In compliance with the
Canadian Privacy Legislation
some supporting forms
may have been removed from
this dissertation.

While these forms may be included
in the document page count,
their removal does not represent
any loss of content from the dissertation.

Blind Adaptive Techniques Used for Smart Antennas in CDMA Systems

By

Mina Labib

A thesis submitted to
The Faculty of Graduate Studies and Research
In partial fulfillment of
The requirements for the degree of
Master of Applied Science (M.A.Sc)

Ottawa-Carleton Institute For Electrical and Computer Engineering
Department of Systems and Computer Engineering
Faculty of Engineering
Carleton University
Ottawa, Ontario
August 2003

© Copyright
Mina Labib, August 2003

National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services

Acquisitions et
services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-88878-9

Our file Notre référence

ISBN: 0-612-88878-9

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Canada

Abstract

The smart antenna system is a promising technique for supporting high traffic demand in the third generation mobile systems. Also, *Direct Sequence Code Division Multiple Access* (DS-CDMA) is considered a strong candidate to support multimedia services in third generation wireless systems as it increases spectrum efficiency. Using Smart Antennas in CDMA wireless systems makes it possible to meet the increasing demand in mobile communication services. In this research, we investigate the performance of the different blind adaptive array algorithms in CDMA systems. Three novel algorithms have been developed, namely the *Sample-based Recursive Least Squares Despread Respread Multi-Target Array* (Sample RLS-DRMTA), the *Affine Projection Despread Respread Multi-Target Array* (AP-DRMTA) and the *Conjugate Gradient Despread Respread Multi-Target Array* (CG-DRMTA). A MATLAB simulation testbed was created to evaluate the performance of these algorithms and compare them with the performance of the *Least Squares Despread Respread Multi-Target Array* (LS-DRMTA), the *Least Squares Despread Respread Multi-Target Constant Modulus Algorithm* (LS-DRMTCMA) and the *Block-based RLS-Despread Respread Multi-Target Array* (Block RLS-DRMTA). Simulations results have shown that the proposed algorithms can outperform both the LS-DRMTA and the LS-DRMTCMA under all conditions and give comparable performance to the Block RLS-DRMTA.

Acknowledgements

The research work reported in this thesis has been carried out at the Department of Systems and Computer Engineering at Carleton University, Ottawa, Canada, from September 2001 to August 2003. I am most grateful to my supervisors, Professor Rafik Goubran and Professor Mohamed El-Tanany. I truly appreciate their belief in me and their invaluable support, encouragement and guidance during the course of the research.

I would like to thank the **Natural Sciences and Engineering Research Council of Canada (NSERC)**, **Canada, Communications and Information Technology Ontario (CITO)**, and **Carleton University** for providing financial support for this research.

I would like to thank Professor Halim Yanikomeroglu, Professor David Falconer, and Professor Ian Marsland for their valuable advices regarding some simulation problems that I have encountered.

I would like to thank my friends in the DSP graduate lab for their helpful discussions, with special thanks to Yasser Mahgoub for so many fruitful discussions. I would also like to thank Danny Lemay for his help providing the necessary facilities to run numerous time-consuming simulations.

Most of all, I would like to thank my parents and my sister for their love and encouragement. I am very grateful for their continuous and immeasurable support.

My deepest gratitude goes to my fiancée Dina Boshra, whose priceless love, caring and patience have brought me here. Her being there waiting for me has made a big difference in my life. I owe all the success I have achieved to her support and encouragement. The Lord has blessed me with her being in my life.

"In God is my salvation and my glory; the rock of my strength,

And my refuge, is in God."

Contents

ABSTRACT	III
ACKNOWLEDGEMENTS	IV
CONTENTS	VI
LIST OF TABLES	X
LIST OF FIGURES	XI
LIST OF ACRONYMS	XIII
LIST OF SYMBOLS	XVI
CHAPTER 1: INTRODUCTION	1
1.1 Motivation.....	1
1.2 Problem Definition and Thesis Contribution.....	2
1.3 Evaluation Criteria	4
1.4 Outline of Thesis.....	5
CHAPTER 2: LITERATURE REVIEW	7
2.1 Smart Antennas: Definition.....	7
2.2 Benefits of Smart Antenna	8
2.2.1 Increasing the Base Station Antenna Gain	8
2.2.2 Robustness to System Perturbation	9
2.2.3 Location Identification	10
2.2.4 Security	11
2.2.5 Improving System Capacity	11
2.3 Challenges to Smart Antennas	12
2.3.1 The Cost	12
2.3.2 Transceiver Complexity	13

2.3.3	Physical Size	13
2.4	Types of Smart Antenna.....	14
2.4.1	Switched Beams.....	14
2.4.2	Adaptive Antennas	16
2.5	Design of Adaptive Antennas	17
2.6	Overview of Beamforming Schemes.....	19
2.7	Overview of Adaptive Algorithms	26
2.7.1	Least Mean Square.....	30
2.7.2	Recursive Least Squares.....	33
2.7.3	Constant Modulus Algorithm.....	35
2.7.4	Affine Projection Algorithm	37
2.7.5	Conjugate Gradient Method.....	39
2.8	Introduction to CDMA Systems.....	41
2.8.1	Key Benefits of CDMA Systems.	42
2.8.2	Structure of CDMA System.....	42
2.8.3	Spreading Code Properties and Generation	43
CHAPTER 3: THE ADAPTIVE BEAMFORMING ALGORITHMS USED IN SIMULATION		45
3.1	Introduction.....	46
3.2	Least Squares Despread Respread Multi-Target Array.....	49
3.2.1	Algorithm Description	49
3.2.2	Analysis.....	50
3.3	Least Squares Despread Respread Multi-Target Constant Modulus Algorithm	53
3.3.1	Algorithm Description	53
3.3.2	Analysis.....	54
3.4	Block Recursive Least Squares Despread Respread Multi-Target Array	55
3.4.1	Algorithm Description	55
3.4.2	Analysis.....	56
3.4.3	The Proposed Sample-based RLS-DRMTA Algorithm.....	58
3.5	The Proposed Affine Projection Despread Respread Multi-Target Array	61
3.5.1	Algorithm Description	61
3.5.2	Analysis.....	61
3.6	The Proposed Conjugate Gradient Despread Respread Multi-Target Array	65
3.6.1	Algorithm Description	65
3.6.2	Analysis.....	65

CHAPTER 4: SIMULATION ENVIRONMENT AND PERFORMANCE RESULTS	68
4.1 Transmitter Model.....	69
4.1.1 BPSK Modulation	69
4.1.2 The Spreading Process	70
4.2 Channel Model	72
4.3 Receiver Model.....	73
4.3.1 Signal Model for Uniform Linear Array	74
4.4 System Parameters and Performance Evaluation Criteria	83
4.5 Simulation Results of the Sample-based RLS-DRMTA	85
4.5.1 Beam Pattern	85
4.5.2 Constellation Diagram	89
4.5.3 Signal Waveform	91
4.6 Simulation Results of the AP-DRMTA	94
4.6.1 Beam Pattern	94
4.6.2 Constellation Diagram	97
4.6.3 Signal Waveform	98
4.7 Simulation Results of the CG-DRMTA	100
4.7.1 Beam Pattern	100
4.7.2 Constellation Diagram	103
4.7.3 Signal Waveform	104
4.8 Graphical User Interface	105
4.9 Summary	105
CHAPTER 5: COMPARISON OF THE PROPOSED AND EXISTING ALGORITHMS	106
5.1 Introduction	106
5.2 Comparison in AWGN channel	106
5.2.1 Convergence Time Evaluation	106
5.2.2 BER Evaluation.....	111
5.3 Comparison in a Time Varying Channel	118
5.3.1 Convergence Time Evaluation	118
5.3.2 BER Evaluation.....	123
5.4 Comparison in terms of Computational Complexity	125

CHAPTER 6: CONCLUSION AND FUTURE WORK.....	127
6.1 Conclusion.....	127
6.2 Future Work.....	129
REFERENCES.....	131
APPENDIX A: THE GUI	144

List of Tables

Table 4-1: System Parameters Used in the Simulation.....	83
Table 5.1: BER Performance in Dynamic Channel (8 users, $E_b/N_o=8$ dB).....	123
Table 5.2: The Computational Complexity for the different algorithms.....	126
Table 6.1: The Recommended Algorithm to Be Used in the Different Conditions.....	128

List of Figures

Figure 1.1: Algorithm Evaluation Methods.....	4
Figure 2.1: Different Types of Smart Antennas.....	14
Figure 2.2: Different Geometries of the Adaptive Antennas.....	18
Figure 2.3: The General Structure of Narrowband Beamformer.....	19
Figure 2.4: Optimum Beamformer Using a Reference Signal.....	24
Figure 2.5: The General Structure of the Adaptive Array.....	26
Figure 3.1: Multi-Target Adaptive Beamformer with M Elements and Q Output Ports...	46
Figure 3.2: Generating the Reference Signal in LS DRMTA Algorithm.....	50
Figure 3.3: Generating the Reference Signal in LS-DRMTCMA Algorithm.....	53
Figure 4.1: A Direct Sequence CDMA Transmitter and Receiver.....	69
Figure 4.2: Illustration of a Plane Wave Incident on a Uniform Linear Array from Direction Angle θ°	74
Figure 4.3: Users' Distribution in Spatial Domain.....	84
Figure 4.4: Beam Pattern for Users 1 and 2 at Angles 0° and 20° Using Sample RLS- DRMTA Algorithm at $E_b/N_0=8$ dB.....	87
Figure 4.5: Beam Pattern for Users 3 and 4 at Angles 40° and 60° Using Sample RLS- DRMTA Algorithm at $E_b/N_0=8$ dB.....	87
Figure 4.6: Beam Pattern for Users 5 and 6 at Angles 80° and 100° Using Sample RLS- DRMTA Algorithm at $E_b/N_0=8$ dB.....	88
Figure 4.7: Beam Pattern for Users 7 and 8 at Angles 120° and 140° Using Sample RLS- DRMTA Algorithm at $E_b/N_0=8$ dB.....	88
Figure 4.8: Constellation Diagram for User 4 at Angle 60° before the Beamforming Processing at $E_b/N_0=8$ dB.....	90
Figure 4.9: Constellation Diagram for User 4 at Angle 60° after the Beamforming Processing Using the Sample RLS-DRMTA Algorithm at $E_b/N_0=8$ dB.....	90
Figure 4.10: Original Bit Waveform.....	92
Figure 4.11: Original Waveform after Spreading.....	92
Figure 4.12: The Corrupted Signal Waveform.....	93
Figure 4.13: The Reconstructed Signal Waveform of User 4 Using Sample RLS-DRMTA Algorithm.....	93
Figure 4.14: Beam Pattern for Users 1 and 2 at Angles 0° and 20° Using APA-DRMTA at $E_b/N_0=8$ dB.....	95
Figure 4.15: Beam Pattern for Users 3 and 4 at Angles 40° and 60° Using APA-DRMTA at $E_b/N_0=8$ dB.....	95
Figure 4.16: Beam Pattern for Users 5 and 6 at Angles 80° and 100° Using APA- DRMTA at $E_b/N_0=8$ dB.....	96
Figure 4.17: Beam Pattern for Users 7 and 8 at Angles 120° and 140° Using APA- DRMTA at $E_b/N_0=8$ dB.....	96

Figure 4.18: Constellation Diagram for User 4 at Angle 60° after the Beamforming Processing Using the AP-DRMTA Algorithm at $E_b/N_0=8\text{db}$	97
Figure 4.19: The Reconstructed Signal Waveform of User 4 Using AP-DRMTA Algorithm.....	99
Figure 4.20: Beam Pattern for Users 1 and 2 at Angles 0° and 20° Using CG-DRMTA at $E_b/N_0=8\text{ dB}$	101
Figure 4.21: Beam Pattern for Users 3 and 4 at Angles 40° and 60° Using CG-DRMTA at $E_b/N_0=8\text{ dB}$	101
Figure 4.22: Beam Pattern for Users 5 and 6 at Angles 80° and 100° Using CG-DRMTA at $E_b/N_0=8\text{ dB}$	102
Figure 4.23: Beam Pattern for Users 7 and 8 at Angles 120° and 140° Using CG-DRMTA at $E_b/N_0=8\text{ dB}$	102
Figure 4.24: Constellation Diagram for User 4 at Angle 60° after the Beamforming Processing Using the CG-DRMTA Algorithm at $E_b/N_0=8\text{ dB}$	103
Figure 4.25: The Reconstructed Signal Waveform for User 4 Using CG-DRMTA Algorithm.....	104
Figure 5.1: Conversion Times of Block-Based Adaptive Algorithm in AWGN Channel.....	108
Figure 5.2: SINR for Sample RLS-DRMTA Adaptive Beamforming Algorithm in AWGN Channel.....	109
Figure 5.3: SINR for AP-DRMTA Adaptive Beamforming Algorithm in AWGN Channel.....	110
Figure 5.4: BER Performance in AWGN with Input $E_b/N_0=8\text{ dB}$	112
Figure 5.5: BER Performance in AWGN with Input $E_b/N_0=4\text{ dB}$	115
Figure 5.6: Conversion Times of Block-Based Adaptive Algorithms in a Dynamic Channel.....	119
Figure 5.7: SINR for Sample RLS-DRMTA Adaptive Beamforming Algorithm in Dynamic Channel.....	121
Figure 5.8: SINR for AP-DRMTA Adaptive Beamforming Algorithm in Dynamic Channel.....	121
Figure A.1: The First Part of the GUI.....	145
Figure A.2: The Second Part of the GUI.....	146
Figure A.3: The Third Part of the GUI at the beginning of the animation.....	147
Figure A.4: The Third Part of the GUI at the end of the animation.....	148

List of Acronyms

APA	Affine Projection Algorithm
AP-DRMTA	Affine Projection Despread Respread Multi-Target Array
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CDMA	Code Division Multiple Access
CG	Conjugate Gradient
CGM	Conjugate Gradient Method
CG-DRMTA	Conjugate Gradient Despread Respread Multi-Target Array
CMA	Constant Modulus Algorithm
DD	Decision Directed
DOA	Direction of Arrival
DPSK	Differential Phase Shift Keying
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
FCC	Federal Communication Commission

FDMA	Frequency Division Multiple Access
FIR	Finite Impulse Response
FSK	Frequency Shift Keying
GSO	Gram-Schmidt Orthogonalization
GUI	Graphical User Interface
LCMV	Linearly Constrained Minimum Variance
LMS	Least Mean Square
LS-CMA	Least Squares Constant Modulus Algorithm
LS-DRMTA	Least Squares Despread Respread Multi-Target Array
LS-DRMTCMA	Least Squares Despread Respread Multi-Target Constant Modulus Algorithm
ML	Maximum Likelihood
MMSE	Minimum Mean Square Error
MT-LSCMA	Multi-Target Least Squares Constant Modulus Algorithm
MUSIC	Multiple Signal Classification
MV	Minimum Variance
MVDR	Minimum Variance Distortionless Response
NLMS	Normalized Least Mean Square
PN	Pseudorandom Noise

PSK	Phase Shift Keying
QPSK	Quadrature Phase Shift Keying
RAKE	
RLS	Recursive Least Squares
Sample RLS-DRMTA	Sample-based Recursive Least Squares Despread Respread Multi-Target Array
SCORE	Spectral Self-Coherence Restoral
SD	Steepest Descent
SD-CMA	Steepest Descent Constant Modulus Algorithm
SDMA	Space Division Multiple Access
SINR	Signal to Interference and Noise Ratio
SNR	Signal to Noise Ratio
TDMA	Time Division Multiple Access
ULA	Uniform Linear Array

List of Symbols

P_r	Received power
P_t	Transmitted power
G_s	Gain of the subscriber's handset antenna
G_b	Gain of the base station
PL	Path loss
d_r	The distance between the transmitter and the receiver.
d_o	The close-in reference distance which is determined from measurements.
$\overline{PL}(d_o)$	The ensemble average scale path loss at distance d_o
n_o	The path loss exponent
X_σ	A zero mean Gaussian distributed random variable (in dB) with standard deviation σ
ω	The weight vector in the beamformer.
λ	The wavelength of the signal
$y(t)$	Output of the beamformer
$x(t)$	Input data to the antenna elements
$r(t)$	The reference signal

$e(t)$	The error signal which is the difference between the reference and the output signal.
M	The number of the antenna elements.
$[.]^T$	Denotes the transpose of the vector.
$[.]^H$	Denotes the Hermitian transpose of the vector.
$[.]^*$	Denotes the conjugate of the value $[.]$.
d	The distance between the elements in the antenna array
i	index for the user
θ_i	Direction of Arrival (DOA) angle for the signal coming from user i
T_s	Sampling period in the FIR filter
f	Temporal frequency of the FIR filter.
$y(k)$	Denotes the sample signal of $y(t)$ at time instant t_k
$x(k)$	Denotes the sample signal of $y(t)$ at time instant t_k
$r(k)$	Denotes the sample signal of $r(t)$ at time instant t_k
$e(k)$	Denotes the sample signal of $e(t)$ at time instant t_k
$P(\omega)$	Mean Output power of the beamformer using the weight vector ω
$E[.]$	The expectation operator of $[.]$.
$\text{sgn}[.]$	The sign of $[.]$.
R	The correlation matrix for the input data

z signal	The cross correlation between the reference signal and the input
μ	Step size.
μ_0	Positive constant in NLMS algorithm.
$J(k)$	The cost function
k	Iteration index
$P_{RLS}(k)$	Inverse correlation matrix in RLS algorithm
$k_{RLS}(k)$	Gain vector in RLS algorithm
δ_{RLS}	A small positive constant in RLS algorithm.
λ	The forgetting factor
I	Identity matrix
L	Number of snapshots of the input data vector to form the block-based algorithms.
N_{conv}	The number of iterations required for the block-based algorithm to converge
p	The projection order in APA algorithm.
δ	Small positive number that serves as the regularization parameter in computing the inverse of the signal autocorrelation matrix in the APA algorithm.
$g_{APA}(k)$	Prefilter vector in APA algorithm.
$\alpha(k)$	The step size that minimizes the cost function $J(\omega)$

η	A positive number ($(\lambda-0.5)< \eta < \lambda$) in the CG algorithm.
$p(k)$	The Direction vector in the CG algorithm.
$g(k)$	Residual vector in the CG algorithm.
$\beta(k)$	The estimated gradient in the CG algorithm.
$c(t)$	The spreading waveform.
$b(t)$	The binary data signal composed of a sequence of symbols b_k .
K	The data block size.
R_s	Sampling rate.
N_s	The number of samples in the chip.
R_c	Chip rate in CDMA.
N	The processing gain.
n	The bit index
$\hat{b}_{i,n}$	The estimated n^{th} bit for user i .
r_{PN}	The reference signal generated from utilizing the PN sequences in the LS-DRMTCMA algorithm
r_{CM}	The reference signal generated from utilizing the constant modulus feature in the LS-DRMTCMA algorithm.
a_{PN}	A positive value less than 1 to weight r_{PN} in the LS-DRMTCMA algorithm.
h	sample index.

s	The block size in the AP-DRMTA algorithm.
j	The iteration index in the AP-DRMTA algorithm.
Z	The autocorrelation matrix for the block data in the AP-DRMTA algorithm.
P_0	The power of the signal.
f_c	Carrier frequency.
b_n	n^{th} data bit.
Ψ_T	A unit rectangular pulse of duration T .
T_b	Bit period.
$c(t)$	The spreading waveform.
c_k	The spreading code and $\in \{+1, -1\}$
T_c	The chip period.
v_i	The CDMA signal of user i .
P_i	Signal power of user i .
$\tilde{u}_{im}(t)$	The pass band signal arriving at the m^{th} element of the antenna array is from user i .
$u_{im}(t)$	The base band representation of the signal arriving at the m^{th} element of the antenna array is from user i .
τ_i	The time delay for user i
Φ_i	The random phase for the signal of user i .

c	Speed of light
m	Index for the antenna element.
β	The phase propagation factor.
$\mathbf{a}(\theta)$	The steering vector.
q	Number of narrow band signals arriving at the antenna array.
$\mathbf{n}(t)$	Noise vector.
$G(\theta)$	The antenna beam pattern
$g(\theta)$	Normalized beamformer response also known as the array factor
θ_{BW}	Null-to null bandwidth of the main lobe of the uniform linear array
Δf	The pass band width Δf of the FIR filter
$\Delta\theta$	The beam width of the main lobe.

