

UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'UNIVERSITÉ DU QUÉBEC À TROIS-RIVIÈRES

COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN GÉNIE ÉLECTRIQUE

PAR
ELIE H. SARRAF

ÉVALUATION DES TECHNIQUES DE RÉCEPTION MULTI-ANTENNES DANS UN
SYSTÈME DS-CDMA

JUIN 2007

Université du Québec à Trois-Rivières

Service de la bibliothèque

Avertissement

L'auteur de ce mémoire ou de cette thèse a autorisé l'Université du Québec à Trois-Rivières à diffuser, à des fins non lucratives, une copie de son mémoire ou de sa thèse.

Cette diffusion n'entraîne pas une renonciation de la part de l'auteur à ses droits de propriété intellectuelle, incluant le droit d'auteur, sur ce mémoire ou cette thèse. Notamment, la reproduction ou la publication de la totalité ou d'une partie importante de ce mémoire ou de cette thèse requiert son autorisation.

Abstract

The *SA* is set to play a significant role in the development of the next-generation wireless communication system. *SAs* are a new technology for wireless systems that use a fixed set of antenna elements in an array.

Technological progress has recently changed the introduction of multiple antenna elements at the receiver unit of wireless access points and mobile terminals from a purely theoretical concept to a practical issue in current and future wireless communication systems.

The signals from these antenna elements are combined to form a movable beam pattern that can be steered, using either digital signal processing, or *RF (Radio Frequency)* hardware, to a desired direction that tracks mobile units (pedestrians, cars,) as they move. This allows the *SA* system to focus the *RF* resources on a particular subscriber, while minimizing the impact of noise, interference, and other effects that can degrade signal quality.

The challenge lies in two approaches, the first is purely mathematical, called signal processing and the second is purely hardware, called implementation. These two approaches are highly correlated in order to achieve good performance especially in a real time requirements or applications, which is the case.

In short, software and hardware engineers are facing an enormous challenge in designing, developing and implementing a reliable system core combining an efficient Algorithm (Software) with an intelligent Architecture (Hardware) at a moderate cost for

the next generation of wireless communications systems. This challenge, this trade-off
this future vision, consist the core of our work.

Résumé

Un des principaux avantages des systèmes des antennes intelligentes réside dans l'augmentation potentielle du nombre d'utilisateurs dans le réseau cellulaire d'une part, et l'accroissement de l'éventail des services offerts par le système cellulaire d'une autre part. L'intérêt de ces systèmes est leur capacité à réagir automatiquement à un environnement complexe dont l'interférence est connue à priori. L'augmentation du nombre d'utilisateurs et l'amélioration de la qualité du service offert représentent un atout pour les futurs systèmes sans fils de la troisième et quatrième génération.

Ces systèmes reposent sur des antennes réseau, des dispositifs pour calculer les angles d'arrivées et des outils numériques de synthèse. Ces derniers attribuent des poids aux éléments de l'antenne réseau afin d'optimiser le signal de sortie.

Une antenne réseau adaptative peut être définie comme un réseau capable de modifier son diagramme de rayonnement. Cette modification est réalisée grâce à un algorithme performant implémenté et apte à répondre aux spécifications désirées. Plus spécifiquement, en considérant un système de communication à temps réel.

La problématique se figure dans deux aspects physiquement différents, mais en terme d'application ces deux aspects sont très dépendants entre eux.

Le premier problème est un aspect logiciel (Software) et mathématique appelé «Traitement de signal», il comprend les méthodes mathématiques appliquées aux antennes intelligentes. Le deuxième problème est un aspect matériel (Hardware) appelé

« Implémentation », il comprend les circuits intégrés (*DSP, FPGA, ASIC, ...*). Dans notre cas on utilise les composantes programmables *FPGA*.

L'objectif de notre étude est de trouver pour chaque aspect la meilleure combinaison dans le but de donner une solution optimale qui lui correspond.

Preface

This thesis is based on a work that was carried out at the Laboratory of Signals and Systems Integration *LSSI*, and at the Laboratory of the Canadian Microelectronics Corporation, Department of Electrical Engineering, Université du Québec à Trois-Rivières.

The purpose of this thesis is to provide a comparative study of the *Smart Antennas (SAs)* aspect, previous, current and future approaches. In addition, this work presents a performance evaluation of the mathematical algorithms and methods used for *SAs*. Moreover, to minimize the time-to-market, this work provides an evaluation of the hardware implementation on *FPGA (Field Programmable Gate Array)* as well. According to the results, the best candidate will be able to drive the performance up and the costs down.

Several thousand years ago, King Solomon wrote "The end of a matter is better than its beginning, and patience is better than pride".

At the end of this matter, I'd like to thank my family for their support and patience. I acknowledge my gratitude to my research advisor Dr. Daniel Massicotte as well as my research co-advisor Dr. Adel-Omar Dahmane, for providing me the tools I needed to complete this work, giving me the opportunity to explore both past and present ideas and issues.

I would also like to acknowledge my colleague and friend Dr. Messaoud-Ahmed Ouameur for his insightful analysis. He was very helpful in guiding me toward a qualitative methodology.

My sincere thanks go to my brother Walid for always helping out in a crisis, and for his financial support.

Ultimately, however all thanks are due to the beginning and end, the Alpha and Omega, the creator, sustainer, and redeemer of my soul, Jesus Christ.

Table of Contents

ABSTRACT	I
RÉSUMÉ	III
PREFACE	V
TABLE OF CONTENTS	VII
LIST OF FIGURES	IX
LIST OF TABLES	XI
LIST OF ACRONYMS	XII
CHAPTER 1 - INTRODUCTION	1
1.1 INTRODUCTION	1
1.2 AIM OF THESIS	3
1.3 OVERVIEW OF CONTENT	4
CHAPTER 2 - SMART ANTENNAS STATE-OF-THE-ART	5
2.1 INTRODUCTION	5
2.2 THE NEED FOR SMART ANTENNAS	6
2.3 MULTIPLE ACCESS SCHEMES	9
2.3.1 <i>Frequency Division Multiple Access (FDMA)</i>	10
2.3.2 <i>Time Division Multiple Access (TDMA)</i>	11
2.3.3 <i>Code Division Multiple Access (CDMA)</i>	12
2.3.4 <i>Space Division Multiple Access (SDMA)</i>	13
2.4 SMART ANTENNA’S SYSTEM.....	15
2.4.1 <i>A First Basic Approach</i>	16
2.4.2 <i>A Second systematic Approach</i>	18
2.4.3 <i>Two different functionality approaches</i>	20
2.5 SWITCHED SMART ANTENNAS (SBA).....	20
2.5.1 <i>Description</i>	20
2.5.2 <i>Advantages</i>	21
2.5.3 <i>Drawbacks</i>	22
2.6 ADAPTIVE SMART ANTENNAS (TBA).....	23
2.6.1 <i>Description</i>	23
2.6.2 <i>Advantages</i>	25
2.6.3 <i>Drawbacks</i>	28
2.7 GENERAL BENEFITS.....	28
2.8 FUTURE CHALLENGES	31
2.9 SUMMARY	32
CHAPTER 3 - PERFORMANCE EVALUATION OF BEAMFORMING TECHNIQUES	33
3.1 INTRODUCTION	33
3.2 SIGNAL MODEL.....	36

