enumeration (RLSE) algorithm. The performance of these algorithms is analyzed in [44]. However, all the above three algorithms were developed under the assumption that all of the signals arrive synchronously at the array, which is not necessarily true in reality.

4.3.4 Other Blind Beamforming Algorithms

In addition to the three classes of blind beamforming algorithms discussed above, there are other blind algorithms that are reported in the literature.

2D RAKE Receiver

A RAKE receiver in a CDMA system combines the multipath components of a signal to improve the system performance [45][46]. However, if the delay between the multipath components is less than the chip period, the components cannot be resolved and the conventional RAKE receiver will fail. In [47][48], Suard et al. proposed the 2D RAKE receiver exploiting the spatial property of each multipath components such that the components with time delay less than the chip period but with different DOAs can still be resolved and optimally combined. The 2D RAKE receiver estimates the pre-processing (before matched filtering) and post-processing array correlation matrices. The appropriately constructed differences of these estimates are then used to estimate the steering vector and the noise covariance matrix, which are then used to get the beamforming weight vectors.

The system capacity improvement obtained by using the 2D RAKE receiver is analyzed in [49][50]. However, the 2D RAKE receiver tries to steer one beam to each multipath component of each user, which will result in a high computational complexity and highly complicated hardware construction.

4.3.5 Decision Directed Algorithms

In a communication system with low bit error rate (BER), the demodulated signal can be used as the reference signal in the adaptation of the weight vector. Such a method of adaptation is said to be decision directed (DD), because the beamformer attempts to adapt by employing its own decisions. The DD algorithm was first used in the adaptive equalization problem [51]. If the BER is small, the decisions made by the receiver are correct enough for the estimate of the error signal to be accurate most of the time. This means that, in general, the adaptive beamformer is able to improve the weight vector by virtue of the correlation procedure built into its feedback control loop. The improved weight vector will, in turn, result in a lower BER and therefore more accurate estimates of the error signal for adaptation, and so on. However, it is also possible for the reverse effect to occur, in which case the weight vector loses acquisition of the channel.

For a BPSK signal, the adaptation equations for steepest-descent DD (SD-DD) algorithm can be obtained by replacing the complex limited signal $y(k) / \hat{y}(k)$ in

the SD-CMA by the decision term $\text{sgn}[\text{Re}(\mathbf{y}(k))]$, where $\text{sgn}(\cdot)$ denotes the signum function. Therefore, the SDDD algorithm can be described by the following equations

$$Y(k) = W^H X(k) \quad (4.65)$$

$$e(k) = y(k) - \text{sgn}[\text{Re}(y(k))] \quad (4.66)$$

$$W(k+1) = W(k) + \mu X(k)e^*(k) \quad (4.67)$$

For a BPSK signal, this algorithm is exactly the same as the SD-DD algorithm. The SD-DD algorithm can also be applied to the multitarget problem. The multitarget steepest-descent decision directed (MT-SDDD) algorithm will be discussed in detail in Chapter 5.

Chapter 5

Adaptive Beamforming Algorithms Used in Project

5.1 Introduction

In Chapter 4, we give a thorough survey of adaptive beamforming algorithms. In this chapter, we will describe the algorithms used in this research in detail. Our research will focus on the blind adaptive beamformer used in the base station in a CDMA system. Since in a CDMA system, multiple users share a radio channel, our adaptive beamforming algorithms should have the ability to separate and extract each user's signal blindly and simultaneously, in other words, the algorithms should be multitarget-type blind algorithms. In this chapter, we will present four algorithms

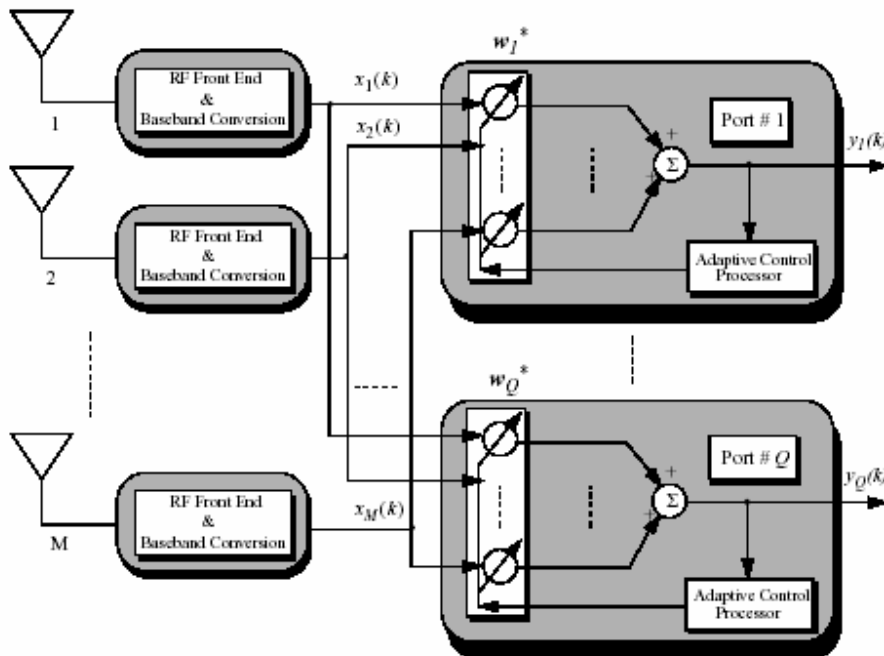
- ? Multitarget least-squares constant modulus algorithm (*MT-LSCMA*)
- ? Multitarget decision-directed (*MT-DD*) algorithm
- ? Least-squares despread respread multitarget array (*LS-DRMTA*)
- ? Least-squares despread respread multitarget constant modulus algorithm (*LS-DRMTCMA*)

The LS-DRMTA and LS-DRMTCMA are two algorithms developed and to be compared with others.

5.2 Multitarget Beamformer

In a CDMA mobile communication system, multiple users occupy the same frequency band. The beamformer in the base station attempts to form a beam directed to each user such that for one desired user, the interference from other directions is reduced. For a system with p users, the beamformer will generate p sets of complex weights, i. e., p weight vectors. These p weight vectors corresponding to p different beampatterns are used to linearly combine the input data vector of the array to generate p outputs in different output ports. Figure 5.1 shows the structure of a Multitarget adaptive beamformer with M antenna elements and Q output ports. In Figure 5.1, $y_1(k), \dots, y_Q(k)$ are the outputs of port 1, \dots, Q , respectively, and w_1, \dots, w_Q are the weight vectors of port 1, \dots, Q , respectively. For some algorithms, the number of output ports, Q , must be less than or equal to the number number of antenna elements, M , but for other algorithms such as LS-DRMTA and LS-DRMTCMA it can be greater than the number of antenna elements and equal to the number of

users. From Figure 5.1, we see that the multitarget beamformer can be viewed as a multi-input multi-output system. There are two problems that the multitarget blind adaptive beamformer needs to solve. The first one is how to generate different weight vectors for each port. For a multitarget non-blind adaptive beamformer, this problem can be solved very easily by using different training signals in different ports. For a multitarget blind adaptive beamformer, however, if all the signals have the same properties, for example, the constant modulus property, and the same cyclic feature, which is just the case in a CDMA system, and if a property-restoration algorithm is used and no other procedure is performed during the adaptation of the weight vectors, all these weight vectors may converge to one vector with the same beampattern and only a phase shift exists between different weight vectors. So some procedures must be implemented to prevent this from happening. The second problem is how to extract each user's signal from the outputs of each port. If the number of users is less than the number of antenna elements, for some algorithms, a sorting procedure has to be performed to relate each user's signal to the correct port output. If the number of users is greater than M , for some algorithms which can generate at most only M weight vectors, i. e., M port outputs, in addition to the sorting procedure, another procedure may need to be performed to extract several users' signals from the same port output.



5.1 A Multitarget adaptive beamformer with M antenna elements and Q output Ports.

5.3 Multitarget Least-Squares Constant Modulus Algorithm

The multitarget least-squares constant modulus algorithm (MT-LSCMA) was first proposed by Agee in [27]. This algorithm contains three principle components: a soft-orthogonalized dynamic LS-CMA, a set of sorting and classification algorithms, and a fast acquisition algorithm. Figure 5.2 shows the structure of a MT-LSCMA adaptive array. In Figure 5.2, the number of output ports is equal to the number of antenna elements, and the weight vectors

W_1, \dots, W_M are initialized with a set of different vectors, for example, the column vectors of a $M \times M$ identity matrix. These vectors are then adapted independently by the dynamic LSCMA which is described by equation (4.56), (4.57), and (4.58). However, since the LS-CMA only utilizes the a priori information that the original signals have a constant envelope, for a CDMA system with all the transmitted signals having the constant modulus property, if there is no procedure implemented, all the weight vectors in different ports may converge to the same beampattern. To avoid this situation, a Gram-schmidt orthogonalization (GSO) procedure is performed as shown in Figure 5.2.

5.3.1 Gram-Schmidt Orthogonalization

The GSO is used to prevent different weight vectors from converging to one with the same beampattern. If two weight vectors intend to converge to one with the same beampattern, the absolute value of their correlation coefficient will become larger (greater than 0.5 and close to 1). The correlation coefficient between two weight vectors w_i and w_j is defined by

$$\rho_{ij} = \frac{W_i^H W_j}{\|W_i\| \|W_j\|} \quad (5.1)$$

In MT-LSCMA, a threshold ρ_{tr} is set for all the correlation coefficients. After several iterations using the LS-CMA, the correlation coefficients ρ_{ij} $i = 2, \dots, M$; $j = 1, \dots, i-1$. are calculated using equation (5.1).

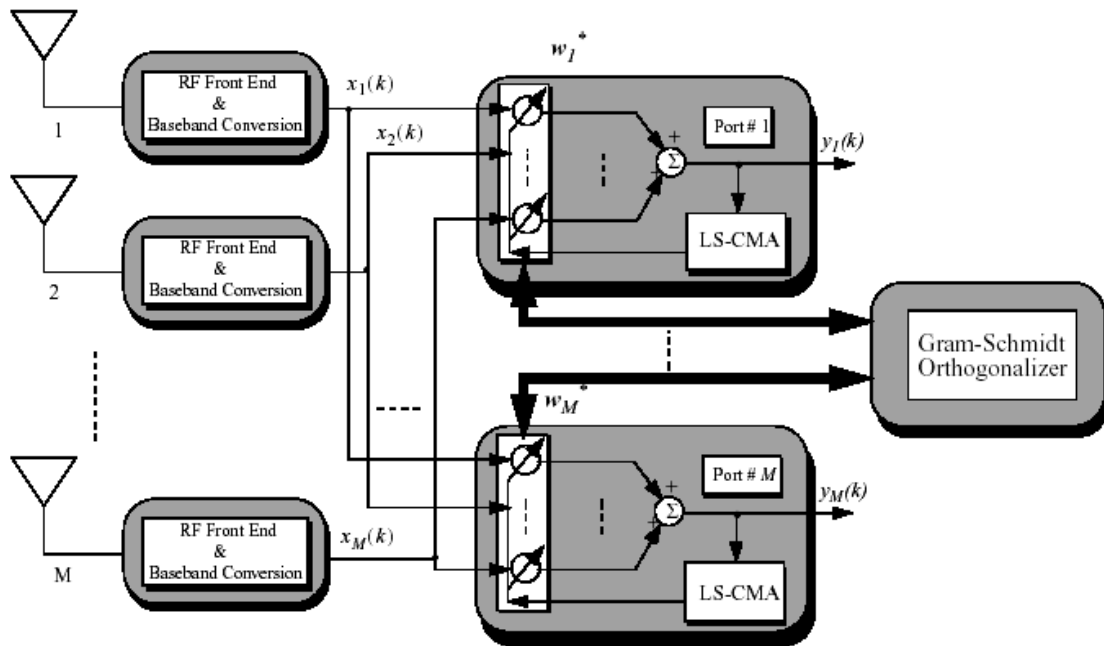


Figure 5.2: Illustration of a multitarget LS-CMA adaptive array.

The orthonormal basis \hat{W} for the range of W is also computed by using Gram-Schmidt orthogonalization procedure, where

$$W = [W_1, W_2, \dots, W_M] \quad (5.2)$$

$$\hat{W} = [\hat{W}_1, \hat{W}_2, \dots, \hat{W}_M] \quad (5.3)$$

The absolute values of the correlation coefficients are then compared to the threshold τ . If for one i , $i = 2, 3, \dots, M$, there exists one $j < i$ such that $|\rho_{ij}| > \tau$, w_i will be replaced by a scaled version of \hat{W}_j , $\hat{W}_j / \|w_i\|$. After the GSO, the weight vectors are again adapted by using the LS-CMA and the above procedure is repeated until the algorithm converges. The orthonormal basis \hat{W} for the range of W can be obtained by using the Gram-Schmidt orthogonalization procedure that is described as follows:

1. Given $W = \{W_1, W_2, \dots, W_M\}$, $i = 1$

$$2. \hat{W}_1 = \frac{W_1}{\|W_1\|}$$

3. $i = i + 1$

$$\rho_{ij} = \frac{W_i^H \hat{W}_j}{\|W_i\| \|\hat{W}_j\|}, \quad j = 1, 2, \dots, M, i = 1 \quad (5.4)$$

$$W_i = W_i - \sum_{j=1}^{i-1} \rho_{ij} \hat{W}_j \quad (5.5)$$

$$\hat{W}_i = \frac{\tilde{W}_i}{\|\tilde{W}_i\|} \quad (5.6)$$

4. Repeat step 3 until $i = M$.

The choice of the threshold ρ_{rr} will affect the convergence speed of the MT-LSCMA. If ρ_{rr} is too small, two weight vectors that will not converge to one having the same beampattern may be mis-identified as those intending to, and one of them will be replaced by the scaled column vector of \hat{W} . This new weight vector must be adapted for another several iterations to converge but may be replaced again by the GSO output vector if the threshold is too low. Apparently, this will decrease the convergence speed of the algorithm or even cause the algorithm to diverge. On the other hand, if ρ_{rr} is too large, two weight vectors that intend to converge to one having the same beampattern may not be identified, and the algorithm may fail to form different beams directed to different users. In the simulations performed in this research, 0.7 is found to be a good value of ρ_{rr} through experiments.

There is no need to perform the GSO once per iteration since the weight vectors adapted after only one iteration are not well converged, and the correlation coefficients then calculated do not well represent the potential convergence of the

weight vectors. Instead, the GSO can be performed once per several iterations (i. e, five iterations) to reduce the computational complexity of the algorithm.

5.3.2 Phase Ambiguity

Since MT-LSCMA uses the LS-CMA to adapt the weight vector for each port, and as shown in section 4.3.2, the CMA-type algorithms suffer the phase ambiguity problem, the phase of the signal at each output port is indeterminate. This problem can be solved by three methods. The first method is to use the differential phase-shift keying (DPSK) modulation in the CDMA system. Since in the DPSK modulation, it is the phase difference between the current symbol and the previous symbol that determines the output data, a phase rotation does not affect the demodulated data. The second method is to send a pilot signal from the mobile to the base station and use the received pilot signal to obtain the phase rotation information. This information can then be used to compensate the phase ambiguity in the output port. The last method is to use the phase-constraint technique [29]. That is, add a phase constraint to each weight vector such that the first element of each weight vector is a real number. For a weight vector \mathbf{w}_i after convergence, the new weight vector \tilde{W}_i generated by using phase-constraint is given by

$$\tilde{W}_i = W_i \exp\{-j \arg(w_{1i})\} \quad (5.7)$$

where $\arg[\]$ denotes the phase function and w_{1i} is the first element of the weight vector \mathbf{w}_i .

5.3.3 Sorting Procedure

In MT-LSCMA, after the algorithm converges, a sorting procedure must be performed to relate the port outputs to each user's signal. In a CDMA system, the pseudo-noise (PN) sequence assigned to each user can be utilized in the sorting procedure. For a CDMA system with p users, the complex envelope of the signal transmitted by the i th user can be expressed as

$$s_i(t) = \sqrt{2p_i} b_i(t) c_i(t) \exp\{j\theta_i\}, \quad i = 1, 2, \dots, p, \quad (5.8)$$

where $p_i(t)$, $b_i(t)$, $c_i(t)$ and $\theta_i(t)$, are the power, the data signal, the spreading signal (PN sequence), and the random phase of the i th user signal, respectively. The data signal

$$b_i(t) = \sum_{n=0}^{N_b-1} b_{in} P_{T_b}(t - nT_b) \quad (5.9)$$

where $b_{in} \in \{-1, 1\}$ is the n th data bit of i th user, and P_{T_b} is a unit rectangular pulse of duration T_b . T_b is the bit period of the CDMA signal. The spreading signal $c_i(t)$ is given by

$$c_i(t) = \sum_{m=0}^{N_c-1} c_{im} P_{T_c}(t - mT_c) \quad (5.10)$$

where $c_{im} \in \{-1, 1\}$ is the m th chip of i th user and P_{T_c} is a unit rectangular pulse of duration T_c . T_c is the chip period of the CDMA signal. The ratio of the bit period to the chip period is called the processing gain and is defined as

$$N_c = \frac{T_b}{T_c} \quad (5.11)$$

Usually, systems are designed to have high processing gain where

$$N_c \gg 1 \quad (5.12)$$

or equivalently,

$$T_b \gg T_c \quad (5.13)$$

The PN codes are also designed to have low cross-correlation and narrow autocorrelation function. Now, let's assume the number of users in the system is less than or equal to the number of the output ports in the beamformer. Also, the multipath and the interference of one desired user are rejected due to the spatial filtering. Let's further assume we have perfect power control in the system, the output of port i in the beamformer is therefore a delayed, scaled, and phase-rotated version of $S_j(t)$ which is corrupted by noise and is given by

$$y_i(t) = \alpha_i \sqrt{2P_j} b_j(t - \tau_j) \exp\{j(\omega_j t - \phi_i)\} + n_i(t) \quad (5.14)$$

where α_i is the scaled factor for user i such that $\alpha_i^2 P_i = \alpha_j^2 P_j$ for $i = j$, τ_j is the time delay of the j th user, ϕ_i is the phase shift in port i due to the phase rotation in LS-CMA, and $n_i(t)$ is the AWGN noise in port i . In equation (5.14), i may or may not be equal to j , and the sorting procedure is used to relate the user index i to the port index j . Let's assume the time delay τ_j is estimated perfectly, and the phase $\phi_i = \phi_j$ is also estimated correctly by using the method discussed in section 5.3.2, the sorting procedure can be performed as shown in Figure 5.3. In Figure

5.3, $y(t)$ is a vector containing the port outputs of the beamformer and is given by

$$Y(t) = [y_1(t), y_2(t), \dots, y_M(t)]^T \quad (5.15)$$

The M output signals in $y(t)$ are first multiplied by the delayed versions of p users' PN codes. So for each arm in Figure 5.3, there are M multiplying outputs corresponding to one user's PN code. These outputs are then integrated over one bit period, and the integration outputs are sampled and their absolute values are compared with each other. Actually, the output of the j th integrator in arm i is the correlation value between $y_j(t)$ and the delayed PN code of user i , $c_j(t - \tau_j)$. Since the PN codes are designed to have low cross correlation, only the output port containing the i th user's signal will have a peak at the integration output. In arm i , the output port with the highest absolute value of the integrate output will be identified as one containing the i th user's signal, and the index of this output port, j_i , will be stored as the output of the sorting procedure.

If the number of users in the system is greater than the number of output ports, i. e., the number of antenna elements for this specific algorithm, one output port may contain several users' signals. Using the sorting procedure shown in Figure 5.3, we may relate one output port with several users. In other words, there exist i_1 and i_2 , where $i_1 \neq i_2$, such that $j_{i1} = j_{i2}$. In this case, the output of port j_{i1} or (j_{i2}) will be fed into both user i_1 and user i_2 's receivers to extract their signals. If the above procedure is performed by using a digital system, the analog signals will be replaced by their discrete time samples and the integration will be substituted by the accumulated sum.

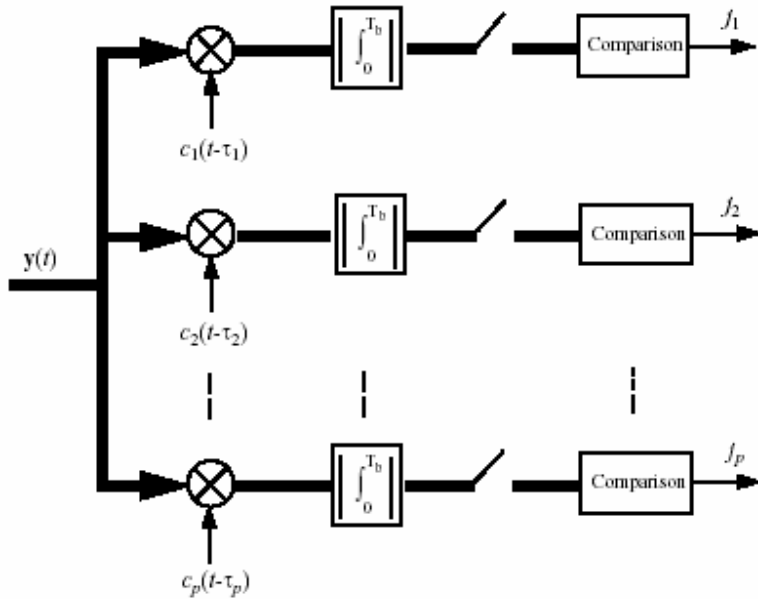


Figure 5.3: Illustration of sorting procedure in MT-LSCMA for a CDMA system.