Chapter 1: Introduction

1.1 Motivation

The main challenge of the next generation in mobile communications is to offer high speed wireless access and wireless multimedia services. It is desired to combine both voice and data services such as multimedia, circuit and packet data services and high speed Internet access. Providing these services to the users requires high data rates and acceptable *Signal to Interference and Noise Ratio* (SINR) at the required data rate. Another important factor that must be taken into consideration is the capacity, which is increasing rapidly nowadays. The number of users is expected to grow even more rapidly with the services that will be offered in the third generation, while the available radio frequency spectrum is limited. We need to look for innovative solutions to meet this expected growth in demand.

The smart antenna has a key role in the next generation of mobile communications. It is one of the most promising technologies used for reducing the co-channel interference and hence increasing the capacity of wireless cellular systems. The basic concept of the smart antenna is to deliver high gain in the direction of the desired user and to null out the interferers. In general, the smart antenna reduces the effect of multipath fading in wireless communications environments, improves the SINR and hence increases the system capacity.

Another approach to increase spectrum efficiency is to use the DS-CDMA as the multiple access scheme. CDMA is in a state of interim standard for third generation wireless communications that enables high speed data and multimedia data transfer.

Combining Smart antenna technology with CDMA will lead to great improvement in the system. A CDMA system is interference limited in nature. All users operate in the same frequency band simultaneously using different codes. Thus, by tracking the user and directing the beam towards this user, as well as nulling out any other interferer, we can improve the capacity of the system.

1.2 Problem Definition and Thesis Contribution

An important research area in the field of smart antenna technology is the area of adaptive beamforming algorithms, which aims at finding the optimum weight vector that enables the antenna to deliver high gain towards the desired user and to null the interferers. Several adaptive algorithms have been discussed in the open literature for use in smart antennas. In the case of CDMA, where all the users occupy the same frequency band, the adaptation problem is generally more complex due to the presence of multi-user interference. It is also preferred that the adaptive algorithm be blind (i.e. work without a training sequence). The LS-DRMTA algorithm and the LS-DRMTCMA algorithm, proposed in literature, do not have a satisfactory performance in terms of the *Bit Error Rate* (BER) and the SINR and eventually do not perform well in the improvement of

capacity. Also, the existing Block RLS-DRMTA algorithm only achieves good performance when the channel is not a time varying channel. Furthermore, the performance of this algorithm depends heavily on the level of the cross correlation between the different codes used in the CDMA systems. Therefore, it is desirable to develop novel adaptive algorithms to overcome the drawbacks of the existing adaptive algorithms.

The objective of this thesis is to develop novel adaptive algorithms for smart antennas used in CDMA systems and to compare the performance of the proposed algorithms with those of the algorithms presented in literature.

Three novel algorithms have been developed, namely the *Sample-based Recursive Least Squares Despread Respread Multi-Target Array* (Sample RLS-DRMTA), the *Affine Projection Despread Respread Multi-Target Array* (AP-DRMTA) and the *Conjugate Gradient Despread Respread Multi-Target Array* (CG-DRMTA).

Simulations results have shown that the proposed algorithms can outperform both the LS-DRMTA and the LS-DRMTCMA under all conditions and give comparable performance to the Block RLS-DRMTA.

1.3 Evaluation Criteria

In order to evaluate the performance of the proposed algorithms, we will examine the performance of each of the proposed algorithms individually, in terms of the antenna beam pattern generated for each user, the constellation diagram before and after using the beamformer suggested by the algorithm and the shape of the signal waveform before and after using the beamformer suggested by the algorithm.

We will then compare the performance in terms of convergence time and BER on using the proposed algorithms with the performance on using other algorithms presented in literature. Figure 1.1 illustrates how we evaluate each of the proposed algorithms.

Figure 1.1 Algorithm Evaluation Methods

1.4 Outline of Thesis

This thesis is organized as follows:

Chapter 2 introduces the fundamentals of the adaptive array, the benefits of using adaptive antenna arrays and the different types of adaptive array antenna. It gives also a global overview of the beamforming schemes and the adaptive techniques that have been used in smart antennas.

Chapter 3 presents the different multi-target adaptive blind beamforming algorithms used in our simulation. First, we introduce the LS-DRMTA algorithm, the LS-DRMTCMA algorithm and the Block RLS-DRMTA algorithm. Next, we introduce the proposed algorithms, the Sample RLS-DRMTA algorithm, the AP-DRMTA algorithm and the CG-DRMTA algorithm. A complete description and analysis for each algorithm is presented.

Chapter 4 presents the simulation environment. A description of the CDMA system is presented, then the transmitter, channel and receiver model are presented. Next, we evaluate the performance of each of the proposed algorithms by presenting the simulation results that show firstly the antenna pattern, then the constellation diagram and finally the shape of the reconstructed waveform.

Chapter 5 presents a comparison between the performance of all the different algorithms discussed in Chapter 3, in terms of the convergence time and the BER, based on the simulation results.

Chapter 6 presents a brief summary and conclusion along with suggested future work.

Chapter 2: Literature Review

In this chapter, we will present a global overview of the smart antennas, the beamforming techniques, the adaptive algorithms used in smart antennas and the CDMA systems.

2.1 Smart Antennas: Definition

Smart antenna system (or simply a smart antenna) can be defined as a spatial diversity processor [1]. It consists of a number of independent antenna elements. The plane for these antenna elements can take any form, usually either linear or circular or rectangular. The distance between these elements may or may not be equal. The objective of the smart antenna is to form a beam that delivers maximum gain to the desired user and to suppress the interfering signals. Many techniques used in smart antenna for beamforming have been developed from adaptive filter theory.

There are several terms used to refer to the various aspects of smart antennas technology, such as intelligent antennas, phased arrays, spatial processing, digital beamforming and adaptive antenna systems [4]. Smart antenna systems are usually categorized into two types based on how they produce their response: switched beam systems and adaptive antennas.

2.2 Benefits of Smart Antenna

2.2.1 Increasing the Base Station Antenna Gain

Smart antennas provide greater gain at the base station, which leads to the following benefits:

- A. **Range Extension:** Smart antennas increase the gain of the base station, which leads to range extension [3].

$$P_r = P_t + G_s + G_b - PL \quad (2.1)$$

Where P_r is the received power, P_t is the transmitted power, G_s is the gain of the subscriber's handset antenna, G_b is the gain of the base station and PL is the path loss, where

$$PL = \overline{PL}(d_o) + 10n_o \log\left(\frac{d_r}{d_o}\right) + X_\sigma \quad (2.2)$$

Where d_r is the distance between the transmitter and the receiver, $\overline{PL}(d_o)$ is the ensemble average scale path loss at distance d_o , d_o is the close-in reference distance which is determined from measurements, n_o is the path loss exponent and X_σ is a zero mean Gaussian distributed random variable (in dB) with standard deviation σ .

From (2.2), we can note that by increasing the gain of the base station, the same received power can be achieved at a longer distance d_r . Since smart antennas allow greater gain compared to conventional antennas, smart antenna systems can provide range extension. This allows reducing the number of base stations, especially in rural areas, which is what the service providers aim to do.

- B. Better Building Penetration:** Clearly, by increasing the antenna gain of the base station, building penetration is improved.
- C. Reducing the Transmitted Power:** By increasing the power gain of the receiving antenna at the base station, it is clear from Equation (2.1) that the transmitted power from the mobile terminal can be reduced. This is very important. Firstly, this fulfils the health requirements that demand reducing the transmitted power from handsets and secondly, it eases demands on the battery of the mobile terminal.

2.2.2 Robustness to System Perturbation

A main problem in CDMA systems is the Near Far Problem, which requires a sophisticated power control process to ensure that all the signals arriving at the base station are approximately of the same power level. Smart antennas help to isolate the uplink signals for different users, which leads to reducing the power control requirements or mitigating the impact of imperfect power control [3]. Also, as the location and the

speed of the mobile user information become available, deciding, to which cell to hand over the user to, becomes much easier.

2.2.3 Location Identification

A very important feature of using smart antenna is identifying the user's location. This new feature provides a lot of benefits, such as:

- A. **Emergency Calls Support:** The recent mandate by the *Federal Communication Commission* (FCC) (which is called "phase 2 E911 requirements" (E911=Enhanced-911 technology)) requires specific accuracy and reliability in determining the position of the mobile user. The accuracy required for handset-based solutions is to be able to identify the location of the calling user within 50 meters radius in at least 67 percent of emergency calls, and within 150 meters radius in at least 95 percent of emergency calls [5]. There are many *Direction Of Arrival* (DOA) algorithms used to identify the location of the user. Most of these algorithms are based on smart antenna techniques.
- B. **Location of Fraud Perpetrators:** A significant portion of a service provider's revenue is lost every year due to fraud. Some solutions have been suggested, such as deactivating the phone number from the switch, which is not enough. However, by identifying the location of the phone, it is so easy to solve this problem [4].

- C. **Location Sensitive Billing:** Traffic statistics can be generated based on the location in addition to the usual time usage statistics. This will enable the service providers to increase their revenue. For example, the service provider can give the user a better call price if the user is in a preferred zone [4].
- D. **On Demand Location Specific Services:** These include real-time traffic updates, tourist information and electronic yellow pages [4].

2.2.4 Security

Smart antennas offer better security to the system because it is more difficult to tap a connection when smart antennas are used. The intruder must be in the same direction as the user, as seen from the base station, to be able to tap a connection [6].

2.2.5 Improving System Capacity

Capacity is a very important issue in mobile communication. Improving the capacity in CDMA systems can be done through three approaches:

- A. Smart antennas refocus coverage patterns to deal with hot spots. Statistics derived from commercial networks showed that usually the three sectors in the cell are not equally loaded. One sector is usually overloaded (reaches 140% of what it should carry), while the other two sectors have free channels (98% and 65%). By using smart antenna, the traffic in each cell can be reloaded [7].

- B. Smart antennas can be used to separate signals spatially. This allows different subscribers to share the same spectral resources, provided that they are spatially separable at the base station. This is called *Space Division Multiple Access* (SDMA). SDMA allows multiple users to operate in the same cell, on the same channel slot provided (whether time, frequency or code) and uses smart antennas to separate the signals [3]. Using SDMA will lead to an increase in the overall capacity of the system [8], [9].
- C. CDMA systems are interference limited. Smart antennas are used to allow subscribers to transmit less power for each link. Thus, the Multiple Access Interference is reduced, which allows increasing the number of simultaneous subscribers that can be supported in each cell. More details can be found in [10] and [11].

2.3 Challenges to Smart Antennas

2.3.1 The Cost

It is so obvious that the cost of smart antennas is much greater than that of the traditional base station antennas. On the other hand, one of the benefits of smart antennas is that it allows range extension in rural areas and thus reduces the number of base stations. Thus, despite the high cost of smart antenna systems in the short term, they will prove cheaper in the long term [12].

2.3.2 *Transceiver Complexity*

It is obvious that a smart antenna transceiver is much more complex than a traditional base station transceiver. Each of the antenna elements needs a separate transceiver chain (down converter and analog to digital converter). In addition, the beamforming algorithms used are usually computationally intensive, which means that there must be a very powerful numeric processor and control system at the base station that uses smart antennas [6].

2.3.3 *Physical Size*

For the antenna array to work in an efficient way, it needs from 6 to 10 antenna array elements with the space between them about half the wavelength of the signal [3]. This makes it applicable to use smart antennas at the base station but difficult to use them at phone handsets. Still, smart antennas can be implemented in case of Wireless LAN systems, where the mobile station is a notebook computer [13].

2.4 Types of Smart Antenna

There are mainly two types of smart antenna, switched beams and adaptive antennas [2], [4], [14], [15]. Both of them aim at delivering higher gain towards the desired user.

Figure 2.1 shows the difference between those two types.

Figure 2.1 Different Types of Antenna

2.4.1 Switched Beams

This system can be considered as an extension to the current cellular sectorization technique, where the cell is divided into 3 sectors; each of them is 120 degrees. The switched beams method further subdivides the sector into several micro-sectors. Each micro-sector has a predetermined fixed beam pattern. This means that this method

establishes a certain number of choices of beam patterns at the time of deployment. During the call, the base station selects the best beam pattern for the user.

As the switched beams method is a normal extension to the deployed technology, this makes it easy to integrate into the existing systems. The switched beams method can increase the range of coverage of the base station from 20% to 200%, depending on the environment [2].

The limitation of the switched beams method stems from the fact that the beam pattern is fixed. Although this method can suppress interference arriving from directions away from the center of the active beam, if the interfering signal is coming from approximately the direction of the center of the selected beam (may be from another cell) and the user is away from the center of the selected beam, the interfering signal can be enhanced far more than the desired signal. In these cases, the quality is degraded for the user. Also, because the beam pattern is predetermined, sensitivity can occasionally vary as the user moves through the sector [2]. In fact, switched beams work best in minimal to moderate co-channel interference environments. CDMA systems are interference limited, so the switched beams systems are not the best choice for CDMA systems.

2.4.2 *Adaptive Antennas*

In this approach, the user in the cell is tracked with an individual beam pattern and the beam pattern keeps updating as the user moves in the cell and also nulls out the interfering signals. Adaptive antennas utilize sophisticated signal processing algorithms to keep tracking the desired signal and distinguish it from multipath and interfering signals. The ability to track the users with main lobes and interferers with nulls maximizes the link budget.

The adaptive antenna concept is like the hearing system in humans. If a person listens to another while the listener's eyes are closed, the listener's brain collects the voice from both ears and can decide the direction from which the speaker is talking. The hearing system and the brain can keep tracking the speaker as he moves, focus on the speaker's voice and filter out any unwanted noise [2].

Adaptive antennas can track the user as he moves and can also track the interferers. Thus, adaptive antennas provide better capability to suppress interferences than switched beams systems and increase the capacity. Also, there are many new services that can be provided when using the adaptive antennas, such as identifying the location of the user, which uses the DOA algorithms based on adaptive antennas.

Due to the advantages of adaptive antennas over switched beams systems, many studies have focused on adaptive antennas. Adaptive antennas will be the focus of this research.

2.5 Design of Adaptive Antennas

The performance of the adaptive antenna depends heavily on the antenna design. In the case of identical array elements, there are two main variables that can be used to control the beam pattern of the antenna array, the size of the array and the geometry of the array.

- **The Size:** The gain of the antenna array depends on the size or the aperture of the array. The gain increases as the size increases. Two factors control the size: the number of antenna elements and the distance between the elements. Concerning the number of elements, it is clear that increasing the number of elements increases the gain. But as the RF hardware associated with each of the elements is expensive, a practical number for antenna elements must be used. Usually, the antenna array consists of six to ten elements [3]. Concerning the spacing between the antenna elements, it has been shown in [16] that increasing the distance between the elements above half the wavelength of the input signal will result in grating lobes, giving the antenna pattern undesired beams, which may lead to amplifying the interference [3]. Also, as the distance between the elements

decreases, the mutual coupling between the elements increases. So it is better to keep the spacing between the elements as $(\lambda/2)$ [17].

- **The Geometry:** The antenna elements can be arranged in different ways: linear, circular and any two dimensional grid. Figure 2.2 shows some examples of the geometries that can be used.

Linear array

Circular array

Square frame array

Triangular array

Figure 2.2: Different Geometries of the Adaptive Antennas.

In practical mobile communication systems, the most frequently used adaptive antenna array is the Uniform Linear Array (ULA) [18]. ULA consists of identical elements arranged in linear form having equal inter-element spacing. The inter-element spacing is equal to $(\lambda/2)$. The ULA is frequently used because it is simple to install and because the practical geometry of the base station masts makes it either difficult or expensive to use different geometries other than the ULA [18].

2.6 Overview of Beamforming Schemes

The term “beamformer” is derived from the fact that the early forms of antenna arrays (spatial filters) have tried to form a pencil beam in order to deliver high gain towards the desired user and to null out the interferers. The techniques that will be explained in this section have been referred to as *one-shot beamforming*. This simply means that the antenna pattern is determined using a single operation [18].

Figure 2.3: The General Structure of Narrowband Beamformer.

Consider a narrow band beamformer, as shown in Figure 2.3, where the input signal at each of the antenna elements is multiplied by a complex weight and summed to form the antenna output. It follows from Figure 2.3 that the antenna output [19] is given by:

$$y(t) = \sum_{m=1}^M \omega_m^* x_m(t) \quad (2.3)$$

where the superscript * denotes the complex conjugate.

Denoting the complex weight in a vector form as

$$\omega = [\omega_1, \omega_2, \dots, \omega_M]^T \quad (2.4)$$

and the input signals at the antenna element as

$$\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_M(t)]^T \quad (2.5)$$

then the output of the array will become

$$y(t) = \mathbf{w}^H \mathbf{x}(t) \quad (2.6)$$

where the superscripts T and H denote the transpose and the Hermitian (complex conjugate) transpose of the vector, respectively.

It was shown in [20] that there is a correspondence between the Finite Impulse Response (FIR) filter and the beamformer. As the FIR filter linearly combines the temporally sampled data, the beamformer linearly combines the spatial samples from each array element to obtain the output of the array. Assuming the distance between the elements is d and the DOA angle for the signal coming from user i is θ_i , then the normalized element spacing, d/λ , can be mapped into the sampling period, T_s in the FIR filter, and $\cos \theta_i$ can be mapped into the temporal frequency f of the FIR filter.

There are different ways to classify the beamformer; one of them is to classify the beamformers into *narrowband beamformer* and *wideband beamformer*. The narrowband beamformer samples the input signal in space only, as shown in Figure 2.3 and by Equation 2.3.

The wideband beamformer samples the input signal in both space and time and is used when the input signal is of wideband compared to the carrier frequency [20].

The beamformers can also be classified according to how the weights are chosen into *data independent* or *statistically optimum beamformers* [20]. The weights in the data independent beamformer do not depend on the data at the array elements. The weights are chosen to form a specified beam pattern for all signal/interference scenarios. The weights in the statistically optimum beamformer are chosen based on the statistics of the data received at the array. The simplest form of the data independent beamformer is the conventional beamformer, which is also known as delay and sum beamformer. The weights here are all of the same magnitude. The phases are selected in order to steer the beam pattern in a specific direction, known as the look direction, which is the direction of the desired user. The main disadvantage of this delay and sum beamformer is that it does not place nulls in the directions of the interferers. Thus, it is not effective in the presence of directional jammers [19].

Another type of data independent beamformer is the null-steering beamformer, which is used to cancel a plane wave coming from a specific direction and thus produce nulls in the direction of interferers. The main disadvantage of the null steering beamformer is that it requires knowing the direction of the interference sources. Also, this beamformer does not maximize the output *Signal to Noise Ratio* (SNR) [19].

As for the statistically optimum beamformer, the weights are chosen based on the statistics of the input signal at each of the elements. The goal when choosing the weights is to optimize the array performance response based on one of the optimization criteria. These criteria are expressed as cost functions, which are inversely associated with the quality of the signal. By minimizing the cost function, the array performance response is optimized. The most popular optimization criteria are *Maximum Signal to Noise and Interference Ratio (SNIR)*, *Maximum Likelihood (ML)* and squared function -based criteria such as *Minimum Variance (MV)* and *Minimum Mean Square Error (MMSE)* [21].

- **Maximum SNIR:** The weights are selected so as to maximize the ratio of the power in the desired signal component to the power in the interference plus the noise components at the output of the array. The main disadvantage of this method is that it requires the knowledge of the DOA of the desired signal and the covariance matrix of the noise components [3].
- **Maximum Likelihood (ML):** The weights are selected so as to maximize the likelihood function, which is the joint probability density function of observations [16]. The ML principle attempts to estimate the data sequence that has been most likely sent, based on the observed data. The ML technique is computationally intensive [3] and that makes this method less attractive in smart antenna receivers, especially in the presence of co-channel interference with delay spread [21].

- **Linearly Constrained Minimum Variance (LCMV):** The weights are selected so as to minimize the variance at the output of the array, subject to linear constraints. In the case of single constraint, this is equivalent to forcing the beam pattern to be constant in the look direction (direction of the desired user). The Minimum Variance Distortionless Response (MVDR) beamformer is a special case of the LCMV beamformer [3], where the weights are chosen by minimizing the mean output power of the beamformer while maintaining a unity response in the look direction, which leads to maximizing the output SINR [22]. The MVDR beamformer is one of the statistically optimum beamformers that uses in its structure the data independent beamformer. The MVDR beamformer is also known as Capon's minimum variance technique [23] and as optimal combiner [19]. The main disadvantage of the MVDR technique is that it requires the knowledge of the DOA of the desired signal [3].
- **MMSE:** The weights are selected so as to minimize the mean square value of the error signal, where the error signal is the difference between the output of the array and some reference signal. The reference signal can be generated either by using a training sequence or by using decision direction algorithms [24]. The main advantage of this technique is that it does not require the knowledge of the DOA of the desired user, which makes it one of the most popular and the most widely used techniques [16]. MSE optimization criteria are the most attractive for smart antenna receivers and will be the optimization criteria used in this research.