3.3	NON ADAPTIVE PILOT-CHANNEL AIDED BEAMFORMING TECHNIQUES	42
3.3.1	<i>Direct Approach (MRC)</i>	42
3.3.2	<i>Direct Approach (DMI)</i>	43
3.3.3	<i>Eigen-Decomposition Approach</i>	44
3.3.4	<i>Code Filtering Approach</i>	45
3.4	ADAPTIVE PILOT-CHANNEL AIDED BEAMFORMING TECHNIQUES	46
3.4.1	<i>Least Mean Squares (LMS)</i>	47
3.4.2	<i>Normalized Least Mean Squares (NLMS)</i>	48
3.4.3	<i>Noise Constrained Least Mean Squares (NC-LMS)</i>	49
3.4.4	<i>Recursive Least Squares (RLS)</i>	50
3.4.5	<i>Set Membership Identification (SMI)</i>	51
3.5	SIMULATIONS RESULTS.....	52
3.5.1	<i>PTR Effect</i>	54
3.5.2	<i>Number of Antenna Effect</i>	55
3.5.3	<i>Number of Users Effect</i>	56
3.6	SUMMARY	57
CHAPTER 4 - FPGA IMPLEMENTATION OF BEAMFORMING TECHNIQUES		58
4.1	INTRODUCTION.....	58
4.2	IMPLEMENTATION OF BEAMFORMING ALGORITHMS	60
4.3	FPGA IMPLEMENTATION OF MRC AND NC-LMS	60
4.3.1	<i>Complexity analysis</i>	60
4.3.2	<i>Proposed architectures</i>	61
4.3.3	<i>Implementation Technique</i>	62
4.3.4	<i>Hardware resources</i>	65
4.3.5	<i>Quantization study</i>	67
4.4	FPGA DESIGN AND IMPLEMENTATION OF DMI	68
4.4.1	<i>Redesign of DMI</i>	68
4.4.2	<i>Complexity analysis</i>	70
4.4.3	<i>Quantization study</i>	71
4.4.4	<i>Proposed architecture</i>	73
4.4.5	<i>Implementation technique</i>	76
4.4.6	<i>Mapping to VLSI architectures</i>	77
4.4.7	<i>Hardware resources</i>	82
4.4.8	<i>Tradeoffs for maximum number of users</i>	83
4.5	SUMMARY	86
CHAPTER 5 - CONCLUSION.....		88
REFERENCES		92
APPENDIX A - RESULTS WITH A LOWER COMPLEXITY PLATFORM.....		99
APPENDIX B - RÉSUMÉ EN FRANÇAIS		116

List of figures

	Page
Chapter 2	
Figure 2.1	Wireless communication environment 6
Figure 2.2	Problems related to wireless communications 8
Figure 2.3	Multiple access schemes 12
Figure 2.4	Concept of a <i>CDMA</i> system 13
Figure 2.5	The new dimension: <i>Space Division Multiple Access</i> 15
Figure 2.6	Two analogies 18
Figure 2.7	SA system and functionality 19
Figure 2.8	Switched-beam-array 21
Figure 2.9	Adaptive-beam-array 24
Figure 2.10	Comparison between the <i>ABA</i> and the <i>TBA</i> 26
Chapter 3	
Figure 3.1	Delay and Sum Beamformer 35
Figure 3.2	PTR effect 54
Figure 3.3	Number of antenna effect 55
Figure 3.4	Number of users effect 56
Chapter 4	
Figure 4.1	Proposed implementation of <i>MRC (a) and NC-LMS (b)</i> 62
Figure 4.2	<i>MRC</i> mounted into Xilinx Blocks 63
Figure 4.3	<i>NC-LMS</i> mounted into Xilinx Blocks 64

Figure 4.4	Quantization study results for <i>MRC</i> and <i>NC-LMS</i> using $N=8$, $K=5$ users, $PTR=0$ dB, mobile speeds of 60Km/h, carrier frequency 2GHz and chip rate of 1.25Mchip/s, and <i>Simulink</i> blocks	68
Figure 4.5	Fixed point results of the DMI technique (24-31). $N = 16$ (Processing gain), $K = 10$ users, $PTR=-6$ dB	72
Figure 4.6	Proposed block diagram of the proposed <i>VLSI</i> architecture for the <i>SD-DMI</i> with the corresponding data flow	75
Figure 4.7	Legend of operators	77
Figure 4.8	Block b1	78
Figure 4.9	Block b2	79
Figure 4.10	Block b3	79
Figure 4.11	Block b4	80
Figure 4.12	Block b5	81
Figure 4.13	Block b6	81
Figure 4.14	Block b7	82

List of tables

		Page
Chapter 4		
Table 4.1	Number of arithmetic operations per iterations for <i>MRC</i> and <i>NC-LMS</i>	61
Table 4.2	Hardware resources for one antenna	65
Table 4.3	Hardware resources for four antennas	66
Table 4.4	Percentage of slices <i>S</i> and multipliers <i>M</i> for four antennas	67
Table 4.5	Arithmetic Complexity of the SD-DMI (proposed) versus <i>RLS</i> , per antenna and per user.....	71
Table 4.6	Time steps derivation from the SD-DMI scheduling of equations (4.2 – 4.9)	74
Table 4.7	Hardware resources for different blocks in the architecture for $L=4$ and $J=10$	82
Table 4.8	Maximum number of users K^{MAX} over different FPGA devices in the case of $L=4$, $J=10$ and $M=1$	86

List of Acronyms

A

ABA	Adaptive Beam Array
AB	Adaptive Beamforming
ASIC	Application Specific Integrated Circuit
ASSP	Application Specified System Processor
A/D	Analogic to Digital

B

BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BRAM	Block Random Access Memory
BS	Base Station

C

CCI	Co-Channel Interference
CDMA	Code Division Multiple Access
CPLD	Complex Programmable Logic Device
CSI	Channel State Information

D

dB	Deci-Bel
DOA	Direction of Arrival
DS-CDMA	Direct Sequence Code Division Multiple Access
DSP	Digital Signal Processor
D/A	Digital to Analogic

E

EM	Electromagnetic
EVD	Eigen Value Decomposition

F

FDMA	Frequency Division Multiple Access
FIR	Finite Impulse Response
FF	Flip Flop
FPGA	Field Programmable Gate Array

H

HDL	Hardware Description Language
-----	-------------------------------

I

IC	Integrated Circuit
IF	Intermediate Frequency
ISI	Inter-Symbol Interference
IPN	Interference Plus Noise
IOB	Input Output Bound
I/O	Input Output

L

LMS	Least Mean Squares
LUT	Look Up Table

M

MAC	Multiplier Accumulator
MCU	Maximum Core Unit

MAI	Multiple Access Interference
MIMO	Multiple Input Multiple Output
MMSE	Minimum Mean Square Error
MRC	Maximum Ratio Combining
MSINR	Maximum Signal plus Interference Noise Ratio
MUD	Multi-user Detector

N

NC-LMS	Noise Constrained Least Mean Squares
NLMS	Normalized Least Mean Squares

O

OBE	Optimal Bounding Ellipsoids
-----	-----------------------------

P

PCS	Personal Communication Service
PE	Process Element
POCR	Performance Over Complexity Ratio
PTR	Pilot to Traffic Power Ratio

Q

QPSK	Quadrature Phase Shift Keying
------	-------------------------------

R

RF	Radio Frequency
RLS	Recursive Least Squares

S

SA	Smart Antenna
----	---------------

SB	Switched Beamforming
SINR	Signal to Interference Noise Ratio
SMI	Set Membership Identification
SNOI	Signal Not Of Interest
SOI	Signal Of Interest
T	
TBA	Tracking Beam Array
TBUF	Tristate Buffer
TDMA	Time Division Multiple Access
U	
ULA	Uniform Linear Array
UMTS	Universal Mobile Telecommunication System
V	
VLSI	Very Large Scale Integration
VSLMS	Variable Step-size Least Mean Square
VHDL	VHSIC Hardware Description Language
VHSIC	Very High Speed Integrated Circuits
W	
WCDMA	Wideband Code Division Multiple Access
WLAN	Wireless Local Area Network
WLL	Wireless Local Loop

Chapter 1

Introduction

1.1 Introduction

Since the dawn of civilization, communication has been of foremost importance to mankind. In the first place, communication was accomplished by sound through voice. However as the distance of communication increased, numerous devices were introduced, such as horns, drums, and so forth.

At some point they used animals to send messages for long distances, e.g. pigeons. In addition to that, visual techniques were injected for even greater distances. For example, signal flags and smoke signals were used in the daytime while fireworks in the night. These optical communications utilize the light portion of the electromagnetic (*EM*) spectrum and it has only been in recent times that the *EM* spectrum, outside the visible region, has been adopted for communication, though the use of the radio.

The radio antenna may be defined as the structure associated with the region of transition between a guided wave and a free-space wave, or vice versa [KRA88]. In

another word, radio antennas coupled *EM* energy from one medium e.g. space to another such as waveguide, wire or coaxial cable.

Consequently, the applications of wireless communication systems such as radios, Bluetooth, cellular networks, *WLAN (Wireless Local Area Network)* and *MIMO (Multiple Input Multiple Output)* has erupted throughout the world and recent years have witness wireless communications relishing its fastest growth period in history.

This huge eruption in the applications of wireless communications is due to an enormous evolution of signal processing algorithms driving up the performance and the quality of service as well, and most importantly the evolution of hardware implementation driving down the design to lower power consumption, lower complexity and lower costs.