The MT-LSCMA can be summarized as follows:

1. Initialize the M weight vectors w_1, \dots, w_M as the column vectors of a $M \times M$ identity matrix.
2. Adapt each weight vector independently using the LS-CMA described in section 4.3.2.
3. After several iterations in LS-CMA, perform the GSO on the resulting weight vectors as described in section 5.3.1.
4. Repeat step 2 and 3 until the algorithm converges.
5. Add the phase constraint or perform phase rotation compensation on the resulting weight vectors as described in section 5.3.2.
6. Perform the sorting procedure described in section 5.3.3 to relate the output ports to each user's signal.

5.4 Multitarget **Decision-Directed Algorithm**

By replacing LS-CMA in the MT-LSCMA with decision-directed (DD) algorithm described in section 4.3.4, we obtain the MT-DD algorithm. The DD algorithm can be performed with either the steepest-descent (SD) method or the least-squares (LS) method. If the DD algorithm in the MT-DD is performed with steepest-descent method, we will call this multitarget-type algorithm the multitarget steepest-descent decision-directed (MT-SDDD) algorithm, and if the DD algorithm in the MT-DD is performed with least-squares method, we will call it the multitarget least-squares decision-directed (MT-LSDD) algorithm. The SD-DD algorithm is described by equation (4.65), (4.66), and (4.67). The LS-DD algorithm can be derived in the same manner as that of the LS-CMA algorithm and can be described as

$$Y(l) = W^H(l)X(l) \quad (5.16)$$

$$= [y(1 \rightarrow lk), y(2 \rightarrow lk), \dots, y((l \rightarrow 1)k)]^T,$$

$$r(l) = [\text{sgn}\{\text{Re}(y(1 \rightarrow lk))\}, \dots, \text{sgn}\{\text{Re}(y((l \rightarrow 1)k))\}]^T \quad (5.17)$$

$$W(l \rightarrow 1) = [X(l)X^H(l)]^{-1} X(l)r^*(l) \quad (5.18)$$

where l is the iteration number, $X(l)$ is defined in equation (4.55), and K is the number of samples in one data block.

Unlike the MT-LSCMA which constrains the signal constellations on the unit circle, the MT-DD algorithm constrains the signal constellations at either +1 or -1. Since the signal of each user has a random phase, this constraint will also cause a phase ambiguity. The methods introduced in section 5.3.2 can be used to compensate this effect.

The MT-SDDD algorithm can be described as follows:

1. Initialize the M weight vectors w_1, \dots, w_M as the column vectors of a $M \times M$ identity matrix.
2. Adapt each weight vector independently using the SD-DD described in section 4.3.4.

3. After a number of iterations in SD-DD, perform the GSO on the resulting weight vectors as described in section 5.3.1.
4. Repeat step 2 and 3 until the algorithm converges.
5. Add the phase constraint or perform phase rotation compensation on the resulting weight vectors as described in section 5.3.2.
6. Perform the sorting procedure described in section 4.3.3 to relate the output ports to each user's signal.

In step 2, the number of iterations needed before the GSO depends on the step size ϵ in the SD-DD algorithm. In our simulation, ϵ is set to 0.001 and the number of iterations before the GSO is set to 1000. In MT-LSDD, the number of iterations needed will be around 5 as that in MT-LSCMA, but for both the MT-LSDD and MT-LSCMA, the matrix inversion has to be calculated. So there is a trade-off between the convergence speed and the computational complexity.

5.5 Least-Squares Despread Respread Multitarget Array

The multitarget adaptive algorithms discussed to date do not utilize any information of the spreading signal of each user in the CDMA system. However, in a CDMA system, it is these spreading signals that distinguish different users occupying the same frequency band. Therefore it will be very useful if the information of these spreading signals can be utilized in the multitarget adaptive algorithm. In this section and section 5.6 we will introduce two algorithms that utilize the information of the spreading signals. The algorithm discussed in this section is called least-squares despread respread multitarget array (LS-DRMTA).

5.5.1 Derivation of LS-DRMTA

In the base station of a CDMA system, the spreading signals of all the users are known beforehand. In the conventional receiver, to detect the i th user's data bits, the received signal is correlated with the time-delayed spreading signal of the i th user, $c_i(t - \tau_i)$, and the correlation output is sent to the detector, which makes a decision based on the correlation output. There exist many techniques to detect the time delay, τ_i , for the i th user, and we are not going to cover this topic in this research. So from now on, we assume that the time delay for each user is detected perfectly unless specified otherwise. If the n th data bit of the i th user is detected correctly by the detector, i. e., $\hat{b}_{in} = b_{in}$, where \hat{b}_{in} is the detector output, the waveform of the i th user's transmitted signal during time period

$[(n - 1)T_b, nT_b)$ can be obtained by resreading the detected data bit b_{in} with the PN sequence of the i th user, $c_i(t)$. This resread signal can then be used in the beamformer to adapt the weight vector for user i . The adaptive algorithm that uses this despread-and-resread technique is referred to as least-squares despread resread multitarget array (LS-DRMTA) in this research. Figure 5.4 shows the structure of a beamformer using the LS-DRMTA and Figure 5.5 shows the block diagram of the LS-DRMTA for user i . Note that in Figure 5.4, the number of beamformer output ports is now equal to the number of users in the system.

In Figure 5.5, $r_i(t)$ is a time-delayed version of the resread signal for user i and is given by

$$r_i(t) = b_{in} c_i(t - \tau_i), \quad (n - 1)T_b \leq t < nT_b \quad (5.19)$$

In a conventional CDMA system, the PN sequence is repeated every bit period, therefore both $c_i(t)$ and $r_i(t)$ have a time period T_b .

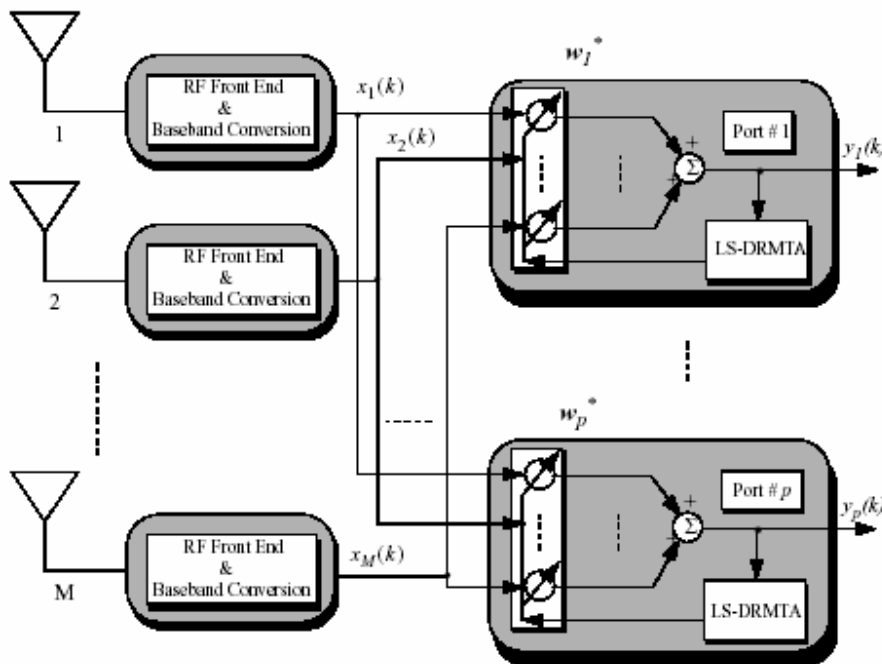


Figure 5.4: Structure of a beamformer using LS-DRMTA.

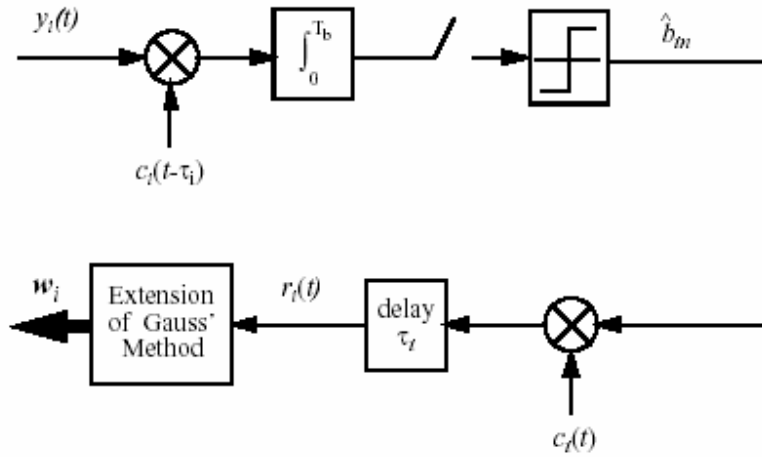


Figure 5.5: LS-DRMTA block diagram for user i.

Let $y_i(k)$ and $r_i(k)$ denote the k th sample of $y_i(t)$ and $r_i(t)$, respectively, in a digital system, the LS-DRMTA tries to adapt the weight vector w_i to minimize the cost function

$$F(w_i) = \sum_{k=1}^K |y_i(k) - r_i(k)|^2$$

$$= \sum_{k=1}^K |w_i^H X(k) - r_i(k)|^2, \quad (5.20)$$

where K is the data block size and is set to be equal to the number of samples in one bit period in LS-DRMTA. So if the signal is sampled with a sampling rate $R_s = N_s R_c$, where R_c is the chip rate of the CDMA signal, and N_s is an integer greater than two, the block size K will be equal to $N_c N_s$, where N_c is the processing gain. Using the extension of Gauss' method described in section 4.3.2 and comparing equation (5.20) with (4.36), we have

$$g_K(w_i) = |y_i(k) - r_i(k)|$$

$$= |w_i^H x(k) - r_i(k)|. \quad (5.21)$$

Substituting equation (5.21) into (4.37), we obtain

$$g(w_i) = \begin{bmatrix} |y_i(1) - r_i(1)| \\ |y_i(2) - r_i(2)| \\ \vdots \\ |y_i(k) - r_i(k)| \end{bmatrix} \quad (5.22)$$

The gradient vector of $g_k(w_i)$ is given by

$$\frac{\partial g_k(w_i)}{\partial w_i} = 2 \frac{\partial g_k(w_i)}{\partial w_i^*} = 2 x(k) \frac{y_i(k) - r_i(k)^*}{|y_i(k) - r_i(k)|}, \quad (5.23)$$

Let

$$v_i(k) = y_i(k) - r_i(k)^* \quad (5.24)$$

equation (5.23) can be expressed as

$$\frac{\partial g_k(w_i)}{\partial w_i} = 2 x(k) \frac{v_i^*(k)}{|v_i(k)|}. \quad (5.25)$$

Substituting equation (5.25) into (4.39), $D(w_i)$ can be expressed as

$$D(w_i) = (g_1(w_i), g_2(w_i), \dots, g_k(w_i))$$

$$X = \left(x(1) \frac{v_i^*(1)}{|v_i(1)|}, x(2) \frac{v_i^*(2)}{|v_i(2)|}, \dots, x(k) \frac{v_i^*(k)}{|v_i(k)|} \right)$$

$$X V_{iCM}, \tag{5.26}$$

where

$$X = (x(1), x(2), \dots, x(k)) \tag{5.27}$$

and

$$V_{iCM} = \begin{pmatrix} \frac{v_i^*(1)}{|v_i(1)|} & 0 & 0 & \dots & 0 \\ 0 & \frac{v_i^*(2)}{|v_i(2)|} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dots & \frac{v_i^*(k)}{|v_i(k)|} \end{pmatrix} \tag{5.28}$$

Using equation (5.26) and (5.22), we have

$$D(w_i)D_H(w_i) = XV_{iCM}V_{iCM}^H X_H$$

$$= XX^H, \tag{5.29}$$

and

$$d(w_i)g(w_i) = XV_{iCM} \begin{bmatrix} |v_i(1)| \\ |v_i(2)| \\ \vdots \\ |v_i(k)| \end{bmatrix}$$

$$= X \begin{bmatrix} v_i^*(1) \\ v_i^*(2) \\ \vdots \\ v_i^*(k) \end{bmatrix}$$

$$= XV_i^*, \tag{5.30}$$

$$= X(y_i \ r_i)^* \tag{5.31}$$

where

$$v_i = [v_i(1), v_i(2), \dots, v_i(k)]^T \tag{5.32}$$

$$y_i = [y_i(1), y_i(2), \dots, y_i(k)]^T \tag{5.33}$$

$$r_i = [r_i(1), r_i(2), \dots, r_i(k)]^T \tag{5.34}$$

The vector y_i is the output data vector for user i and r_i is the estimate of the signal waveform of user i over one bit period. Substituting equation (5.29) and (5.31) into equation (4.42), we obtain

$$\begin{aligned}
 W_i(l+1) &= w_i(l) + \eta X X^H \{X(y_i(l) - r_i(l))^*\} \\
 &= w_i(l) + \eta X X^H \{X X^H w_i(l) + \eta X X^H \{X r_i(l)\}^*\} \\
 &= \eta X X^H \{X r_i^*(l)\},
 \end{aligned} \tag{5.35}$$

where $y_i(l)$ and $r_i(l)$ are the output data vector and estimate of signal waveform of user i over one bit period corresponding to the weight vector w_i in the l th iteration, respectively.

Similar to the dynamic LS-CMA, LS-DRMTA can adapt the weight vectors using different input data blocks in each iteration. Let

$$X(l) = [x(1-lk), x(2-lk), \dots, x((l+1)k)]^T, \quad l = 0, 1, \dots, L, \tag{5.36}$$

where L is the number of iterations required for the algorithm to converge, and K is the number of data samples per bit (NcNs) if the data samples over one bit period are all used for the adaptation. In Figure 5.4, the LS-DRMTA for the i th user can be described by the following equations:

$$\begin{aligned}
 y_i(l) &= w_i^H(l) X(l) \\
 &= [y_i(1-lk), y_i(2-lk), \dots, y_i((l+1)k)]^T,
 \end{aligned} \tag{5.37}$$

$$b_{ii} = \text{sgn} \left\{ \text{Re} \sum_{k=1-lk}^{(l+1)k} y_i(k) c_i(k - k_{\tau_i}) \right\} \tag{5.38}$$

$$r_i(l) = \sum_{k=1-lk}^{k_{\tau_i}} b_{ii} c_i(k - k_{\tau_i}), \dots, \sum_{k=(l+1)k}^{k_{\tau_i}} b_{ii} c_i(k - k_{\tau_i})^T, \tag{5.39}$$

$$W_i(l+1) = X(l) X^H(l) \{X(l) r_i^*(l)\} \tag{5.40}$$

where $c_i(k)$ is the k th sample of the spreading signal of user i , $k\tau_i$ is the number of samples corresponding to τ_i , the delay of user i , and \hat{b}_i^l is the estimate of l th bit for user i . The accumulated sum in equation (5.38) is equivalent to integration in the continuous time domain.

We can summarize the LS-DRMTA as follows

1. Initialize the p weight vectors W_1, W_2, \dots, W_p as p identical $M \times 1$ column vectors with the first element equal to 1 and the other elements equal to 0.
2. Calculate the array output vector using equation (5.37).
3. Despread the i th user's signal and estimate the n th data bit using equation (5.38).
4. Respread the estimate data bit with the PN code of user i to get a estimate of the signal waveform of user i over time period $[(n - 1)T_b, nT_b)$ using equation (5.39).
5. Adapt the weight vector W_{i1} of user i using equation (5.40).
6. Repeat step 2 to 5 until the algorithm converges.

In equation (5.39), when $l = 0$, there may be some time indices less than zero, but since the PN sequence has a time period of T_b , or equivalently, a period of K in the discrete time domain, the index k , where $k < 0$, can be replaced by the index $k + K$. In LS-DRMTA, if the data bit is not estimated correctly at the beginning of the algorithm, from equation (5.39) and (5.40) we see that the resulting weight vector may have a phase shift of π , but this does not affect the resulting beampattern. The beampattern will still have a higher gain in the DOA of the desired signal and the interference from other directions will be rejected. Therefore if the DPSK modulation is used in the system, the data bit estimation error at the beginning of the algorithm does not affect the demodulated data bit.

5.5.2 Advantages of LS-DRMTA

Since the LS-DRMTA utilizes the information of each user's spreading signal to adapt the weight vectors, it has several advantages over the two algorithms, MT-LSCMA, and MT-SDDD, discussed in previous sections.

The first advantage of the LS-DRMTA is that there is no need to perform the GSO procedure. This can be seen by comparing Figure 5.4 with Figure 5.2. In

MT-LSCMA and MT-SDDD, the weight vectors are adapted by using the same property of the entire user's signal, and the GSO procedure is used to prevent different weight vectors from converging to one having the same beampattern. In LS-DRMTA, however, different users' weight vector are adapted to minimize a different cost function, which is a sum of the squared error between the port output and the respread signal of the desired user over one bit period. Since each different user has a different spreading signal, the weight vector of each different user is updated with a different tendency and thus will not converge to one having the same beampattern.

The second advantage of the LS-DRMTA is that there is no need to perform the sorting procedure. In MT-LSCMA and MT-SDDD, the sorting procedure is used to relate each output port to each user. In LS-DRMTA, since different weight vectors are adapted with different PN sequences, the weight vector adapted by using the i th user's spreading signal will be corresponding to the i th user. Therefore there is no need to perform the sorting procedure.

The third advantage of the LS-DRMTA is that the number of output ports of the beamformer is not limited by the number of antenna elements of the array. In MT-LSCMA and MT-SDDD, due to the GSO procedure, the number of output ports should be less than or equal to the number of antenna elements. If the number of users in the system is greater than the number of antenna elements, several users have to share one output port, and the interference cannot be reduced to a very low level. In LS-DRMTA, since there is no GSO needed, the number of output ports can be equal to the number of users even if the number of users is greater than the number of antenna elements. Since each user has its own output port, the interference can be reduced to a lower level. Furthermore, if the system is expanded, more output ports can be added to the beamformer easily by adding more weight vectors adapted with the PN sequences of the new users. As shown in Figure 5.4, the new output ports use the same array input data vector, so no new RF front end and baseband conversion are needed, which will reduce the cost of the adaptive antenna system.

The fourth advantage of the LS-DRMTA is that the computational complexity of the LS-DRMTA is lower than that of the MT-LSCMA and MT-SDDD. In MT-LSCMA and MT-SDDD, both the GSO procedure and the sorting procedure have to be performed to extract different user's signal. From the previous discussion we see that these two procedures are very computationally intensive. LS-DRMTA, on the other hand, eliminates these two procedures and therefore requires lower computational complexity.

The last advantage of the LS-DRMTA is that it can work under the condition where the SINR is low and both the MT-LSCMA and MT-SDDD cannot work. In LS-DRMTA, the PN sequence is used in the adaptation of the weight vector, and these PN sequence may spread the interference over a large frequency band and despread the desired signal. Also the noise is averaged over one bit period in LS-DRMTA and their effect on the desired signal is reduced. So the LS-DRMTA can work in a low SINR situation.

5.6 Least-Squares Despread Respread Multitarget Constant Modulus Algorithm.

In LS-DRMTA, we utilize the spreading signal of each user in a CDMA system to adapt the weight vectors of the beamformer. In MT-LSCMA, on the other hand, we utilize the constant modulus property of the transmitted signal to update the weight vectors. One intuitive thought is to combine the spreading signal and the constant modulus property of the transmitted signal to adapt the weight vector. The algorithm using this kind of combination in the adaptation of the weight vector is referred to as least-squares despread respread multitarget constant modulus algorithm (LS-DRMTCMA) in this research. Figure 5.6 shows the structure of a beamformer using the LS-DRMTCMA and Figure 5.7 shows the block diagram of the LS-DRMTCMA for user i .

5.6.1 Derivation of LS-DRMTCMA

The derivation of the LS-DRMTCMA is very similar to that of the LS-DRMTA. In LS-DRMTA, the cost function that the algorithm wants to minimize is given in equation (5.41)

$$F(w_i) = \sum_{k=1}^K |y_i(k) - r_i(k)|^2$$

$$= \sum_{k=1}^K |W_i^H X(k) - r_i(k)|^2, \quad (5.41)$$

where

$$r_i(k) = \sum_{m=1}^M c_m(k - k_{\tau_i}), \quad (n-1)K + 1 \leq k \leq nK, \quad (5.42)$$

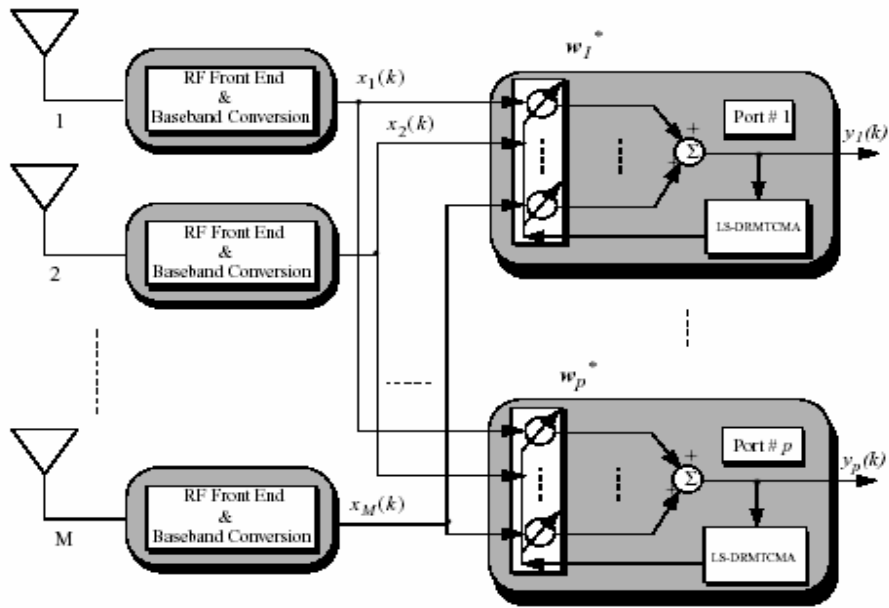


Figure 5.6: Structure of a beamformer using LS-DRMTCMA.