The beamformer that uses the MSE as optimization criteria is known as the optimum beamformer with reference signal [19] and the resulting solution is known as Wiener filter. The structure of this beamformer is shown in Figure 2.4 [19].

Figure 2.4: Optimum Beamformer Using a Reference Signal.

Let $y(k)$, $x(k)$, $r(k)$ and $e(k)$ denote the sample signal of $y(t)$, $x(t)$, $r(t)$ and $e(t)$ at time instant t_k , respectively. Then the output of the beamformer $y(k)$ is given by

$$y(k) = \omega^H x(k) \quad (2.7)$$

If the components of $x(k)$ can be modeled as zero mean stationary processes, then the mean output power [19] is given by

$$P(\omega) = E[y(k)y^*(k)] = \omega^H \mathbf{R} \omega \quad (2.8)$$

where $E[\cdot]$ denotes the expectation operator and $\mathbf{R}$ is the correlation matrix for the input data given by

$$\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^H(k)] \quad (2.9)$$

Weights are selected to minimize the MSE, which is given by

$$MSE = E[|e(k)|^2] = E[|r(k)|^2] + \omega^H \mathbf{R} \omega - 2\omega^H \mathbf{z} \quad (2.10)$$

where $\mathbf{z}$ is the correlation between the reference signal and the input signal and is given by

$$\mathbf{z} = E[\mathbf{x}(k)r(k)] \quad (2.11)$$

The MSE surface is the quadratic function of ω . To get the minimum MSE, set the gradient of MSE with respect to ω to zero. This leads to the famous Wiener-Hopf equation

$$\mathbf{R}\omega_{opt} = \mathbf{z} \quad (2.12)$$

Solving (2.12) directly to get the optimum ω leads to

$$\omega_{opt} = \mathbf{R}^{-1}\mathbf{z} \quad (2.13)$$

The optimum weight given by (2.13) is called Wiener solution [19]. Equation (2.13) has a solution provided that $\mathbf{R}$ is non-singular, which will be achieved as long as the noise component at the array input has non-zero variance [3].

2.7 Overview of Adaptive Algorithms

The optimum weight vector in (2.13) requires the knowledge of the correlation matrix $\mathbf{R}$ of the input signal and requires also the knowledge of the cross correlation vector $\mathbf{z}$ between the input data and the reference signal. Both of them are usually unknown. In addition, the solution requires the calculation of the inverse matrix of the correlation matrix $\mathbf{R}$, which is computationally intensive. Furthermore, the correlation matrix $\mathbf{R}$ is a time varying matrix due to the changes in the environment (either the user is moving or movements in the surroundings). This motivates the need to use an adaptive algorithm to get the optimum weight vector [20]. The general structure of the adaptive algorithm used in smart antennas is shown in Figure 2.5.

Figure 2.5: The General Structure of the Adaptive Array

The adaptive algorithm is used to obtain the optimum weight vector in a recursive manner. The algorithm starts with a predetermined set of *initial conditions*. In a stationary environment and after a certain number of iterations, the algorithm *converges*

to the Wiener solution. In a non-stationary environment, the algorithm tries to *track* the variation in the statistics of the input data [24].

Since there is a mapping between the narrow band beamformer and the FIR filter, most of the algorithms in the literature for adaptive filters can be used for antenna arrays. The choice of one algorithm over the other is determined by one or more of the following factors [24]:

- **Rate of Convergence:** This is the number of iterations required by the algorithm to converge close enough to the Wiener solution in response to stationary inputs.
- **Misadjustment:** After the algorithm converges, this parameter provides a quantitative measure of the amount by which the final value of the mean square error, averaged over an ensemble of the adaptive filter, deviates from the mean square error of the Wiener solution [25].
- **Tracking:** This is the ability of the algorithm to track the statistical variations in the non-stationary environment.
- **Robustness:** For the algorithm to be robust, it should give small estimation error, in case the input data are disturbances with small energy.
- **Computational Requirements:** This factor includes

1. The number of operations (i.e., multiplications, division and addition) required to make one complete iteration of the algorithm.
2. The size of memory required to store the data and the program.
3. The investment required to program the algorithm on a computer or a DSP processor.

There are mainly two different types of adaptive algorithms: the non-blind adaptive algorithms and the blind adaptive algorithms [18].

- **Non-Blind Adaptive Algorithms:** A training sequence is used in the adaptation process. Examples of this type of algorithm are the Least Mean Square (LMS) algorithm and the Recursive Least Squares (RLS) algorithm.
- **Blind Adaptive Algorithms:** Blind adaptive algorithms do not require training sequences. Instead, they exploit some known properties of the desired signal. The blind adaptive algorithms can be categorized into:[26]
 - **Algorithms Based on DOA Estimation:** In the *Direction of Arrival* (DOA) algorithms, the DOA of the signal is first estimated using prior knowledge of the array response (i.e., the array manifold or special array structure). The optimum beamformer is then constructed so as to deliver high gain towards the desired user and to null out the interferers. Examples of this type of

algorithm are the *Multiple Signal Classification* (MUSIC) algorithm [27] and the *Estimation of Signal Parameters via Rotational Invariance Techniques* (ESPRIT) algorithm [28]. One drawback to these techniques is that the performance depends heavily on the prior spatial information (the array manifold). In many practical situations the spatial information is either not available or unreliable. Another drawback to these techniques is that the number of the DOAs estimations is limited by the number of antenna elements. This makes these techniques have very limited use in practical communication systems, especially in multi-path CDMA systems, where the number of users' signals exceeds by far the number of antenna elements [18].

- Algorithms Based on Property-Restoral Techniques: Digital communication signals possess unique properties. The adaptive algorithm can make use of these properties in the adaptation process by trying to restore them after the signal has been corrupted by noise and fading through the mobile channel. One algorithm that uses this technique is the *Spectral Self-Coherence Restoral* (SCORE) algorithm. The SCORE algorithm tries to restore the spectral self-coherence property. This property exists due to the correlation between the signal and certain frequency-shifted versions of the signal itself. The correlation exists because of periodicities in the second order statistics of the signal [29]. Another algorithm that uses this technique is the *Constant*

Modulus Algorithm (CMA). The CMA uses the constant modulus nature of the signal to obtain the optimum weight vector [30] [31].

- o Decision Directed (DD) Algorithms: In these algorithms, the beamformer uses the demodulated signal as a reference signal to update the weight vectors. When the error rate is small, the adaptive beamformer is able to improve the weight vector using a feedback control loop. These algorithms need a low error rate or relatively high SNR to perform well [18]. An algorithm was proposed in [32] to allow soft transition between the CMA and the DD algorithm for *Phase Shift Keying* (PSK) modulation.

In the following sections we will review the basic adaptive algorithms that will be used though the rest of this thesis.

2.7.1 Least Mean Square

Least Mean Square (LMS) is the simplest adaptive algorithm [25]. It is capable of achieving satisfactory performance under the right conditions. That is why it is very widely used [24]. The optimization criterion used in the LMS algorithm is the minimum mean square error criterion. The mean square error can be visualized as a “bowl-shaped surface” [16]. This bowl is referred to as the *error-performance surface* [24]. The adaptive processor has the task of seeking the “bottom of the bowl” to reach the MMSE. One way to do so is the “hill climbing” approach that is used in the *steepest descent*

method. In this approach, the weight vector is updated in the direction of the negative of the gradient vector of the error-performance surface. This should eventually lead to the minimum mean square error [26]. The LMS algorithm is like the steepest descent method except that updates to the weight vector are made in the negative direction given by an estimated gradient vector rather than the actual gradient vector [16].

The LMS algorithm can be described by the following three equations [19] (Figure 2.5)

$$y(k) = \omega^H \mathbf{x}(k) \quad (2.14)$$

$$e(k) = r(k) - y(k) \quad (2.15)$$

$$\omega(k+1) = \omega(k) + \mu \mathbf{x}(k) e^*(k) \quad (2.16)$$

where $\omega(k+1)$ denotes the new weights computed at $(k+1)$ iteration and μ is a positive real-valued parameter, called the step size. The step size controls the convergence characteristics of the algorithm [19].

The LMS is a member of the *stochastic gradient* family of adaptive algorithms, since the instantaneous estimate of the gradient vector is a random vector that depends on the input data vector [26].

The main advantage of the LMS algorithm is its simplicity, as it requires $2M$ complex multiplications per iteration, where M is the number of antenna elements [20]. Also in a

non-stationary environment, the LMS algorithm can *track* the bottom of the error-performance surface provided that the input data vary slowly compared to the learning rate of the LMS algorithm [24]. The main limitation of the LMS algorithm is that its convergence characteristics depend on the eigen structure of the correlation matrix of the input data vector. When the eigen values are widely spread, convergence can be slow [20].

There are several modifications that have been proposed to improve the convergence speed of the LMS algorithm. In [33], Naguma and Noda proposed the *Learning Identification* algorithm as a modification of the LMS. It is known as the Normalized LMS (NLMS) [34]. It was shown in [35] and [36] that NLMS converges faster than the LMS does.

In the NLMS algorithm, the constant step size has been replaced by a data dependent step size. This means the step size changes with the input data signal.

At iteration k , the step size is given by

$$\mu(k) = \frac{\mu_o}{\mathbf{x}_k^T \mathbf{x}_k} \quad (2.17)$$

where μ_o is a positive constant and $\mathbf{x}_k$ is the input vector at instant k .

In the constrained LMS that was first proposed in [37], the weights are subjected to constraints at each iteration. The constraint is delivering unit gain response in the look direction.

2.7.2 Recursive Least Squares

The Recursive Least Squares (RLS) algorithm uses the method of *Least Squares* to update the weight vector. In the Least Squares method, the weight vector is chosen to minimize the cost function. The cost function is the sum of error squares over a time window [26].

At time k , the cost function is given by

$$J(k) = \sum_{h=1}^k \lambda^{k-h} |e(h)|^2 \quad (2.18)$$

Where $e(h)$ is defined by (2.15), and λ is a positive constant close to, but less than, one and is called the forgetting factor. This factor determines how quickly the previous data are de-emphasized. The value of λ does not affect the convergence rate but determines the tracking ability of the RLS algorithm. By decreasing λ , a better tracking ability can be achieved. However, if λ is too small, the algorithm will become unstable. A typical value for λ is from 0.8 to less than 1 [25].

By minimizing (2.18) to get the optimum weight vector and applying the matrix inversion lemma, we get the following set of equations that describe the RLS algorithm [24]

$$\mathbf{k}_{RLS}(k) = \frac{\lambda^{-1} \mathbf{P}_{RLS}(k-1) \mathbf{x}(k)}{1 + \lambda^{-1} \mathbf{x}^H(k) \mathbf{P}_{RLS}(k-1) \mathbf{x}(k)} \quad (2.19)$$

$$e(k) = r(k) - \omega^H(k-1) \mathbf{x}(k) \quad (2.20)$$

$$\omega(k) = \omega(k-1) + \mathbf{k}_{RLS}(k) e^*(k) \quad (2.21)$$

$$\mathbf{P}_{RLS}(k) = \lambda^{-1} \mathbf{P}_{RLS}(k-1) - \lambda^{-1} \mathbf{k}_{RLS}(k) \mathbf{x}^H(k) \mathbf{P}_{RLS}(k-1) \quad (2.22)$$

The initial value of $\mathbf{P}_{RLS}(k)$ can be set to

$$\mathbf{P}_{RLS}(0) = \delta_{RLS}^{-1} \mathbf{I} \quad (2.23)$$

Where $\mathbf{I}$ is the $M \times M$ identity matrix, and δ_{RLS} is a small positive constant.

The main advantage of the RLS algorithm is its fast convergence rate compared to that of the LMS algorithm [24]. However, this fast convergence rate is achieved at the expense of the increased computational complexity. The RLS requires $4M^2 + 4M + 2$ complex multiplications per iteration, where M is the number of antenna elements [20].

2.7.3 Constant Modulus Algorithm

The *Constant Modulus Algorithm* (CMA) is one of the property-restoral based algorithms. Some communication signals, such as the *phase-shift keying* (PSK) signal, have a constant envelope which gets distorted when the signal is transmitted through the channel. The CMA adjusts the weight vector so as to minimize the variation of the envelope at the output of the array.

The *Steepest Descent CMA* (SD-CMA), in the simplest form, tries to minimize the cost function

$$J(k) = E[|y(k) - 1|^2] \quad (2.24)$$

The SD-CMA [30], [31] can be described by the following set of equations

$$y(k) = \omega^H \mathbf{x}(k) \quad (2.25)$$

$$e(k) = y(k) - \frac{y(k)}{|y(k)|} \quad (2.26)$$

$$\omega(k+1) = \omega(k) - \mu \mathbf{x}(k) e^*(k) \quad (2.27)$$

In [38] Agee proposed the *Least Squares CMA* (LS-CMA), where the LS-CMA tries to minimize the cost function

$$J(\omega) = \sum_{k=1}^L \|y(k) - 1\|^2 \quad (2.28)$$

The LS-CMA can be implemented either statically or dynamically. The static LS-CMA uses one data block in updating the weight vector, while the dynamic LS-CMA uses different input data blocks in updating the weight vector [26]. Let $\mathbf{X}(k)$ denote the block of input data at iteration k , $\mathbf{X}(k)$ contains L snapshots of the input data vector.

$$\mathbf{X}(k) = [x(1+kL), x(2+kL), \dots, x((k+1)L)] \quad k=0,1,\dots,N_{\text{conv}} \quad (2.29)$$

Where N_{conv} is the number of iterations required for the algorithm to converge. The dynamic LS-CMA can be described by the following set of equations

$$\mathbf{y}(k) = [\omega^H(k) \mathbf{X}(k)]^T = [y(1+kL), y(2+kL), \dots, y((k+1)L)]^T \quad (2.30)$$

$$\mathbf{r}(k) = \left[\frac{y(1+kL)}{|y(1+kL)|}, \frac{y(2+kL)}{|y(2+kL)|}, \dots, \frac{y((k+1)L)}{|y((k+1)L)|} \right]^T \quad (2.31)$$

$$\omega(k+1) = [\mathbf{X}(k) \mathbf{X}^H(k)]^{-1} \mathbf{X}(k) \mathbf{r}^*(k) \quad (2.32)$$

There are several modifications that have been proposed to the CMA. In [39], linearly constrained CMA has been proposed which transforms a priori knowledge of the signal into a set of linear constraints on the weight vector. In [40], Marquardt method is used with CMA. Marquardt method is a combination of the non-linear least squares method

and the steepest descent method. In [41], *recursive CMA* was proposed, which uses an exponentially squared error as a cost function over a time window.

2.7.4 Affine Projection Algorithm

The Affine Projection Algorithm (APA) is a non-blind algorithm, which was first proposed by Ozeki et al. [42] as a generalization for the NLMS algorithm [43]. The weight vector update in the NLMS can be viewed as a one dimensional affine projection, while in the APA, the weight vector update is obtained from projection on an affine subspace with multiple dimensions [44].

The APA tries to solve the underdetermined system of the following equations in an exact way while minimizing the norm of $\Delta\omega = \omega(k) - \omega(k-1)$ [45].

$$\mathbf{X}^T(k)\omega(k) = \mathbf{r}(k) \quad (2.33)$$

$$\mathbf{X}(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \dots, \mathbf{x}(k-(p-1))] \quad (2.34)$$

$$\mathbf{r}(k) = [r(k), r(k-1), \dots, r(k-(p-1))]^T \quad (2.35)$$

where $\mathbf{x}(k)$ is the column vector for the input signal at each of the elements of the array. $\mathbf{x}(k)$ is of size $M \times 1$, where M is the number of antenna elements. This means that the matrix $\mathbf{X}(k)$ is of size $M \times p$, where p is a positive integer called the projection order and controls the performance of the APA. $\mathbf{r}(k)$ is the vector containing p snapshots of the reference signal.

The complex version of the APA can be described by the following set of equations [44] [46]

$$\mathbf{e}(k) = \mathbf{r}(k) - \mathbf{X}^T(k) \boldsymbol{\omega}^*(k-1) \quad (2.36)$$

$$\mathbf{R}(k) = [\mathbf{X}^H(k) \mathbf{X}(k) + \delta \mathbf{I}]^{-1} \quad (2.37)$$

$$\mathbf{g}_{APA}(k) = \mathbf{R}^{-1}(k) \mathbf{e}^*(k) \quad (2.38)$$

$$\boldsymbol{\omega}(k) = \boldsymbol{\omega}(k-1) + \mu \mathbf{X}(k) \mathbf{g}_{APA}(k) \quad (2.39)$$

where δ is a small positive number that serves as the regularization parameter in computing the inverse of the signal autocorrelation matrix. And μ is a scalar quantity called the step size. $\mathbf{g}_{APA}(k)$ is called the prefilter vector because it filters the row vectors of $\mathbf{X}(k)$ to synthesize the adjustment vector [46].

The main advantage of the APA is that its performance lies in the middle between the performances of the NLMS and the RLS. It has faster convergence than the NLMS with an input signal that has a correlation matrix of widely spread eigen values, and it has less computational complexity than RLS [47].

The family of APA algorithms has been investigated extensively in recent years for acoustical echo cancellation [47], [48] and active noise control [49]. It has been very recently used in coherent interference suppression using adaptive arrays [50] and in

adaptive beamforming [51]. It has been reported in [52] that it can be also used in adaptive equalization and CDMA applications to minimize the training sequence.

2.7.5 Conjugate Gradient Method

The Conjugate Gradient (CG) algorithm is one of the non-blind algorithms. Its performance can be considered as intermediate, lying somewhere in between the performance of both the Steepest Descent (SD) method and the RLS algorithm. The CG has a faster convergence rate than the SD method and has lower computational complexity than the RLS algorithm [53].

The objective of the CG algorithm is to solve the Wiener-Hopf equation

$$\mathbf{R}\boldsymbol{\omega} = \mathbf{z} \quad (2.40)$$

Where $\mathbf{R}$ is the $M \times M$ correlation matrix of the input data vector $\mathbf{x}(k)$ and $\mathbf{z}$ is the cross correlation between the input data vector and the reference signal $\mathbf{r}(k)$.

To solve (2.40), the CG indirectly minimizes the cost function given by [54]

$$J(\boldsymbol{\omega}) = \frac{1}{2} \boldsymbol{\omega}^T(k) \mathbf{R} \boldsymbol{\omega}(k) - \mathbf{z}^T \boldsymbol{\omega}(k) \quad (2.41)$$

The way $\mathbf{R}$ and $\mathbf{z}$ are calculated will directly influence the performance of the algorithm. There are two ways to compute $\mathbf{R}$ and $\mathbf{z}$. The first one is the *Finite Sliding*

Data Window method, where the data samples inside a window of a specific finite length are used. Both $\mathbf{R}$ and $\mathbf{z}$ are computed by time ensemble average [54].

The second way to compute $\mathbf{R}$ and $\mathbf{z}$ is the *Exponentially Decaying Data Window* method, where both $\mathbf{R}$ and $\mathbf{z}$ will be given by

$$\mathbf{R}(k) = \lambda \mathbf{R}(k-1) + \mathbf{x}(k) \mathbf{x}^T(k) \quad (2.42)$$

$$\mathbf{z}(k) = \lambda \mathbf{z}(k-1) + \mathbf{d}(k) \mathbf{x}(k) \quad (2.43)$$

In this case, the CG algorithm can be described by the following set of equations [54]

Initial conditions: $\boldsymbol{\omega} = \mathbf{0}, \mathbf{g}(0) = \mathbf{z}(0), \mathbf{p}(1) = \mathbf{g}(0)$

$$\alpha(k) = \eta \frac{\mathbf{p}^T(k) \mathbf{g}(k-1)}{\mathbf{p}^T(k) \mathbf{R}(k) \mathbf{p}(k)} \quad (2.44)$$

$$\boldsymbol{\omega}(k) = \boldsymbol{\omega}(k-1) + \alpha(k) \mathbf{p}(k) \quad (2.45)$$

$$\mathbf{g}(k) = \lambda \mathbf{g}(k-1) - \alpha(k) \mathbf{R}(k) \mathbf{p}(k) + \mathbf{x}(k) (\mathbf{r}(k) - \mathbf{x}^T(k) \boldsymbol{\omega}(k-1)) \quad (2.46)$$

$$\beta(k) = \frac{(\mathbf{g}(k) - \mathbf{g}(k-1))^T \mathbf{g}(k)}{\mathbf{g}^T(k-1) \mathbf{g}(k-1)} \quad (2.47)$$

$$\mathbf{p}(k+1) = \mathbf{g}(k) + \beta(k) \mathbf{p}(k) \quad (2.48)$$

where $\alpha(k)$ is the step size that minimizes the cost function $J(\omega)$ given by (2.41), η is a positive number $(\lambda - 0.5) < \eta < \lambda$, $\mathbf{p}(k)$ is the direction vector, $\beta(k)$ is the estimated gradient and $\mathbf{g}(k)$ is the residual vector (where ∇J denotes the gradient of the function J .)

defined by
$$\mathbf{g}(k) = \mathbf{z} - \mathbf{R}\omega(k) = -\nabla J(\omega(k)) \quad (2.49)$$

In the literature, there are many adaptive filtering algorithms based on the CG method of optimization that were reported in [53] [54] [55]. In recent years, the CG method has been used in adaptive beamforming for smart antennas [56][57][58]. In [59] the Constrained CG algorithm to be used in beamforming was proposed. In [60] the CG method is used for mobile user tracking.