FPGAs have historically been found in high-end professional broadcast systems, network surveillance cameras and medical imaging equipment; their flexibility has made them very suitable for digital signal processing applications. Today the design flow for *FPGA* has been largely characterized by hardware centric approach.

The requirement is the exposure of the high computational efficiency of *FPGAs* matched by high bandwidth concurrent memory access and rich on-chip interconnectivity, combined with complete programmability. These requirements make *FPGAs* well suited for high efficient implementation of signal processing, packet processing and high performance computing applications [BOL06].

In our view, the hardware implementation aspect is much more crucial than the signal processing aspect, ever since the key of *time-to-market* is related to the hardware implementation and its efficiency whether it meets the real time requirements or not, and most importantly whether the cost is affordable or not.

1.2 Aim of thesis

The demand for high performance, better quality, lower complexity wireless communication systems, has led to the research and studies in this exciting topic.

Communications has become the key to momentous changes in the organization of businesses and industries worldwide as they themselves adjust to the shift toward an information economy. Consumers are in demand of more and more high-tech services, and these services require the developing of sophisticated algorithms and at the same time require an intelligent hardware implementation.

In this project we face the problem of cellular systems in the third generation of mobile communications. Thus, we evaluate existing methods, even we examined methods that were never used for *SAs*, we, and at some point, re-designed an existing algorithm, and this re-design enables us to explore more efficiency and parallelism in the algorithm which results in larger bandwidth and higher quality of service though more reliable communication.

The major goal of this work is to provide a performance evaluation of the existing numerical algorithms and methods used for *SAs*. The evaluation is in terms of measuring the *Bit Error Rate (BER)*, the *Pilot to Traffic Power Ratio (PTR)*, the number of antenna elements L , and the number of users K to be served by the *base station (BS)*. For an insightful analysis of the results and for better understanding a *DS-CDMA (Direct Sequence Code Division Multiple Access)* platform based on real time requirements has been developed and deployed for simulations. The purpose of this evaluation is to give an overview of the used methods, and to choose the appropriate method favorable for hardware implementation.

For hardware evaluation, three methods were chosen based on their complexity and performance wherein two different techniques have been used for implementation; rapid prototyping method and regular method or manual coding. The purpose of this evaluation is to give an overview of the two techniques and to find the suitable architecture for the suitable algorithm that meets our real time requirements.

1.3 Overview of content

The contents are organized as follows:

Chapter 2 provides an in-depth description of the *SA* technology. It discusses the need of this technology and where to use it, different access schemes, their characteristics and limitations. This chapter presents a description of the *SA* system, different approaches, advantages, drawbacks and future challenges.

Chapter 3 describes the mathematical aspect of the *SA* system, or the backbone of the system, so called *Beamforming*. This chapter discusses the *DS-CDMA* platform which has been used for simulations. Then it presents an in-depth description of the used algorithms, adaptive and non-adaptive approaches. Simulations are carried out for different schemes and parameters.

Chapter 4 describes the *FPGA* implementation of the three chosen algorithms. The implementation techniques, their characteristics and efficiency, then it describe the proposed architectures, the hardware resources, and the quantization study as well in order to determine the precision of the algorithm.

Lastly, chapter 5 draws a summary and concludes the thesis with future developments.

Chapter 2

Smart Antennas State-Of-The-Art

2.1 Introduction

With the limitation of the valuable *EM* spectrum, engineers are looking for new frontiers for the wireless communications battle. It would be very easy if we allocate for each user a frequency channel, but for a fixed bandwidth of spectrum, there is a fundamental limit on the number of radio channels that are realized by a mobile or a cellular communication system operating over the bandwidth [BHO01], [BOU00], and [BEL02a]. This can be considered as one of many reasons for the interest in *SA* technology.

A second reason is the multi-path phenomenon, when the transmitted radio signal is reflected by physical features/structures [BOU00] such as buildings, cars and other users. This reflection creates, as it is shown in Figure 2.1, multiple signal paths between the base station and the user terminal, and the consequences of this multi-path phenomenon are very dramatic on the wireless communications.

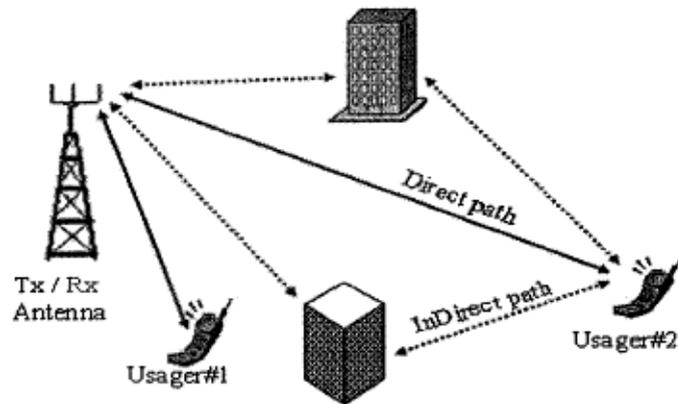


Figure 2.1: Wireless communication environment.

The interest in this promising technology is increasing since spatial processing is considered as the last frontier in the battle for cellular system capacity with a limited amount of the radio spectrum [BOU00] as mentioned above i.e. limited channel bandwidth satisfying a growing demand for a large number of mobiles on communications channels [BEL02a].

Unlike previous published work, which covered each area separately (The communications problems, the antenna array design, and the adaptive algorithms) for *SAs*, this work presents a very global overview of *SAs*, from the needs of their employment, to the performance of the adaptive algorithms used in *SAs*.

2.2 The need for smart antennas

For a better understanding of the analogy of signal propagation in a telecommunication model, a model of wireless communication is illustrated in Figure 2.1, this model contains a base station and two users communicating with each other.

One of the major problems is what we call *Co-Channel Interference (CCI)* [IECOE], [ALE04]. It occurs when the same carrier frequency reaches the same receiver from two separate transmitters i.e. the signals that miss an intended user can become interference for users on the same frequency in the same or adjoining cell. Here is a very good illustration of this type of interference. Envision a perfectly pool of water into which a stone is dropped. The waves that radiate outward from that point are uniform and diminish in strength evenly. This pure omni-directional broadcasting equate to one caller's signal originating at the terminal and going uplink. It is interpreted as one signal everywhere it travels.

Picture now a *Base Station (BS)* at some distance from the wave origin. If the pattern remains undisturbed, it is not a challenge for a base station to interpret the waves. But as the signal's waves begin to bounce off the edges of the pool, they come back (perhaps in a combination of directions) to intersect with the original wave pattern. As they combine, they weaken each other's strength. These are multi-path interference problems [IECOE]. The problem becomes more serious when a few more stones have being dropped in different areas of the pool, equivalent to other calls starting. How could a base station at any particular point in the pool distinguish which stone's signals were being picked up and from which direction? This multiple-source problem is called *Co-Channel Interference*.

There are two-dimensional analogies: to fully comprehend the distinction between callers and/or signal in the earth's atmosphere, a base station must possess the intelligence to place the information it analyzes in a true spatial context.

Another major problem is the multi-path. In a wireless system, the transmitted signal interacts with the environment in a very complex way. There are reflections from

large objects, diffractions from sharp edges and scattering waves, the result of these interactions is the presence of many signal components, which are referred to as multi-path signal at the receiver [ZHA01]. It is one of the major and serious problems in wireless communications systems and can be a limiting factor in a system [ALE04], [SHI97] (see Figure 2.1).

Many problems are associated with multi-path. One problem resulting from having unwanted reflected signals is that the phases of the waves arriving at the receiving station do not match. The phase of a radio wave is simply an arc of radio wave, measured in degrees, at a specific time [IECOE] (see Figure 2.2 (a)).