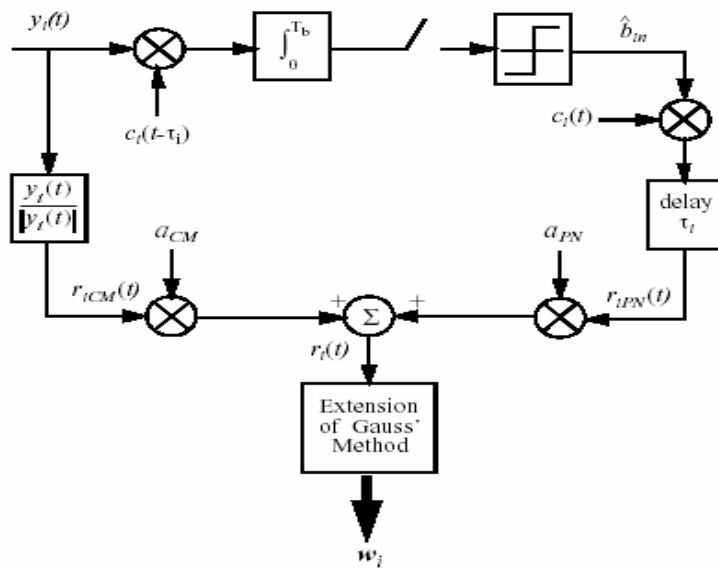


Figure 5.7: LS-DRMTCMA block diagram for user i .

In LS-DRMTCMA, the cost function that the algorithm wants to minimize has the same form as that shown in equation (5.41). However, as illustrated in Figure 5.7, $r_i(k)$ now becomes the sum of the weighted respread signal and the weighted complex-limited output.

$$r_i(k) = a_{PN} r_{iPN}(k) + a_{CM} r_{iCM}(k) \quad (5.43)$$

where $r_{iPN}(k)$ is the respread signal of user i and is given in equation (5.42), r_{iCM} is the complex-limited output of user i and can be expressed as

$$r_{iCM}(k) = \frac{y_i(k)}{|y_i(k)|}, \quad (5.44)$$

and a_{PN} and a_{CM} are the real positive weight coefficients for the respread signal and the complex-limited output of user i , respectively. The coefficients a_{PN} and a_{CM} should satisfy the condition:

$$a_{PN} + a_{CM} = 1, \quad a_{PN}, a_{CM} > 0 \quad (5.45)$$

Using the extension of Gauss' method, and repeating the procedure performed in section 5.5.1, we obtain the following equations for LS-DRMTCMA:

$$y_i(l) = \sum_{k=1}^L w_i^H(l) X(l) \quad (5.46)$$

$$b_{il} = \text{sgn} \left\{ \text{Re} \left[\sum_{k=1}^{(l-1)k} y_i(k) c_i(k - k_{\gamma_i}) \right] \right\} \quad (5.47)$$

$$r_i(l) = \sum_{k=1}^{(l-1)k - k_{\gamma_i}} b_{il} c_i(k - k_{\gamma_i}) + \sum_{k=1}^{(l-1)k - k_{\gamma_i}} b_{il} c_i((l-1)k - k_{\gamma_i}), \quad (5.48)$$

$$r_{iCM} = \left[\frac{y(1-lk)}{|y(1-lk)|}, \frac{y(2-lk)}{|y(2-lk)|}, \dots, \frac{y((1-l)k)}{|y((1-l)k)|} \right]^T \quad (5.49)$$

$$r_i(l) = a_{PN} r_{iPN}(l) + a_{CM} r_{iCM}(l) \quad (5.50)$$

$$W_i(l+1) = [X(l)X^H(l)]^{-1} X(l) r_i^*(l) \quad (5.51)$$

From the above equations we see that if a_{CM} is set to zero, the LS-DRMTCMA becomes the LS-DRMTA, therefore the LS-DRMTA can be viewed as a special case of the LS-DRMTCMA.

Also we see that if a_{PN} is set to zero and the GSO procedure is performed during the adaptation, the algorithm becomes MT-LSCMA. The choice of a_{PN} and a_{CM} can affect the resulting beampattern and thus the performance of the system. We can summarize the LS-DRMTCMA as follows

1. Initialize the p weight vectors W_1, W_2, \dots, W_p as p identical $M \times 1$ column vectors with the first element equal to 1 and the other elements equal to 0.
2. Calculate the array output vector using equation (5.46).
3. Despread the i th user's signal and estimate the n th data bit using equation (5.47).
4. Respread the estimate data bit with the PN code of user i to get a estimate of the signal waveform of user i over time period $[(n-1)T_b, nT_b)$ using equation (5.48).
5. Calculate the complex-limited output vector of user i using equation (5.49).
6. Calculate the reference signal vector for user i by summing up the weighted respread signal vector and the weighted complex-limited output vector using equation (5.50).
7. Adapt the weight vector W_i of user i using equation (5.51).
8. Repeat step 2 to 7 until the algorithm converges.

5.6.2 Advantages of LS-DRMTCMA

Since the LS-DRMTCMA utilize both the PN sequence and the constant modulus property of the transmitted signal, it possesses all the advantages of LS-DRMTA and has other advantages that the LS-DRMTA does not possess. The most important advantage is that it can achieve a much lower BER than the LS-DRMTA. We will discuss this point in Chapter 6. However, since LS-DRMTCMA utilizes the complex-limited output vector of each user in the adaptation of the weight vectors, this advantage is achieved at the expense of additional complexity.

CHAPTER 6

Results Obtained and Discussion

6.1 Introduction

In Chapter 5 we present four multitarget-type adaptive beamformer algorithms for the base station in a CDMA system. In this chapter, we will compare these four algorithms under different conditions. The following two cases will be considered and we shall try to discuss with each of them:

1. AWGN channel
2. Multipath channel.

Our comparison of the algorithms will focus on the BER performance of different algorithms. The convergence characteristics of the different algorithms will also be presented.

6.2 Description of System Parameters

The system we are going to consider is a direct sequence CDMA (DS-SS) system with a processing gain, N , equal to 15. There are several reasons for choosing such a small processing gain. The first reason is that we want to reduce the simulation time and at the same time compare the performance of different adaptive algorithms in a CDMA system. The second one is that we want to focus on the enhancement of the system capacity obtained by utilizing the adaptive array, not on the enhancement obtained by using a long PN sequence for each user.

The modulation scheme used in the system is the binary phase-shift keying (BPSK), the carrier frequency f_c is equal to 2.05 GHz, and the data bit rate R_b is equal to 128 kbps. An 8-element uniform linear array with half wave-length spacing between the elements is used in the base station of the system to perform spatial filtering in the reverse link (from the mobile to the base station). The sampling rate is four times the chip rate, in other words, there are four data samples per chip. So for the LS-DRMTA and LS-DRMTCMA, the data block size is equal to 60, which is the number of samples per bit.

6.3 Results in AWGN Channel

Let's first consider one case where there are 8 users in the system transmitting CDMA signals from different directions. The DOAs of the signals are equally spaced between -70° and 90° . Figure 6.1 shows the distribution of these 8 users. We assume that there is no multipath and the radio channel only introduces the additive white Gaussian noise (AWGN). We also assume perfect power control in the base station, so all the signals impinging on the array have the same power. The input signal to noise ratio per bit (E_b/N_0) is set to 20dB. Table 6.1 shows the signal parameters of the user's entire signal.

Table 6.1 Signal Parameters of 8 Users Transmitting Signals from Different directions.

User #	Power (dbW)	DOA(deg)
1	0	-70
2	0	-47
3	0	-24
4	0	-1
5	0	22
6	0	45
7	0	68
8	0	90
$E_b / N_b = 20\text{db}$.		

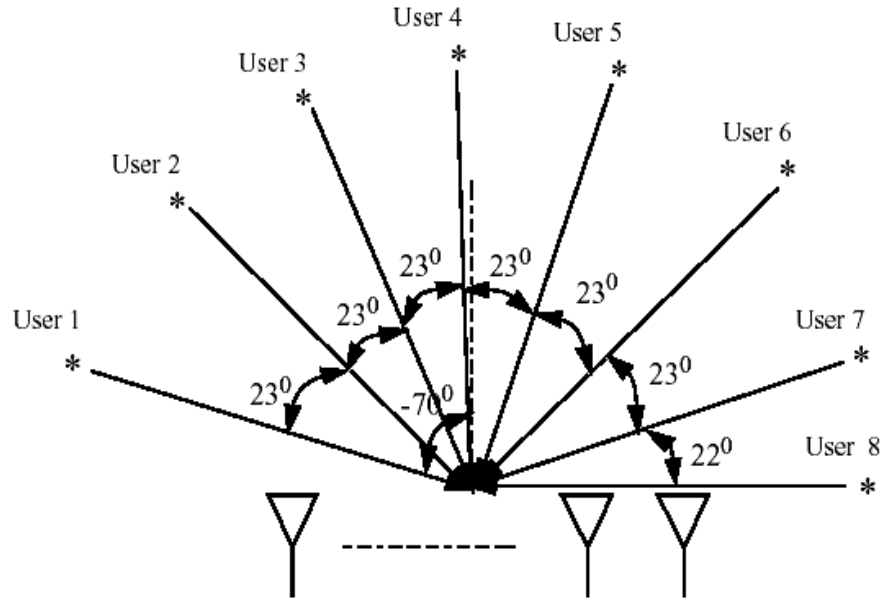


Figure 6.1: Illustration of eight users with DOAs equally spaced between -70° and 90° .

6.3.1 General Result

Figures 6.2 and 6.3 show the beam patterns of all the 8 users generated by using the LS-DRMTCMA with $a_{PN} / a_{CM} = 2$. From Figures 6.2 and 6.3 we see that except for the signals with DOAs near the endfire of the array most of the signals can be extracted by nulling out all the other interference. Due to the existence of noise, the nulls are not constructed perfectly, but the desired user could still have a gain about 20 dB over the interference as shown in Figures 6.2 and 6.3. For the signals with DOAs near the endfire of the array (e. g., signals of user 1 and user 8), since the beamwidth of the beam near the endfire is wider than that of the beam steered to other direction, two or more signals may fall into one main beam depending on the angle separation of the signals. In such a case, although the interference is not rejected completely due to the wide beamwidth of the main beam directed to the endfire, most of the interference coming from other directions is rejected, thus the overall interference level is reduced. The reason for the wider beamwidth along the end fire of the array can be explained as follows.

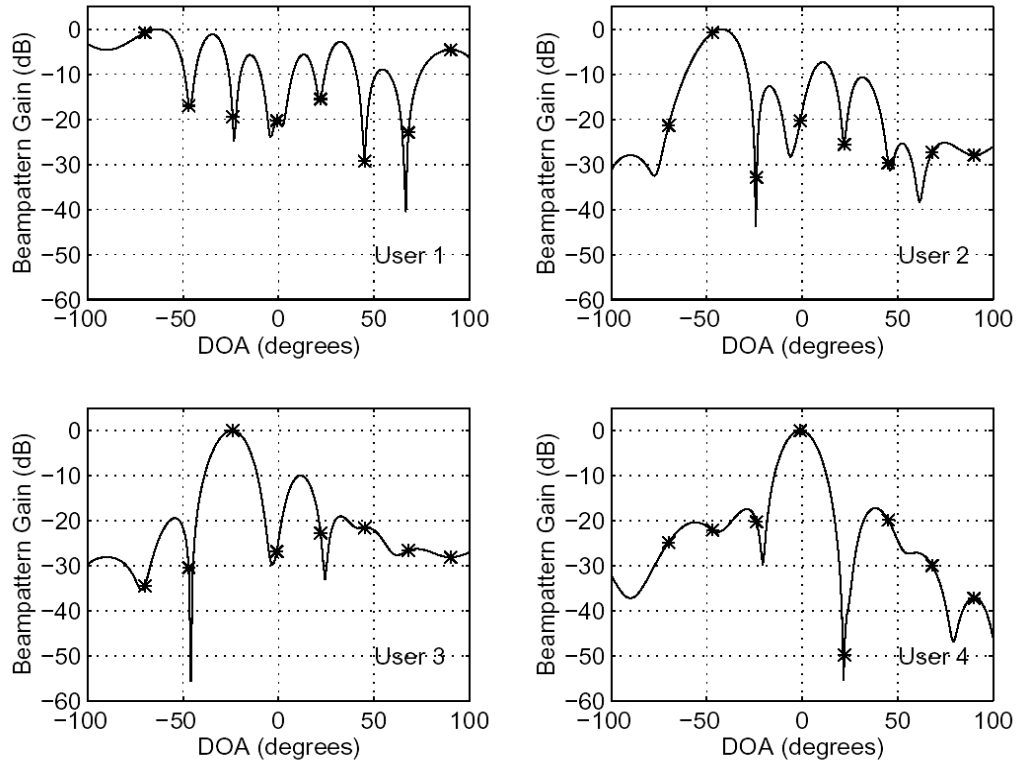


Figure 6.2: Beampatterns corresponding to different users generated by using LS-DRMTCMA. * denotes user in the system. The signal parameters are shown in Table 6.1.

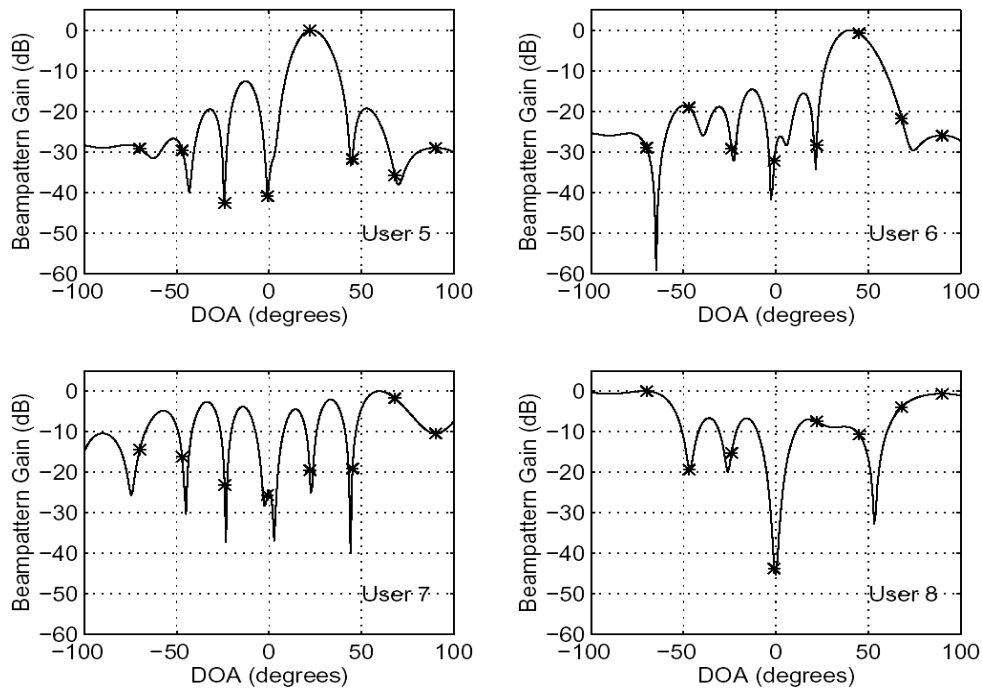


Figure 6.3: Beam patterns corresponding to different users generated by using LS-DRMTCMA .

As shown in Chapter 3, the sine of the DOA θ corresponds to the temporal frequency f of a FIR filter input, and the beam pattern of an adaptive array can be viewed as the counterpart of the magnitude response of an FIR filter. So the passband-width Δf of a FIR filter corresponds to the term $\sin \theta - \sin (\theta - \Delta \theta)$ in the array, where $\Delta \theta$ is the beamwidth of the main beam. If we want to keep $\sin \theta - \sin (\theta - \Delta \theta)$ constant for all θ (like keeping the passband-width of the FIR filter constant over all the frequency band), we have

$$\sin \theta - \sin(\theta - \Delta \theta) = \Delta f \quad (6.1)$$

The term in the left side of equation (6.1) can be expressed as

$$\sin \theta - \sin(\theta - \Delta \theta) = \Delta \theta \cos \theta' \quad (6.2)$$

where θ' is a value in $(\theta - \Delta \theta, \theta)$. Substituting equation (6.2) into (6.1) we obtain

$$\beta \approx \frac{\beta f}{\cos \theta} \quad (6.3)$$

From equation (6.3) we see that for a fixed βf , β will change with θ . For DOA near the end fire of the array, θ , and therefore β , are close to $\pi/2$, so for a fixed passband-width, β will be large since $\cos \theta$ is close to 0. On the other hand, for DOA near the broadside of the array, θ is close to 0, and β will be small since $\cos \theta$ is close to 1. Therefore the beamformer can construct a narrow beam directed to the broadside of the array and a wide beam directed to the endfire of the array.

The signal constellations of user 5 before and after the beamformer processing are shown in Figure 6.4 and Figure 6.5, respectively. In Figure 6.5, we have compensated the random phase effect by multiplying the beamformer output of user 5 by the complex conjugate of $\exp \{ j\theta_5 \}$, where θ_5 is the random phase of user 5. Comparing Figure 6.4 and Figure 6.5, we see that the interference from different DOAs is indeed rejected and the signal constellation is reconstructed. Figures 6.6, 6.7, and 6.8 show the original signal waveform, the corrupted signal waveform, and the reconstructed signal waveform of user 5 over three bit-periods, respectively. Comparing Figures 6.6, 6.7 and 6.8, we see that the signal is reconstructed in the beamformer output. Note that the signal shown in Figure 6.8 is the undespread CDMA signal which will be fed into the matched filter to make a decision on the transmitted data bit. The rejection of the interference will result in a low BER.

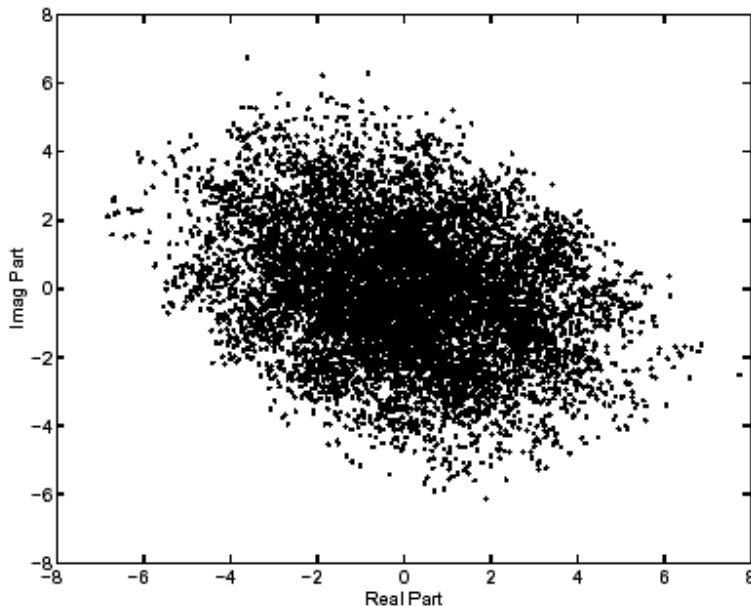


Figure 6.4: Signal constellation of user 5 before the beamformer processing. The signal parameters are shown in Table 6.1.

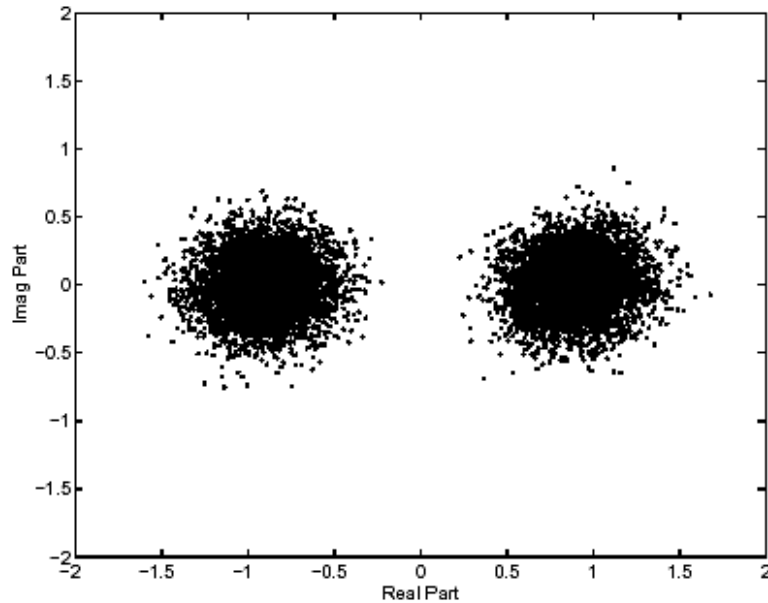


Figure 6.5: Signal constellation of user 5 after the beamformer processing. The signal parameters are shown in Table 6.1.

Figure 6.9 shows the convergence curve of the MT-LSDD in port 6 of the beamformer. The CM criterion shown in Figure 6.9 is defined in equation (5.28). From Figure 6.9 we see that due to the GSO procedure, after 1000 iterations, the weight vector in port 6 is replaced by the GSO output and the CM criterion rises sharply but then decreases as the iteration goes on. For the MT-SDDD, it will take about 8000 iterations (samples) for all the weight vectors in the 8 ports to converge. The LS-DRMTCMA, on the other hand, does not need to perform the GSO procedure and therefore can eliminate the sharp rise in the convergence curve. Figure 6.10 shows the least-squares mean squared error (LS MSE) defined in equation (5.41) vs. the iteration number in port 6 of the beamformer for the LS-DRMTCMA which is described in equations (5.46)-(5.51). From Figure 6.10 we see that the algorithm converges after 4 to 5 iterations. Since in the LS-DRMTCMA and LS-DRMTA, the data block size in each iteration is equal to the number of data samples per bit, which is equal to 60 in the simulation, the number of samples required for the LS-DRMTCMA to converge is 240 to 300, which is far less than the number of samples required for the MT-SDDD to converge.