2.8 Introduction to CDMA Systems

There are three basic multiple access schemes; *Frequency Division Multiple Access* (FDMA), *Time Division Multiple Access* (TDMA) and *Code Division Multiple Access* (CDMA). In FDMA, each user is assigned a different frequency slot. In TDMA, each user is assigned a different time slot. CDMA is a wireless access technique where all the users occupy the same spectrum at the same time, but each user is assigned a distinct *signature sequence* (also called *spreading waveform*).

The *Direct Sequence –Code Division Multiple Access* (DS-CDMA) system is the system used in our research. Thus, we will discuss it in more detail.

2.8.1 Key Benefits of CDMA Systems.

CDMA was first commercialized in the late 1980s by Qualcomm Inc., based in San Diego, California [61]. Since then, the CDMA systems have gained increased popularity due to many advantages that CDMA offers, such as the inherent security, the multipath rejection and the increasing spectrum efficiency, which leads to increasing the capacity [62] [63]. Also, because the capacity of CDMA systems is only interference limited (unlike the capacity of FDMA and TDMA systems, which is primarily bandwidth limited), any reduction in interference results directly into an increase in capacity [64].

Due to these advantages, CDMA is a very promising technique. Thus, CDMA is used as the air interface technique in the third generation cellular systems, such as WCDMA and CDMA2000, which are designed so as to support a voice quality comparable to that of the Public Telephone Switched Network and to support high data bit rates [65][66][67][68].

2.8.2 Structure of CDMA System

The CDMA technique is based on the spread spectrum techniques, used originally in military applications. Spread spectrum, by definition [69], spreads the message signal into a much wider bandwidth than the minimum bandwidth necessary to send the message signal. In CDMA, spreading the message signal is accomplished by multiplying it by a spreading code sequence at the transmitter. This makes the signal appear similar to

random noise. At the receiver, recovering the message signal (despreading) is accomplished by multiplying the received signal by a synchronized replica of the spreading code sequence. This enables the receiver to recover the data of the desired user while the other users' signals will appear as random noise to the receiver [70].

2.8.3 *Spreading Code Properties and Generation*

The spreading codes, which are also called *Pseudorandom Noise* (PN) sequences, play an important role in the implementation of CDMA systems, as the selection of the set of codes directly influences the system performance (SNR, BER and multipath signal rejection) and the number of simultaneous users (system capacity) [71].

CDMA codes are designed to have a very low cross correlation because the ability of a receiver to detect the desired signal in the presence of interference from other users relies to a great extent on the correlation properties of the used spreading codes [72]. The correlation properties of the spreading codes depend on the type of code used and its length.

The codes found in practical CDMA systems can be categorized into orthogonal codes (e.g. Walsh-Hadamard codes) and non-orthogonal codes (e.g. Maximal length codes, Gold codes and Kasami codes).

In Walsh-Hadamard codes, the full sequence cross correlation is typically zero. But this does not hold for the partial cross correlation of the sequences. This means that the

advantage of using orthogonal codes is almost lost. Walsh codes of length 64 have been used in IS-95 CDMA [3].

Maximal length sequences (which are also known as *m-sequences*) are, by definition, the largest codes that can be generated by a given shift register or a delay element of a given length [73]. Maximal length sequences are not orthogonal but they do have a narrow auto correlation peak [72].

Gold codes can be obtained by properly combining two *m-sequences* which form a *preferred pair* that leads to the first Gold code [72].

Kasami codes can be obtained by combining a Gold code with a decimated version of the one of the two *m-sequences* that forms this Gold code. Kasami codes type is one of the important types of spreading codes because of the very low cross correlation between the codes and because of the large number of codes it supplies [73].

Chapter 3: The Adaptive Beamforming Algorithms Used in Simulation

In this chapter, we will describe the adaptive beamforming techniques used in this research in detail. The research will focus on the blind adaptive algorithms used in the base station for the uplink (the base station as a receiver) in CDMA systems. In this chapter we will present six algorithms

- *Least Squares Despread Respread Multi-Target Array Algorithm (LS-DRMTA)*
- *Least Squares Despread Respread Multi-Target Constant Modulus Algorithm (LS-DRMTCMA)*
- *Block-based RLS- Despread Respread Multi-Target Array (Block RLS-DRMTA)*
- *Sample-based RLS- Despread Respread Multi-Target Array (Sample RLS-DRMTA)*
- *Affine Projection- Despread Respread Multi-Target Array (AP-DRMTA)*
- *Conjugate Gradient- Despread Respread Multi-Target Array (CG-DRMTA)*

The last three algorithms are the algorithms we propose in this research based on the Despread-Respread techniques in order to improve the performance of the beamforming.

3.1 Introduction

In the case of a CDMA system, where multiple users share the same frequency band, the adaptation problem is generally more complex due to the presence of multi-user interference. The adaptive algorithm should have the ability to separate and extract each user's signal simultaneously (i.e. multi-target type algorithms). Also, it is preferred that the algorithm works without using a training sequence (i.e. blind algorithms) because using a training sequence reduces the spectral efficiency of the system.

Figure 3.1: A Multi-Target Adaptive Beamformer with M Antenna Elements and Q Output Ports

The multi-target type algorithms have parallel and independent beamformers. The number of these beamformers is dependent on the algorithm used to adapt the weights. In

general, if there are M antenna elements in the beamformer, there will be Q output ports, as shown in Figure 3.1.

One problem associated with the multi-target beamformer is how to generate different weight vectors in each port. If the adaptive algorithm used in the multi-target beamformer can not distinguish a particular user from the other users, the weight vectors in the different ports may converge to one vector with the same beam pattern with just a phase shift difference between the weight vectors. In this case, a procedure must be implemented to prevent this from happening. The procedure called *Gram-Schmidt Orthogonalization* (GSO) adds more computational complexity to the algorithm.

In [74], Agee proposed the Multi-Target Least Squares Constant Modulus Algorithm (MT-LSCMA). This algorithm is based on the Least Squares Constant Modulus Algorithm (LS-CMA) in a multi-target form. The LS-CMA uses a priori information that the received signals have a constant envelope only. Thus, it can not distinguish a particular user from the rest of the users. This algorithm suffers from high computational complexity and slow convergence rate, as it needs to perform the GSO procedure and a sorting procedure to extract each user's signal from the outputs of each port.

In [75][76], Rong et al. proposed two algorithms, the *Least Squares Despread Respread Multi-Target Array algorithm* (LS-DRMTA) and the *Least Squares Despread Respread Multi-Target Constant Modulus Algorithm* (LS-DRMTCMA). These

algorithms utilize the spreading code of the CDMA users to generate a reliable reference signal to adapt the weight vector.

Different algorithms based on the *Despread Respread* technique have been proposed. In [77], Dominique et al. proposed a combination of *Despread Respread* technique to generate the reference signal and LMS adaptive algorithm to update the weight vector. However, this algorithm suffers from the low convergence rate relative to the LS-DRMTA due to using the LMS algorithm in adaptation. In [78], the Block-based RLS *Despread Respread Multi-Target Array algorithm* (Block RLS-DRMTA) was proposed. In this algorithm, after generating the reference signal, it uses a combination of an exponentially decaying data window and variable step size to update the weight vector.

In [79], Cavalcanti and Romano pointed out that the performance of the LS-DRMTA depends directly on the level of cross-correlation among the spreading codes used in the system. They proposed a combination of DRMTA algorithm and auxiliary multi-user detection technique to reduce the capture effect. In [80][81], a combination of the LS-DRMTA algorithm and the RAKE receiver was proposed to improve the performance of the LS-DRMTA, assuming that the time delay of each multipath in the channel is known.

The major assumption in the *Despread Respread* techniques is that the synchronization has been achieved prior to the beamforming. Thus, some techniques have to be used to perfectly detect the time delay of the desired user, such as the techniques in [82][83][84]. The *Despread Respread* technique is very similar to the

demod-remod technique, which has been used to reject interference in QPSK and FSK signals [85][86].

3.2 Least Squares Despread Respread Multi-Target Array

3.2.1 Algorithm Description

The main idea of the LS-DRMTA [26][75][76] is to generate the reference signal using the spreading code. In the base station of a CDMA system, the spreading signals of all users are known. In a conventional receiver, in order to detect the i^{th} user's data bit, the received signal is multiplied by the time-delayed version of the spreading code for user i , $c_i(t-\tau_i)$. The multiplier output is sent to the detector to make the bit decision. In the despread-respread technique, the received signal is despread by multiplying it by $c_i(t-\tau_i)$ and the bit decision is made. Then the same time-delayed version of the spreading code for user i , $c_i(t-\tau_i)$ is used to respread the estimated data bit to generate the reference signal, as shown in Figure 3.2.

In the LS-DRMTA algorithm, after generating the reference signal, the error between the reference signal and the output of the array is minimized using the least squares approach.

Figure 3.2: Generating the Reference Signal in LS-DRMTA Algorithm.

3.2.2 Analysis

The LS-DRMTA algorithm tries to adapt the weight vector to minimize the cost function [75]

$$J(\omega_i) = \sum_{k=1}^K |y_i(k) - r_i(k)|^2 = \sum_{k=1}^K |\omega_i^H \mathbf{x}(k) - r_i(k)|^2 \quad (3.1)$$

where $y_i(k)$ is the output of the beamformer for user i and given by (2.7) and $r_i(k)$ is the reference signal generated as in Figure 3.2, ω_i is the $M \times 1$ weight vector for user i and $\mathbf{x}(k)$ is the $M \times 1$ input data vector (M is the number of antenna elements). K is the data block size and is set to be equal to the number of samples in one bit period. So if the sampling rate $R_s = N_s R_c$, where R_c is the chip rate in the CDMA signal and N_s is the

number of samples in the chip, the block size K will be equal to NN_s , where N is the processing gain.

The dynamic LS-DRMTA adapts the weight vectors using a different data block during each iteration. Each block represents one data bit. Let us first define $\mathbf{X}(n)$, which is the block of input signal during one bit n (where n is the bit index)

$$\mathbf{X}(n) = [\mathbf{x}(1+nK), \mathbf{x}(2+nK), \dots, \mathbf{x}((1+n)K)] \quad (3.2)$$

and the reference signal generated during the same bit n , $\mathbf{r}_i(n)$ is an $1 \times K$ vector

$$\mathbf{r}_i(n) = [r_i(1+nK), r_i(2+nK), \dots, r_i((1+n)K)] \quad (3.3)$$

The cost function $J(\omega_i)$ in the matrix notation during the n th bit is

$$J(\omega_i(n)) = \|\omega_i^H(n)\mathbf{X}(n) - \mathbf{r}_i(n)\|^2 \quad (3.4)$$

where $\|\cdot\|^2$ denotes the norm of vector $[\cdot]$. Using the extension of Gauss' method [26], we can summarize the LS-DRMTA by the following equations

$$\begin{aligned} \mathbf{y}_i(n) &= \omega_i^H(n)\mathbf{X}(n) \\ &= [y_i(1+nK), y_i(2+nK), \dots, y_i((1+n)K)] \end{aligned} \quad (3.5)$$

$$\hat{b}_{i,n} = \text{sgn}(\text{Re}(\mathbf{y}_i(n)\mathbf{c}_i^H)) \quad (3.6)$$

$$\mathbf{r}_i(n) = \hat{b}_{i,n} \mathbf{c}_i \quad (3.7)$$

$$\boldsymbol{\omega}_i(n+1) = [\mathbf{X}(n)\mathbf{X}^H(n)]^{-1} \mathbf{X}(n)\mathbf{r}_i^H(n) \quad (3.8)$$

where $\hat{b}_{i,n}$ denotes the estimated n th bit, $\mathbf{c}_i$ is the spreading waveform that contains K samples, and $\text{sgn}[\cdot]$ denotes the sign of $[\cdot]$.

The main advantages of the LS-DRMTA algorithm are the low computational complexity and the fast convergence rate relative to the LS-MTCMA algorithm. Another important advantage is that the number of the output ports of the beamformer is not limited to the number of array elements.

The disadvantage of the LS-DRMTA is that it requires synchronization. The algorithm will even suffer if proper synchronization is not achieved. Another drawback of this algorithm is its sensitivity to the initial data bit estimation, which can result in a phase ambiguity of π for *Binary Phase Shift Keying* (BPSK). This problem can be eliminated by using *Differential Phase Shift Keying* (DPSK) as the modulations scheme [18].

3.3 Least Squares Despread Respread Multi-Target Constant Modulus Algorithm

3.3.1 Algorithm Description

The LS-DRMTCMA was first proposed in [26][75][76]. This algorithm combines the CM type technique with the despread-respread technique. The motivation behind combining both of them is to get better performance by using the processing gain of the system (despread-respread technique) and the constant modulus property of the signal.

The reference signal in the LS-MTCMA is generated as in Figure 3.3

Figure 3.3: Generating the Reference Signal in LS-DRMTCMA Algorithm.

As shown in Figure 3.3, the reference signal is generated as the weighted sum of the respread data bit and the output of the array after a complex limiter operation.

3.3.2 Analysis

The derivation of the LS-DRMTCMA is very similar to that of the LS-DRMTA, with a change to the way of generating the reference signal.

The reference signal for user i at the n th $\mathbf{r}_i(n)$ is given by

$$\mathbf{r}_i(n) = a_{PN} \mathbf{r}_{i,PN}(n) + (1 - a_{PN}) \mathbf{r}_{i,CM}(n) \quad (3.9)$$

where a_{PN} is a positive value less than 1 (to achieve best performance it should be from (0.3-0.6)) and $\mathbf{r}_{PN}$ and $\mathbf{r}_{CM}$ are given by

$$\mathbf{r}_{i,PN}(n) = \hat{b}_{i,n} \mathbf{c}_i \quad (3.10)$$

$$\mathbf{r}_{i,CM}(n) = \left[\frac{y_i(1+nK)}{|y_i(1+nK)|}, \frac{y_i(2+nK)}{|y_i(2+nK)|}, \dots, \frac{y_i((1+n)K)}{|y_i((1+n)K)|} \right] \quad (3.11)$$

where $\hat{b}_{i,n}$ is given by (3.6) and $y_i(h+nK)$ is the output sample h in the n th bit, as given in (3.5). The weight vector is given by

$$\boldsymbol{\omega}_i(n+1) = [\mathbf{X}(n)\mathbf{X}^H(n)]^{-1} \mathbf{X}(n)\mathbf{r}_i^H(n) \quad (3.12)$$

The LS-DRMTCMA has the same advantages as the LS-DRMTA, since it utilizes the PN sequence as the LS-DRMTA does, with the additional advantage that it can achieve a lower bit error rate than the LS-DRMTA, because the LS-DRMTCMA algorithm utilizes

the constant modulus property of the transmitted signal (in addition to the PN sequence), which helps in deriving a better reference signal. The better performance comes at the expense of increased complexity [76].

3.4 Block Recursive Least Squares Despread Respread Multi-Target Array

3.4.1 Algorithm Description

The LS-DRMTA and LS-DRMTCMA algorithms make use of the array input vectors corresponding to the current bit only. They may lose some useful information carried in the previous bit vectors, especially when the channel is a fast time varying channel. In [78], the authors proposed the Block-based RLS-DRMTA algorithm, which uses a sliding window cost function similar to that used in the RLS algorithm. A forgetting factor accordingly weighs the old input data at the antenna array.

3.4.2 Analysis

The reference signal $r_i(n)$ is generated as shown in Figure 3.2 and by (3.7).

The cost function used to update the weight vector is given by

$$J(\omega_i(n)) = \sum_{j=1}^n \lambda^{n-j} \|\omega_i^H(j)X(j) - r_i(j)\|^2 \quad (3.13)$$

where λ is the forgetting factor.

In the Block RLS-DRMTA algorithm, the matrix inversion lemma method is not used to update the weight vector, in order to reduce the computational complexity. The weight vector is updated according to the gradient method

$$\omega_i(n+1) = \omega_i(n) - \frac{\mu_i(n)}{2} \frac{\partial J(\omega_i(n))}{\partial \omega_i(n)} \quad (3.14)$$

where $\mu_i(n)$ is a variable step size that minimizes the cost function. It is introduced to avoid the slow convergence when using a constant step size.

The following equations can summarize the Block RLS-DRMTA (note that $\mathbf{R}$ is the auto correlation matrix for the input data and $\mathbf{z}$ is the cross correlation vector between the reference signal and the input signal).

$$\mathbf{R}(n+1) = \sum_{j=1}^n \lambda^{n-j} \mathbf{X}(j) \mathbf{X}^H(j) \quad (3.15)$$

or recursively $\mathbf{R}(n+1) = \lambda \mathbf{R}(n) + \mathbf{X}(n) \mathbf{X}^H(n) \quad (3.16)$

$$\mathbf{z}_i(n+1) = \sum_{j=1}^n \lambda^{n-j} \mathbf{X}(j) \mathbf{r}_i^H(j) \quad (3.17)$$

or recursively $\mathbf{z}_i(n+1) = \lambda \mathbf{z}_i(n) + \mathbf{X}(n) \mathbf{r}_i^H(n) \quad (3.18)$

$$\mathbf{e}_i(n+1) = \{\lambda \mathbf{I} - \mu_i(n) \mathbf{R}(n)\} \mathbf{e}_i(n) + \mathbf{X}(n) \{\boldsymbol{\omega}_i^H \mathbf{X}_i(n) - \mathbf{r}_i(n)\} \quad (3.19)$$

$$\mu_i(n+1) = \frac{\mathbf{e}_i^H(n+1) \mathbf{e}_i(n+1)}{\mathbf{e}_i^H(n+1) \mathbf{R}(n+1) \mathbf{e}_i(n+1)} \quad (3.20)$$

$$\boldsymbol{\omega}_i(n+1) = \{\mathbf{I} - \mu_i(n+1) \mathbf{R}(n+1)\} \boldsymbol{\omega}_i(n) + \mu_i(n+1) \mathbf{z}_i(n+1) \quad (3.21)$$

The main advantage of the Block RLS-DRMTA is the fast convergence relative to the LS-DRMTA and LS-DRMTCMA. It also achieves better bit error rate in a slow changing channel and at high value of E_b/N_0 .

To get to know the drawback of this algorithm, let us recall the excellent work that has been done by Poor and Wang [87]. They have shown that using a *Decision-Directed* RLS adaptive algorithm in interference suppression in a DS-CDMA system gives a high level of output SINR (close to the optimum). However, this only happens under certain

conditions. Otherwise, the algorithm can suffer from catastrophic error propagation in dynamic channels or at low level of input SNIR [88].

Despread-Respread technique can be considered as one of the decision-directed adaptive algorithms. Thus, the Block RLS-DRMTA has poor tracking capabilities in a dynamic channel and is non-robust to changes in the mobile environment, where users enter and leave the system randomly and non-robust to estimation errors. Thus, any error in estimating the transmitted bit (either due to low level of signal to interference and noise ratio or due to high cross-correlation between codes) will propagate and will degrade the performance.

3.4.3 *The Proposed Sample-based RLS-DRMTA Algorithm*

One way to increase the robustness of the Block RLS-DRMTA is to use the algorithm sample by sample instead of being a block-based algorithm. The reason for this is that if we update the weight vector sample by sample we will improve the performance in a time varying channel. Also, the robustness of the algorithm will increase, as any error will accumulate for a specific number of sample indices instead of accumulating for the same number of block indices.