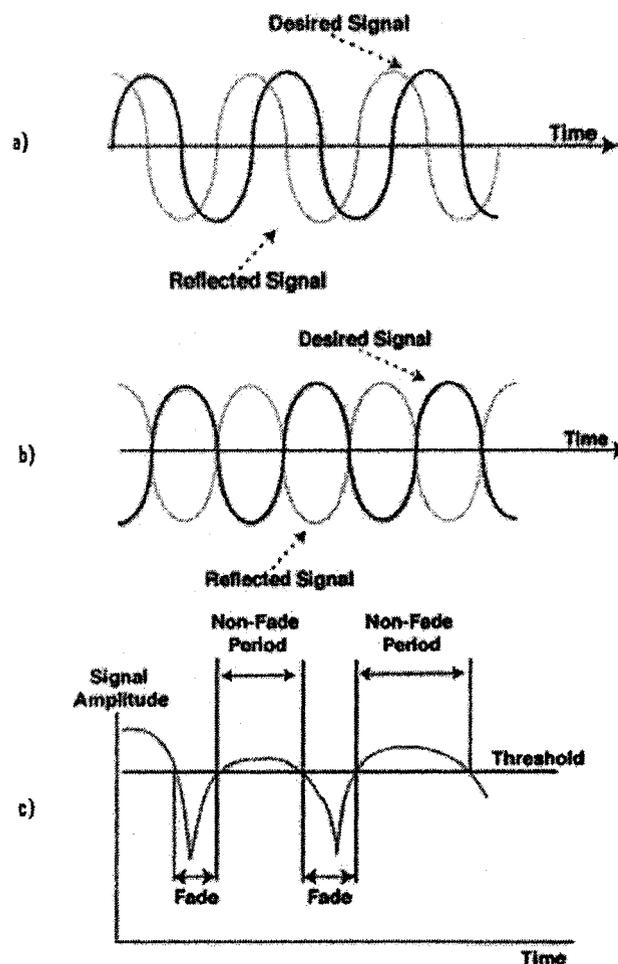


Figure 2.2: Problems related to wireless communications. (a) Signals out of phase, (b) Phase cancellation, and (c) Fading.

Another resulting problem is phase cancellation which occurs when waves of two multi-path signals are rotated to exactly 180° out of phase; the signals will cancel each other (see Figure 2.2 (b)). The effect is more a concern when the control channel signal is canceled out resulting in a black hole, a service area in which call set-ups will occasionally fail [IECOE].

The fading also result from the multi-path problem. It is a reduction in signal strength when the waves of multi-path signals are out of phase (see Figure 2.2 (c)), this phenomenon is known as “Rayleigh fading” or “fast fading” causing the received signal to fluctuate downward, causing a momentary, but periodic degradation in quality [IECOE].

Another resulting problem is the delay spread. It occurs in multi-path propagation environments when a desired signal, arriving from different directions, becomes delayed due to different travel distances [CHR00] (see Figure 2.1). This effect has a critical impact on link quality. The resulting effect of this delay spread is an *Inter-Symbol Interference (ISI)* [ALE04], or bits crashing into one another and the receiver cannot sort out. When this occurs, the *BER* rises and eventually causes noticeable degradation in signal quality [IECOE].

2.3 Multiple access schemes

Due to the recent development of wireless communication systems, the range of frequencies available for wireless communication technologies can be utilized in various ways/schemes, and this is referred to as multiple access schemes. These techniques are adopted to allow numerous users to share simultaneously a finite amount of signal spectrum.

The distribution of spectrum is required to achieve this high system capacity by simultaneously allocating the available bandwidth (or available amount of channels) to multiple users. This must be accomplished without severe degradation in the performance of the system in order to achieve high quality communications.

Conventionally, there are three major access schemes used to share the available bandwidth in a wireless communication. Nonetheless they are known as the *Frequency Division Multiple Access (FDMA)*, *Time Division Multiple Access (TDMA)*, and the *Code Division Multiple Access (CDMA)*.

In [BHO01], the authors present the evolution of *SAs* for the first, second and third generation. Here we present a brief overview of each technique, advantages and limitations, preparing for presenting the new technology known as the *Space Division Multiple Access (SDMA)*.

2.3.1 Frequency Division Multiple Access (FDMA)

FDMA is the most widespread multiple-access scheme for land mobile communication system due to its ability to discriminate channels effortlessly by filters in the frequency domain. In *FDMA*, every subscriber is allocated to an individual unique frequency band or channel (see Figure 2.3 (a)) where the allocated system bandwidth is divided into bands of W_{ch} and guard space between adjacent channels to prevent spectrum overlapping that may be resulted from carrier frequency instability.

Besides, when a user sends a call request, the system will assign one of the available channels to the user, in which, the channel is used exclusively by the user during a call. However, the system will reassign this channel to a different user when the previous call is terminated.

One of the most important advantages in *FDMA* system is there isn't any need for synchronization or timing control and therefore, the hardware is simple. In addition, there is only a need for flat fading consideration as for anti-fading technique because the bandwidth of each channel in the *FDMA* is sufficiently narrow. On the other hand there are many problems associated with *FDMA* systems and they are:

- Inter-modulation interference increases with the number of carriers.
- Variable rate transmission is difficult because such terminal has to prepare a lot of modems. For the same reason, composite transmission of voice and non-voice data is also difficult.
- High Q-value for the transmitter and receiver filters is required to guarantee high channel selectivity.

2.3.2 Time Division Multiple Access (*TDMA*)

In the basic *TDMA* protocol, the transmission time is divided into frames of equal duration, and each frame is divided into the same number of time slots having equal duration. Each slot position within a frame is allocated to a different user and this allocation stays over the same sequence frames.

Each user occupies a cyclically repeating time slot, so a channel may be thought of as a particular time slot that reoccur every frame. *TDMA* systems transmit data in a buffer-and-burst method: thus, the transmission for any user is non-continuous. This implies that digital data and digital modulation must be with *TDMA* [SEU99].

A *TDMA* frame with four time slots per frame is illustrated in Figure 2.3 b, with the shaded areas representing the guard times in each slot in which transmission is

prohibited. It is essential to have the guard times as it prevents transmissions of different (spatially distributed) users from overlapping due to transmission delay differences.

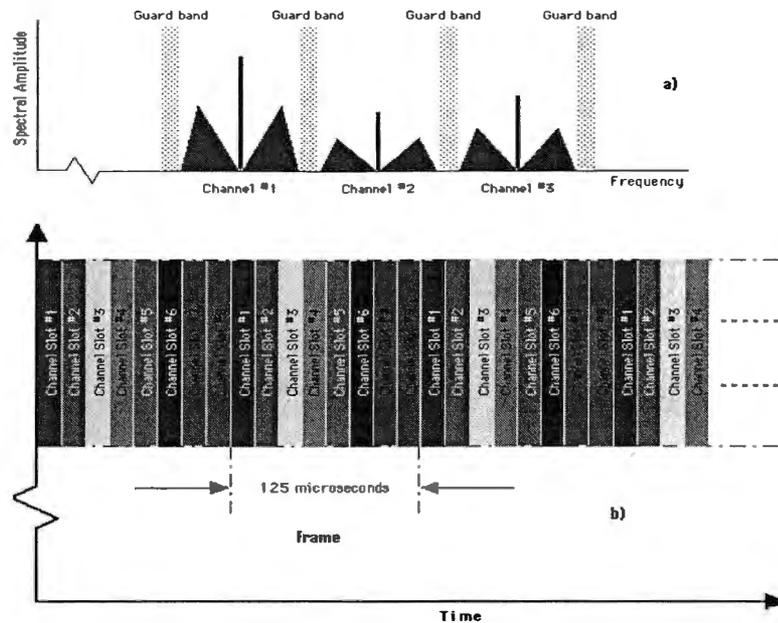


Figure 2.3: Multiple access schemes (a) *FDMA*, (b) *TDMA*.

2.3.3 Code Division Multiple Access (*CDMA*)

In *CDMA* systems, the signal is multiplied by a very large bandwidth signal called the spreading signal. The spreading signal is a pseudo-noise code sequence that as a chip rate which is in orders of magnitude greater than the data rate of the message [LEB99]. Having its own pseudorandom codeword, all subscribers in a *CDMA* system use the same carrier frequency and may transmit simultaneously. A *CDMA* is illustrated in Figure 2.4. The most distinct feature of *CDMA* system is that all the terminals share the whole bandwidth, and each terminal signal is discriminated by the code.

In a *CDMA* mobile communication system, since each subscriber is assigned its own (orthogonal) code, the signal of each subscriber is separated by individually

correlating the received signal with the code sequence assigned to each subscriber performing a matched-filtering operation [SEU99].