6.3.2 BER Performance for AWGN Channel

In this section, we will compare the BER performance of different algorithms under different conditions. We will consider two E_b / N_0 cases, $E_b / N_0 = 8$ dB, and $E_b / N_0 = 4$ dB. For each E_b / N_0 case, we will also consider two different DOA distribution cases. One case is the non-crowded case with all the DOAs of the signals equally spaced between -70° and 90° . Another one is the crowded case with all the DOAs equally spaced between 0° to 90°

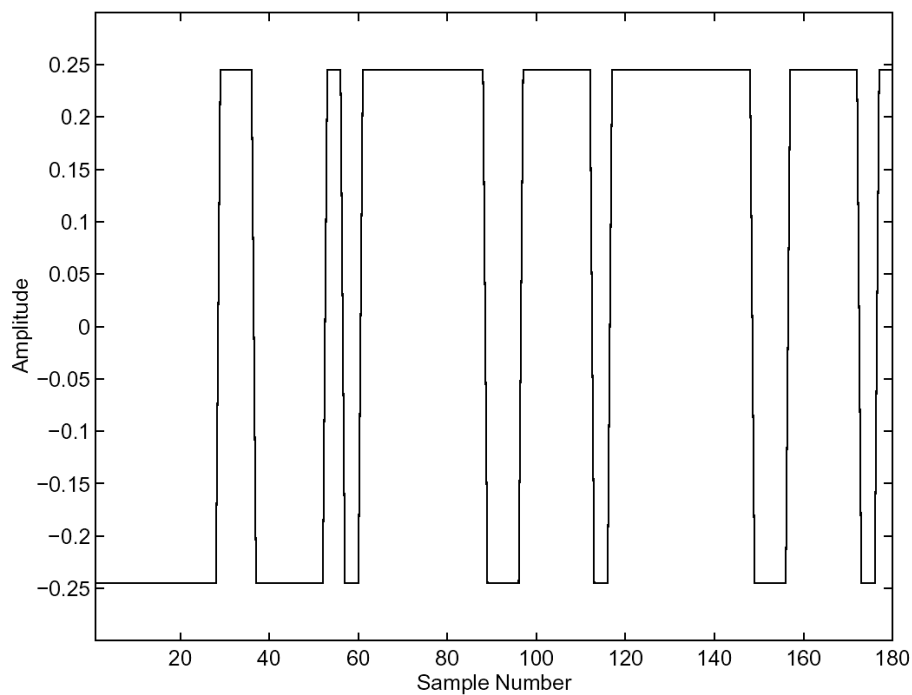


Figure 6.6: Original signal waveform of user 5. The signal parameters are shown in Table 6.1.

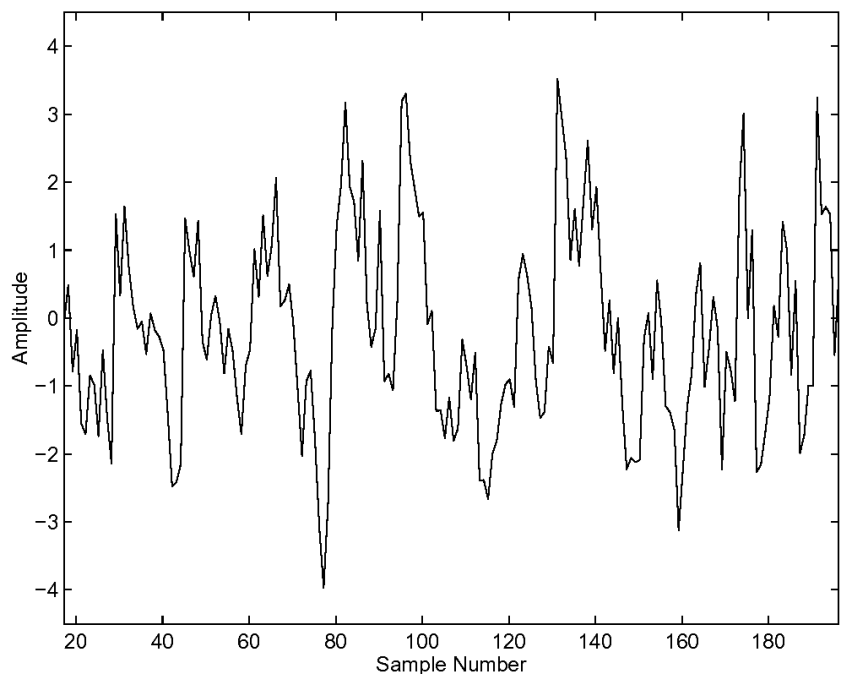


Figure 6.7: Corrupted signal waveform of user 5. The signal parameters are shown in Table 5.1.

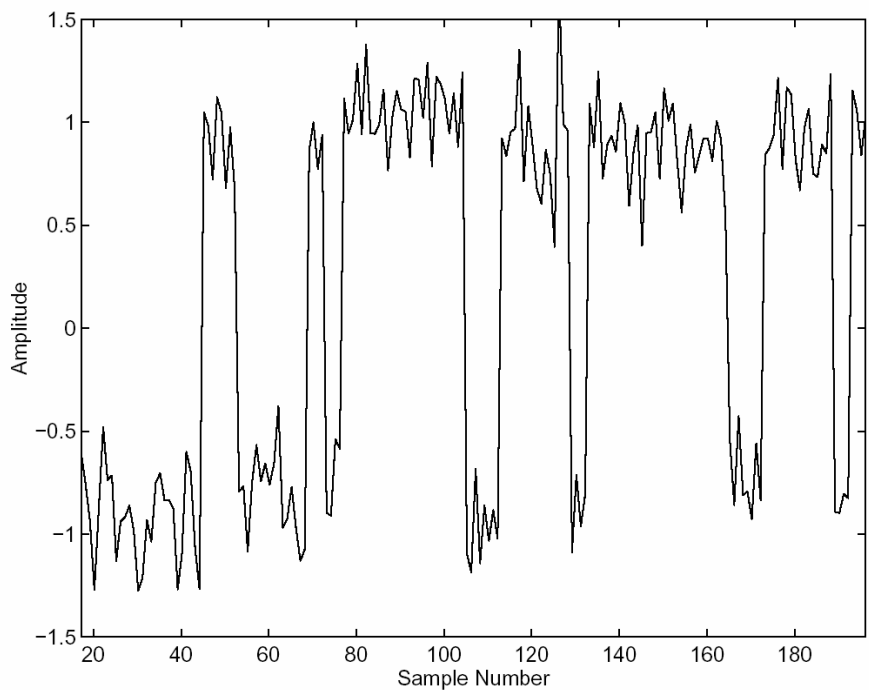


Figure 6.8: Reconstructed signal waveform of user 5. The signal parameters are shown in Table 6.1.

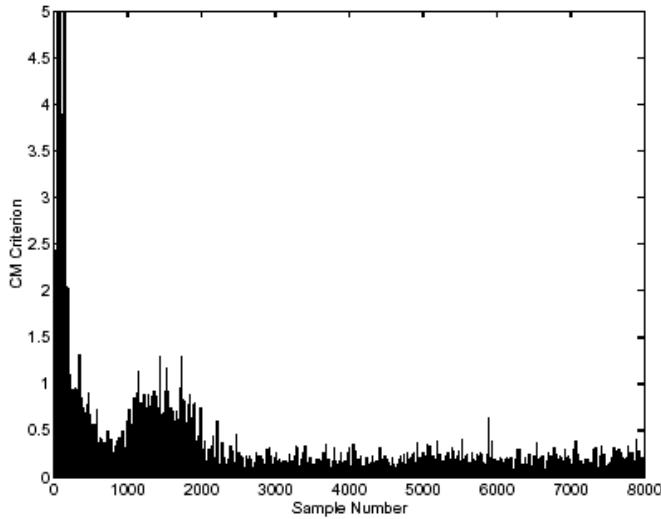


Figure 6.9: Convergence curve for MT-SDDD in port 6 of the beamformer. The signal parameters are shown in Table 6.1.

The DOA distribution of all the users for both the non-crowded and crowded cases is illustrated in Figure 6.11. In the following simulation cases, we assume perfect power control, so all the signals impinging on the array have the same power unless specified otherwise. Figure 6.12 shows the BER performance of different algorithms for $E_b / N_0 = 8$ dB, non-crowded DOA case. In Figure 6.12, to generate the BER for the different adaptive algorithms, we transmit different random bit streams for each user. In the receiver, we use the beamformer adapted by each algorithm to extract each user's signal and then feed the output of the beamformer into the matched filter to despread the signal and estimate the transmitted data bit. The estimates of the data bits are compared to the original transmitted data bits and the BER of each user is calculated. The BER averaged over all the users is then calculated and used in the BER performance plot. From Figure 6.12 we see that the MT-LSCMA cannot work under the low E_b / N_0 condition, and its BER performance is very close to that of the conventional receiver. For the MT-SDDD, when the system is under-loaded (with the number of users less than the number of antenna elements of the array), the BER increases sharply as the number of users increases. This is because the MT-SDDD uses the estimate of the data sample as the desired signal to adapt the weight vectors of the beamformer. When the number of users increases, the error rate of the data sample estimate becomes larger, and the algorithm cannot adapt the weight vectors correctly, therefore the improvement due to the spatial filtering becomes smaller and the BER increases. However, comparing the BER performance of the MT-SDDD with that of the conventional receiver, we see that

a large improvement can still be achieved when the system is under-loaded. When the system is fully-loaded (with the number of users equal to the number of antenna elements of the array) or overloaded (with the number of users greater than the number of antenna elements of the array), the BER changes smoothly as the number of users increases, and the improvement of the BER performance over that of the conventional receiver is small. This is because under

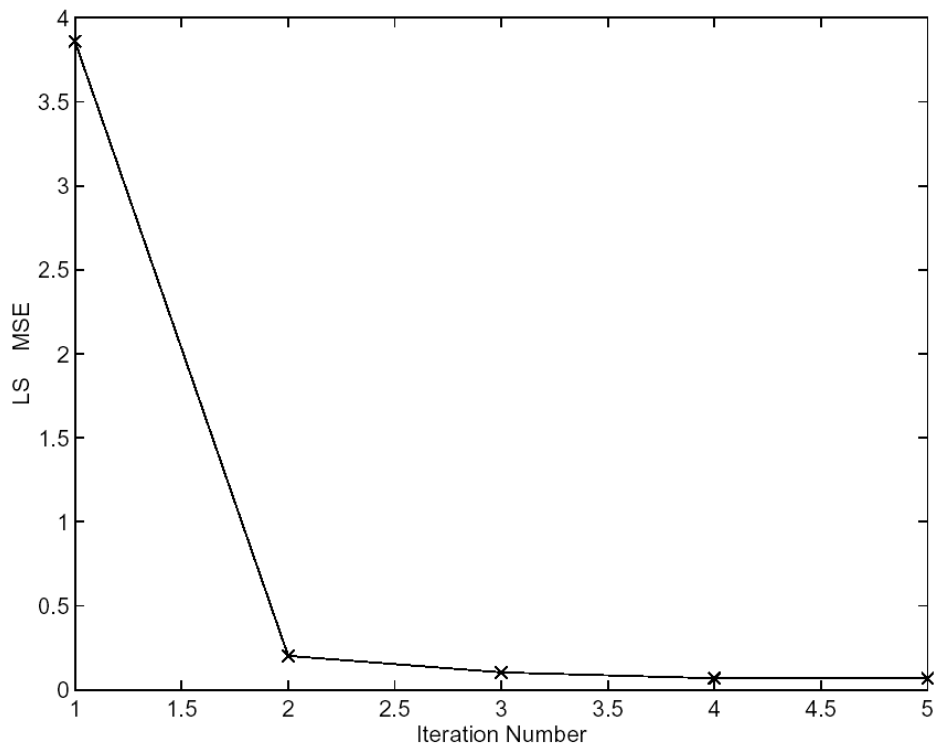
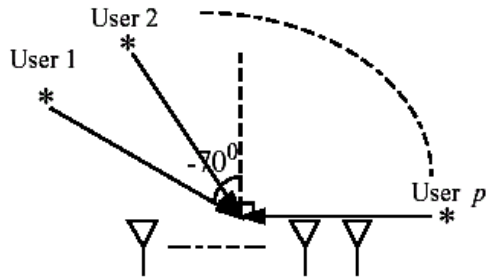


Figure 6.10: Convergence curve for LS-DRMTCMA in port 6 of the beamformer. The LS-DRMTCMA is described in equations (5.46)-(5.51). The data block size in each iteration is equal to 60. The signal parameters are shown in Table 6.1.

Non-Crowded DOA Case



Crowded DOA Case

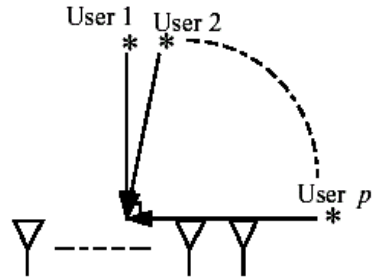


Figure 6.11: DOA distribution of all the users for both the non-crowded and crowded cases.

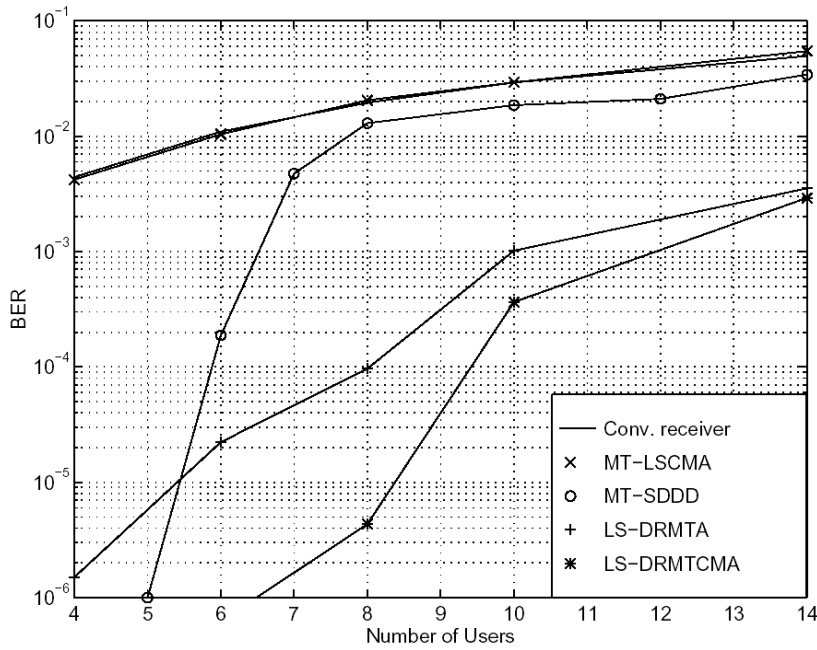


Figure 6.12: BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 8$ dB, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

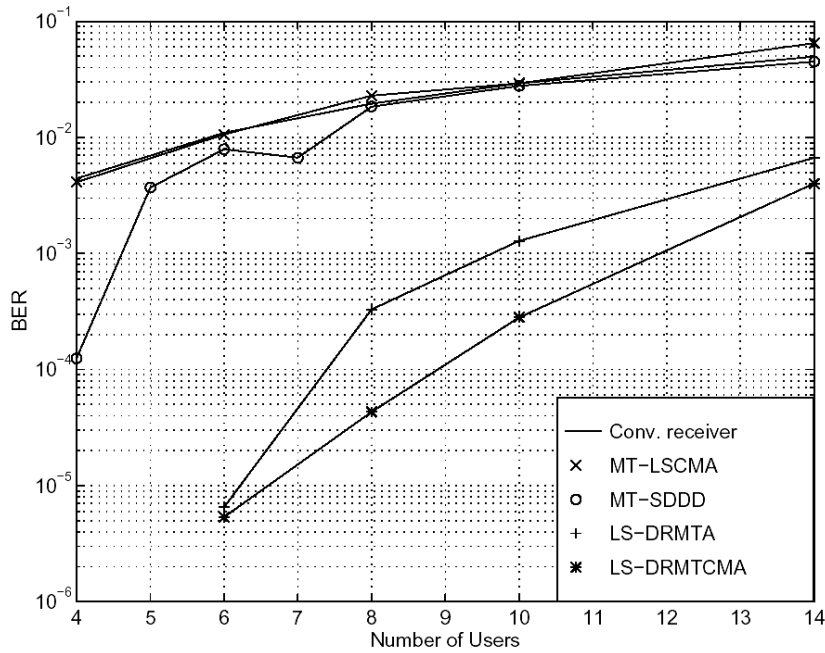


Figure 6.13: BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 8$ dB, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

the fully-loaded and over-loaded situations, the MT-SDDD cannot form deep nulls in the DOAs of the interference. Also, several signals may fall into a main beam of one output port, therefore the interference level cannot be reduced to a very low point, and the BER performance is thus close to that of the conventional receiver.

For LS-DRMTA and LS-DRMTCMA, however, since these two algorithms utilize the information of the PN sequences of all the users to adapt the weight vectors, they can construct deeper nulls in the DOAs of the interference than the MT-SDDD, therefore the BER of these two algorithms is much lower than that of the MT-SDDD. Figures 6.14 and 6.15 show the beampatterns of user 5 generated by using the LS-DRMTCMA and the MT-SDDD algorithm, respectively. Comparing Figures 6.14 and 6.15 we see that the LS-DRMTCMA can generate deeper null in the DOAs of the interference than the MT-SDDD, therefore can reduce the interference to a lower level. Also, since the LS-DRMTCMA uses the constant modulus property of the transmitted signal in addition to the PN sequences of all the users to adapt the weight vectors, it can achieve a lower BER than the LS-DRMTA. However, the improvement of the LS-DRMTCMA over the

LS-DRMTA becomes smaller when the system is over-loaded. This is because under the over-loaded situation, it is the interference falling into the main beam of each output port that dominates the overall interference level. Although the LS-DRMTCMA can form deeper nulls in some DOAs of the interference than the LS-DRMTA, once there is some interference falling into the main beam, the overall interference levels of these two algorithms become almost the same, and thus the BER performance is very close. It is the beamwidth of the main beam and the DOA distribution of the signals that determines the number of interference falling into the main beam. For an 8-element uniform linear array with element spacing equal to half wavelength, from equation (3.47), we obtain the peak-to-null beamwidth of the main beam directed to the broadside of the array

$$\theta_H \approx \arcsin\left(\frac{\lambda}{Md}\right) \approx \arcsin\left(\frac{\lambda}{8 \cdot \frac{\lambda}{2}}\right) \approx 14.5^\circ \quad (6.4)$$

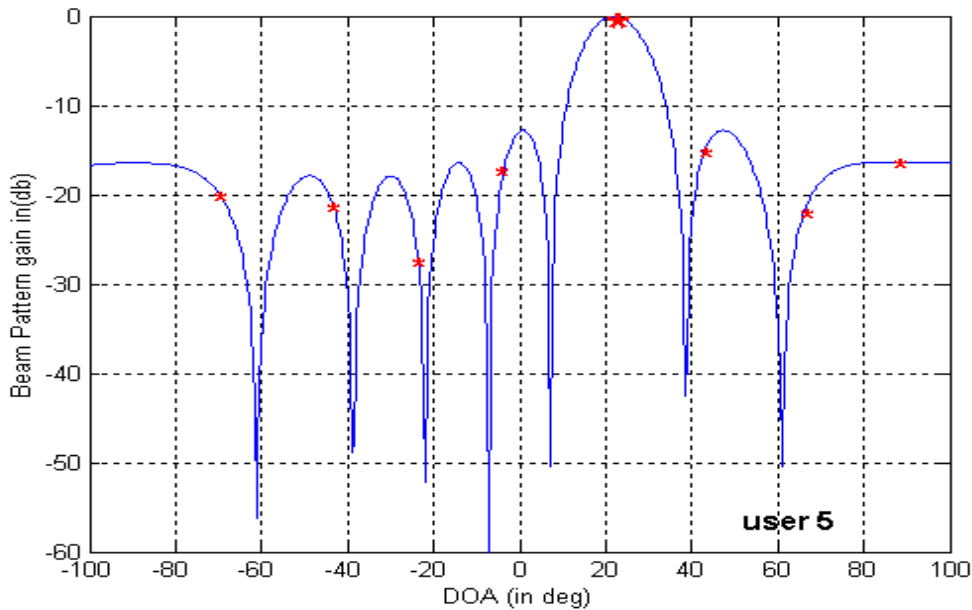


Figure 6.14 Beampattern of user 5 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

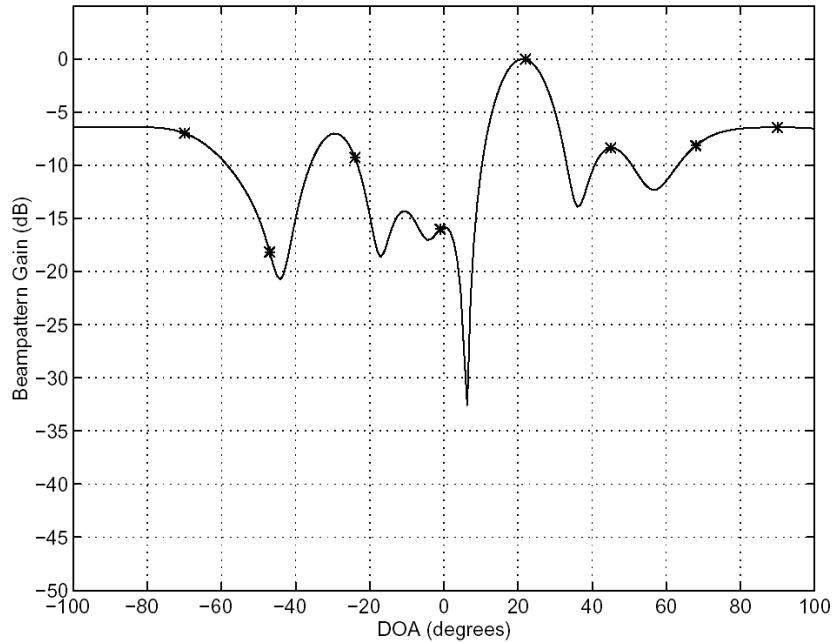


Figure 6.15: Beampattern of user 5 generated by using MT-SDDD. In this case, $E_b/N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

Since as shown in section 6.3.1, the beamwidth of the main beam directed to the broadside of the array is always smaller than those of the main beams directed to other directions, if the angle separation between the DOAs of the received signals is less than 14.5° , at least one interference will fall into the main beam of the desired user.