The Sample RLS-DRMTA tries to minimize the cost function

$$J(\mathbf{w}_i(h)) = \sum_{j=1}^h \lambda^{h-j} \|\mathbf{w}_i^H(j)\mathbf{x}(j) - r_i(j)\|^2 \quad (3.22)$$

where h is the sample index, not the bit index (n) as given by (3.13), ω_i is the $M \times 1$ weight vector for user i and $\mathbf{x}(h)$ is the $M \times 1$ input data vector (M is the number of antenna elements).

Following the same steps as in the Block RLS-DRMTA leads to the following set of equations which describe the algorithm

$$\mathbf{R}(h+1) = \sum_{j=1}^h \lambda^{h-j} \mathbf{x}(j) \mathbf{x}^H(j) \quad (3.23)$$

or recursively
$$\mathbf{R}(h+1) = \lambda \mathbf{R}(h) + \mathbf{x}(h) \mathbf{x}^H(h) \quad (3.24)$$

$$\mathbf{z}_i(h+1) = \sum_{j=1}^h \lambda^{h-j} \mathbf{x}(j) r_i^*(j) \quad (3.25)$$

or recursively
$$\mathbf{z}_i(h+1) = \lambda \mathbf{z}_i(h) + \mathbf{x}(h) r_i^*(h) \quad (3.26)$$

$$\mathbf{e}_i(h+1) = \{\lambda \mathbf{I} - \mu_i(h) \mathbf{R}(h)\} \mathbf{e}_i(h) + \mathbf{x}(h) \{\omega_i^H \mathbf{x}_i(h) - r_i(h)\} \quad (3.27)$$

$$\mu_i(h+1) = \frac{\mathbf{e}_i^H(h+1) \mathbf{e}_i(h+1)}{\mathbf{e}_i^H(h+1) \mathbf{R}(h+1) \mathbf{e}_i(h+1)} \quad (3.28)$$

$$\omega_i(h+1) = \{\mathbf{I} - \mu_i(h+1) \mathbf{R}(h+1)\} \omega_i(h) + \mu_i(h+1) \mathbf{z}_i(h+1) \quad (3.29)$$

The main advantage of the Sample RLS-DRMTA is that because it updates the weight vector sample by sample, it converges faster than the Block RLS-DRMTA.

Updating the weight vector sample by sample improves the performance in a fast time varying channel. Also, it is more robust against decision errors as decision errors do not accumulate for a long period, as in the case of the Block RLS-DRMTA. The improvement in the performance comes at the expense of increasing the computational complexity.

3.5 The Proposed Affine Projection Despread Respread Multi-Target Array

3.5.1 Algorithm Description

In this section we propose a new *multi-target* adaptive algorithm to be used in CDMA systems called the Block Affine Projection-DRMTA (Block AP-DRMTA). This algorithm uses the *despread-respread* technique to generate the reference signal and then uses the Affine Projection Algorithm to generate the weight vector. The *Affine Projection Algorithm* (APA) was first proposed by Ozeki et al. [42]. Since then it has been considered a very promising technique due to its good convergence properties compared to the LMS algorithm, as well as its low computational complexity compared to the RLS algorithm. This algorithm has been used in CDMA applications in order to minimize the training sequence [52].

3.5.2 Analysis

Here we will adapt the Block APA given by Tanaka et al. [46] to be suitable for use in a CDMA system and with the *despread-respread* technique.

First, the reference signal $r_i(n)$ is generated as shown in Figure 3.2 for the n^{th} bit. During the n^{th} bit, we use the APA algorithm to update the weight vector. First, let us recall how we generate the reference signal.

$\mathbf{X}$, is the block of input signal during one bit n (where n is the bit index, the index n has been dropped for simplicity)

$$\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(K)] \quad (3.30)$$

and $\mathbf{r}_i$ is the reference signal generated for user i during the same bit n . $\mathbf{r}_i$ is an $1 \times K$ vector

$$\mathbf{r}_i = [r_i(1), r_i(2), \dots, r_i(K)] \quad (3.31)$$

$$\begin{aligned} \mathbf{y}_i &= \boldsymbol{\omega}_i^H \mathbf{X} \\ &= [y_i(1), y_i(2), \dots, y_i(K)] \end{aligned} \quad (3.32)$$

$$\hat{b}_{i,n} = \text{sgn}(\text{Re}(\mathbf{y}_i \mathbf{c}_i^H)) \quad (3.33)$$

$$\mathbf{r}_i(n) = \hat{b}_{i,n} \mathbf{c}_i \quad (3.34)$$

The error vector $\mathbf{e}$ for user i during the n^{th} bit is defined as

$$\mathbf{e} = \mathbf{r}_i - \mathbf{y}_i \quad (3.35)$$

We divide $\mathbf{X}$ into small blocks, each of them of size p . This forms the matrix $\mathbf{Z}_i$, which will be of size $M \times p$. $\mathbf{Z}_i$ is given by

$$\mathbf{Z}_i = [\mathbf{x}(p), \mathbf{x}(p-1), \dots, \mathbf{x}(1)] \quad (3.36)$$

There are several ways to form the matrix Z_2 that will control both the performance and the computational complexity. We can form the next block by changing the whole block and taking the next p samples. In this case, our experiments have shown that the achieved performance is not good. We can form the next block by changing one sample, dumping the first time sample and taking a new one. This way, the algorithm will suffer from high computational complexity. Thus, the best way is to form the Z_2 matrix by dumping a small block of samples of size s smaller than p and taking a new block of size s (where $s < p$).

$$Z_2 = [x(p+s), x(p-1+s), \dots, x(s+1)] \quad (3.37)$$

$$\text{and} \quad Z_j = [x(p+j*s), x(p-1+j*s), \dots, x(j*s+1)] \quad (3.38)$$

where $j=1, 2, \dots, \text{floor}((K-1)/s)$, and $\text{floor}(g)$ is the floor function that gives the greatest integer not greater than g .

The above equations mean that, each matrix Z_j will be of size p and the overlapping between Z_j and Z_{j+1} will be $(p-s)$.

So during the n^{th} bit, we generate the reference signal as in (3.34). The algorithm can thus be described by the following set of equations

$$r_{i,j} = [r_i(p+j*s), r_i(p-1+j*s), \dots, r_i(j*s+1)] \quad (3.39)$$

$$\mathbf{y}_{i,j} = \boldsymbol{\omega}_{i,j-1}^H \mathbf{Z}_j \quad (3.40)$$

$$\mathbf{e}_{i,j} = \mathbf{r}_{i,j} - \mathbf{y}_{i,j} \quad (3.41)$$

$$\mathbf{R}_j = \mathbf{Z}_j^H \mathbf{Z}_j \quad (3.42)$$

$$\mathbf{g}_{i,j \text{ APA}} = \mathbf{R}_j^{-1} \mathbf{e}_{i,j}^* \quad (3.43)$$

$$\boldsymbol{\omega}_{i,j} = \boldsymbol{\omega}_{i,j-1} + \mu \mathbf{Z}_j \mathbf{g}_{i,j \text{ APA}} \quad (3.44)$$

where μ is the step size and $\boldsymbol{\omega}_{i,j}$ is the weight vector for user i at iteration j .

$\boldsymbol{\omega}_{i,j}$ is used with the incoming $\mathbf{Z}_{j+1}$ to form $\mathbf{y}_{i,j+1}$ during the n^{th} bit. At the bit $(n+1)$, we use the last updated weight vector to get the beamformer output.

The main advantage of the AP-DRMTA algorithm is that it gives good performance in a dynamic channel. It offers good tracking capabilities in a time varying channel. It is also robust against the changes in the mobile environment. The computational complexity is less than that of the Sample RLS-DRMTA algorithm but more than that of the Block RLS-DRMTA. The main drawback of this algorithm is the slow convergence rate compared to both the Block RLS-DRMTA and the Sample RLS-DRMTA.

3.6 The Proposed Conjugate Gradient Despread Respread Multi-Target Array

3.6.1 Algorithm Description

In this section, we propose a new *multi-target* adaptive algorithm to be used in CDMA systems called Conjugate Gradient - DRMTA (CG-DRMTA). This algorithm uses the *despread-respread* technique to generate the reference signal and then uses the Conjugate Gradient method to generate the weight vector. Recently, the Conjugate Gradient Method (CGM) has been proposed for adaptive beamforming in CDMA systems and it has shown a good performance in a multipath fading channel [89]. Thus, generating the reference signal using the despread-respread technique, then using the CGM for updating the weight vector, seems to be a very promising technique.

3.6.2 Analysis

If we use the model of sample-by-sample CGM proposed in [54], the algorithm will suffer from high computational complexity, as in the Sample RLS-DRMTA algorithm. Thus, we adapt the algorithm proposed in [54] to change it to a *block-based* algorithm.

In the CG-DRMTA algorithm, we will update the weight vector bit by bit, so we will generate the auto correlation matrix $\mathbf{R}$, and the cross correlation matrix $\mathbf{z}$, using the block of the input data and block of the reference signal as in (3.15) and (3.17), respectively.

$$\mathbf{R}(n+1) = \sum_{j=1}^n \lambda^{n-j} \mathbf{X}(j) \mathbf{X}^H(j) \quad (3.45)$$

or recursively $\mathbf{R}(n+1) = \lambda \mathbf{R}(n) + \mathbf{X}(n) \mathbf{X}^H(n) \quad (3.46)$

$$\mathbf{z}_i(n+1) = \sum_{j=1}^n \lambda^{n-j} \mathbf{X}(j) \mathbf{r}_i^H(j) \quad (3.47)$$

or recursively $\mathbf{z}_i(n+1) = \lambda \mathbf{z}_i(n) + \mathbf{X}(n) \mathbf{r}_i^H(n) \quad (3.48)$

We will set the initial conditions: $\boldsymbol{\omega}_i = [1, 0, 0, \dots, 0]^T$, $\mathbf{g}(0) = \mathbf{z}(0)$, $\mathbf{p}(1) = \mathbf{g}(0)$

$$\alpha(n) = \eta \frac{\mathbf{p}^T(n) \mathbf{g}(n-1)}{\mathbf{p}^T(n) \mathbf{R}(n) \mathbf{p}(n)} \quad (3.49)$$

$$\boldsymbol{\omega}_i(n) = \boldsymbol{\omega}_i(n-1) + \alpha(n) \mathbf{p}(n) \quad (3.50)$$

$$\mathbf{g}(n) = \lambda \mathbf{g}(n-1) - \alpha(n) \mathbf{R}(n) \mathbf{p}(n) + \mathbf{X}(n) (\mathbf{r}_i(n) - \mathbf{X}^T(n) \boldsymbol{\omega}_i(n-1)) \quad (3.51)$$

$$\beta(k) = \frac{(\mathbf{g}(n) - \mathbf{g}(n-1))^T \mathbf{g}(n)}{\mathbf{g}^T(n-1) \mathbf{g}(n-1)} \quad (3.52)$$

$$\mathbf{p}(n+1) = \mathbf{g}(n) + \beta(n) \mathbf{p}(n) \quad (3.53)$$

where $\alpha(n)$ is the step size that minimizes the cost function $J(\omega)$ given by (2.41), η is a positive number ($(\lambda-0.5) < \eta < \lambda$), a typical value for η is from 0.6 to 0.8 [54]. And $p(n)$ is the direction vector and $g(n)$ is the residual vector defined (2.49).

The main advantage of this algorithm is the low computational complexity. Also, it offers good tracking abilities and it is robust against the changes in the mobile environment.

The drawback of this algorithm is the slow convergence rate.

Chapter 4: Simulation Environment and Performance Results

In this chapter, we describe the system model and the environment in which we have run the simulation; we will study the model for the transmitter, the channel and the receiver. We then present the simulation results for the performance of the proposed algorithms. We present the results to show how each algorithm will perform individually. A comparison between the different algorithms (the novel and the existing ones) will be presented in chapter 5.

In the simulation, we will consider the case of a Single Cell system, which means we will take into consideration the effect of interference coming from the same cell only.

Also, perfect power control is assumed, which means that all the signals impinging on the array have the same power level.

It is worth mentioning that in the base station of the CDMA system, the spreading code sequences of all the users are known.

First, we will study the model of the transmitter, the channel and the receiver of the CDMA system. A simplified block diagram for our system is shown in Figure 4.1 [3].

Figure 4.1: A Direct Sequence CDMA Transmitter and Receiver.

4.1 Transmitter Model

4.1.1 BPSK Modulation

The modulation scheme used in the simulation is *Binary Phase Shift Keying* (BPSK). BPSK is now widely used in both military and commercial communication systems. The expression for the n^{th} bit of the BPSK signal is [70].

$$s_n(t) = \sqrt{2P_0} \cos[2\pi f_c t + n\pi] \quad n=0, 1 \quad (4.1)$$

where P_0 is the power of the signal, f_c is the carrier frequency and n is the data bit index. The BPSK signal may be expressed as

$$s(t) = \sqrt{2P_0} b(t) \cos(2\pi f_c t) \quad (4.2)$$

$$b(t) = \sum_{n=-\infty}^{\infty} b_n \Psi_{T_b}(t - nT_b) \quad (4.3)$$

$b(t)$ is the binary data sequence composed of a sequence of symbols b_n , $b_n \in \{+1, -1\}$ and it is the n^{th} data bit and Ψ_{T_b} is a unit rectangular pulse of duration T_b .

4.1.2 The Spreading Process

The signature sequence is used at the transmitter side to spread the message signal. The signature sequence also allows the receiver to demodulate the message transmitted by the desired user and separate it from the messages of the other users [90].

The *spreading waveform* often has much larger bandwidth compared to the bandwidth of the message signal. Also, it must be independent of the message signal [3].

The spreading waveform $c(t)$ may be expressed as

$$c(t) = \sum_{n=-\infty}^{\infty} \sum_{l=0}^N c_l \Psi_{T_c} \left(\frac{t - (l + nN)T_c}{T_c} \right) \quad (4.4)$$

where $c_l \in \{+1, -1\}$, N is the number of chips sent in the spreading code sequence, T_c is the chip period and Ψ_{T_c} is a unit rectangular pulse of duration T_c .

The CDMA signal of user i may be expressed as

$$v_i(t) = \sqrt{2P_i} c_i(t) b_i(t) \cos(2\pi f_c t) \quad (4.5)$$

where $b(t)$ and $c(t)$ are given by (4.3) and (4.4) respectively. And P_i is the power of the signal for user i .

The term *spreading waveform* refers to the signal $c(t)$ which is the binary signal having the value or +1 or -1 during the duration of T_c while the term *spreading code* refers to the binary sequence c_i , which is either +1 or -1.

We will use a set of Kasami codes in the simulation. The length of the code used in the simulation is 15 chips [26]. The reasons for choosing a code with such a short length are:

1. Reducing the simulation time and thus using this time in comparing the performance of different adaptive algorithms.
2. Focusing on the enhancement of the system capacity obtained by using the adaptive array, not on the enhancement obtained by using a long PN sequence for each user.

In the simulation, code-on-pulse modulation will be used, which means that one PN sequence fits exactly in one bit. That is to say, the processing gain (which is the ratio of the chip rate to the bit rate) $N = 15$.

4.2 Channel Model

The signal received at the base station from user i , after passing through the channel [3], is given by

$$\tilde{u}_i(t) = \sqrt{2P_i} c_i(t - \tau_i) b_i(t - \tau_i) \cos(2\pi f_c t + \phi_i) \quad (4.6)$$

where τ_i is the time delay for user i and ϕ_i is the random phase for the signal of user i .

Note that since perfect power control has been assumed in our system, the value of P_i will be the same for all users.

In the simulation we assume

1. The channel is an ideal channel that just introduces *Additive White Gaussian Noise* (AWGN). There is no multipath or fading in the channel.
2. The time delay τ_i for each user is detected perfectly. There are many techniques proposed to detect the time delay of the signal, such as the techniques in [82][83][84].

We will use two different channels in our simulation, a static channel and a dynamic channel. In the static channel, the DOA of the signal is constant, while in the dynamic channel, the DOAs of the signals change at a constant angular speed.

4.3 Receiver Model

At the base station, for the adaptive array we will assume that

1. The spacing between the elements is half the wavelength of the input signal.
This makes the spacing between the array elements small, such that there are no amplitude variations between the signals received at the different elements.
2. There is no mutual coupling between the antenna elements.
3. We use narrowband processing, as the signal bandwidth is small relative to the carrier frequency.

At the base station, either a linear array or circular array can be used. The main disadvantage of the circular array is the presence of high side lobe levels in its beam pattern [91], while the main disadvantage of the linear array is that it forms a wide main lobe when the direction of the main lobe is the same as the direction of the antenna array. As the linear array is simpler in analysis and more frequently used, we will use a linear array antenna that consists of identical elements with equal inter-element spacing. This structure is called *Uniform Linear Array* (ULA).

4.3.1 Signal Model for Uniform Linear Array

At the base station, consider an M elements uniformly spaced linear array with inter-element spacing equal to d , aligned with the x -axis. Consider also a plane wave for user i arriving at the array from a direction θ° , relative to the direction of the array, as shown in Figure 4.2. The angle θ is the azimuth angle and completely specifies the DOA [3].

Figure 4.2: Illustration of a plane wave incident on a Uniform Linear Array from direction angle θ°

The received signal at the first element may be expressed as in (4.6), but neglecting the time delay in the channel for now, it yields [19][26]

$$\tilde{u}_{i1}(t) = \sqrt{2P_i} c_i(t) b_i(t) \cos(2\pi f_c t + \phi_i) \quad (4.7)$$

It is convenient to use the complex envelope representation of the signal,

$$u_{i1}(t) = \sqrt{2P_i} c_i(t) b_i(t) \exp(-j\phi_i) \quad (4.8)$$

The received signal at the first element and its complex envelope representation can be related by

$$\tilde{u}_{i1}(t) = \text{Re}[u_{i1}(t) \exp(2j\pi f_c t)] \quad (4.9)$$

where $\text{Re}[\cdot]$ represents the real part of $[\cdot]$.

The first element is taken as the reference point. From Figure 4.2, we can see that the difference in the distance between the m^{th} element and the first element is equal to $(m-1)d\cos(\theta)$. The time taken by the plane wave to arrive at the m^{th} element is given by

$$\tau(\theta) = \frac{d \cos \theta}{c} (m-1) \quad (4.10)$$

where c is the velocity of light.

The signal arriving at the m^{th} element of the antenna array is given by

$$\begin{aligned} \tilde{u}_{im}(t) &= \sqrt{2P_i} c_i(t) b_i(t) \cos(2\pi f_c t - 2\pi f_c \tau(\theta) + \phi_i) \\ &= \tilde{u}_{i1}(t) \exp(-2j\pi f_c \tau(\theta)) \end{aligned} \quad (4.11)$$

The base band representation is given by

$$u_{im}(t) = u_{i1} \exp(-2j\pi f_c \tau(\theta)) \quad (4.12)$$

Note that in (4.11) and (4.12), because the bandwidth of the spread message signal $b(t)c(t)$ is small compared to the carrier frequency f_c , the effect of $\tau(\theta)$ on the spread message signal has been neglected. The effect of the time delay can be represented by a phase shift.

Now substitute (4.10) in (4.12) to get

$$\begin{aligned} u_{im}(t) &= u_{i1} \exp(-2j\pi f_c \frac{d \cos \theta}{c} (m-1)) = u_{i1} \exp(-2j\pi f_c \frac{c}{\lambda} \frac{\lambda \cos \theta}{2c} (m-1)) \\ &= u_{i1} \exp(-j \frac{2\pi}{\lambda} d(m-1) \cos \theta) = u_{i1} \exp(-j\beta d(m-1) \cos \theta) \end{aligned} \quad (4.13)$$

where $\beta = 2\pi/\lambda$ is the phase propagation factor.

We can express (4.13) in matrix notation by

$$\mathbf{u}_i(t) = \mathbf{a}(\theta) u_{i1} \quad (4.14)$$

$$\text{where} \quad \mathbf{u}_i(t) = \begin{bmatrix} u_{i1}(t) \\ u_{i2}(t) \\ \vdots \\ u_{iM}(t) \end{bmatrix} \quad (4.15)$$

and

$$\mathbf{a}(\theta) = \begin{bmatrix} 1 \\ e^{-j\beta d \cos \theta} \\ \dots \\ e^{-j\beta d (M-1) \cos \theta} \end{bmatrix} \quad (4.16)$$

The vector $\mathbf{a}(\theta)$ is called the *steering vector*. $\mathbf{a}(\theta)$ is also called *direction vector*, *array response vector* or *array manifold vector*. The collection of steering vectors for all angles is referred to as *array manifold* [26].