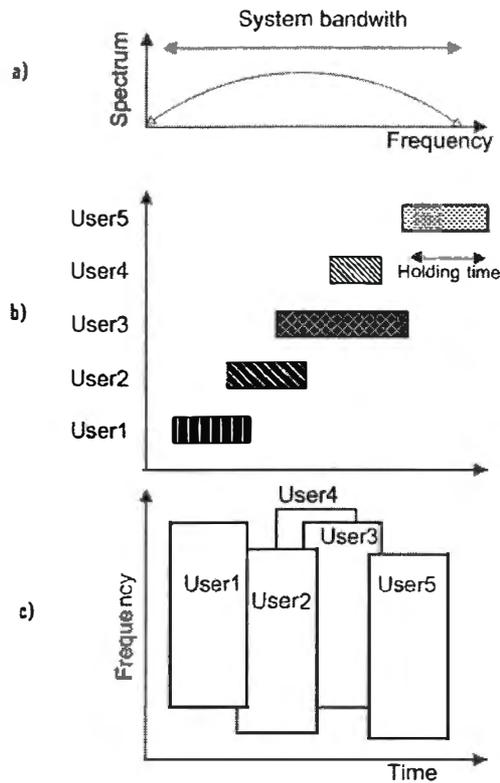


Figure 2.4: Concept of a *CDMA* system. (a) Spectrum of a *CDMA* system, (b) a call initiation and holding model for five-user case, (c) channel allocation for each user.

When each user sends a call request to the base station, the base station assigns one of the spreading codes to the user. When five users initial and hold the calls as shown in Figure 2.4 (b), time and frequency are occupied as shown in Figure 2.4 (c). Therefore *CDMA* requires a larger bandwidth as compared to *FDMA* and *TDMA*. Furthermore, there is also a need for code synchronization in *CDMA* system.

2.3.4 Space Division Multiple Access (*SDMA*)

In addition to these techniques, *SAs* provide a new method of multiple access schemes to the users, which is known as the *SDMA*.

As its name implies, *SDMA* technology exploits information collected in the spatial dimension in addition to the temporal dimension to achieve significant improvements in wireless information transmission. Spatially selective transmission and reception of *RF* energy provides substantial increases in wireless system capacity, coverage and quality [ROY97].

The *SDMA* scheme, which is commonly referred to space diversity, uses *SAs* to provide control of space by providing virtual channels in an angle domain. With the use of this approach, simultaneous calls in various different cells can be established at the same carrier frequency.

The *SDMA* scheme is based upon the fact that a signal arriving from a distant source reaches different antennas in an array at different times due to their spatial distribution, and this delay is utilized to differentiate one or more users in one area from those in another area [GOD97a].

Its advanced spatial processing capability enables it to locate many users, creating a different sector for each user, as shown in Figure 2.5, this means that more than one user can be allocated simultaneously to the same physical communications channel in the same cell, with only an angular separation.

This technology dramatically improves the interference suppression capability while it greatly increases frequency reuse, resulting in increased capacity and reduced infrastructure cost. Basically, capacity is increased not only through inter-cell frequency reuse, but also through intra-cell frequency reuse [BEL02A].

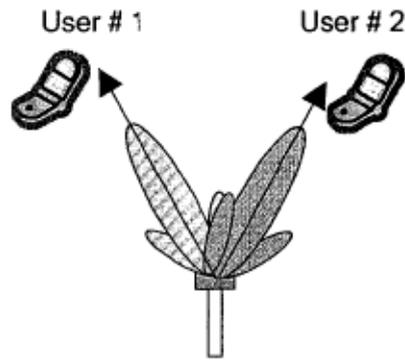


Figure 2.5: The new dimension: *Space Division Multiple Access*.

Moreover this technique enables an effective transmission to take place in one cell without affecting the transmission in another cell. Without the use of an array, this can be accomplished by having a separate base station for each cell and keeping cell size permanent, while the use of space diversity enables dynamic changes of cell shapes to reflect the user movement. Thus an array of antennas constitutes to an extra dimension in this system by providing dynamic control in space and needless to say, it leads to improved capacity and better system performance.

In conclusion, we refer to this technology or to this new dimension, as one of Qualcomm's founders Andrew Viterbi stated: "Spatial processing remains as the most promising, if not the last frontier, in the evolution of multiple access systems" [ROY97].

2.4 Smart antenna's system

First of all, antennas have been the most neglected of all the components in personal communications systems. Yet, the manner in which energy is distributed into and collected from surrounding space has a profound influence on the efficient use of spectrum, the cost of establishing new networks, and the service quality provided by those networks [IECOE].

The term “antenna” itself comprises only the mechanical construction transforming free *EM* waves into *RF* signals traveling on a shielded cable and vice versa, most often it is called the “radiating element” [BHO01], [KRA88], i.e. it is the port through which *RF* energy is coupled from the transmitter to the outside world and, in inverse, to the receiver to the outside world [IECOE].

The *SA* system combines multiple antenna elements with a signal processing capability to optimize its radiation and/or reception pattern automatically in response to the signal environment [IECOE]. In other words, we define a *SA* system as an array of antenna elements [BHO], [CHR00] connected to either an analog receiver or a digital signal processor, whose radiation pattern adapts to the current signal environment [BHO01].

2.4.1 A First Basic Approach

The term *SAs* was born in the early 1990s when the well-developed adaptive arrays used in the military were brought by several scientists into mobile communications. To be more precise, *SAs* entered research in civil mobile communications before the 1990s under the original name adaptive antenna arrays [KAI05]. So, the fundamental theory of *SAs* is not new, in fact, they have been applied in defense-related systems [BHO01], [BEL02a], [IECOE] since World War II [BEL02a]. As a result, recently, the application of *SAs* has been suggested for mobile communications systems, to overcome the problem mentioned in section 2.2.

The most enquiring is about its notation, the fact they are smart, but what makes them smart?

In truth, antennas are not smart – antenna systems are smart [BEL02a], [IECOE]. Actually, it is the digital signal processing, along with the antennas, which make the system smart [BEL02a].

How this smartness can be understood? Where this smartness come from? A very good two illustrations show that the closest system to an *SA* is the human ear [BEL02a] and [IECOE], besides the functionality of many engineering system is readily understood when it is related to our human body. For instance, close your eyes and converse with someone as they move about the room, you will notice that you can determine their location without seeing them because of the following:

- You hear the speaker's signals through your two ears, your acoustic sensors.
- The voice arrives at each ear at a different time.
- Your brain, a specialized signal processor, does a large number of calculations to correlate information and compute the location of the speaker.

In addition to that, your brain adds the strength of the signals from each ear together, so you perceive sound in one chosen direction as being twice as loud as everything else [BEL02a], [IECOE]. Furthermore, if additional speakers join in the conversation, the brain can tune out unwanted interferers, and concentrate on one conversation at a time. Conversely, the listener can respond back to the same direction as the desired speaker by orienting his or her transmitter – his or her mouth – towards the speaker [BEL02a]. As a result, 8, 10, or 12 ears can be employed to help fine-tune and turn up signal information [IECOE], or in the equivalent electrical system, an array of 8, 10 or 12 antenna elements can be used.

2.4.2 A Second systematic Approach

Electrical *SA* systems do the same things as the ears in the reception case and as the mouth in the transmission case and in both cases a digital signal processor instead of the brain (see Figure 2.6).

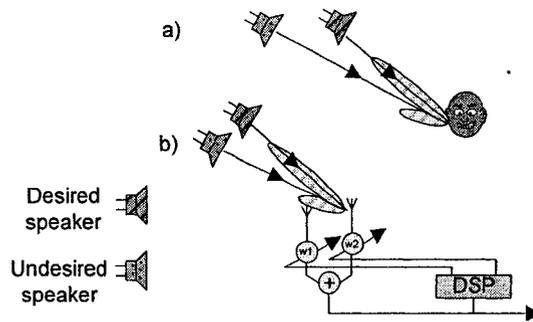


Figure 2.6: Two analogies (a) human analogy, (b) *SA* analogy.

Therefore, after the digital signal processor receives the time delays from each antenna element, it computes the *Direction-Of-Arrival (DOA)* of the *Signal Of Interest (SOI)*. It then adjusts the excitations (the amplitudes and phases of the signal) to produce a radiation pattern that focuses on the *SOI*, while turning out any *Signal Not Of Interest (SNOI)* [BEL02a] which resumes the effect of the smartness of the system. This scenario is very well illustrated in Figure 2.7.