The BER performance of different algorithms for the $E_b / N_0 = 8$ dB, crowded DOA case is shown in Figure 6.13. From Figure 6.13 we see that the MT-LSCMA still cannot work under this crowded DOA situation. It was found from experiments that the MT-LSCMA can only work for high E_b / N_0 (e. g., $E_b / N_0 = 20$ dB) and far under-loaded (e. g., number of users equal to 4) case. Comparing Figure 6.13 with Figure 6.12, we see that the BER increases as the DOAs of the signals becomes crowded for almost all the test cases. This is because more and more interference can fall into the main beam of one output port if the DOAs of the signals becomes crowded. For example, if there exist 8 users in the system, in the non-crowded DOA case, the angle separation between the DOAs of the received signals is equal to 23° , which is larger than $\theta_H = 14.5^\circ$, therefore no interference will fall into the main beam of the desired user, at least for those close to the broadside of the array.

However, in the crowded DOA case, the angle separation becomes 12.86° , which is smaller than $\theta_H = 14.5^\circ$, thus even for the desired user close to the broadside of the array, two interference will fall into the main beam of the desired user. Figures 6.16 and 6.17 show the beampatterns of user 4 generated by using LS-DRMTCMA for both the non-crowded and crowded case, respectively. Comparing Figures 6.16 and 6.17, we see that in the crowded case, two more interference fall into the main beam of user 4, thus the interference level increases and the BER becomes higher. From Figures 6.12 and 6.13, we see that when the number of users is equal to 8, the BER for LS-DRMTCMA in the non-crowded DOA case is about 4×10^{-6} while the BER for LS-DRMTCMA in the crowded DOA case becomes approximately 4×10^{-5} , which is 10 times of that in the non-crowded DOA case. In this situation, however, the LS-DRMTCMA and LS-DRMTA can still achieve a large improvement over the BER performance of the MT-SDDD.

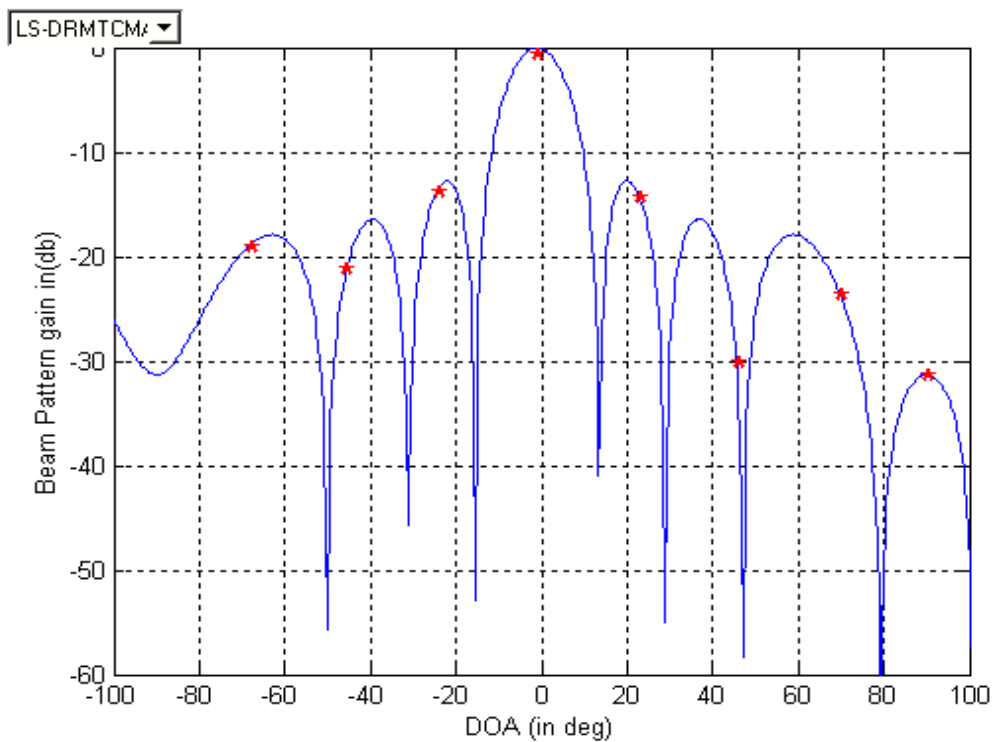


Figure 6.16: Beampattern of user 4 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

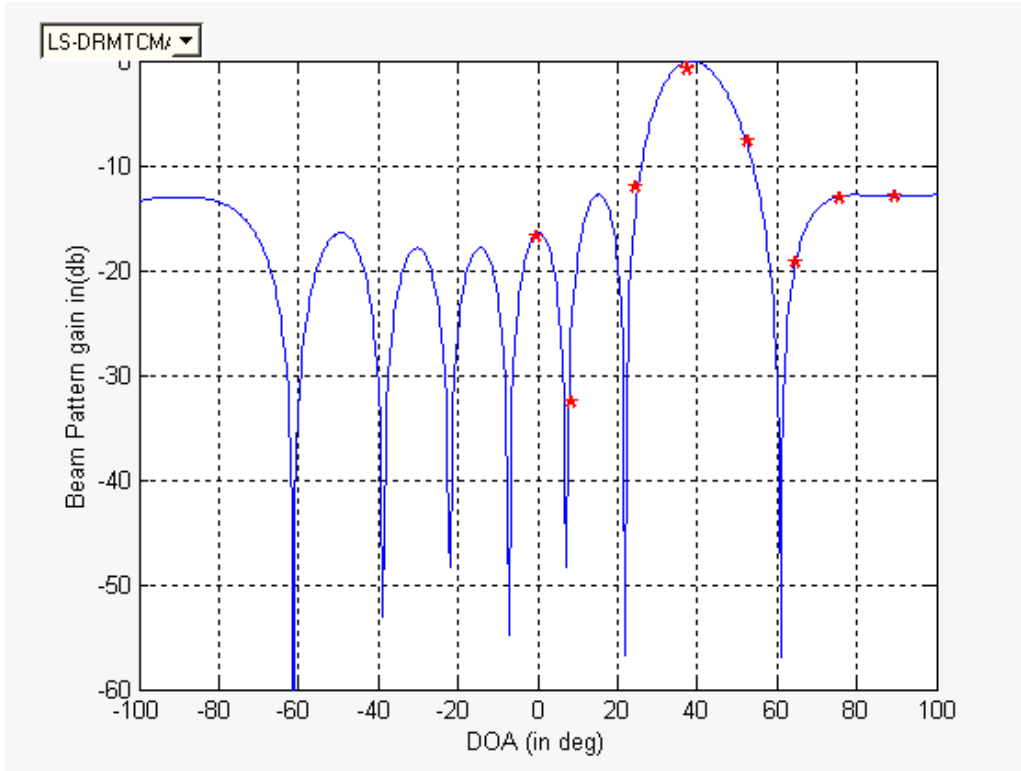


Figure 6.17: Beampattern of user 4 generated by using LS-DRMTCMA. In this case, $E_b / N_0 = 8$ dB, the number of users is equal to 8, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

The BER performance of different algorithms for the $E_b / N_0 = 4$ dB, non-crowded and crowded DOA cases are shown in Figure 6.18 and Figure 6.19, respectively. Comparing Figures 6.18 and 6.19 with Figures 6.12 and 6.13, we see that the BER increases as the E_b / N_0 decreases. From Figures 6.18 and 6.19, we see that in the low E_b / N_0 case, the LS-DRMTCMA and LS-DRMTCMA can still outperform the MT-SDDD for both non-crowded and crowded DOA cases. Comparing the BER curves of LS-DRMTCMA and LS-DRMTCMA in Figures 6.18 and 6.19, we see that the improvement of LS-DRMTCMA over LS-DRMTCMA does not degrade very much as the number of users increases, which is different from that in the $E_b / N_0 = 8$ dB case. This is because for such a low E_b / N_0 , in addition to the interference, the noise also plays an important role in determining the BER. Since the difference between the abilities of the LS-DRMTCMA and LS-DRMTCMA to reduce the noise changes little as the number of users increase, the improvement of LS-DRMTCMA over LS-DRMTCMA does not degrade very much as the number of users increases. One point to note is that when the number of users is equal to 4, the BER of the crowded DOA case is less than that of the non-crowded DOA case. The reason is that when the number of users is equal to 4,

the angle separation between the users is large for both the crowded and non-crowded DOA cases, so there is no interference falling into the main beam near the broadside of the array, and only the users near the endfire of the array can affect each other. The beampattern of user 4 which is near the endfire of the array for both the non-crowded and crowded cases are illustrated in Figures 6.20 and 6.21, respectively. For the crowded DOA case, the DOAs of the signals are equally spaced between 0° and 90° , so there is only one user near the endfire of the array. For the non-crowded case, on the other hand, the DOAs of the signals are equally spaced between -70° and 90° , thus there are two users near the endfire of the array and they interference with each other, which results in a higher BER than in the crowded DOA case.

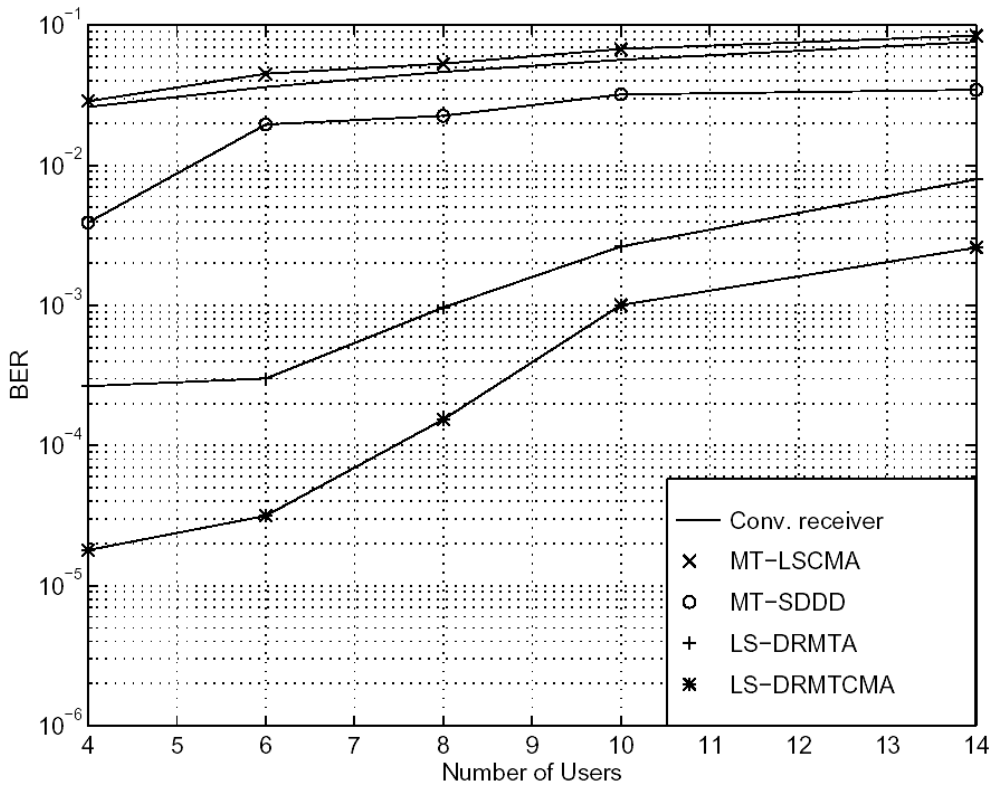


Figure 6.18 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 4$ dB, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

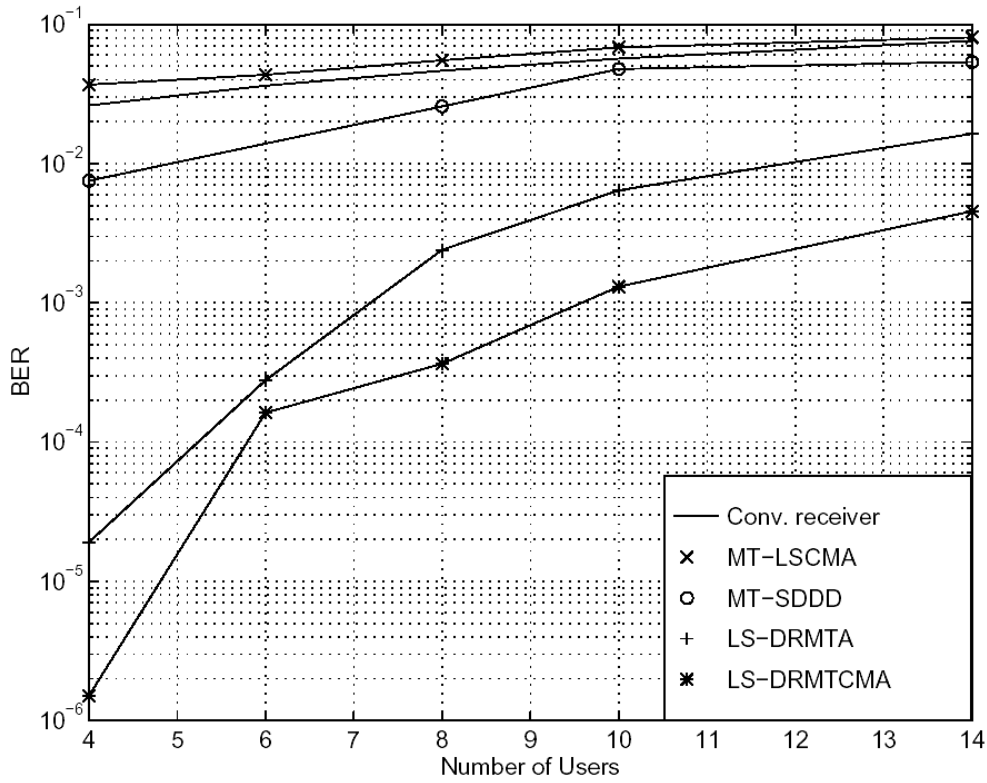


Figure 6.19 BER performance of different adaptive algorithms. In this case, $E_b / N_0 = 4$ dB, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

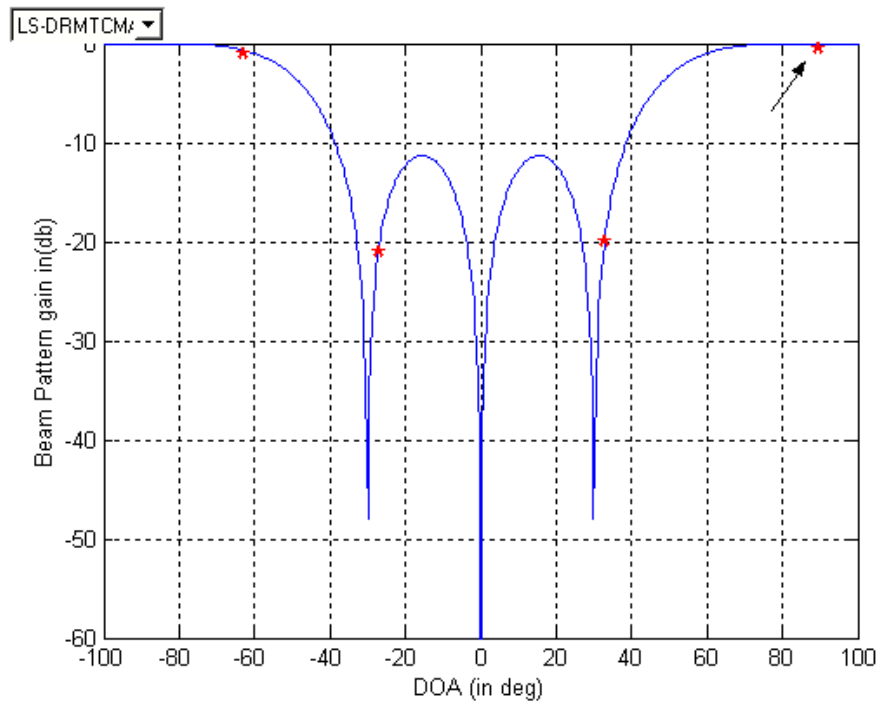


Figure 6.20 Beampattern of user 4 generated by using LS-DRMTCMA in the non-crowded DOA case. In this case, $E_b / N_0 = 4$ dB, the number of users is equal to 4, the DOAs of all the users are equally spaced between -70° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

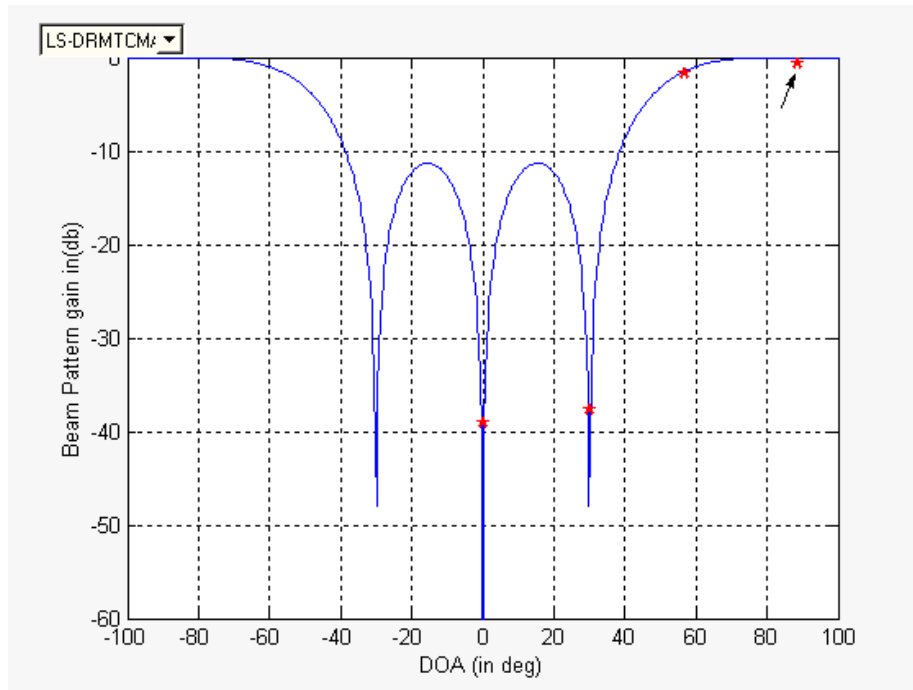


Figure 6.21 Beampattern of user 4 generated by using LS-DRMTCMA in the crowded DOA case. In this case, $E_b / N_0 = 4$ dB, the number of users is equal to 4, the DOAs of all the users are equally spaced between 0° and 90° . The ratio of the coefficients a_{PN} / a_{CM} used in the LS-DRMTCMA is set to 2.

6.4 BER Performance in Multipath Environment

In a wireless radio channel, the transmitted signal may arrive at the receiver through different paths with different time delays. These multipaths will cause the intersymbol interference (ISI) and degrade the BER performance of the system. However, if these multipaths are coming from different DOAs, we can use an adaptive array in the receiver to extract the path with the strongest power and reject the other ones, therefore reducing the ISI and improving the BER performance. In this section, we will examine the BER performance of different algorithms in the multipath environment.

The channel model used is the 2-ray resolvable channel where each user has two multipaths. The reason for using the 2-ray resolvable channel is that the bandwidth of the transmitted signal, i. e., the direct sequence spread spectrum signal, is very high compared to the bandwidth of the wireless radio channel, therefore the multipaths are resolvable in time. The first issue we need to consider for the channel model is the DOAs of the multipaths. In the 2-ray channel model, the two multipaths have different DOAs, and we want to investigate the effect of varying the angle separation between the multipaths on the performance of different algorithms. In the simulation, we consider two different multipath angle separation cases. In the first case, the DOA difference between the two multipaths is set to 10° while in the second case, the DOA difference is set to 20° . As shown in section 6.3.2, for an 8-element uniform linear array with element spacing equal to half wavelength, the peak-to-null beamwidth, θ_H , of the main beam directed to the broadside of the array is approximately equal to 14.5° . By setting the multipath angle separation equal to 10° and 20° , which are less and greater than θ_H , respectively, we can investigate the performance of different algorithms under the condition when both multipaths fall into the main beam and the condition when only one multipath fall into the main beam. The second issue we need to consider for the channel model is the time delay between the two multipaths. For different multipath angle separation cases, the time delay between the two multipaths should be varied. It was shown in [56] and [57] that the time delay between the multipaths with close DOAs tend to be small and that between the multipaths with far separated DOAs tend to be large. Since we also want to investigate the performance of different algorithms under the condition when the time delay between the two multipaths is less and greater than the chip period T_c , for the first case with small DOA difference, we set the time delay between the two multipaths to $0.5T_c$ while for the second case with large DOA difference, we set the time delay between the two multipaths to $1.5T_c$. The last issue we need to consider for the channel model is the power ratio between the two multipaths.