Now, we will extend the previous analysis from studying the effect of one user to including the effect of having many users in the cell. Assume there are q narrow band signals $u_1, u_2 \dots u_q$, all centered around the same carrier frequency f_c , impinging on the array with a DOA θ_i , $i=1,2,\dots,q$. These q signals may express different signals coming from different users or may express time-delayed versions of the same signal due to multipath [26]. The received signal at the array will be the summation of the q signals and the noise. The received signal is given by

$$\mathbf{x}(t) = \sum_{i=1}^q \mathbf{a}(\theta_i) u_i(t) + \mathbf{n}(t) \quad (4.17)$$

where $\mathbf{a}(\theta_i)$ is given by (4.16) and $\mathbf{n}(t)$ denotes the $M \times 1$ vector of the noise at the array elements.

We can express (4.17) in matrix notation as

$$\mathbf{x}(t) = \mathbf{A}(\theta)\mathbf{u}(t) + \mathbf{n}(t) \quad (4.18)$$

where $\mathbf{A}(\theta)$ is the $M \times q$ matrix of the steering vector.

$$\mathbf{A}(\theta) = [\mathbf{a}(\theta_1), \dots, \mathbf{a}(\theta_q)] \quad (4.19)$$

and

$$\mathbf{u}(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_q(t) \end{bmatrix} \quad (4.20)$$

Equation (4.18) represents the most commonly used *narrowband input* signal [19].

As explained before, the collection of steering vectors for all angles is referred to as *array manifold*, or *antenna beam pattern*. The antenna beam pattern is the gain that the antenna delivers as a function of the angle, which is given in the case of linear array by [20]

$$G(\theta) = |g(\theta)| \quad (4.21)$$

where

$$g(\theta) = \sum_{m=1}^M \omega_{i,m} e^{-j\beta(m-1)d \cos(\theta)} \quad (4.22)$$

Where M is the number of antenna elements, $\omega_{i,m}$ is the weight factor at the m^{th} element for user i , β is the propagation vector in direction θ and d is the distance between two successive elements in the array.

Using $G(\theta)$, we may define the *normalized beamformer response* [26]

$$g_n(\theta) = \frac{G(\theta)}{\max\{G(\theta)\}} \quad (4.23)$$

$g_n(\theta)$ is also known as the *array factor* or the *normalized beam pattern*. In literature, the term “Beam Pattern” usually refers to the normalized beam pattern $g_n(\theta)$. This term is what will be used throughout this thesis.

There are three main issues that need to be addressed concerning the beam pattern of the linear array antenna:

- The symmetry of the beam pattern about the axis of the uniform linear array.
- The effect of the number of elements and the distance between elements on the beam pattern.
- The end fire problem in the beam pattern.

Concerning the symmetry of the beam pattern about the direction of the array, we can see from (4.22) that the beam pattern $g(\theta)$ depends on $\cos(\theta)$. Since $\cos(\theta) = \cos(-\theta)$, then $g(\theta) = g(-\theta)$. Therefore, the beam pattern of the uniform linear array is symmetrical about the axis of the array. Arrays with non-uniform spacing between elements or arrays with other geometries, such as circular arrays, do not exhibit this kind of symmetry [3].

The antenna beam pattern describes the ability of the beamformer to spatially discriminate the input signal. This ability depends on the size of the aperture of the array; as the aperture increases, the ability of the beamformer to spatially discriminate the input signal improves. What matters is not the absolute aperture size, but its size in wavelength [20].

It was shown in [26] that the null-to-null beam width of the main lobe of the uniform linear array (when directed normal to the axis of the array and equal-weight for each element is assumed) is given by

$$\theta_{BW} = 2 \sin^{-1} \left(\frac{\lambda}{Md} \right) \quad (4.24)$$

We can see from (4.24) that the beam width of the main lobe depends on the number of antenna elements M and the ratio (λ/d) . It is desired to have this beam width as small as possible in order to have a very narrow beam towards the desired user and to null out the interferers. Thus, we need to increase the number of antenna elements M and increase d in comparison to λ . It has been shown in chapter 2 that increasing d above $\lambda/2$ results in grating lobes so the optimum value for d is $\lambda/2$.

From (4.24), when using $d = \lambda/2$ and $M = 8$ elements, the null-to-null beam width of the main lobe will be around 29° . This will be the beam width when the beam is directed normal to the direction of the axis of the array. When the antenna pattern is in the same

direction as the axis of the array, the beam width will be wider than this value due to the end fire problem.

About the end fire problem, it has been stated in chapter 2 that there is a correspondence between the FIR filter and the beamformer. The correspondence is closest when the array geometry is linear. The normalized element spacing, d/λ , can be mapped into the sampling period, T_s , in the FIR filter, and $\cos \theta_i$, where θ_i is the DOA of the signal of user i , can be mapped into the temporal frequency f of the FIR filter. The beam pattern of the adaptive array can be viewed as equivalent to the magnitude response of the FIR filter. It was shown in [26] that the pass band width Δf of the FIR filter corresponds to the term $[\cos \theta - \cos(\theta - \Delta\theta)]$ in the array, where $\Delta\theta$ is the beam width of the main lobe. In order to keep $[\cos \theta - \cos(\theta - \Delta\theta)]$ constant for all values of θ , we have

$$\cos \theta - \cos(\theta - \Delta\theta) = \Delta f \quad (4.25)$$

We can express the left side of (4.25) as

$$\cos \theta - \cos(\theta - \Delta\theta) = \Delta\theta \sin \theta' \quad (4.26)$$

where θ' is a value between $\theta - \Delta\theta$ and θ .

By substituting (4.25) into (4.26)

$$\Delta\theta = \frac{\Delta f}{\sin \theta'} \quad (4.27)$$

We can see from (4.27) that for a fixed pass bandwidth Δf , $\Delta\theta$ (the beam width of the main lobe) depends on θ' . For signals of DOA normal to the direction of the array (i.e. θ' is close to 90°), the beam width of the main beam will be minimum. Whereas, when the DOA of the signal is in the same direction as the array (i.e. θ' is close to 0°), the beam width of the main beam will be maximum because $\sin \theta'$ is close to 0. Therefore, the linear array beamformer constructs a narrow beam directed to the broadside of the array and a wide beam directed to the end fire of the array [26].

4.4 System Parameters and Performance Evaluation Criteria

In the previous sections, we have discussed in detail the transmitter model, the channel model and the receiver model. Table 4.1 summarizes the system, transmitter, channel and receiver parameters that we will use in our simulation.

The system used is asynchronous *Direct Sequence Code Division Multiple Access* (DS-CDMA) system, where each of the users has a different time delay but the time delay for the desired user can be perfectly detected. As stated before, in the simulation we will consider the case of a Single Cell, which means we will take into consideration the effect of interference coming from the same cell only.

Table 4.1: System parameters used in simulation.

Transmitter Parameters		Channel & Receiver Parameters	
Carrier Frequency	2.05 GHz	Channel	AWGN channel
Modulation Scheme	BPSK	Multipath or Fading	No
Data Bit Rate	128 Kbps	Power Control	Perfect power control
Multiple Access Scheme	CDMA	Time Delay	Perfectly detected
Processing Gain	15	Antenna Geometry	Uniform linear array
Number of Samples/Chip	4	Distance between Elements	$\lambda/2$
Sampling Frequency	7.68 MHz	Number of Antenna Elements	8

The carrier frequency f_c is equal to 2.05 GHz. The data bit rate is 128 kbps. The sampling rate is equal to 4 samples per chip, with 15 chips per bit. This makes the sampling frequency equal to $60 \times 128 \text{ k} = 7.68 \text{ MHz}$ [26].

There are 8 users in the cell. The DOA of the signals from these users is uniformly distributed between angle 0° and angle 140° , as shown in Figure 4.3.

Figure 4.3: Users' Distribution in Spatial Domain.

In the following sections, the performance of the proposed algorithms will be evaluated. We will first evaluate the performance of the proposed algorithm by checking the antenna beam pattern as given in (4.22). Then we will check the signal constellation of a specific user before and after using the beamformer suggested by the algorithm and finally we will check the signal waveform corresponding to one specific user before and after using the beamformer suggested by the algorithm.

4.5 Simulation Results of the Sample-based RLS-DRMTA

4.5.1 Beam Pattern

The beam pattern when using the Sample-based RLS-DRMTA algorithm and with using the forgetting factor λ equal to 0.99 at $E_b/N_0 = 8$ dB is shown in Figures 4.4, 4.5, 4.6 and 4.7.

We can clearly see that except for the users with DOA near the end fire of the array, most of the interferers have much lower gain than the gain for the desired user with an average of 20dB.

In the upper part of Figure 4.4, user 1 at angle 0 is the desired user. As has been explained earlier, the beam width of the beam generated along the direction of the linear array is wider than the beam generated towards any other direction. This is the reason why the gain for interferer 2 at angle 20 is somewhat high, as the direction of the interferers is close to the direction of the array. However, the gain for the rest of the interferers at directions apart from the direction of the linear array is very low. Thus, overall, there is significant increase in the total SINR. The same situation exists in the lower part of Figure 4.5, where user 2 is the desired user and users 1 and 3 are the interferers having high gain.

If we look at the upper part of Figure 4.6, where user 5 at angle 80° is the desired user, we see that the desired user has high gain compared to all the rest of the interferers

(17 dB higher than 2 of the interferers and around 20 dB higher than the other 5 interferers). This is because user 5 is at a direction apart from the direction of the array.

In the lower part of Figure 4.7, user 8 at angle 140° is the desired user. We notice that in the case of user 8, the gain for the desired user is higher than the gain of one of the interferers by 10 dB only. This is because user 8's direction is close to the direction of the axis of the array, so it suffers from the end-fire problem.

Figure 4.4: Beam Pattern for Users 1 and 2 at Angles 0° and 20° Using Sample RLS-DRMTA Algorithm at $E_b/N_0=8$ dB (each red star and red line present the direction of one specific user)

Figure 4.5: Beam Pattern for Users 3 and 4 at Angles 40° and 60° Using Sample RLS-DRMTA Algorithm at $E_b/N_0=8$ dB (each red star and red line present the direction of one specific user)

Figure 4.6: Beam Pattern for Users 5 and 6 at Angles 80° and 100° Using Sample RLS-DRMTA Algorithm at $E_b/N_0=8$ dB (each red star and red line present the direction of one specific user)

Figure 4.7: Beam Pattern for Users 7 and 8 at Angles 120° and 140° Using Sample RLS-DRMTA Algorithm at $E_b/N_0=8$ dB (each red star and red line present the direction of one specific user)

4.5.2 Constellation Diagram

The constellation diagram is a measure of the ability of the receiver to demodulate the received signal; the real part is plotted against the imaginary part of the received signal before and after using the beamformer.

The constellation diagram for user 4 at angle 60° , without using any adaptive beamforming technique is shown in Figure 4.8. The constellation diagram for the same user using the Sample RLS-DRMTA algorithm with $\lambda = 0.99$, is shown in Figure 4.9. Note that the $E_b/N_0 = 8$ dB in both cases.

We can note, in the case of using the Sample RLS-DRMTA there is no point close to the threshold level. Thus, using the beamformer of the Sample RLS-DRMTA algorithm can make it a much easier task to extract the desired signal.

Figure 4.8: Constellation Diagram for User 4 at Angle 60° before the Beamforming Processing at $E_b/N_0=8\text{dB}$

Figure 4.9 :Constellation Diagram for User 4 at Angle 60° after the Beamforming Processing using the Sample RLS-DRMTA Algorithm at $E_b/N_0=8\text{dB}$

4.5.3 Signal Waveform

In this section, we will examine the signal waveform before and after using the Sample RLS-DRMTA beamforming algorithm with $\lambda=0.99$.

We will examine the signal waveform for user 4 at angle 60° in the case of 8 users present, assuming no multipath, no fading and perfect power control. The input E_b/N_o is equal to 20 dB.

Figure 4.10 shows the original bit waveform for three bits. Figure 4.11 shows the original waveform after spreading with the spreading code. Figure 4.12 shows the corrupted signal waveform, which is the sum of all the impinging signals at the first element of the array. This is the signal that will be received if one omni-directional antenna is used.

Figure 4.13 shows the reconstructed signal waveform for user 4. Note that this figure represents the non-despread CDMA signal before using the matched filter to make the decision on the transmitted data bit.

Comparing Figure 4.11 and Figure 4.13, we see that using the Sample RLS-DRMTA adaptive beamforming enables reconstructing the signal. Figure 4.13 can be considered the same as Figure 4.11, with added noise as seen in the figures.

Figure 4.10: Original Bit Waveform **Figure 4.11: Original Waveform after Spreading**

Figure 4.12: The Corrupted Signal Waveform

Figure 4.13: The Reconstructed Signal Waveform of user 4 using Sample RLS-DRMTA algorithm.

4.6 Simulation Results of the AP-DRMTA

4.6.1 Beam Pattern

The antenna beam patterns for each of the eight users when using the AP-DRMTA beamformer algorithm are plotted. The projection order that has been used is 3 and the block is 2 with $\mu = 0.01$. The choice of these parameters' values is based on the simulation results. Simulation results have shown that these parameter values result in a very good performance. As stated earlier, we use a processing gain of 15 with 4 samples per chip. That makes each bit have 60 samples. Then the number of iterations per bit will be $\text{floor}(59/2) = 29$ iterations.

Figures 4.14, 4.15, 4.16 and 4.17 represent the antenna beam patterns for the 8 users when using the AP-DRMTA algorithm at input $E_b/N_o = 8$ dB.

We can see how the algorithm results in a beam pattern that delivers higher gain to the desired user than to the other users acting as interferers to the system. When users 1, 2 and 3 are the desired users, they suffer from the end fire problem. The rest of the users, when they are the desired users, have gain of 15 dB or more than the other users. For example, in the upper part of Figure 4.16, user 5 at angle 80° is the desired user; he has an increased gain in the amount of 20 dB higher than all the other users.

Figure 4.14: Beam Pattern for Users 1 and 2 at Angles 0° and 20° using APA-DRMTA at $E_b/N_0=8$ dB

Figure 4.15: Beam Pattern for Users 3 and 4 at Angles 40° and 60° using APA-DRMTA at $E_b/N_0=8$ dB

Figure 4.16: Beam Pattern for Users 5 and 6 at Angles 80° and 100° using APA-DRMTA at $E_b/N_0=8$ dB

Figure 4.17: Beam Pattern for Users 7 and 8 at Angles 120° and 140° using APA-DRMTA at $E_b/N_0=8$ dB

4.6.2 Constellation Diagram

The constellation diagram for user 4 at angle 60° is shown in Figure 4.18 at input $E_b/N_0 = 8$ dB using the AP-DRMTA algorithm using the parameters specified above. The constellation diagram before using the AP-DRMTA algorithm is shown in Figure 4.8.

Figure 4.18: Constellation Diagram for user 4 at angle 60° after the Beamforming Processing Using the AP-DRMTA Algorithm at $E_b/N_0=8$ dB

From figure 4.18, we can see that there is no point close to the threshold level, which makes the decision-making very clear.

4.6.3 Signal Waveform

In this section, we will examine the signal waveform before and after using the AP-DRMTA beamforming algorithm for user 4 at angle 60° at input $E_b/N_o=20$ dB, with the parameters stated earlier.

The original bit waveform, the original signal after spreading and the corrupted signal are shown in Figure 4.10, Figure 4.11 and Figure 4.12 respectively.

Figure 4.19 shows the reconstructed signal waveform using the AP-DRMTA adaptive beamformer. Comparing Figure 4.19 and Figure 4.11, it is clear that the original signal can be reconstructed upon using the AP-DRMTA, as the signal in Figure 4.19 has the same form as the signal in Figure 4.11 with added noise. Thus the message signal will be easily retrieved upon multiplying by the CDMA code at the receiver site.

Figure 4.19: The Reconstructed Signal Waveform of User 4 Using APA-DRMTA Algorithm.

4.7 Simulation Results of the CG-DRMTA

4.7.1 Beam Pattern

The antenna beam patterns for each of the eight users using the CG-DRMTA beamformer algorithm with $\lambda = 0.99$ and $\eta = 0.7$ are plotted.

Figures 4.20, 4.21, 4.22 and 4.23 represent the antenna beam pattern diagrams when using the CG-DRMTA algorithm at input $E_b/N_0 = 8$ dB.

From the figures, we can see that the algorithm is able to form a beam that has maximum gain towards the desired user (around 15 dB higher than the gain for the interferers) and minimum gain towards the interferers, except for the first three users, because of the end fire problem as discussed before. As seen from the upper part of Figure 5.22 for example, when user 5 at angle 80° is the desired user, he has gain higher by around 17 than the gain for 3 of the interferers and by 20 dB than the gain of the rest of the users.

Figure 4.20 : Beam Pattern for Users 1 and 2 at Angles 0° and 20° using CG-DRMTA at $E_b/N_0 = 8$ dB

Figure 4.21 : Beam Pattern for Users 3 and 4 at Angles 40° and 60° using CG-DRMTA at $E_b/N_0 = 8$ dB

Figure 4.22 : Beam Pattern for Users 5 and 6 at Angles 80° and 100° using CG-DRMTA at $E_b/N_0=8$ dB

Figure 4.23 : Beam Pattern for Users 7 and 8 at Angles 120° and 140° using CG-DRMTA at $E_b/N_0=8$ dB

4.7.2 Constellation Diagram

The constellation diagram for user 4 at angle 60° is shown in Figure 4.24 at input $E_b/N_0 = 8$ dB when using the CG-DRMTA algorithm to perform the antenna beam pattern. The constellation diagram before using the CG-DRMTA algorithm is shown in Figure 4.8.

Figure 4.24: Constellation Diagram for User 4 at Angle 60° after the Beamforming Processing using the CG-DRMTA Algorithm at $E_b/N_0 = 8$ dB

From Figure 4.23, we can see that there is no point close to the threshold level, which makes the decision-making very clear.

4.7.3 Signal Waveform

In this section, we will examine the signal waveform before and after using the CG-DRMTA beamforming algorithm for user 4 at angle 60° at input $E_b/N_0=20$ dB. In the CG-DRMTA $\lambda = 0.99$ and $\eta = 0.7$ are used.

The original bit waveform, the original signal after spreading and the corrupted signal are shown in Figure 4.10, Figure 4.11 and Figure 4.12 respectively.

Figure 4.25 shows the reconstructed signal waveform using the CG-DRMTA adaptive beamformer. Comparing Figure 4.25 and Figure 4.11, it is clear that CG-DRMTA enables reconstructing the original signal.

Figure 4.25: The Reconstructed Signal Waveform for User 4 Using CG-DRMTA Algorithm

4.8 Graphical User Interface

A *Graphical User Interface* (GUI) has been created to illustrate the performance of the different algorithms. The beam pattern of each of the adaptive algorithms has been used as an evaluation criterion to show and compare the performance of the different adaptive algorithms. There are three parts of the GUI. The first part of the GUI deals with the signal through the AWGN channel (from the mobile station to the smart antenna at the base station). The second part of the GUI deals with the different algorithms used in our simulation; it shows the beam pattern of the algorithm specified by the user. The third part of the GUI shows an animation for how the algorithm changes with the time as the mobile stations moved in the direction specified by the user. More details about the GUI can be found in Appendix A.

4.9 Summary

In this chapter, we have presented the simulation environment in which we have run our simulation and the system parameters used in the simulation. We have then illustrated the performance of the three proposed algorithms in terms of the beam pattern, the constellation diagram and the ability to reconstruct the signal. We have shown that the three algorithms are able to deliver high gain towards the desired user and to null out the interferers.

Chapter 5: Comparison of the Proposed and Existing Algorithms

5.1 Introduction

In the previous chapter we have presented the simulation model along with the simulation results for each of the proposed algorithms individually. In this chapter we will present a comparison between the different algorithms (those that exist and those that are proposed) in terms of the convergence time and the *Bit Error Rate* (BER). We will compare the algorithms in two different channels, the AWGN channel where there is no multipath effect, no fading and the DOAs of the signals at the array are constant and the dynamic channel where the DOAs of the signal at the array change with time. Also we will compare the different algorithms in terms of the computational complexity.

5.2 Comparison in AWGN channel

In this section, we will investigate the performance of the different algorithms in the AWGN channel. We will compare the different algorithms in terms of the convergence time, and in terms of the BER at two different input levels of E_b/N_o .