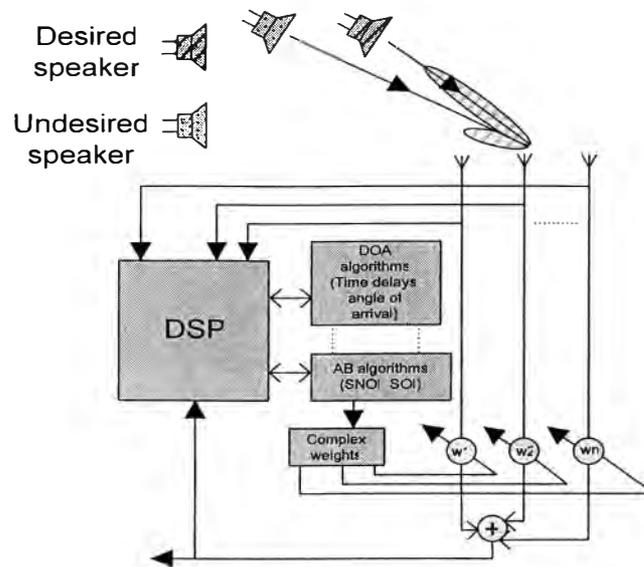


Figure 2.7: SA system and functionality.

An *SA* system is generally co-located with a *BS*, as we mentioned earlier in this section, it combines an antenna array with a digital signal-processing capability to transmit and receive in an adaptive spatially sensitive manner.

This digital signal processor is called the control unit, or the *SA*'s intelligence, it receives signal input from a smart scanning receiver and outputs to an antenna controller. The processor controls feed parameters of the antenna, based on several inputs, in order to optimize the communication link Here in the following sub-section we present the types of the *SA* system, for the evolution process from the conventional base station (cell sectoring or splitting), passing by diversity techniques (switched diversity and diversity combining), to the *SA*'s base stations, which is beyond the scope of this thesis, and for more in-depth details, the reader is referred to [BEL01], and [IECOE].

2.4.3 Two different functionality approaches

Terms commonly heard today that embrace various aspects of a *SA* system technology include intelligent antennas, phased array, *SDMA*, spatial processing, digital beamforming, adaptive antenna systems [IECOE], antenna array [SHI97], fully adaptive, and others.

So, depending on the type of *SA* system, different optimization criteria can be used at the transmit/receive patterns are more than just the “antenna”, but rather a complete transceiver concept [BHO01]. Thus, *SA* systems can be customarily [IECOE] categorized as either switched or adaptive [BHO01], [BEL02a], [IECOE], [CHR00].

2.5 Switched smart antennas (SBA)

2.5.1 Description

The switched approach is called switched beam [BHO01], Switching-Beam Array [SEU99], switched-beam system [BEL02a], [CHR00]. In this work we refer it as the *Switched Beam Array* or *SBA*.

In the *SA* system, the *SBA* approach is considered as an extension of the current cellular sectorization scheme, in which a typical sectorized cell site is composed of three 120-degree macro-sectors, this approach subdivides the macro-sectors into micro-sectors [CHR00]. The switched approach forms multiple fixed beams with enhanced selectivity in specific area or in particular directions.

These antenna systems will detect signal strength, and select one of the best [BHO01], [BEL02a], [IECOE], [CHR00], predetermined, fixed beams for the subscribers as they move throughout the coverage sector [BEL02a], [IECOE]. So,

during the call, the system monitors the signal strength and switch to other micro-sectors, if required [CHR00], which is illustrated in Figure 2.8 (a). Thus, instead of modeling the directional antenna pattern with the metallic properties and physical design of single element, the switched approach system couple the outputs of multiple antennas in such manner that it forms finely sectorized or directional beams with more spatial selectivity than the conventional, single-element approach [IECOE].

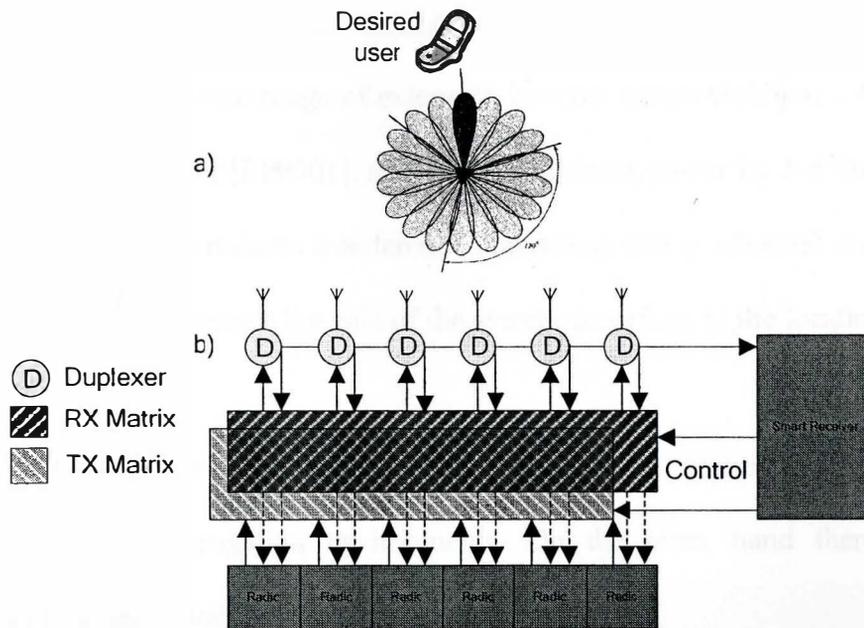


Figure 2.8: Switched-beam-array (a) functionality (b) block diagram.

2.5.2 Advantages

Concerning the materials of the system, it is a simple technique and comprises only a basic switching function between separate directive antennas or predefined beams of an array [BHO01] (see Figure 2.8). A block diagram of *SBA* systems is illustrated in Figure 2.8 (b). The switch matrix should be quick and accurately switch the subscriber's channel to the correct beam in which the user best signal, with no degradation to voice quality [BHO01].

The receivers take the RF energy from the multi-beam antennas and choose the appropriate beam [BEL02a] that gives the best *Signal to Interference Noise Ratio (SINR)* [BHO01] or the one having the strongest signal [CHR00]. For more in-depth details information about this, please refer to [BHO01], [SEU99].

The switched approach offer numerous advantages of more elaborate *SA* systems at a fraction of the complexity and expense [CHR00]. Consequently, there are number significant improvements with such a system.

They provide some range of extension benefits which yields in a 40% increase in the range of the sector [BHO01], reducing transmitting power by 3-6 dB, and for the base station too, which reduces interference introduced into co-channel cells [BHO01], which consequently increases the gain of the system according to the location of the user [BEL02a].

Another very practical benefit of this approach is offering reduction in delay spread in certain propagation environments. On the other hand there are many limitations to system based on the switched approach.

2.5.3 Drawbacks

Since the beams are predetermined, the intended user may not be in the center of the beam [BEL02a]. The signal strength varies as the user moves through the sector, specially if the user moves the edge of the beam, the signal strength degrade rapidly before the user switched to another beam [BHO01] , [CHR00]. Another limitation is when a switched beam system does not distinguish between a desired user and an interfering one [BHO01], [CHR00], or in other words if the interferer is located at the

same angle of the desired, it may be enhanced more than the desired user [BHO01], [BEL02a], [SEU99] this results in poor quality due to low *SINR*.

So the interference suppression is limited by the antenna bandwidth of the switched beam [BHO01]. However, compared to conventional sectorized cells *SBA* systems can increase the range of a base station from 20% to 200%, depending on the circumstances of operation. The additional coverage means that an operator can achieve a substantial reduction in infrastructure costs [CHR00].

2.6 Adaptive smart antennas (TBA)

2.6.1 Description

The second approach is called digitally adaptive *beamforming (AB)*, phased arrays, adaptive antenna array [BHO01], adaptive array [BEL02a], [CHR00], tracking-beam-array [SEU99]. In this work we refer it as the *Adaptive Beam Array (ABA)*. This approach deals with the communication and the base station in a very different way, in effect adding a dimension in space.

Using a variety of new signal processing algorithms, the adaptive approach systems provide more degrees of freedom [BEL02a], since they adapt the radiation pattern to the *RF* signal environment as it changes in real time.