As shown in [58], the amplitude of each single multipath varies little over a small range of distance. Also, since the data bit rate of the system is very high (128kbps), we will consider the amplitude of each multipath unchanged over the simulation period. In the simulation, we want to investigate the effect of varying the power ratio between the multipaths on the performance of different algorithms. So three different power ratios of the first path to the second path, 0 dB, 6dB, and 10dB, are considered for each multipath angle separation case. The signal parameters of the multipaths in different simulation cases are shown in Table 6.2.

Table 6.2 Signal Parameters of Multipaths

Case #	DOA difference Between Multipaths	Time Delay Between Multipaths	Power Ratio (db) (First Path/second path)
1	10^0	$0.5T_C$	0
2	10^0	$0.5T_C$	6
3	10^0	$0.5T_C$	10
4	20^0	$1.5T_C$	0
5	20^0	$1.5T_C$	6
6	20^0	$1.5T_C$	10

The BER performance of different algorithms for cases 1, 2 and 3 are illustrated in Figures 6.22, 6.23, and 6.24, respectively. Comparing Figures 6.22, 6.23, and 6.24 with Figure 6.12, we see that the BER performance is indeed degraded by the multipath in all the cases. Comparing Figures 6.22, 6.23, and 6.24, we note that decreasing the power ratio between the multipaths will result in a higher BER. This is what we expect since in these three simulation cases, the angle separation between the multipaths is equal to 10^0 , which is less than $\theta_H = 14.5^0$, hence both of the multipaths fall into the main beam of the output port, and a lower power ratio between the multipaths means a higher ISI level, which will result in a worse BER performance. However, from Figures 6.22, 6.23, and 6.24, we see that even for such a small multipath angle separation, using the adaptive array in the receiver can still reduce the multipath effect and improve the BER performance, although the improvement becomes smaller when the power ratio between the multipaths decreases.

The BER performance of different algorithms for cases 4, 5 and 6 are illustrated in Figures 6.25, 6.26, and 6.27, respectively. From Figures 6.25, 6.26, and 6.27, we see that although the angle separation between the multipaths is now equal to 20° , which is greater than $\theta_H = 14.5^\circ$, as in the close multipath DOA case, decreasing the power ratio between the multipaths will also result in a higher BER. The reason is that the beamformer cannot ideally form a null in the direction of the second path, so decreasing the power ratio between the multipaths still will result in a higher ISI level. Also since the angle separation is now equal to 20° , the multipaths of one user may fall into the main beam constructed for another user, hence decreasing the power ratio between the multipaths will increase the interference level of the desired user. The higher ISI and interference level will then cause a worse BER performance. Comparing Figures 6.25, 6.26, and 6.27 with Figures 6.22, 6.23, and 6.24, we see that the BER of the well-separated multipath DOA case is lower than that of the close multipath DOA case, and the improvement obtained by using the adaptive array is larger in the well-separated DOA case. The reason is that in the close DOA case where the DOAs of the multipaths are separated by only 10° , all the multipaths tend to fall into the main beam of the output port, and thus the adaptive array cannot reject the multipath effectively. However, in the well-separated multipath DOA case, for most of the users, only one multipath can fall into the main beam of the output port, therefore the multipath effect can be reduced by the array to a very low level.

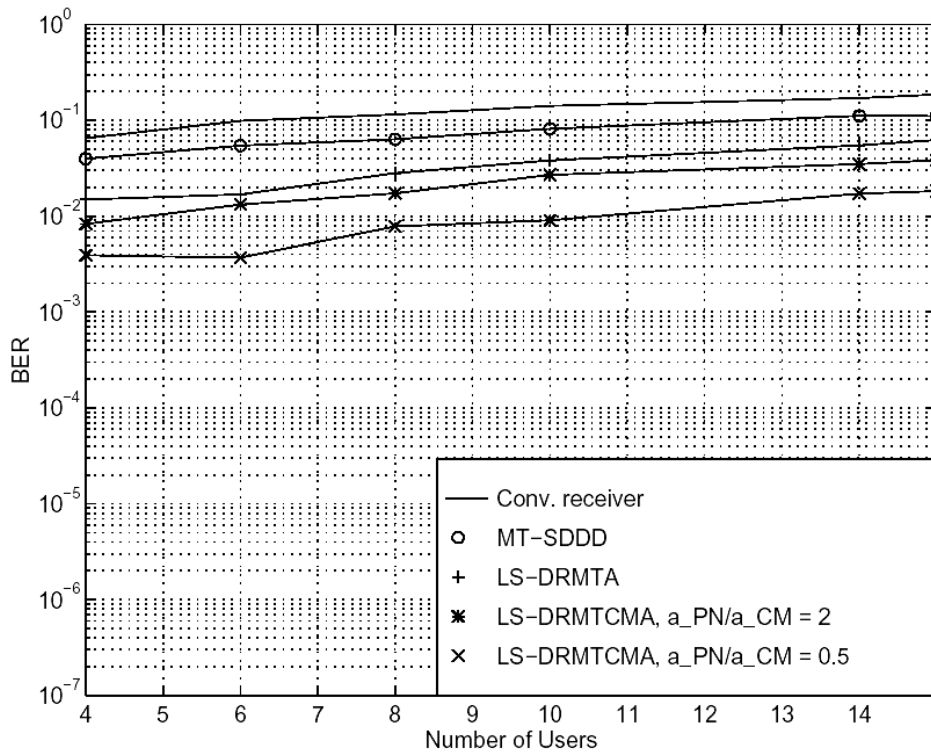


Figure 6.22 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 0 dB, and the time delay between these two paths is $0.5T_c$.

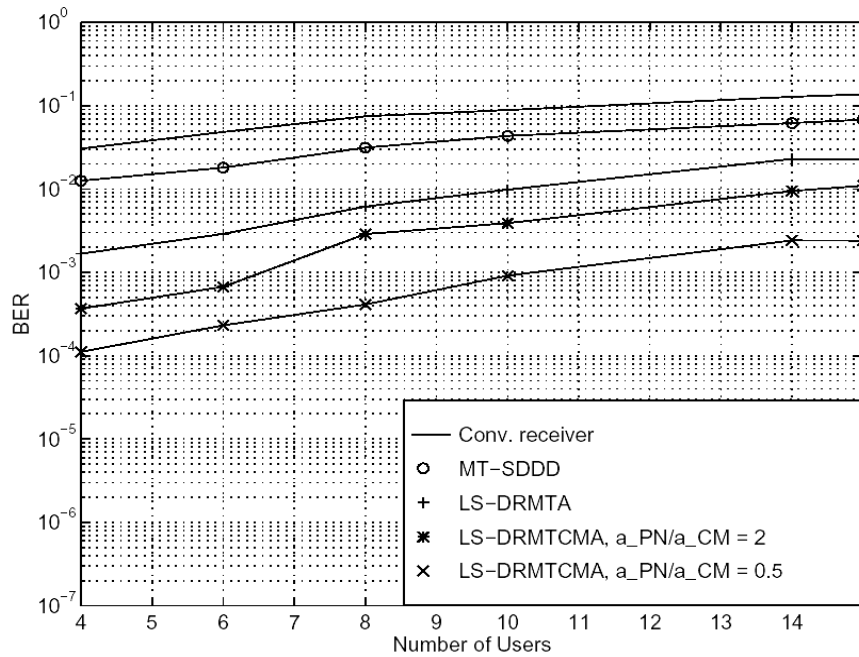


Figure 6.23 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 6 dB, and the time delay between these two paths is $0.5T_c$.

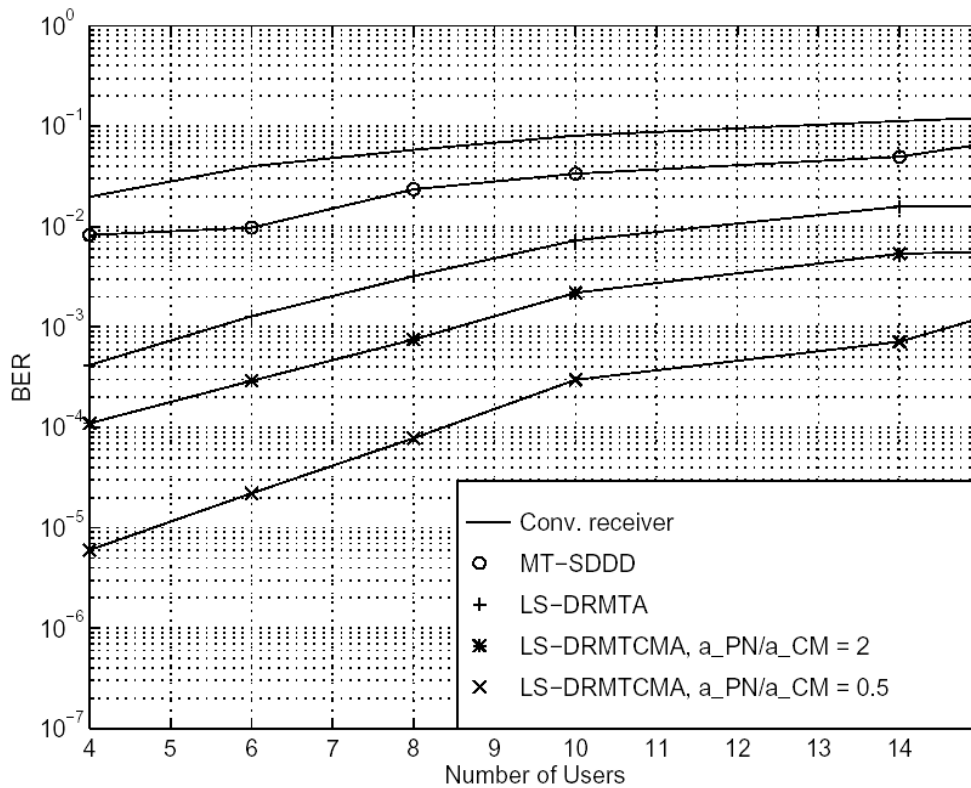


Figure 6.24 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 10° less than that of the first path. The power ratio of the first path to the second path is 10 dB, and the time delay between these two paths is $0.5T_c$.

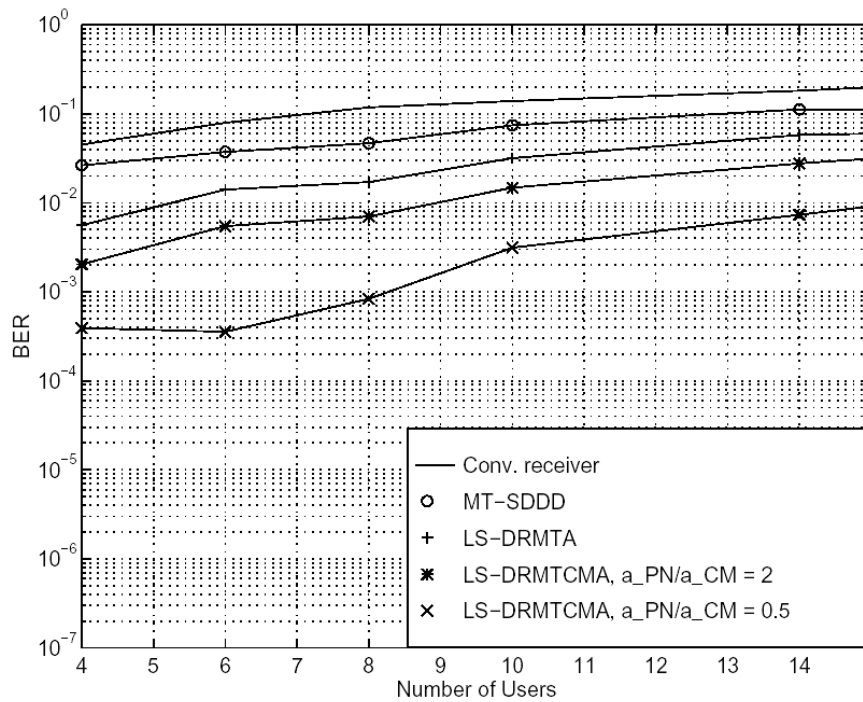


Figure 6.25 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 0 dB, and the time delay between these two paths is $1.5T_c$.

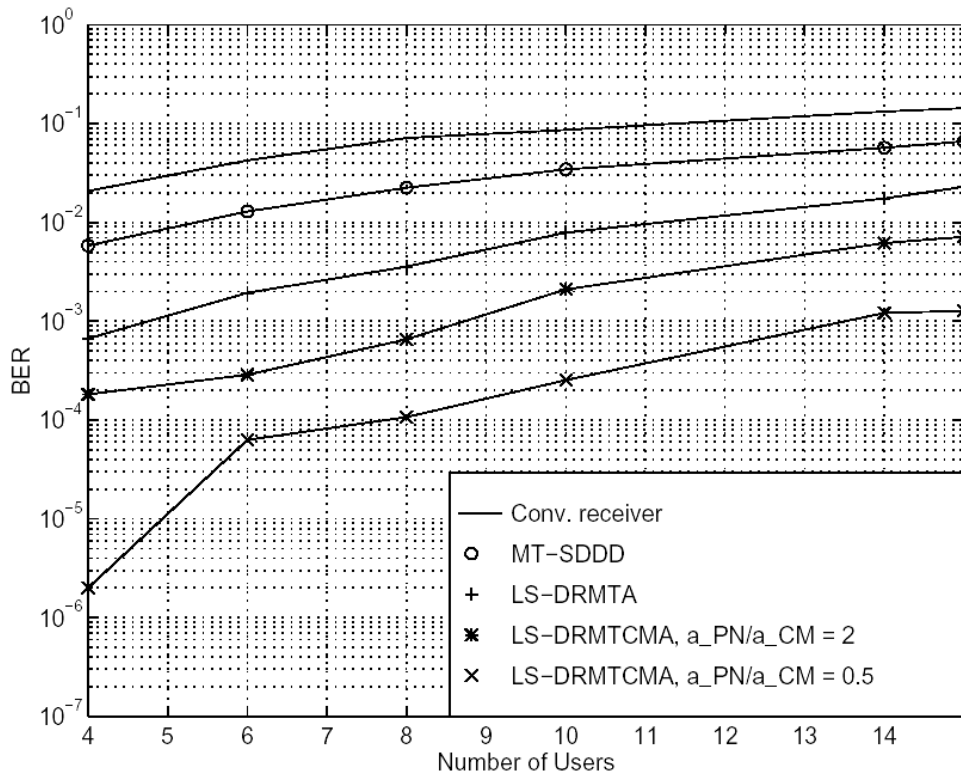


Figure 6.26 BER performances of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 6 dB, and the time delay between these two paths is $1.5T_c$.

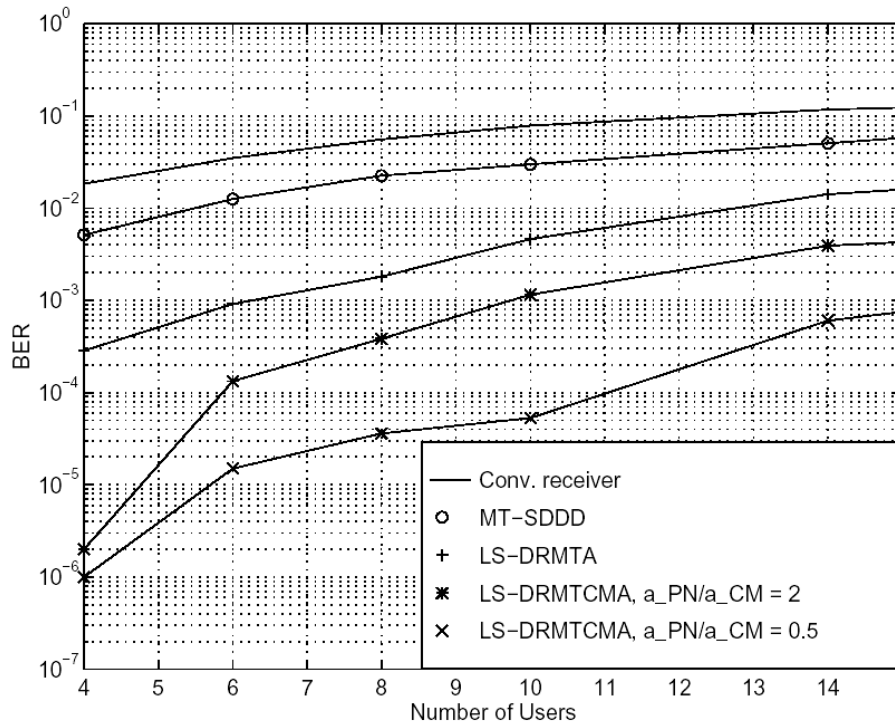
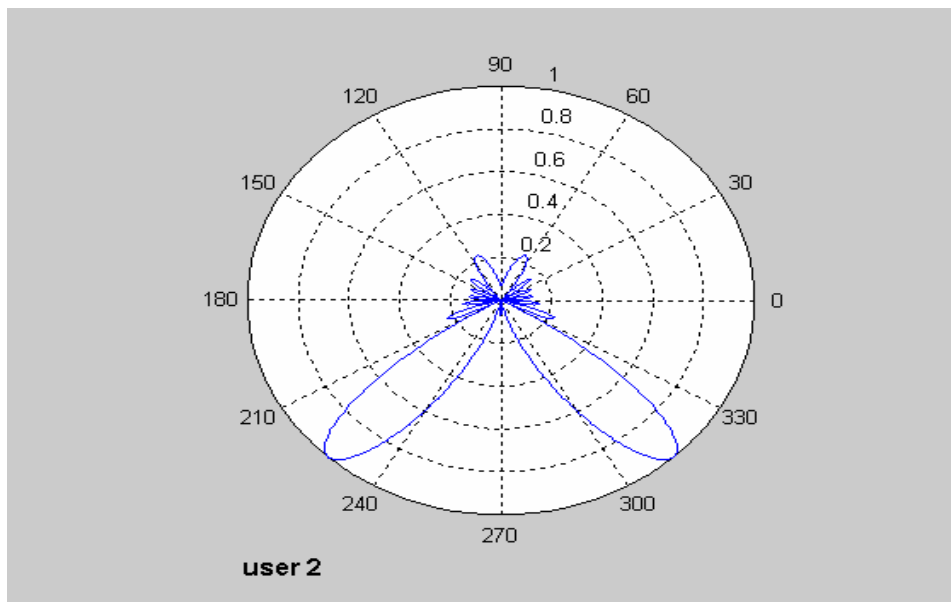
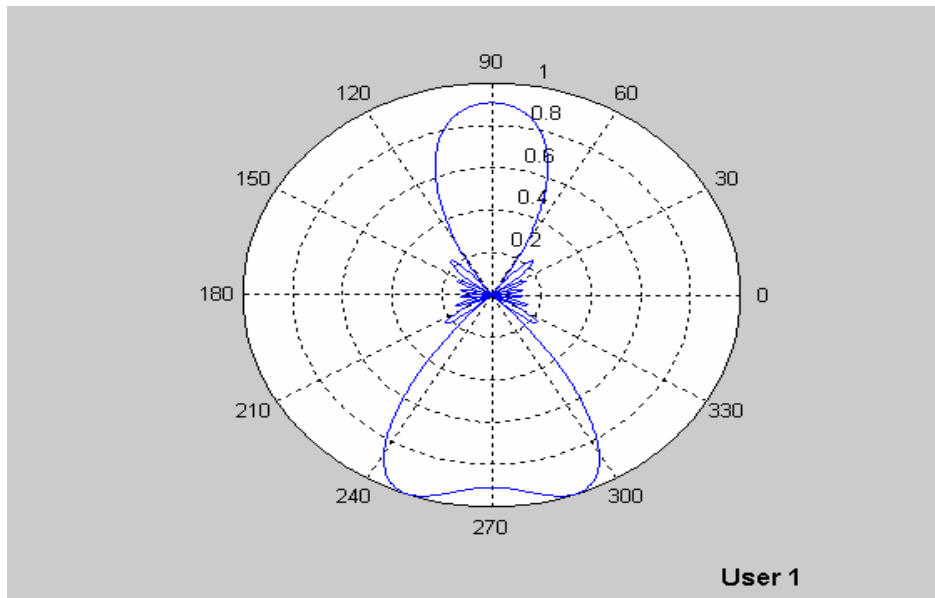
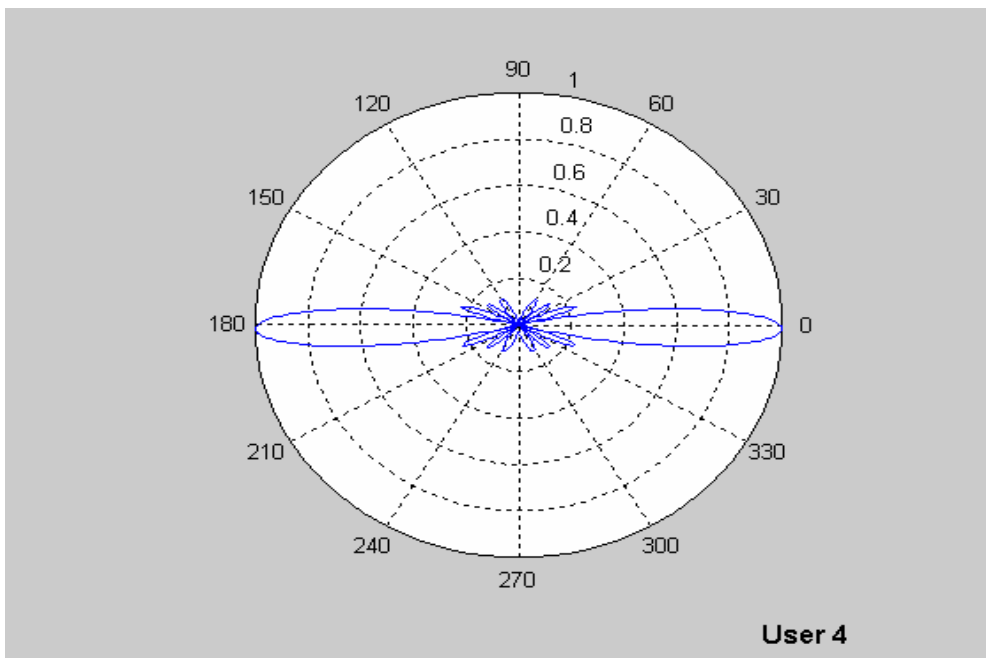
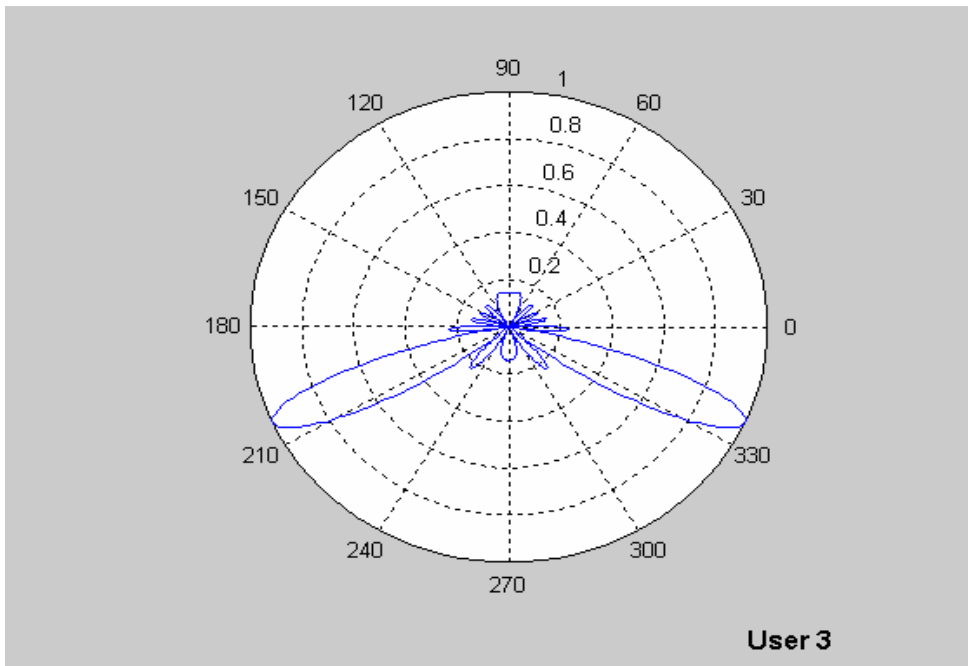
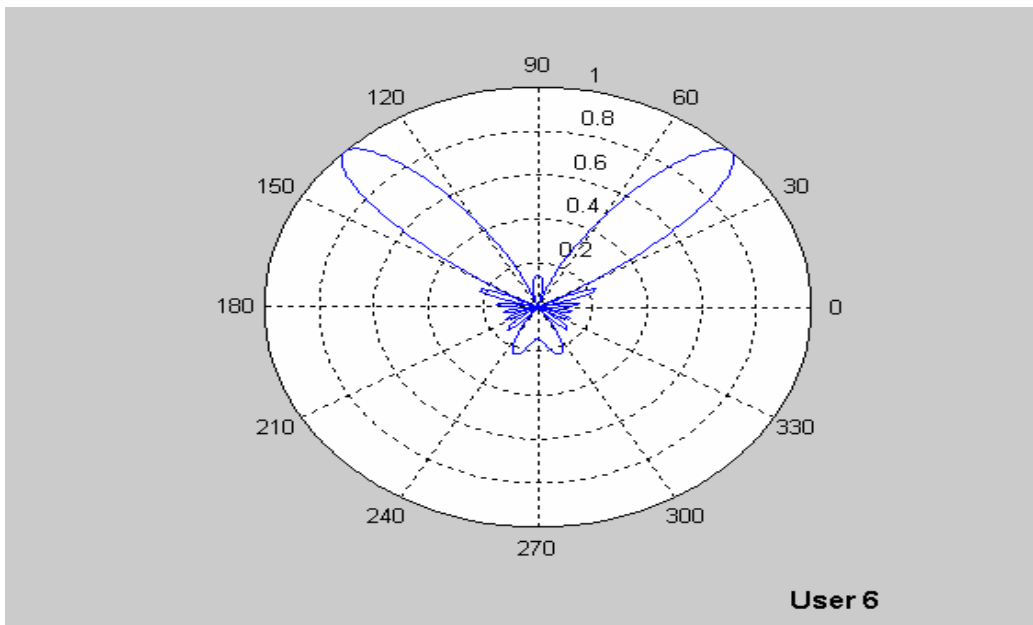
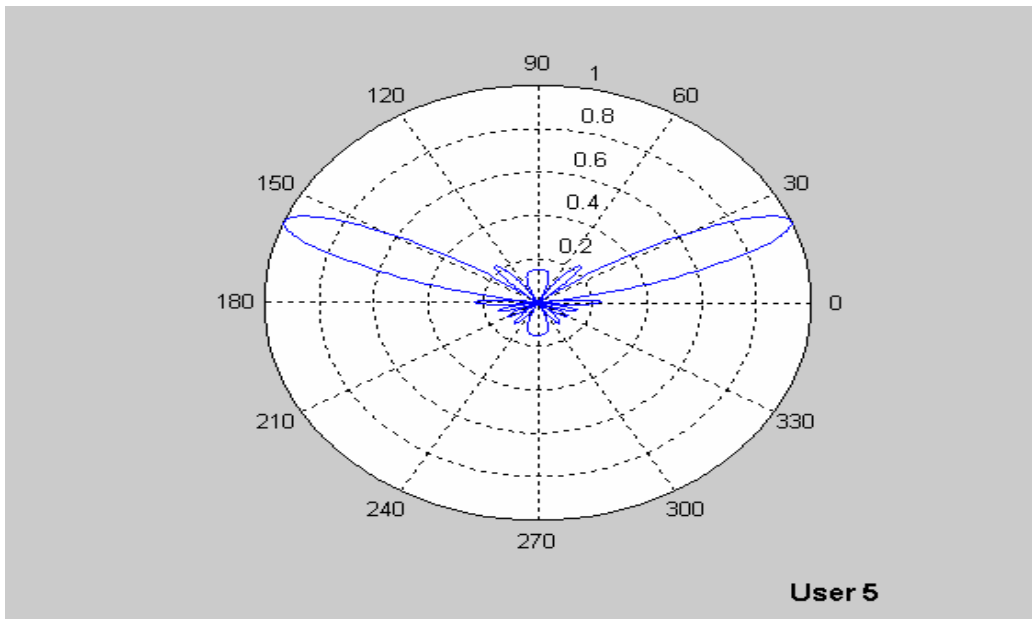