5.2.1 Convergence Time Evaluation

The parameter that will be used to check the convergence time is the output SINR. The SINR is calculated by averaging 100 independent Monte-Carlo runs for each user and then averaging the SINR level for all users under the following conditions:

- There is no multipath effect and no fading in the channel.
- Perfect power control is assumed.
- The time delay of each user can be perfectly detected.
- There are 8 users in the cell. The DOAs of the signals from these users is uniformly distributed between the angle 0° and the angle 140° , as shown in Figure 4.3
- E_b/N_0 is equal to 8 dB

Under the above conditions, the simulation results for the block-based algorithms are shown in Figure 5.1. The block-based algorithms use one iteration per block (one block represents one bit, the duration of each bit is $7.8125 \mu\text{s}$). Because of this we have been able to plot the LS-DRMTA, the LS-DRMTCMA, the RLS-DRMTA and the CG-DRMTA algorithms on the same graph. The simulation has been performed using 500 iterations (500 bits) but only the first 120 iterations have been plotted to show the convergence behavior, as after the first 120 iteration there is no additional information that can be obtained.

We can clearly see that the LS-DRMTA gives the lowest output SINR. The LS-DRMTCMA gives greater SINR than the LS-DRMTA but still less than Block RLS-DRMTA and the CG-DRMTA. The reason for this is that the first two algorithms do not

utilize the information carried in the previous data vectors. Nevertheless, these two algorithms have fast convergence.

Figure 5.1: Convergence Times of Block-Based Adaptive Algorithm in AWGN Channel
(Note that each iteration time represents one block (bit) time, which is $7.8125 \mu\text{s}$.)

Both the Block RLS-DRMTA and the CG-DRMTA show good performance in terms of large SINR. The Block RLS-DRMTA has an advantage over the CG-DRMTA as it converges a little bit faster. The Block RLS-DRMTA requires approximately 18 iterations to converge with a duration of $18 \times 7.8125 \mu\text{s} = 140.625 \mu\text{s}$.

The Sample-based RLS-DRMTA algorithm uses one iteration per sample (which means 60 iterations per bit). Thus the duration of one iteration for the Sample-based RLS-DRMTA will be different from the duration of one iteration for the block-based

algorithms. The iteration duration for the sample-based RLS-DRMTA is $7.8125/60 \mu\text{s} = 130.208 \text{ ns}$. We have used 500 bits to simulate the output SINR, but the simulation result shown in Figure 5.2 is for the first 20 bits only. This is sufficient to show the convergence behavior. Because the block size is 60, the number of iterations for 20 bits is equal to $60 \times 20 = 1200$ iterations. The duration of these 1200 iterations is equal to $1200 \times 130.208 \text{ ns} = 156.250 \mu\text{s}$.

Figure 5.2: SINR for Sample RLS-DRMTA Adaptive Beamforming Algorithm in AWGN Channel (Note that each iteration time represents one sample time, which is $= 130.208 \text{ ns}$.)

We can see that the Sample-based RLS-DRMTA has a faster convergence than the Block-based RLS-DRMTA algorithm, as it converges after approximately 150 iterations of duration $150 \times 130.208 \text{ ns} = 19.531 \mu\text{s}$. The faster convergence is achieved at the expense of additional computational complexity.

The convergence curve of the AP-DRMTA is shown in Figure 5.3. Projection order of 3 and block size of 2 are used (as stated in section 4.7). Thus, the number of iterations per bit time will be equal to 29 iterations. The first 200 bits of the simulated 500 bits are shown. Thus the total number of iterations will be $29 \times 200 = 5800$ iterations. We can see that the AP-DRMTA has the same level of SINR as that of the Sample-based RLS-DRMTA but it has slower convergence rate, even slower than the convergence rate of the CG-DRMTA. It converges after approximately 4000 iterations, of duration $4000 \times 7.8125/29 \mu\text{s.} = 1.0775 \text{ ms.}$)

Figure 5.3: SINR for AP-DRMTA Adaptive Beamforming Algorithm in AWGN Channel
(Note that each 29 iterations represent one bit time, which is 7.8125 $\mu\text{sec.}$)

5.2.2 BER Evaluation

To evaluate the performance of the different algorithms, we will compare the BER performance that results upon using each algorithm. The BER performance is obtained in an AWGN channel at two different levels of E_b/N_o , 8 dB and 4 dB. The users are equally distributed between angle 0° and 160° while the number of users changes from 8 to 14 users.

In the simulation, we transmit different bit streams for each user. At the receiver, we use the beamformer to get each user's signal. The output of each beamformer is then fed to the matched filter to despread the signal and estimate the transmitted data bit. The estimates of the data bits are compared to the original data bits transmitted and the BER for each user is calculated. The BER is then averaged over all the users, to average out the level of cross-correlations between the different PN sequences. This averaged BER is the one used in the plots. Note that the BER is computed using the converged weight vector.

First, the BER performance of the different algorithms when the input E_b/N_o is equal to 8 dB is shown in Figure 5.4.

Figure 5.4: BER Performance in AWGN with Input $E_b/N_0=8$ dB

We can see that both the LS-DRMTA and the LS-DRMTCMA algorithms give the worst BER, as they do not make use of the information carried in the previous data.

When the number of users is between 12 and 14, the level of the cross correlations between the codes is quite high. Thus, in the case of the Block RLS-DRMTA algorithm, any error in taking the decision will propagate and result in the degradation of the performance. This is the reason for the bad performance of the Block RLS-DRMTA. The CG-DRMTA is more robust against the decision errors. Thus, it has better performance than the Block RLS-DRMTA for this number of users. Using the sample-based algorithms improves the performance. This is why the Sample RLS-DRMTA and the AP-DRMTA have better performance than the two previously discussed algorithms for this number of users. The AP-DRMTA algorithm gives better performance than the Sample-based RLS-DRMTA because the AP-DRMTA algorithm utilizes the information in the previous data only in the range of the projection order, which decreases the amount of the error propagation.

Thus, for a large number of users we can see that the performance of the three proposed algorithms surpasses the performance of the other algorithms.

However, when the number of users decreases to 10, the level of cross correlation decreases and hence the performance of the block RLS-DRMTA begins to improve sharply. Unlike with 12 and 14 users, it even gives less BER than the proposed CG-

DRMTA algorithm, but still the proposed sample-based algorithms achieve better performance. Comparing the Sample RLS-DRMTA and the AP-DRMTA, we can see that the Sample RLS-DRMTA algorithm gives better performance as now the level of cross correlations decreases, so there are few errors to propagate. One can conclude that utilizing more of the past data will improve

the performance. That is to say, at this number of users, the two proposed sample-based algorithms result in better performance than the Block-based RLS-DRMTA algorithm, while the Block-based RLS-DRMTA algorithm gives better performance than the proposed CG-DRMTA algorithm.

When the number of users decreases to 8, we find that the Block RLS-DRMTA begins to achieve good performance comparable to that of the Sample RLS-DRMTA algorithm and better than that of the AP-DRMTA algorithm. Although the proposed Sample RLS-DRMTA algorithm still outperforms the Block RLS-DRMTA algorithm, in this case of small number of users and large E_b/N_o , it may be preferable to use the Block RLS-DRMTA to avoid the computational complexity of the sample-based algorithms.

The BER performance of the different algorithms when the input E_b/N_o is equal to 4 dB and the users are equally distributed between angle 0° and 160° is shown in Figure 5.5.

Figure 5.5: BER Performance in AWGN with Input $E_b/N_0=4$ dB

First, by comparing Figure 5.4 and 5.5, we see that the BER increases as the E_b/N_o decreases. We also notice that the BER performance for most of the algorithms does not improve very much as the number of users decreases, as much as it did when $E_b/N_o = 8$ dB. The reason for this is that at such low E_b/N_o , the system becomes noise limited rather than interference limited. Thus, it is the noise rather than the number of users (interferers) that determines the BER for low values of E_b/N_o . We can see from Figure 5.5 that, in the case of 10 to 14 users, the proposed CG-DRMTA algorithm gives better BER performance than the Block RLS-DRMTA, because at such low E_b/N_o , the CG-DRMTA is more robust than the Block RLS-DRMTA to the estimation error (which increases at such low E_b/N_o). Even when the number of users is 8, the improvement offered by the Block RLS-DRMTA over the proposed CG-DRMTA is not as much as the improvement offered when E_b/N_o was 8 dB.

Comparing the block-based algorithms with the proposed sample-based algorithms, we can see that the sample-based algorithms still achieve superior performance compared to that of the block-based ones even when the number of users is small (case of 8 users). The proposed AP-DRMTA algorithm achieves better BER performance than that of the Block RLS-DRMTA regardless of the numbers of users. The reason for this is that the Block RLS-DRMTA is non-robust to the estimation errors whereas the AP-DRMTA algorithm is robust.

By comparing the two proposed algorithms, the AP-DRMTA and the Sample RLS-DRMTA, we find that the AP-DRMTA gives better performance than the Sample RLS-DRMTA in the case of 10 to 14 users, as the AP-DRMTA utilizes the number of the past samples equivalent to the projection order and hence reduces the effect of the estimation error. In the case of 8 users, the Sample RLS-DRMTA achieves a slightly better BER performance but at the expense of doubling the computational complexity.

5.3 Comparison in a Time Varying Channel

In the simulation, we have examined different scenarios for the users' motion in the space, where the DOAs of the signals at the antenna array change with time in a random manner. All possible scenarios can be simulated on the GUI we have created (Appendix A).

The case we will consider here to show the convergence time and the BER for the different algorithms is the case where the DOAs of the signal change with angular speed with time, where all the users move in the same direction. The reason for choose such case is to simulate the different algorithms in the same conditions have been used in [78].

5.3.1 Convergence Time Evaluation

The parameter that will be used to check the convergence time is the SINR. The SINR is calculated by averaging 100 independent Monte-Carlo runs in a dynamic channel. In the dynamic channel, the DOAs of the signals change at angular speed of $16\pi/s$. We calculate the change of the angle at every sample.

We will use the same assumptions stated in section 5.2.1 and with the same level of E_b/N_0 , which is equal to 8 dB. There are 8 users in the cell. At time $t = 0$, the DOA of the signals from these users is uniformly distributed between angle 0° and angle 140° , as shown in Figure 4.3. The DOA of the signals changes with time; each sample will come from a different angle than the previous sample with an angular shift of 0.000375° .

First, the convergence times for the block-based algorithms (LS-DRMTA, LS-DRMTCMA, Block RLS-DRMTA, CG-DRMTA) are shown in Figure 5.6. Again the first 120 bits of the 500 bits used in simulation are plotted to show the convergence behavior. At the beginning of the simulation, the 8 users will be uniformly distributed between angle 0° and angle 140° . At the end of the 120 bits, the users will be uniformly distributed between angle 2.7° and angle 142.7° .

Figure 5.6: Conversion Times of Block-Based Adaptive Algorithms in a Dynamic Channel
(Note that each iteration time represents one block (bit) time, which is $7.8125 \mu\text{s}$.)

From Figure 5.6, we can see that the LS-DRMTA algorithm still gives the worst performance in terms of the SINR as was in the case of the static channel.

Here we can see that the LS-DRMTCMA algorithm gives a large level of SINR. It is comparable to the level of SINR obtained when using the Block RLS-DRMTA and the CG-DRMTA algorithms. The reason for this is that the preservation of the constant modulus enables the LS-DRMTCMA algorithm to derive a better reference signal, especially in a fast time varying channel. In such channel, utilizing the information carried in the previous data will not increase the ability of the block-based algorithms to extract the desired signal.

With regard to the convergence time, the LS-DRMTCMA algorithm offers a good convergence rate compared to both the Block RLS-DRMTA algorithm and the CG-DRMTA algorithm. It convergences after approximately 6 iterations which has a duration of $6 \times 7.8125 \mu\text{s} = 46.857 \mu\text{s}$. The Block RLS-DRMTA algorithm still has a faster convergence rate than the CG-DRMTA algorithm.

The convergence curve for the Sample RLS-DRMTA algorithm is shown in Figure 5.7, where the first 20 bits of the 500 bits used in the simulation have been plotted. At the end of the 20 bits, the 8 users will be uniformly distributed between angle 0.45° and angle 140.45° . We can see that Sample RLS-DRMTA has a fast convergence rate, the convergence rate is still in the range of 150 iterations which have a duration of $150 \times 130.208 \text{ ns} = 19.531 \mu\text{s}$. We can notice that the level of the SINR is slightly greater than the level of the SINR of the block-based algorithms.

The convergence time for the AP-DRMTA is shown in Figure 5.8, where the first 200 bits of the 500 bits used in the simulation have been plotted. At the end of the 200 bits, the 8 users will be uniformly distributed between angle 4.5° and angle 144.5° . The AP-DRMTA suffers from slow convergence rate compared to the other algorithms.

Figure 5.7: SINR for Sample RLS-DRMTA Adaptive Beamforming Algorithm in Dynamic Channel (Note that each iteration time represents one sample time, which is = 130.208 ns)

Figure 5.8: SINR for AP-DRMTA Adaptive Beamforming Algorithm in Dynamic Channel (Note that there are 29 iterations in one bit time)

5.3.2 BER Evaluation

In this section, we will show the BER performance, where there are 8 users in the cell. At time $t = 0$, the DOA of the signals from these users is uniformly distributed between angle 0° and angle 140° . The input $E_b/N_0 = 8$ dB. Table 5.1 shows the BER performance of the different algorithms in a static channel (angular velocity is equal to 0) and in dynamic channel where the DOAs of the signals are changing. Two different values for the angular velocity have been used in the simulation: $2\pi/s$ and $16\pi/s$.

Table 5.1: BER Performance in Dynamic Channel (8 users, $E_b/N_0 = 8$ dB)

Angular Velocity Algorithm	0	$2\pi/s$	$16\pi/s$
LS-DRMTA	4.37×10^{-4}	0.0081	0.0106
LS-DRMTCMA	5.72×10^{-5}	6.2×10^{-4}	7.4×10^{-4}
Block RLS-DRMTA	$5. \times 10^{-8}$	0.0041	0.0053
Sample RLS-DRMTA	4×10^{-8}	8×10^{-8}	9×10^{-8}
AP-DRMTA	1×10^{-7}	3.3×10^{-4}	3.7×10^{-4}
CG-DRMTA	2×10^{-6}	0.0043	0.0049

We can see from Table 5.1 that the LS-DRMTA algorithm gives the lowest BER for both values of angular velocity.

We can also note from Table 5.1 that there is a great degradation in the BER performance of both the Block RLS-DRMTA and the CG-DRMTA algorithms. The reason for this is that the Block RLS-DRMTA and the CG-DRMTA algorithms make use

of the information carried in all the previous data weighed by the forgetting factor. In a fast time varying channel, like the one we use in this part of the simulation, this may lead to taking a wrong decision. Moreover, we can see that the LS-DRMTCMA algorithm has a better BER performance than the LS-DRMTA, the Block RLS-DRMTA and the CG-DRMTA algorithms. The reason for this is that for the DRMTA algorithms type, any erroneous decision in the Despread-Respread block propagates through the adaptation process (the probability of taking an erroneous decision increases in case of a dynamic channel more than for a static channel). On the other hand in the LS-DRMTCMA algorithm, any erroneous decision in the Despread-Respread block is weighted and hence does not affect the performance as much as in the DRMTA type algorithms.

On the other hand we notice that there is only a slight degradation in the BER performance of the Sample RLS-DRMTA algorithm, as the algorithm updates the weight vector sample by sample, which enables it to track the changes in the channel.

The AP-DRMTA algorithm has a better BER performance than the LS-DRMTCMA, However, the improvement in the performance is slightly low, while there is a large increase in the computational complexity.

5.4 Comparison in terms of Computational Complexity

In this section, we will compare the different algorithms in terms of the computational complexity based on the number of operations each algorithm requires. In our simulation, the number of antenna elements M has been taken as 8, the number of samples in one bit K is 60. In the AP-DRMTA, the projection order p is 3 and the block size s is 2.

Table 5.2 shows the number of operations each algorithm requires for one user during one bit.

We can see from Table 5.2 that CG-DRMTA algorithm has the least number of operations, which means the CG-DRMTA has the less computational complexity of the all algorithms. The amount of the computational complexity for each algorithm was confirmed by the simulation running time for that algorithm.

In the different algorithms, the interference rejection was done on sample basis. If we reject the interference only on the decision making instant (on which we decide which bit was transmitted), the amount of the computational complexity of the algorithms will be reduced.

Table 5.2: The Computational Complexity for the different algorithms

Algorithm	Number of Complex Multiplications	Number of complex additions.
LS-DRMTA	$(2K+3)M^2+(K+4)M=8384$	$(2K+1)M^2+(K+4)M=8256$
LS-DRMTCMA	$(2K+3)M^2+(K+4)M+2K=8504$	$(2K+1)M^2+(K+4)M+K=8316$
Block RLS-DRMTA	$(K+8)M^2+(3K+4)M=5824$	$(K+5)M^2+(2K+4)M+K=5212$
Sample RLS- DRMTA	$K*(10M^2+6M)=41280$	$K(8M^2+6M)=33600$
AP-DRMTA	$29*[PM^2+(2P+1)M+4P^2+4P]=$ 8584	$29*[PM^2+(2P+1)M+2P^2+5P]=$ 8149
CG-DRMTA	$(K+3)M^2+(3K+8)M=5536$	$(K+3)M^2+(2K+9)M=5064$

Chapter 6: Conclusion and Future Work

6.1 Conclusion

In this thesis, we have developed three novel blind adaptive algorithms to be used for the smart antennas in CDMA systems. We have provided a detailed analysis for each of these three algorithms and have created a MATLAB simulation testbed to show the performance of these algorithms. The performance of these algorithms has been compared with the performance of the algorithms presented in literature. The comparison has been done in terms of the convergence time and the BER in two different channels (Static AWGN Channel and Dynamic Channel), and also in terms of the computational complexity.

We have proposed the Sample RLS-DRMTA algorithm. Simulation results have shown that this algorithm has fast convergence and gives low BER in all conditions. However, this comes at the expense of very high computational complexity.

We have proposed the AP-DRMTA algorithm. Simulation results have shown that this algorithm gives a lower BER compared to the block-based algorithms. But this algorithm suffers from slow convergence rate. The computational complexity of this algorithm lies somewhere in the middle between the block-based algorithms and the Sample RLS-DRMTA algorithm.

We have proposed the CG-DRMTA algorithm to overcome the high computational complexity of the Sample RLS-DRMTA algorithm, the slow convergence rate of the AP-DRMTA algorithm and the non-robustness of the Block DRMTA algorithm. The CG-DRMTA algorithm offers a low computational complexity, as it is one of the block-based algorithms. It has a faster convergence rate than the AP-DRMTA algorithm. Also, it offers a lower BER than the Block RLS-DRMTA algorithm for high number of users or low level of E_b/N_0 .

Based on computational complexity/BER performance tradeoff, Table 6.1 shows the recommended algorithm for each set of simulation conditions.

Table 6.1: The Recommended Algorithm to Be Used in the Different conditions.

Condition	Recommended Algorithm
AWGN Channel, $E_b/N_0 = 8$ dB, Small Number of Users	Block RLS-DRMTA
AWGN Channel, $E_b/N_0 = 8$ dB, Large Number of Users	AP-DRMTA
AWGN Channel, $E_b/N_0 = 4$ dB, Small Number of Users	CG-DRMTA
AWGN Channel, $E_b/N_0 = 4$ dB, Large Number of Users	AP-DRMTA
Dynamic Channel, $E_b/N_0 = 8$ dB, Small Number of Users, Angular Velocity= $2\pi/s$	LS-DRCMA, Sample RLS-DRMTA
Dynamic Channel, $E_b/N_0 = 8$ dB, Small Number of Users, Angular Velocity= $16\pi/s$	LS-DRCMA, Sample RLS-DRMTA

In the Dynamic Channel, the advantage of the large improvement in BER performance of the Sample RLS-DRMTA algorithm over the LS-DRMTCMA algorithm came at the cost of a relatively large increase in computational complexity. This is why we recommend using either algorithm.

6.2 Future Work

Some suggestions for future work based on this thesis are listed below

1. We have seen from the results that the Block RLS-DRMTA, the Sample RLS-DRMTA, the AP-DRMTA and the CG-DRMTA each has some advantages and some disadvantages. Also, the performance of each of these algorithms depends on the environment. Thus, different algorithms can be implemented at the base station and a certain algorithm has to be developed to switch between these algorithms according to the channel or the level of the input E_b/N_0 , in order to optimize the performance with the level of complexity.
2. We can study the effect of using different values of the forgetting factor in the Sample RLS-DRMTA, different projection orders and different block sizes in case of the AP-DRMTA algorithm.
3. We can combine the proposed algorithms in this thesis with the RAKE receiver as done in [80] to improve the performance of the algorithms in the case of multipath

channel. It is also possible to combine the proposed algorithms with the auxiliary multi-user detection technique to reduce the capture effect as in [79].