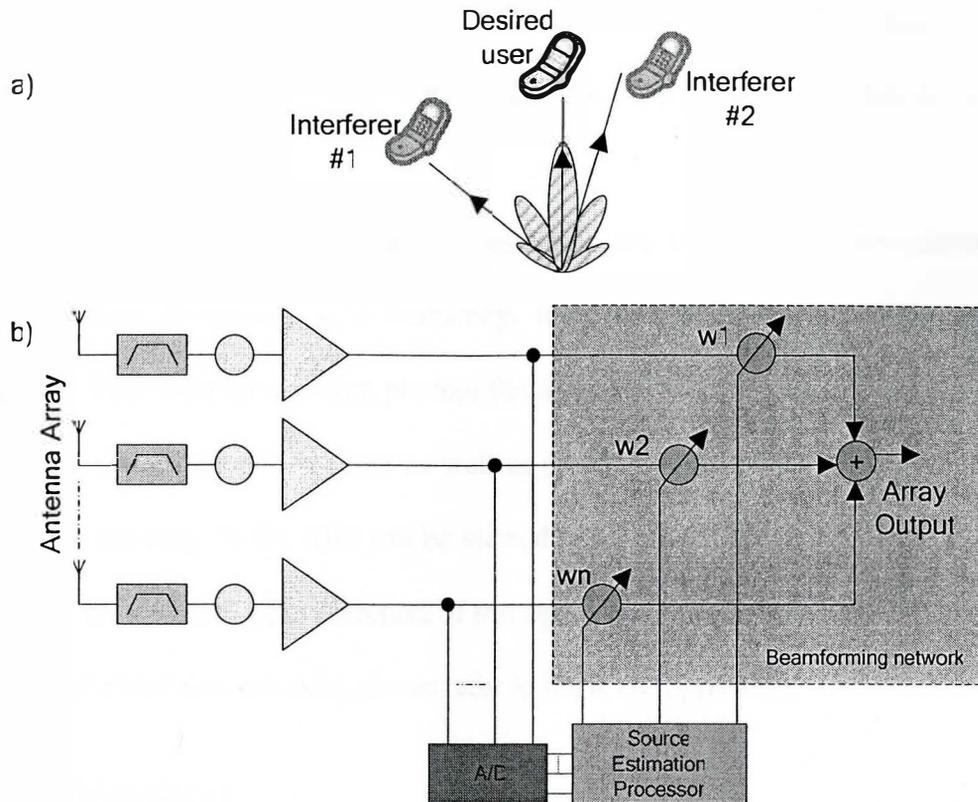


Figure 2.9: Adaptive-beam-array, (a) functionality, (b) block diagram.

This system takes advantage of its ability to effectively locate and track various types of signals to dynamically minimize interference and maximize intended reception [IECOE]. In other words, this system can direct the main beam towards the pilot signal i.e. the desired user or the *SOI* while suppressing the antenna pattern in the direction of interferers or *SNOI* [BEL02a], as illustrated in Figure 2.9 (a). Moreover, the adaptive array system can customize an appropriate radiation pattern for each user.

In conclusion this system continuously differentiates between the desired signals, multipath and interfering signals as well as calculates their *DOA* by utilizing sophisticated signal processing algorithms. The technique constantly updates its transmitting approach based on changes in both the desired and interfering signal locations. It ensures that signal links are maximized by tracking and providing user with

main lobes and interferers with nulls, because they are neither macro-sectors nor predefined patterns as the *SBA* system. A block diagram of this system is illustrated in Figure 2.9 (b).

The *RF* signals from the M antennas are coherently down-converted to an *Intermediate Frequency (IF)* frequency, low enough for quality digitization of the signals. The *beamformer* then process the digital outputs for each channel, the process includes amplitude adjustments as well as phasing adjustments, which results in beam and null steering. So the *ABA* can be viewed as a spatial filter in which the pass and stop band is created along the direction of the signal and interference respectively [BHO01], for more about this scenario, please refer to [BHO01], [SEU99].

2.6.2 Advantages

Although both system attempt to increase gain with respect to the location of the users, however only the adaptive system [BHO01], [IECOE], [SEU99] is able to contribute optimal gain while simultaneously identifying, tracking, and minimizing interfering signals, which can be seen from Figure 2.9 (a), that only the main lobe is directed towards the user while a null, being directed at an interferer.

Figure 2.10 illustrates the beam patterns that each system might choose, in a scenario involving one desired signal and two co-channel interferers. The switched-beam system is depicted on the left, while the adaptive system is on the right. Both systems direct their major lobe in the general direction of the *SOI*, but the adaptive system chooses a more-accurate placement, providing greater signal enhancement.

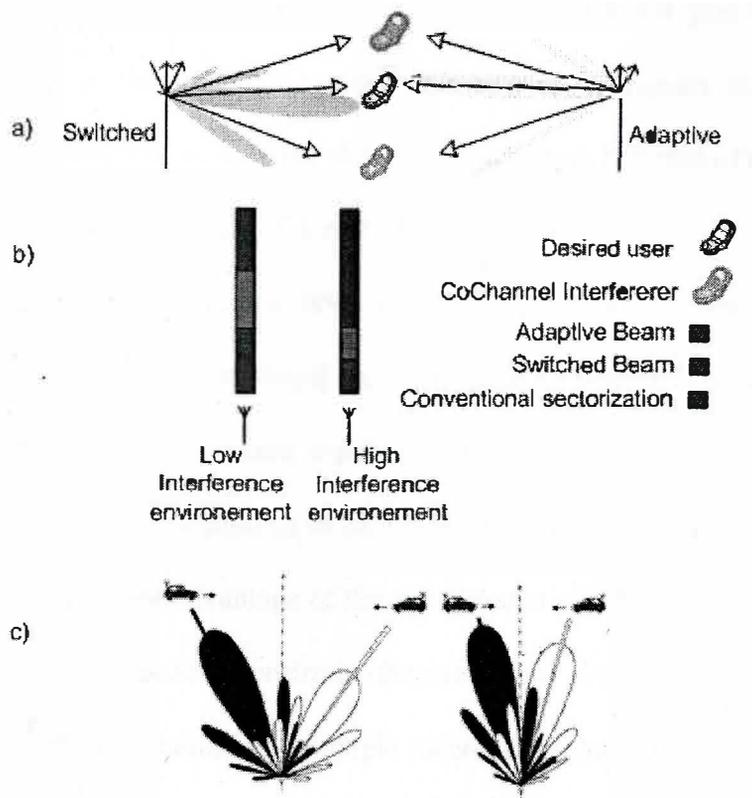


Figure 2.10: Comparison between the *ABA* and the *TBA* a) Comparison of lobes, b) comparison for different environments, c) comparison for user tracking.

In Figure 2.10 (a), the beamforming lobes and nulls that *SBA* (left) and *ABA* (right) systems might choose for identical user signals and co-channel interference. While in Figure 2.10 (b), the coverage patterns for *ABA* and *SBA* for two different environments, low noise (left) and high noise (right).

Figure 2.10 (c) describes an *ABA* supporting two users on the same conventional channel simultaneously in the same cell. The dark beam pattern is used to communicate with the user on the left, while right beam is used to communicate with the user on the right. In the right figure, there is an illustration of how the beam patterns for the *ABA* of the figure on the left are updated to accommodate the motion of the users.

Similarly, the interfering signals arrive at places of lower gain outside the main lobe, but, again, in the adaptive system, the interfering signals receive maximum suppression [CHR00]. Figure 2.10 (b) shows a comparison, in terms of relative coverage area, of conventional sectorized, *SBA*, and *ABA*.

In the presence of a low level of interference both types of *SAs* provide significant gains over the conventional sectorized systems. When a high level interference is present the interference rejection capability of the *ABA* provides significantly more coverage than either the conventional or *SBA* system [BEL02a], [IECOE], [CHR00].

Another important advantage of the next generation of adaptive-antenna systems is its capability to “create” spectrum. Because of the accurate tracking and robust interference-rejection capabilities, multiple users can share the same conventional channel within the same cell. System capacity increases through inter-cell frequency reuse patterns, as well as intra-cell frequency reuse. Figure 2.10 (c) shows how this technology can be used to accommodate two users on the same conventional channel, simultaneously, in the same cell.

The dark beam pattern is used to communicate with the user on the left, while the light beam pattern is used to communicate with the user on the right. It should be noted that every pattern has nulls in the direction of the other user. As the users move, the beam patterns are constantly updated, to insure these positions.

It is this ability to continuously modify the beam pattern with respect to both lobes and nulls that separates the *ABA* approach from the *SBA* type. As interfering signals move throughout the sector, the switched beam pattern is not altered, because it only responds to movements of the signal of interest.