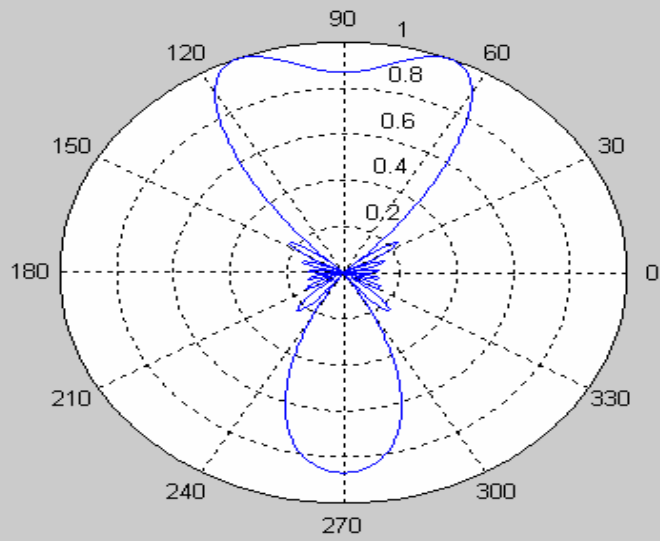
Figure 6.27 BER performance of different adaptive algorithms in multipath environment. In this case, $E_b / N_0 = 8$ dB, the DOAs of the first paths of all the users are equally spaced between -70° and 90° . The DOA of the second path is 20° less than that of the first path. The power ratio of the first path to the second path is 10 dB, and the time delay between these two paths is $1.5T_c$.

The following polar plots are a clear simulation results that shows the ability of the beamformer in tracking a user as the user changes its direction of arrival. The signal processor uses the LS-DRMTA and LS-DRMTCMA algorithms and these shows their interference nulling capability towards the undesired user.

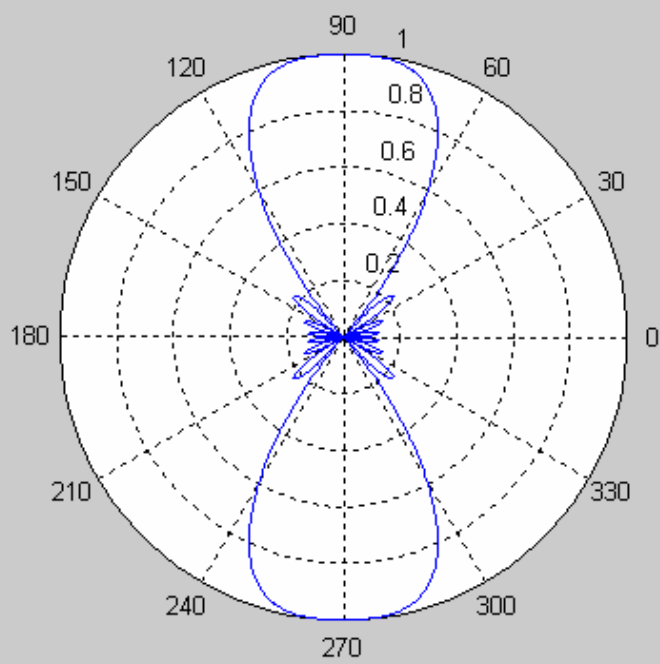








User 7



User 8

6.5 Conclusion

In this chapter, we present the performance of different adaptive array algorithms in a CDMA system. We compare the BER performance of different algorithms in various channel environments (e. g. the AWGN channel and in the multipath environment). From the comparisons we see that the LS-DRMTA and the LS-DRMTCMA, the two algorithms developed, can outperform the other algorithms in both channel environments.

We've first tried to show the performance capability of these four algorithms in different environment. And we see that the BER of the two algorithms (i.e. LS-DRMTA and LS-DRMTCMA) is better than that of the other two (i.e. MT-LSCMA and MT-DD). This is clearly shown by the observation in the simulation. Hence we see that LS-DRMTA and LS-DRMTCMA has the ability to form deep nulls towards the interfering signals and developing major radiation pattern towards the desired signal under both the non-crowded and crowded situation.

Chapter 7

Summary and Future Work

7.1 Summary

In this thesis, four adaptive algorithms are discussed for the beamformer used in a CDMA system. We provide a derivation of these algorithms and create a MATLAB simulation testbed to compare the performance of these algorithms. The BER performance of all these algorithms is compared under different conditions (e. g, the AWGN channel, and the multipath environment). It was shown from the simulation results that the two algorithms, LS-DRMTA and LS-DRMTCMA, can outperform the other algorithms in all the test conditions no matter if the system is over-loaded (i. e., even if the number of users is greater than the number of antenna elements of the array). It was also shown that the LS-DRMTA and LS-DRMTCMA does not need to perform the GSO and sorting procedure which are required in the MT-LSCMA and MT-SDDD, therefore can reduce the system complexity. We also show that unlike that in the MT-LSCMA and MT-SDDD, the number of output ports is not limited by the number of antenna elements in LS-DRMTA and LS-DRMTCMA, which can result in a lower interference level in the beamformer output and make the expansion of the system easier. It has also been tried to show the interference nulling capability of these two algorithms and creating major radiation pattern towards the desired user. This is clearly shown from the polar plots and we can observe that the radiation pattern follows the user as it changes the DOA and this shows the adaptive nature of the beamformer. We also examine the convergence property of different algorithms and show that the two algorithms can converge faster than the other algorithms. In this thesis, we also provide a detailed survey of the adaptive beamformer algorithms, which is very useful for the researchers working in this area.

7.2 Future Work

A number of possibilities exist for future work based on this thesis. These possibilities are outlined below.

1. Currently the two algorithms (LS-DRMTA and LS-DRMTCMA) are only simulated in the workstation using the MATLAB code. It will be useful if these

two algorithms can be implemented in a DSP chip and the 8-element antenna array can be constructed for field trial measurement.

2. In this thesis, the performance of all the algorithms is evaluated by using a uniform linear array. In the future, we can use different array geometries, e. g., a circular array, to examine the performance of the algorithms.

3. The spreading signal used in this research is a short PN sequence (e. g., only 15 chips per bit). In a realistic CDMA system, a longer PN sequence is always used. To reduce the computational complexity of the algorithms, we can use only part of the PN sequence in the LS-DRMTA and LS-DRMTCMA for the adaptation. The effect of using only one segment of the PN sequence on the performance of the algorithms should be examined in the future.

4. In this research, we only use a simple channel model to evaluate the performance of the algorithms. It may be useful if a channel model including the DOA, the time delay, the power level, and the time-varying property of each multipath can be used in the simulation.

5. In the simulation of the multipath case, only the path with the strongest power is extracted by the LS-DRMTA and LS-DRMTCMA. In the future, one can use several time delayed versions of the respread signal in the LS-DRMTA and LS-DRMTCMA to extract several multipaths of the transmitted signal, and combine the beamformer with a RAKE receiver to further enhance the capacity of the system.

In this research, we assume a perfect power control for all the users. In the future, one can examine the performance of these algorithms under imperfect power control conditions. It is believed that the algorithms should have the ability to combat the power variation of the signals.

Finally, in this thesis, all the beamforming algorithms are only used in the base station for the reverse link. It will be a challenge to use the weight vectors generated by the algorithms in the reverse link for the beamforming in the forward link, since the reverse link and forward link are always working at different frequencies.

CHAPTER 8

APPENDIX A

Computer Programs source codes

(Partially given)

% This matlab function implements the Multiple Signal Classification (Music) algorithm .It is applied to (the spatial problem of) the estimation of the direction of arrivals for the purpose of adaptive beamforming. (Proposed by Schmidt.)

```
lambda=0.1;
kappa=2*pi/lambda;
% d=Element_spacing;
d=lambda/2;           %d=Element_spacing;
```

%Generate the signals.

```
Total_signal=zeros(No_signals,No_samples);
k=1:No_signals;
for i=1:No_signals,
    theta=180*(2*rand-1);
    theta=theta*pi/180;
    Total_signal(i,:)=(sqrt(2)*sin(2*sin(2*Freq_signal(I)*pi*k)))+theta);
    %Total_signal(i,:)=randBW(No_samples);
end
```

%Generate noise-which is added to the signal at each of the rx.elements.

```
noise=zeros(No_elements,No_samples);
SNR=20;
noise_var=1/(10^(SNR/10));
for i=1:No_samples,
```

```

    noise(i :, )=sqrt(noise_var)*randn(1,No_samples);
end

%Generate the Array prpagation vector for each of the signals.


---


prop_matrix=zeros(No_elements,No_signals);
for i=1:No_samples,
    k=0:No_elements-1;
    temp=exp(j.*k*kappa*d*sin(Angle_arrivals(i)));
    prop_matrix(:,i)=temp;
end
%waveform received at the elements.
U=zeros(No_elements,No_samples);
for i=1:No_samples,
    u(:,i)=prop_matrix*Total_signal(:,i)+noise(:,i);
end

%Estimate the Input covariance matrix.
Cov_matrix=zeros(No_elements,No_elements);
for i=1:No_samples,
    temp=u(:,i)*(u(:,i)');
end
cov_matrix=cov_matrix/No_samples;

%Find the Eigen values and the corresponding eigen vectors.
[Eig_matrix, Eig_value]=eig(cov_matrix,'nobalance');

[sorted_matrix,sorted_value,count_signals,v_matrix]=Eigsort(Eig_matrix,Eig_value,
No_elements,noise_var);
figure(1);
plot([1:No_elements],abs(fliplr(sorted_value)),'-+');
grid on
hold on
plot([1:No_elements],noise_var*[ones(1,No_elements)],'-o');
xlabel('index');
ylabel('Eigen value');
title('Magnitude of Eigen values of the correlation matrix of rxd.signal');

%plot the spectrum.
Plotspectrum(v_matrix,No_elements,kappa,d);

for i=1:No_elements,

```

```

    value(i)=Eig_value(i,i);
end
sorted_value=sort(value);
count=0;
for i=1:No_elements,
    if (sorted_value(I)<1,5*noise_var)
        count=count+1;
    end
end
count-signals=No_elements-count;
v_matrix=zeros(No_elements,count);
sorted_matrix=zeros(No_elements,No_elements);
for i=1:No_elements,
    for k=1:No_elements,
        if(sorted_value(i)==value(k))
            sorted_matrix(:,i)=Eig_matrix(:,k);
        end
    end
end
for i=1:count
    v_matrix(:,count-i+1)=sorted-matrix(:,i);
end

```

%This matlab function plots the spectrum for the music algorithm.

```

f=zeros(1,10001);
i=1;
for theta=-90:180/10000:90
    k=0:No_elements-1;
    temp=exp(j.*k*kappa*d*sin(theta*pi/180));
    a=temp.';
    f(i)=(a'*a)/(a'*v_matrix*v_matrix'*a);
    i=i+1;
end
theta=-90:180/10000:90;
f=abs(f)./max(abs(f));
figure(2)
plot(theta,20*log10(f));
gridon
xlabel('angle');
ylabel('Magnitude (in db)');
title('plot of the spatial spectrum');

```

%This matlab function generates Random Binary waveform using randn function.

```
function[result]=randBW(No_samples)
    array=randint(1,No_samples,[0 1]);
    array=2.*array-1;
    result=array;
```

% LMS(N,lambda,d,nsamples,SNR,sType,Theta_s,SIR,Theta_I,IntType,Mu)

```
k=2*pi/lambda;
```

```
Theta_s=Theta_s*pi/180;
Theta_I=Theta_I*pi/180;
```

```
signal power=1;
N_int=length(SIR);
```

% signal

% steering vector.

```
Apv_s=zeros(N,nsamples);
for index=1:nsamples,
    Apv_s(:,index)=steeringvector(N,k,d,Theta_s(index)).';%0<=n<=N-1.
End
D_n=GensigAdaptive(sType,signalpower,nsamples,Apv_s);
N_n=GensigAdaptive(nType,10^(-SNR/10),nsamples,Apv_s);
S_n=D_n+N_n;
```

% interferers

```
I_total=zeros (N,nsamples);
for index=1:N_Int,
    SV=zeros(N,nsamples);
    for timeIndex=1:nsamples,
        SV(:,timeIndex)=steeringvector(N,k,d,Theta_I(index,timeIndex)).';
    end
    I_n=GensigAdaptive(IntType(index),10^(-SIR(index)/10),nsamples,SV);
    I_Total=I_Total+I_n;
end;
```

```
% Rx signal=signal+interference.
X_n=S_n+I_Total;
```

```
% weights
```

```
w=ones(N,nsamples);
```

```
%Desired signal.
```

```
Desired=D_n(1,:);
```

```
%.....LMS Filtering.....%
```

```
Error=zeros(1,nsamples);
for index=1:nsamples,
    error(index)=Desired(index)-X_n(:,index)*w(:,index);
    w(:,index+1)=w(:,index)+Mu*Error(index)*x_n(:,index);
end
```

```
%plot the array factor..
```

```
subplotAF(N,k,d,w(:,1),1,2,2,1,'r','f');
axis([-90,90,-50,0]);
subplotAF(N,k,d,w(:,round(nsamples/64)),1,2,2,2,'r','f');
axis([-90,90,-50,0]);
subplotAF(N,k,d,w(:,round(nsamples/32)),1,2,2,3,'r','f');
axis([-90,90,-50,0]);
subplotAF(N,k,d,w(:,round(nsamples/16)),1,2,2,4,'r','f');
axis([-90,90,-50,0]);
```

```
subplotAF(N,k,d,w(:,1),1,2,2,1,'p','f');
subplotAF(N,k,d,w(:,round(nsamples/64)),1,2,2,2,'p','f');
subplotAF(N,k,d,w(:,round(nsamples/32)),1,2,2,3,'p','f');
subplotAF(N,k,d,w(:,round(nsamples/16)),1,2,2,4,'p','f');
```

```
plotAf(N,k,d,w(:,round(nsamples)),3,'p','h');
plotAf(N,k,d,w(:,round(nsamples)),4,'r','h');
```

%Generate signal.

%Generates an input signal:

```
%function output=Generatesignal(Type,power,nsamples,sterringvector,F,Fs)
```

```
%switch Type
```

```
case 'c',%constant signal.
```

```
    signal=sqrt(power)*ones(1,nsamples);
```

```
case 'g',%Gaussian.
```

```
    signal=sqrt(power)*randn(1,nsamples);
```

```
case 'r',%Rayleigh Fading.
```

```
    signal=ghFade(nsamples,power);
```

```
case 's',%sinusoidal.
```

```
    signal=sqrt(2*power)*sin(pi*0.4*((1:nsamples)-1));
```

```
end;
```

```
%steering vector.
```

```
N_elements=length(sterringvector);
```

```
output=zeros(N_elements,nsamples);
```

```
for spaceIndex=1:N_elements,
```

```
    output(spaceIndex,:)=signal*sterringvector(spaceIndex);
```

```
end
```

%sterring vector

```
%computes and returns the sterring vector for a uniform linear array(ULA)
```

```
% SV=sterringvector(N,k,d,Theta);
```

```
%parameters.
```

```
%N=number of elements in the ULA.
```

```
%k=wave number.
```

```
%d=spacing between the antenna elements.
```

```
%Theta=Angle of Incidence of the signal.
```

```
sv=exp(j*k*d*(N-1)*sin(Theta));
```

%Calculation of covariance Matrix

%computes the time-averaged covariance Matrix(R) of the antenna array output.