4. In this research, certain assumptions have been made in simulating the algorithms. The most important assumptions are that there is no multipath effect or fading in the channel, the time delay of the desired user is perfectly detected and perfect power control in the system. It may be worthwhile to study, on an individual basis, the algorithms proposed in this thesis in systems where these assumptions have not been made.
5. Finally, in this thesis, all the beamforming techniques are used at the base station only for the reverse link. It will be a challenge to modify the algorithms so as to work at the mobile terminal.

References

- [1] T. Chulajata and H. M. Kwon, "An adaptive array antenna with no phase calibration for cdma2000 reverse link," *Proc. IEEE Military Communication Conference*, vol. 2, pp. 816 -820, 2000.
- [2] M. Chryssomallis, "Smart Antennas," *IEEE Antennas and Propagation Magazine*, vol. 42, no. 3, pp. 129-136, June 2000.
- [3] J. C. Liberti and T. S. Rappaport, *Smart Antennas for Wireless Communication*, New York: Prentice Hall, 1999.
- [4] G. V. Tsoulos, "Smart antennas for mobile communication systems: benefits and challenges," *IEEE Electronics & Communication Engineering Journal*, vol. 11, issue: 2, pp. 84 -94, April 1999.
- [5] S. Wang, M. Green and M. Malkawi, "Mobile positioning technologies and location services," *Proc. IEEE Radio and Wireless Conf., RAWCON 2002*, pp. 19-12, 2002.
- [6] P. H. Lenhe and M. Pettersen, "An overview of smart antenna technology for mobile communications systems," *IEEE Communications Surveys and Tutorials*, vol. 2, no. 4, fourth quarter 1999.
<http://www.comsoc.org/livepubs/surveys/public/4q99issue/lehne.html>
- [7] M. J. Feuerstein, "Applications of smart antennas in cellular networks," *Proc. IEEE International Symposium on Antennas and Propagation*, vol. 2, pp 1096-1099, Aug. 1999.

- [8] M. Lotter and P. van Rooyen, "An overview of Space Division Multiple Access techniques in cellular systems," *Proc. IEEE South African Symposium on Comm. and Signal Processing SCOMSIG 98*, pp. 161-164, Sept. 1998.
- [9] R. Roy, "An overview of smart antenna technology and its application to wireless communication systems," *IEEE Inter. Conf. on Personal Wireless Comm.*, pp. 234-238, Dec. 1997.
- [10] A. Naguib, A. Paulraj and T. Kailath, "Capacity improvement with base-station antenna arrays in cellular CDMA," *IEEE Trans. On Veh. Technol.*, vol. 43, no. 3, pp. 691-698, Aug. 1994.
- [11] Z. Yong and F. Zhenghe, "Capacity analysis of CDMA networks with smart antenna," *IEEE Canadian Conf. on Elec. and Computer Eng.*, vol. 2, pp. 1333-1336, 2001.
- [12] S. Bellofiore, J. Foutz, C. A. Balanis and A. Spanias, "Smart-Antenna systems for mobile communication networks part 1: overview and antenna design," *IEEE Antennas and Propagation Magazine*, vol. 44, no. 3, pp. 145-154, June 2002.
- [13] K. K. Wong, K. B. Lataief and R. D. Murch, "Investigating the performance of smart antenna system at the mobile and base stations in the down and uplinks," *Proc. IEEE VTC'98*, vol. 2, pp. 880-884, May 1998.

- [14] N. Herscovici, C. Christodoulou and A. El-Zooghby, "Potentials of smart antennas in CDMA systems and uplink improvement," *IEEE Antennas and Propagation Magazine*, vol. 43, no. 53, pp. 172-177, Oct. 2001.
- [15] A. Bhode and P. Perini, "An overview of smart antenna technology for wireless communication," *Proc. IEEE Aerospace Conf.*, vol. 2, pp. 2/875-2/883, 2001.
- [16] S. U. Pillai, *Array Signal Processing*, New York: Springer-Verlag, 1989.
- [17] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, New York: John Wiley & Sons, 1981.
- [18] P. van Rooyen, M. Lotter and D. van Wyk, *Space-Time processing for CDMA mobile communications*, Kluwer Academic Publishers, 2001.
- [19] L. C. Godara, "Application of antenna array to mobile communications. 2: Beamforming and direction of arrival consideration," *IEEE Proc.*, vol.85, pp 1195-1245. Aug. 1997.
- [20] B. D. Van Veen and K. M. Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE ASSP Magazine*, pp. 4-24, April 1988.
- [21] A. Boukalov and S. G. Haggman, "System aspects of smart antenna technology in cellular wireless communications-An overview," *IEEE Trans. on Microwave Theory and Techniques*, vol. 48, no. 6, pp. 919-929, Jun. 2000.

- [22] J. Winters, "Optimum combining in digital mobile radio with co-channel interference," *Proc. IEEE on Selected Areas in Comm.*, vol. 2, no. 4, pp. 528-539, Jul. 1984.
- [23] J. Capon, "High resolution frequency-wavenumber spectral analysis," *Proc. of the IEEE*, vol. 57, no. 8, pp. 1408-1418, Aug. 1969.
- [24] S. Haykin, *Adaptive Filter Theory*, Upper Saddle River, New Jersey: Prentice Hall, 3rd edition, 1996.
- [25] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Englewood Cliffs, New Jersey: Prentice Hall, 1996.
- [26] Z. Rong, "Simulation of adaptive array algorithms for CDMA systems." Master's Thesis, Virginia Tech., Blacksburg, VA, September 1996.
- [27] R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Ant. and Prop.*, vol. AP-34, no. 3, pp. 276-280, 1986.
- [28] 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.
- [29] B. G. Agee, S. V. Schell, and W. A. Gardner, "Spectral Self-Coherence Restoral: A new approach to blind adaptive signal extraction using antenna arrays," *Proc. IEEE*, vol.78, no. 4, pp. 753-767, 1990.

- [30] J. R. Treichler and B. 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.
- [31] 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.
- [32] K. Hilal and P. Duhamel, "A blind equalizer allowing soft transition between the constant modulus and decision directed algorithm for PSK modulated signal," *Proc. IEEE Int. Conf. Communication*, vol. 2, pp. 1144-1148, 1993.
- [33] J. Nagumo and A. Noda, "A learning method for system identification," *IEEE Trans. on Automatic Control*, vol. 12, no. 3, pp. 282-287, Jun 1967.
- [34] R. R. Bitmead and B. D. Anderson, "Performance of adaptive estimation algorithms in dependant random environments," *IEEE Trans. on Automatic Control*, vol. AC-25, pp. 788-794, 1980.
- [35] T. Hsia, "Convergence analysis of LMS and NLMS adaptive algorithms," *Proc. IEEE ICASSP*, vol.8, pp. 667-670, Apr. 1983.
- [36] D. T. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms," *IEEE Trans. Signal Processing*, vol. 41, pp. 2811-2825, 1993.
- [37] O. L. Frost, III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, pp. 926-935, 1972.

- [38] 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.
- [39] M. J. Rude and L. J. Griffiths, "Incorporation of linear constraints into the constant modulus algorithm," *Proc. IEEE ICASSP*, pp. 968-971, 1989.
- [40] N. Kikuma, M. Fujimoto, N. Inagaki, "Rapid and stable optimization of CMA adaptive array by Marquardt method," *IEEE International Symposium on Antennas and Propagation*, vol. 1, pp. 102-105, 1991.
- [41] R. Pickholtz and K. Elbarbary, "The recursive constant modulus algorithm: A new approach for real-time array processing," *Proc. 27th Asilomar Conf. on Signals, Systems and Computers*, pp. 627-632, 1993.
- [42] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Comm. Jpn.*, vol. 67-A, No.5, pp 19-27, 1984.
- [43] S. L. Gay, "A fast converging, low complexity adaptive filtering algorithm," *Proc. of 3rd Inter. Workshop on Acoustic Echo Control*, pp. 223-226, Sep. 1993.
- [44] Q. G. Liu, B. Champagne and K. C. Ho, "On the use of a modified Fast Affine Projection algorithm in subbands for acoustic echo cancellation," *Proc. IEEE Workshop on Digital Signal Processing.*, pp. 354-357, Sep. 1996.

- [45] G. Rombouts and M. Moonen, "A sparse block exact affine projection algorithm," *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 2, pp. 100-108, Feb. 2002.
- [46] M. Tanaka, S. Makino and J. Kojima, "A block exact fast affine projection algorithm," *IEEE Trans. on Speech and Audio Processing*, vol. 7, no. 1, pp 79-86, Jan. 1999.
- [47] M. Tanaka, Y. Kaneda and S. Makino, "Fast projection algorithm and its step size control," *IEEE ICASSP-95*, pp. 945-948, 1995.
- [48] S. Gay and S. Tavathia, "The fast affine projection algorithm," *Proc. IEEE Inter. Conf. Acoustics, Speech, Signal Processing*, vol. 5, pp. 3023-3026, May 1995.
- [49] S. C. Douglas, "The fast affine algorithm for active noise control," *Proc. 29th Asilomar Conf. on Signals, Systems and Computers*, vol. 2, pp. 1245-1249, 1995.
- [50] Y. R. Zheng, R. A. Goubran and M. El-Tanany, "Coherent interference suppression with an adaptive array using spatial affine projection algorithm," *IEEE VTC' 2000*, vol. 1, pp 105-109, 2000.
- [51] Y. R. Zheng and R. A. Goubran "Adaptive beamforming using affine projection algorithms," *Proc. IEEE 5th Inter. Conf. on Signal Processing, WCCC-ICSP 2000*, vol. 3, pp. 1929-1932, 2000.

- [52] H. Basbes, Y. B. Jemaa and M. Jaidane, "Exact convergence analysis of affine projection algorithm: the finite alphabet inputs case," *Proc. IEEE ICASSP'99*, vol. 3, pp. 1669-1672, Mar. 1999.
- [53] G. K. Boray and M. D. Srinath, "Conjugate gradient techniques for adaptive filters," *IEEE Trans. on Circuits and Systems*, vol. 39, no. 1, Jan. 1992.
- [54] P. S. Chang and A. N. Willson Jr., "Analysis of conjugate gradient algorithms for adaptive filtering," *IEEE Trans. on Signal Processing*, vol. 48, pp. 409-418, Feb. 2000.
- [55] T. Bose and M. Q. Chen, "Conjugate gradient method in adaptive bilinear filtering," *IEEE Trans. on Signal Processing*, vol. 43, pp. 1503-1508, Jan. 1995.
- [56] G. D. Mandyam, N. Ahmed and M. D. Srinath, "Adaptive beamforming based on the conjugate gradient algorithm," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 343-347, Jan. 1997.
- [57] D. Shim and F. Alam, "A new adaptive downlink beamforming method for WCDMA system," *IEEE VTC'01*, vol. 1, pp. 157-161, 2001.
- [58] S. Choi, Y. L. Lee and K. Hirasawa, "Real-time design of a smart antenna system utilizing a modified conjugate gradient method for CDMA-based mobile communications," *IEEE VTC'97*, vol. 1, pp. 183-187, May 1997.

- [59] J. K. Apolinario, Jr., and M. L. R. de Campos, "The constrained conjugate gradient algorithm," *IEEE Signal Processing Letters*, vol. 7, no. 12, pp. 351-354, Dec. 2000.
- [60] P. Karttunen and B. Baghaie, "Conjugate gradient based signal subspace mobile user tracking," *IEEE VTC'99*, vol. 2, pp. 1172-1176, Jul. 1999.
- [61] R. J. Sanchez, "CDMA digital cellular overview and field trial results," *IEEE International Symposium on Technologies for Wireless Application Digest, MMT-S* 1995, pp. 37-43, Feb. 1995.
- [62] R. L. Pickholtz and L. B. Milstein, "Spread spectrum for mobile communications," *IEEE Trans. on Veh. Technol.*, vol. 40, no. 2, pp. 313-322, May 1991.
- [63] W.C.Y. Lee, "Overview of cellular CDMA," *IEEE Trans. on Veh. Technol.*, vol. 40, no. 2, pp. 291-302, May 1991.
- [64] 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.
- [65] M. W. Oliphant, "The mobile phone meets the internet," *IEEE Spectrum*, pp. 20-28, Aug. 1999.

- [66] T. Ojanpera and R. Prasad, "An Overview of air interface multiple access for IMT-2000/UMTS," *IEEE Communications Magazine*, vol. 36, pp. 88-95, Sept. 1998.
- [67] E. Dahlman, B. Gudmundson, M. Nilsson, and J. Skold, "UMTS/IMT-2000 based on Wideband CDMA," *IEEE Communications Magazine*, vol. 36, pp. 70-80, Sept. 1998.
- [68] E. Dahlman, P. Beming, J. Knutsson, F. Ovesjo, M. Persson, and C. Roobol, "WCDMA- The radio interface for mobile multimedia communications," *IEEE Trans. on Veh. Technol.*, vol. 47, no. 4, pp. 1105-1118, Nov. 1998.
- [69] J. S. Thompson, P.M. Grant, and B. Mulgrew, "Smart antenna arrays for CDMA systems," *Proc. IEEE on Personal Communications*, vol. 3, no. 5, pp. 16-25, Oct. 1996.
- [70] B. Sklar, *Digital Communication: Fundamentals and Applications*, Englewood Cliffs, New Jersey: Prentice Hall, 1988.
- [71] K. H. Karkkaninen and P. A. Leppanen, "Comparison of the performance of some linear spreading code families for asynchronous DS/SSMA systems," *Proc. IEEE Military Communication Conference, MILCOM'91*, vol. 2, pp. 784-790, Nov. 1991.

- [72] A. M. Turkmani and U. S. Goni, "Performance evaluation of Maximal-length, Gold and Kasami codes as spreading sequences in CDMA systems," *IEEE 2nd International Conf. on Universal Personal Comm.*, vol. 2, pp. 979-974, Oct. 1993.
- [73] E. H. Dinan and B. Jabbari, "Spreading codes for direct sequence CDMA and Wideband CDMA cellular networks," *IEEE Communications Magazine*, vol. 36, no. 9, pp. 48-54, Sept. 1998.
- [74] B. A. Agee, "Blind separation and capture of communication signals using a Multitarget Constant Modulus beamformer," *proc. IEEE Military Comm. Conf.*, Oct. 1989 pp 19.2.1 -19.2.7.
- [75] Z. Rong, S. T. Rappaport, P. Petrus and J. H. Reed, "Simulation of Multitarget Adaptive Array Algorithm for Wireless CDMA systems," *IEEE VTC'97*, Vol.1, pp 1-5.
- [76] Z. Rong, S. T. Rappaport, P. Petrus and J. H. Reed, "Despread-Respread Multi-Target Constant Modulus Array for CDMA systems," *IEEE Comm. Letters*, vol.1, no.4, pp. 114-116, July, 1997.
- [77] F. Dominique and J. H. Reed, "Despread data rate update multi-target adaptive array for CDMA signals," *IEEE Electronics Letters*, vol. 33, no. 2, pp. 119-121, Jan. 1997.
- [78] Z. Du, P. Gong, and W. Wu, "Block-based RLS Despread Respread Multi-Target Array: algorithm and performance," *IEEE VTC'2001*, Vol.1, pp 190-193. 2001.

- [79] F. R. Cavalcanti and J. M. Romano, "Reducing capture effects in despread respread multi-target adaptive arrays in CDMA systems," *IEEE Global Telecommunications Conf., GLOBECOM'99*, vol. 4, pp. 2212-2216, Dec. 1999.
- [80] A. R. de Matos, L. M. de Mendonca, and A. G. D'Assuncao, "A novel Multitarget adaptive array algorithms for wireless CDMA systems," *IEEE Inter. Sym. on Antenna and Propagation*, vol. 4, pp. 10-13, July 2001.
- [81] A. R. de Matos, L. M. de Mendonca, and A. G. D'Assuncao, "Adaptive array algorithms for wireless CDMA systems for multipaths," *Proc. IEEE Conf. on Microwave and Optoelectronics, IMOC 2001*, vol. 1, pp. 31-34, Aug. 2001.
- [82] C. Carlemalm and A. Logothetis, "Blind signal detection and parameter estimation for an asynchronous CDMA system with time-varying number of transmission paths," *IEEE 9th Workshop on Statistical Signal and Array Processing*, pp. 296-299, Sept. 1998.
- [83] S. Bensley and B. Aazhang, "Subspace-based channel estimation for code division multiple access communication systems," *IEEE Trans. on Comm.*, vol. 44, no. 8, pp. 1009-1020, 1996.
- [84] E. Strom, S. Parkvall, S. Miller and B. Ottersten, "Propagation delay estimation in asynchronous direct sequence code division multiple access systems," *IEEE Trans. on Comm.*, vol. 44, no. 4, pp. 84-93, 1996.

- [85] C. Weber and W. Alem, "Demod-Remod coherent tracking receiver for QPSK and SQPSK," *IEEE Trans on Comm.*, vol. 28, no. 12, pp. 1945-1954, Dec. 1980.
- [86] R. Gooch, C. Jorgensen and M. Ready, "The Demod-Remod technique for demodulation co-channel FSK signals," *Proc. 25th Asilomar Conf. on Signals, Systems and Computers*, vol. 2, pp. 704-709, Nov. 1991.
- [87] H. V. Poor and X. Wang, "Code-aided interference suppressions for DS/CDMA communications-Part II: Parallel blind adaptive implementations," *IEEE Trans on Comm.*, vol. 45, pp. 1112-1122, Sept. 1997.
- [88] B. N. Thanh, V. Krishnamurthy, and R. J. Evans, "Detection-aided recursive least squares adaptive multi-user detection in DS/CDMA," *IEEE Signal Processing Letters*, vol. 9, no. 8, pp. 229-232, Aug. 2002.
- [89] N. A. Mohamed and J. G. Dunham, "Low-complexity combined antenna array and interference cancellation DS-CDMA receiver in multi-path fading channels," *Proc. IEEE on Selected Areas in Communications*, vol. 20, no. 2, pp. 248-256, Feb. 2002.
- [90] J. G. Proakis, *Digital Communications*, New York: McGraw-Hill, 1995.
- [91] J. Litva and T. K. Lo, *Digital Beamforming in Wireless Communications*. Boston, MA: Artech House, 1996.

Appendix A: The GUI

A *Graphical User Interface* (GUI) has been created to illustrate the performance of the different algorithms. There are three parts of the GUI. The first part of the GUI deals with the signal through the AWGN channel (from the mobile station to the smart antenna at the base station). The second part of the GUI deals with the different algorithms used in the simulation; it shows the beam pattern on using the algorithm specified by the user of the GUI. The third part of the GUI shows an animation for how the algorithm changes with time as the mobile stations move in the direction specified by the user of the GUI.

Figure A.1 shows the first part of the GUI, where the user of the GUI decides the number of mobile stations to be used in the simulation (1- 8 mobile stations), the number of antenna elements in the Uniform Linear Array and the SNR. Then for each of the mobile stations the following parameters should be specified: the distance between the mobile station and the base station, the DOA of the mobile station's signal at the base station, the velocity of the mobile station and the direction of movement of this mobile station with respect to the x-axis.

The user of the GUI can choose what to draw, either the original transmitted signal for a specific mobile station or the combination of the signals at a specific antenna element. The number of bits used for each user is 200 bits. In drawing the original transmitted signal we draw 20 bits just to show how the CDMA signal of the mobile

station looks like. When the user chooses to go to the second level of the GUI a new window opens.

Figure A.1: The First Part of the GUI

In the second part of the GUI (which is a new window), the user of the GUI can choose the mobile station that will be treated as the desired mobile station, the algorithm that will be used to form the beam pattern and how to plot the beam pattern; either in polar or normal form. Figure A.2 shows the second part of the GUI.

Figure A.2: The Second Part of the GUI

If the user chooses to run the animation, a new window opens. This is the third part of the GUI.

Figure A.3: The Third Part of the GUI at the beginning of the animation.

In the third part of the GUI, the animation for how the mobile stations move is shown in the left part and the animation for how the beamformer tracks the desired mobile station with high gain is shown in the right part.

In this simulation we still assume that the channel is an AWGN channel. We repeat simulating the 200 bits of the transmitted signal for each of the mobile stations at the new positions for the mobile stations that have been specified by user in the first part of the

GUI. We use 30 frames (each frame represents one second) for each algorithm in the animation simulation.

Figure A.3 shows a typical example for the third part of the GUI at the beginning of the animation, where there are 4 mobile stations at angles 30° , 60° , 90° , and 120° . The CG-DRMTA algorithm is used.

Figure A.4 shows the same case at the end of the animation where the mobile stations move to the directions of 0° , 90° , 180° , and 270° respectively.

Figure A.4: The Third Part of the GUI at the end of the animation.