%parameters;

%signal vector=Antenna Array output.

%timestart=starting time index.

%timeEnd=Ending time Index.

%function R =calccovarianceMatrix(signalvector,timestart,timeend)

signalvector=2*sin(pi/2);

timestart=0.1;

timeend=0.3;

N=length(signalvector(:,1));

%Initialize R.

R_n=zeros(N,N);

%compute the averaged auto-correlation matrix.

for Timeindex=timestart:timeend,

 value=signalvector(:,Timeindex)*signalvector(:,Timeindex);

 R_n=R_n+value;

end;

%scale the result.

R=R_n/(timeend-timestart+1);

%Cross correlation vector.

%computes the (time-averaged) cross correlation vector between the

%reference signal and the antenna array output.

%parameters.

%Refsignal=The reference signal.

%Rxsignal=The Antenna Array output(received signal).

%tstart=starting time index.

%tend=Ending time index.

%function r=CrossCorrelationVector(Refsignal,Rxsignal,tstart,tend).

N=(length(Rxsignal(:,1)));

%Initialize cross correlation vector.

```

r_n=('zeros(N,1)');
%compute the vector using time-averaging.
for TimeIndex=tstart:tend,
    value=Refsignal(Timeindex)*Rxsignal(:,Timeindex);
    r_n=r_n+value;
end;
%scale the result.
r=r_n/(tend-tstart+1);

```

%This matlab program gives Beam Pattern of an equal weight beam former with polar and rectangular coordinate.

%Function definition.

% Function Beam Pattern=Normalized Beam Pattern (N,d,theta).

%N - Number of array elements.

%d - element spacing.

%lambda - wavelength=1m.

```

m=8;
lambda=1;
theta=-pi:pi/500:pi;
d=lambda/2;
thetanot=pi/2;
temp=pi*d*(sin(theta))/lambda;

g=sin(m*temp).*exp(-j*(m-1)*temp)./sin(temp);
g=g/max(g);

```

```

figure(1);
polar(theta,abs(g))
grid on
title('Beam Pattern of an equal weight beam former with polar coordinate');

```

```

figure(2);
plot(theta*180/pi,20*log10(abs(g)))
grid on
title('Beam Pattern of an equal weight beam former with rectangular coordinate');
xlabel('DOA(in deg)');

```

```
ylabel('Beam Pattern gain (in db)');  
axis([-180 180 -40 0]);  
hold on
```

% This Matlab program will form a radiation pattern towards each desired signal arriving on the antenna according to a given Direction of arrival (DOA) .

```
%N - Number of array elements.  
%d - element spacing.  
%thetanot - steering angle.  
%lambda - wavelength.=1m.
```

%Function definition.

```
function Afactor=ArrayFactorsSinc(N,d,thetanot)
```

```
bkcoloftx=[1 0 0]  
bkcolofed=[1 0 1]  
%thetanot=pi/2; % DOA of the desired signals.
```

```
N=8;  
thetanot=0;
```

```
lambda=1;  
d=lambda/2;  
theta=-pi:pi/500:pi;  
kappa=2*pi/lambda;  
temp=kappa*d*(sin(theta)-sin(thetanot))/2;
```

```
%calculate Array Factor using the closedform Sinc expression
```

```
f=exp(j*temp*(N-1)).*sin(N*temp)./sin(temp);  
f=f/max(f);
```

```
%plot - Rectangular and polar.
```

```
figure(1);  
%subplot(2,1,1),  
polar(theta,abs(f));  
grid on
```

```

%title('polar plot of the Array Factor using the closedform Sinc Expression');

figure(2);
%subplot(2,1,2),

plot(theta*180/pi,20*log10(abs(f)));
grid on
%title('Array Factor using the closedform Sinc Expression');

xlabel('DOA (in deg)');
ylabel('Beam Pattern gain in(db)');
axis([-180 180 -50 0]);

```

This Matlab program will give an estimation of the mixing matrix and signal separation

```

=====
B = [ 1 0 0 ; 0 1 1 ; 0 -i i ] ;
Bt = B' ;
M=8;
nem=8;
m=8;

Ip= zeros(1,nem) ;
Iq= zeros(1,nem) ;
g= zeros(3,nem) ;
G= zeros(2,2) ;
vcp= zeros(3,3);
D= zeros(3,3);
la= zeros(3,1);
K= zeros(3,3);
angles = zeros(3,1);
pair= zeros(1,2);
c= 0 ;
s= 0 ;

%init;
encore= 1;
V= eye(m);

```

```

% Main loop
while encore, encore=0;
for p=1:m-1,
for q=p+1:m,

    Ip = p:m:nem*m ;
    Iq = q:m:nem*m ;

    % Computing the Given angles

    %g=[M(p,Ip)-M(q,Iq);M(p,Iq); M(q,Ip)];
    [vcp,D] = eig(real(B*(g*g')*Bt));
    [la, K] = sort(diag(D));
    angles = vcp(:,K(3));
    if angles(1)<0 , angles= -angles ; end ;
    c= sqrt(0.5+angles(1)/2);
    s= 0.5*(angles(2)-j*angles(3))/c;

    if abs(s)>s, %%% updates matrices M and V by a Given s rotation
        encore = 1 ;
        pair = [p;q] ;
        G = [ c -conj(s) ; s c ] ;
        V(:,pair) = V(:,pair)*G ;
        M(pair,:)=G*M(pair,:);
        M(:,[Ip Iq]) = [ c*M(:,Ip)+s*M(:,Iq) -conj(s)*M(:,Ip)+c*M(:,Iq) ] ;
    end%% if
end%% q loop
end%% p loop
end%% while

%% estimation of the mixing matrix and signal separation
A= IW*V;
S= V'*Y;

return ;

```

%This matlab program gives a radiation pattern of the received signals in a form circular pattern

```
disp(' ');%Number of elements in an array
disp('please enter the number of elements in the linear array.')
N=input(['Number of elements in the array=']);
disp(' ')%Element spacing
disp('Please enter the element spacing (spacing normalized to wavelength).')
disp('1) 1/2')
disp('2) 2/3')
zz=input(['Element spacing=']);
if zz==1
    var=0.0375;
elseif zz==2
    var=0.05;
end
t=-179:1:180;
rad=t*pi/180;          %teta=t*pi/180;
d_phi=2*pi/N;        %delta phi
phi_1=pi/N;
radius=var/sin(phi_1); %in m
c=3*10^8;            %constant
f=2*10^9;           %key in frequency
lambda=c/f;
k=2*pi/lambda;
for a=1:360
    a1=a*pi/180;
    for i=1:N
        phi=(i-1)*d_phi;
        term=j*(k*radius*cos(a1-phi));
        pattern(i)=exp(term);    %its uniform excitation
    end
    y(a)=sum(pattern);
    z(a)=abs(y(a));
end
polar(rad,z);
```

%This matlab program computes the radiation pattern based upon a given algorithm

```
%N - Number of array elements.
%d - element spacing.
%thetanot - steering angle.
%lambda - wavelength.=1m.

%Function definition.
function Afactor=ArrayFactorsSinc(N,d,thetanot)

bkcoloftx=[1 0 0]
bkcolofed=[1 0 1]

N=18;      % No.of elements in the linear array.

thetanot=pi/3; % DOA of the desired signals.

lambda=1;
d=lambda/2;
theta=-pi:pi/500:pi;
kappa=2*pi/lambda;
temp=kappa*d*(sin(theta)-sin(thetanot))/2;

%calculate Array Factor using the closedform Sinc expression

f=exp(j*temp*(N-1)).*sin(N*temp)./sin(temp);
f=f/max(f);

%plot - Rectangular and polar.

figure(1);

prompt = {'Enter thetanot:','Enter Resprctive Alg.:'};
title = 'Input for peaks function';
lines = 1;
def = {'20','MT-LSCMA'};
```

```

answer = inputdlg(prompt,title,lines,def);

    prompt = {'Enter thetanot:', 'Enter Resprctive Alg.:'};
    title = 'Input for peaks function';
    lines = 1;
    def = {'20', 'MT-DD'};
    answer = inputdlg(prompt,title,lines,def);

    prompt = {'Enter thetanot:', 'Enter Resprctive Alg.:'};
    title = 'Input for peaks function';
    lines = 1;
    def = {'20', 'LS-DRMTA'};
    answer = inputdlg(prompt,title,lines,def);

    prompt = {'Enter thetanot:', 'Enter Resprctive Alg.:'};
    title = 'Input for peaks function';
    lines = 1;
    def = {'20', 'LS-DRMTCMA'};
    answer = inputdlg(prompt,title,lines,def);

%subplot(2,1,1),
polar(theta,abs(f));
grid on
%title('polar plot of the Array Factor using the closedform Sinc Expression');

figure(2);

%subplot(2,1,2),

hpop = uicontrol('Style', 'popup',...
    'String', 'MT-LSCMA | MT-DD | LS-DRMTA | LS-DRMTCMA |',...
    'Position', [20 360 90 50],...
    'Callback', 'setmap');

plot(theta*180/pi,20*log10(abs(f)));
grid on
%title('Array Factor using the closedform Sinc Expression');

xlabel('DOA (in deg)');
ylabel('Beam Pattern gain in(db)');
axis([-50 180 -50 0]);

```

Appendix B

Differentiation with Respect to a Vector

An issue commonly encountered in the study of optimization theory is that of differentiating a cost function with respect to a parameter vector of interest. The purpose of Appendix B is to address the more difficult issue of differentiating a cost function with respect to a complex-valued parameter vector. We begin by introducing some basic definitions [10].

B.1 Basic Definitions

Consider a complex function $f(w)$ that is dependent on a parameter vector w . When w is complex valued, there are two different mathematical concepts that require individual attention: (1) the vector nature of w , and (2) the fact that each element of w is a complex number.

Dealing with the issue of complex numbers first, let x_k and y_k denote the real and imaginary parts of the k th element w_k of the vector w ; that is,

$$w_k = x_k + jy_k \quad (\text{B.1})$$

We thus have a function of the x_k and y_k real quantities. Hence, we may use equation (B.1) to express the real part x_k in terms of the pair of complex conjugate coordinates w_k and w_k^* as

$$x_k = 1/2(w_k + w_k^*) \quad (\text{B.2})$$

and express the imaginary part y_k as

$$y_k = 1/2j(w_k - w_k^*) \quad (\text{B.3})$$

where $*$ denotes complex conjugation. The real quantities x_k and y_k are functions of both w_k and w_k^* . It is only when we deal with analytic functions f that we are permitted to abandon the complex-conjugated term w_k^* by virtue of the Cauchy-Riemann equations.

However, most functions encountered in physical sciences and engineering are not analytic. The notion of a derivative must tie in with the concept of a differential. In particular, the chain rule of changes of variables must be obeyed. With these important points in mind, we may define certain complex derivatives in terms of real derivatives, as shown by

$$\frac{\partial}{\partial w} = \frac{1}{2} \left(\frac{\partial}{\partial x_k} + j \frac{\partial}{\partial y_k} \right) \tag{B.4}$$

and

$$\frac{\partial}{\partial w^*} = \frac{1}{2} \left(\frac{\partial}{\partial x_k} - j \frac{\partial}{\partial y_k} \right) \tag{B.5}$$

The derivatives defined herein satisfy the following two basic requirements:

$$\frac{\partial w_k}{\partial w_k} = 1 \tag{B.6}$$

$$\frac{\partial w_k}{\partial w_k^*} = \frac{\partial w_k^*}{\partial w_k} = 0 \tag{B.7}$$

(An analytic function f must satisfy $\frac{\partial f}{\partial z^*} = 0$ everywhere, where z is a complex variable.)

The next issue to be considered is that of differentiation with respect to a vector. Let w_0, \dots, w_{M-1} denote the elements of an $M \times 1$ complex vector w . We may extend the use of equation (B.4) and (B.5) to deal with this new situation by writing

$$\frac{\partial}{\partial w} = \frac{1}{2} \begin{pmatrix} \frac{\partial}{\partial x_0} + j \frac{\partial}{\partial y_0} \\ \frac{\partial}{\partial x_1} + j \frac{\partial}{\partial y_1} \\ \vdots \\ \frac{\partial}{\partial x_{M-1}} + j \frac{\partial}{\partial y_{M-1}} \end{pmatrix} \tag{B.8}$$

and

$$\frac{\partial}{\partial \mathbf{w}^*} = \frac{1}{2} \begin{pmatrix} \frac{\partial}{\partial x_0} - j \frac{\partial}{\partial y_0} \\ \frac{\partial}{\partial x_1} - j \frac{\partial}{\partial y_1} \\ \vdots \\ \frac{\partial}{\partial x_{M-1}} - j \frac{\partial}{\partial y_{M-1}} \end{pmatrix} \quad (\text{B.9})$$

where we have $w_k = x_k + jy_k$ for $k = 0, 1, \dots, M-1$. We refer to $\frac{\partial}{\partial \mathbf{w}}$ as a derivative with respect to the vector \mathbf{W} , and to $\frac{\partial}{\partial \mathbf{w}^*}$ as a conjugate derivative also with respect to the vector \mathbf{w} . These two derivatives must be considered together. They obey the following relations

$$\frac{\partial \mathbf{w}}{\partial \mathbf{w}} = \mathbf{I} \quad (\text{B.10})$$

and

$$\frac{\partial \mathbf{w}}{\partial \mathbf{w}^*} = \frac{\partial \mathbf{w}^*}{\partial \mathbf{w}} = \mathbf{0} \quad (\text{B.11})$$

where \mathbf{I} is the identity matrix and $\mathbf{0}$ is the null matrix. For the subsequent use, we will adopt the definition of (B.9) as the derivative with respect to a complex-valued vector.

B.2 Examples

In this section, we illustrate some applications of the derivative defined in equation (B.9). Two examples will be considered.

Example 1 Let \mathbf{x} and \mathbf{w} denote two complex-valued $M \times 1$ vectors. There are two inner products, $\mathbf{x}^H \mathbf{w}$ and $\mathbf{w}^H \mathbf{x}$, to be considered. Let $c_1 = \mathbf{x}^H \mathbf{w}$. The conjugate derivative of c_1 with respect to the vector \mathbf{w} is

$$\frac{\partial c_1}{\partial \mathbf{w}^*} = \frac{\partial}{\partial \mathbf{w}^*} \mathbf{X}^H \mathbf{W} = \mathbf{0} \quad (\text{B.12})$$

where $\mathbf{0}$ is the null vector. Consider next $c_2 = \mathbf{w}^H \mathbf{x}$. The conjugate derivative of c_2 with respect to \mathbf{w} is

$$\frac{\partial c_2}{\partial \mathbf{w}^*} = \frac{\partial}{\partial \mathbf{w}^*} \mathbf{W}^H \mathbf{X} = \frac{\partial}{\partial \mathbf{w}^*} \mathbf{X}^T \mathbf{W}^* = \mathbf{X} \quad (\text{B.13})$$

Example 2 Consider next the quadratic form

$$c = \mathbf{W}^H \mathbf{R} \mathbf{W} \quad (\text{B.14})$$

where \mathbf{R} is a Hermitian matrix. The conjugate derivative of c (which is real) with respect to \mathbf{w} is

$$\frac{\partial c}{\partial \mathbf{w}} = \frac{\partial}{\partial \mathbf{w}^*} (\mathbf{W}^H \mathbf{R} \mathbf{W}) = \mathbf{R} \mathbf{W} \quad (\text{B.15})$$

CHAPTER 9

References

- [1] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. Viterbi, L. A. Weaver, and C. Wheatly, "On the capacity of a cellular CDMA system," *IEEE Trans. on Veh. Technol.*, vol. 40, no. 2, pp. 303-312, May 1991.
- [2] J. C. Liberti and T. S. Rappaport, "Analytical results for reverse channel performance improvements in CDMA cellular communications systems employing adaptive antennas," *IEEE Trans. on Veh. Technol.*, vol. 43, no. 3, pp. 680-690, August 1994.
- [3] S. C. Swales, M. A. Beach, D. J. Edwards, and J. P. McGeehn, "The performance enhancement of multibeam adaptive base station antennas for cellular land mobile radio systems," *IEEE Trans. on Veh. Technol.*, vol. 39, no. 1, pp. 56-67, February 1990.
- [4] S. U. Pillai, *Array Signal Processing*, Springer-Verlag, New York, 1989.
- [5] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, John Wiley & Sons, New York, 1981.
- [6] Simon Haykin, Ed., *Advances in Spectrum Analysis and Array Processing*, vol.3, Prentice Hall, Englewood Cliff, New Jersey, 1995.
- [7] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine*, pp. 4-24, April 1988.
- [8] Alan V. Oppenheim and Ronald W. Schaffer, *Discrete-Time Signal Processing*, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
- [9] R. T. Compton, Jr., *Adaptive Antennas, Concept and Performance*, Prentice Hall, Englewood Cliffs, New Jersey, 1988.
- [10] Simon Haykin, *Adaptive Filter Theory*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
- [11] B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems," *Proc. IEEE*, pp. 2143-2159, December 1967.

- [12] B. Widrow and Samuel D. Stearns, Adaptive Signal Processing, Prentice Hall, Englewood Cliffs, New Jersey, 1985.
- [13] T. S. Rappaport, Wireless Communications: Principles and Practice, Prentice Hall, Englewood Cliff, New Jersey, 1996.
- [14] R. O. Schmidt, \Multiple emitter location and signal parameter estimation," IEEE Trans. Ant. and Prop., vol. AP-34, no. 3, pp. 276-280, 1986.
- [15] R. Roy and T. Kailath, \ESPRIT{estimation of signal parameters via rotational invariance techniques," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol.37, pp. 984-995, July 1986.
- [16] Y. Wang and J. R. Cruz, \Adaptive antenna arrays for cellular CDMA communication systems," Proc. IEEE ICASSP, pp. 1725-1728, July 1995.
- [17] John R. Treichler and Brian G. Agee, \A new approach to multipath correction of constant modulus signals," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-31, no. 2, pp. 459-471, April 1983.
- [18] John R. Treichler and M. G. Larimore, \New processing techniques based on the constant modulus adaptive algorithm," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-33, no. 2, pp. 420-431, April 1985.
- [19] Takeo Ohgane, \Characteristics of CMA adaptive array for selective fading compensation in digital land mobile radio communications," Electronics and Communications in Japan, Part 1, vol. 74, no. 9, pp. 43-53, 1991.
- [20] M. G. Larimore and J. R. Treichler, \Convergence behavior of the constant modulus algorithm," Proc. IEEE ICASSP, pp. 13-16, April 1983.
- [21] M. G. Larimore and J. R. Treichler, \Noise capture properties of the constant modulus algorithm," Proc. IEEE ICASSP, pp. 30.6.1-30.6.4, April 1985.
- [22] J. O. Smith and B. Friedlander, \Global convergence of the constant modulus algorithm," Proc. IEEE ICASSP, pp. 30.5.1-30.5.4, April 1985.
- [23] B. G. Agee, \Convergent behavior of modulus-restoring adaptive arrays in Gaussian interference environments," Proc. 22nd Asilomar Conf. on Signals, Systems and Computers, pp. 818-822, 1988.
- [24] R. Gooch and J. Lundell, \The CM array: An adaptive beamformer for constant modulus signals," Proc. IEEE ICASSP, vol. 4, pp. 2523-2526, April 1986.

- [25] B. G. Agee, "The least-squares CMA: A new technique for rapid correction of constant modulus signals," Proc. IEEE ICASSP, pp. 19.2.1-19.2.4, 1986.
- [26] C. S. Beightler, D. T. Phillips, and D. J. Wilde, Foundations of Optimizations, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
- [27] Zhigang Rong \ Adaptive Array Algorithms communication \Using adaptive beamforming algorithms.
- [28] B. G. Agee, "Blind separation and capture of communication signals using a multitarget constant modulus beamformer," Proc. IEEE Communication 1989.
- [29] I. Chiba, W. Chujo, and M. Fujise, "Beam-space CMA adaptive array antennas," Electronics and Communications in Japan, Part 1, vol. 78, no. 2, pp. 85-95, 1995.
- [30] I. Chiba, T. Takahashi, and Y. Karasawa, "Transmitting null beam forming with beam space adaptive array antennas," Proc. IEEE Veh. Technol. Conf., pp. 1498-1502, 1994.
- [31] S. Talwar, M. Viberg, and A. Paulraj, "Blind estimation of multiple co-channel digital signals arriving at an antenna array,"
- [32] S. Talwar and A. Paulraj, "Performance analysis of blind digital signal copy algorithms," Proc. IEEE Military Communication Conference, vol. 3, pp. 123-127, 1994.
- [33] U. Grob, A. L. Welti, E. Zollinger, R. Kung, and H. Kaufmann, "Micro-cellular direct sequence spread-spectrum radio system using N-path RAKE receiver," IEEE J. Select. Areas Commun., vol. 8, no. 5, pp. 772-780, June 1990.
- [34] A. Higashi and T. Matsumoto, "Combined adaptive RAKE diversity (ARD) and coding for DPSK DS/CDMA mobile radio," IEEE J. Select. Areas Commun., vol. 11, no. 7, pp. 1076-1084, September 1993.
- [35] B. Suard, A. F. Naguib, G. Xu, and A. Paulraj, "Performance of CDMA mobile communication systems using antenna arrays," Proc. IEEE ICASSP, vol. 4, pp. 153- 156, April 1993.

DECLARATION

I, the undersigned, here by declare that this thesis is my original work, has not been presented for a degree any other university that all sources used for the thesis has been duly acknowledged.

Name : Dereje Hadgu Hailu

Signature : _____

Place : Addis Ababa

Date of submission: _____

This thesis has been submitted for examination with my approval as a university advisor.

Advisor Name

Signature