Unlike the *SBA* approach, the *ABA* system is able to continue to distinguish between the signal and the interferer, and to allow them to get substantially closer than in the *SBA* system, while maintaining enhanced *SINR* levels. The most-sophisticated adaptive SA will hand off any two co-channel users, whether they are inter-cell or intra-cell, before they get too close, and begin to interfere with each other [CHR00].

2.6.3 Drawbacks

Studies and evaluations in terms of performance and complexity of both systems have been made [SEU99] and showed the advantage in terms of performance of the *ABA* over the *SBA*. However Despite all those advantages, the *ABA* systems have many drawbacks, unlike the *SBA* which is fairly easy to integrate in to the base station architecture, the integration of the *ABA* is difficult and complex [BHO01], [BEL02a], [SEU99].

It should be noted that in many references they consider the *ABA* as the *SA*, knowing that both approaches are smart, and that the *ABA* is much smarter than the *SBA*.

This is what dragged us down to this challenge, to find an algorithm that is very powerful and to prove its low complexity.

2.7 General benefits

An understanding of signal propagation environment and channel characteristics is significant to the efficient use of a transmission medium. In recent years, there have been signal propagation problems associated with conventional antennas and interference and all its consequences are the major limiting factors in the performance of

wireless communications. Thus the introduction of *SAs* is considered to have the potential of leading to a large increase in wireless communication systems performance.

The benefits of *SA* systems can be applied to *ABA* more than to *SBA* can be summarized and regrouped as follows:

- Increased range and coverage: [BOU00],[BEL02a], [IECOE], [ALE04], [CHR00], [JIN05] *SA* systems provide enhanced coverage through range extension, focusing the energy sent out into the cell, hole filling, and better building penetration. Given the same transmitter power at the base station and subscriber unit, *SA* can increase range by increasing the gain of the base station. Lower power requirements also enable a greater battery life and smaller/lighter handset size
- Increased capacity: [BEL02a], [ALE04], [SHI97], [CHR00], [JIN05] *SA* systems can also improve system capacity by mitigating interference and allowing transmission of different data streams from different antennas. They can be used to allow the subscriber and the base station to operate at the same range as a conventional system, but at low power. This may permit *FDMA* and *TDMA* systems to be re-channelized to reuse frequency channels more frequently than conventional systems using fixed antennas, since the *SINR* is much greater when *SAs* are used, so with *SA* the *SINR* is maximized when not only the desired wave or signal but also interfering waves arrive [OHI02]. In *CDMA* systems if *SAs* are used to allow subscribers to transmit less power for each link, the *Multiple Access Interference (MAI)* is reduced, which increases the number of simultaneous subscribers that can be supported in each cell.

- Better Security: [BEL02a] In a society that is becoming more dependent on conducting business and transmitting personal information, security is an important issue. The employment of *SA* systems diminishes the risk of connection tapping. The intruder must be situated in the similar direction as the user as seen from the base station.
- Reduced interference and multipath fading: [BOU00], [BEL02a], [IECOE], [SHI97], [CHR00], [JIN05] In wireless communications, *SAs* systems can be used to reduce multipath fading, in the transmitting mode by forming beams in certain direction and nulls in others, and in the receiving mode by knowing the directional location of the signal's source, and utilizing interference cancellation, thereby canceling some of the delayed arrivals. Usually, in the transmitting mode, the array focuses energy in the required direction, which helps to reduce multipath reflections and the delay spread. In the receiving mode, however, the array provides compensation in multipath fading by adding the signals emanating from other clusters after compensating for delays, as well as by canceling signals emanating from directions other than that of the desired user.
- Reduced expenses: [IECOE], [CHR00] because of the power efficiency provided by the *SA* system, by combining the inputs of multiple elements to optimize available processing gain towards the user, which will result in lower amplifier costs and power consumption.
- Better services: [BEL02a] the usage of the *SA* systems enables the network to have access to spatial information about the users. This information can be send to assess the positions of the users much more precisely than in existing network.

This can be applied in services such as emergency calls and location-specific billing.

2.8 Future challenges

Although *SA* systems are favorable in many ways, there are also drawbacks which include a more complex transceiver [BOU00], [BEL02a] structure compared to traditional base station transceiver and a growing need for development of efficient algorithm for real optimizing and signal tracking. Thus, *SAs* base stations will no doubt be much more expensive than conventional base stations and the advantages should always be evaluated against the cost, which consist a big challenge for engineers, that's why they should be user where they are truly needed.

Concerning the future challenges of *SA* systems, there are many enquiries.

- What are the strategies for next generation wireless systems [ALE04]?
- Trends and challenges of *SAs* [ALE04]?
- What will be the characteristics of the *SA* in the future?
- When will *SAs* be ready to market [KAI05]?

In conclusion we can say that the next generation of *SA* is the capability to “create” spectrum, so multiple users can share the same conventional channel [CHR00], and the ability to reject all kinds of interference, and to locate and track the desired users maintaining a high *SINR*, and a nulls for all interfering users.

In general, the next-generation of wireless systems require signal processing techniques capable of operating in wide variety of scenarios with respect to propagation, traffic, interference, user mobility, antenna configuration, radio access technology and *Channel State Information (CSI)* reliability.

Yet, the biggest challenge for us takes two approaches. The first is to develop an efficient adaptive algorithm, giving the best performance, in different scenarios, and the second approach is to implement it in a very low complexity, guarding always a high *Performance Over Complexity Ratio (POCR)*.

2.9 Summary

A global overview of *SA* systems was studied and discussed. The reasons explaining the needs for this technology are given. The analysis of the multiple access schemes is given too, from the *FDMA* to *SDMA*. A basic and a systematic approach of the *SA* were given. The advantages and disadvantages of the main categories of *SAs*, switched beam array *SBA* and adaptive beam array *ABA* were presented and discussed.

The next chapter presents the research for the adaptive algorithm leading the *SA* system to give his best response whether towards a desired user or an interferer, in any wireless communication environment.

The best candidate will play the role of the brain in the human analogy, and the role of the *Digital Signal Processor (DSP)* in the *SA* analogy. This candidate will push the system to its best performance ever in any wireless environments.

Chapter 3

Performance Evaluation of Beamforming Techniques

3.1 Introduction

There is a very growing need for improvements that increase coverage, capacity, quality [ROY97] and reliability in many wireless communication systems. Optimal exploitation of new dimensions through *SAs* technology has the potential to meet these needs while providing operators the ability to offer new value-added services to their costumers. *SAs* are recognized as a key technology in 3G networks. They offer a mixed service capacity gain of more than 100% and hence reduce to less than half the number of base stations required.

A *SA's* ability to abate both multipath and *CCI* makes it one of the leading candidates [JIN05] and one of the most promising technologies for the enabling of high capacity wireless networks. Since *SAs* are more expensive than conventional base stations, they should be used where they are truly needed. All characteristics and advantages were discussed in Chapter 1.

The most important contributions of an *SA* is its capability of changing its pattern in real time in order to track the user with its maximum gain while trying to minimize the interference by placing the nulls of its radiation pattern at location of the other users [PEN02].

Many questions came to the mind. How the *SA* is able to do that? What stands behind these scenarios? How does the antenna steer its beam toward the user? Is there an engine behind that?

In fact, there is a complex mathematical process standing behind that physical reaction. This process is called *beamforming*, which is referred as the heart of *SA* or the main core of it. We describe it as the backbone of the *SA* system.

The *beamforming* is a process in which each user's signal is multiplied with complex weights that adjust the magnitude and the phase of the signal from each antenna. Hence the array forms a transmit beam in the desired direction and minimizes the output in other directions.

There are two types of *beamforming* constituting the two types of *SAs* presented in chapter 1, the *Switched-Beam-Array (SBA)* where the complex weights are selected from a library of weights that form beams in specific, predetermined directions, this process is called *Switched Beamforming (SB)*; and the *Adaptive-Beam-Array (ABA)* where the weights are computed and adaptively updated in real time, iteratively, this process is called *Adaptive Beamforming (AB)*.

Due to the numerous advantages of the *AB* over the *SB*, the *SB* is no further discussed in this work, and for more about this process the reader is referred to [GOD04], [SEU99], and [BHO01].

For better understanding, consider a *SA* system with M antenna elements, whose signals are processed adaptively in order to exploit the spatial dimension of the mobile radio channel. Figure 3.1 illustrates an example of a *narrow beamformer* or *Delay and Sum Beamformer*. In order to accomplish the tracking in the real time such system has to have information about location of all users in the system relative to the array and adjust the weight to make the phase and magnitude changing.

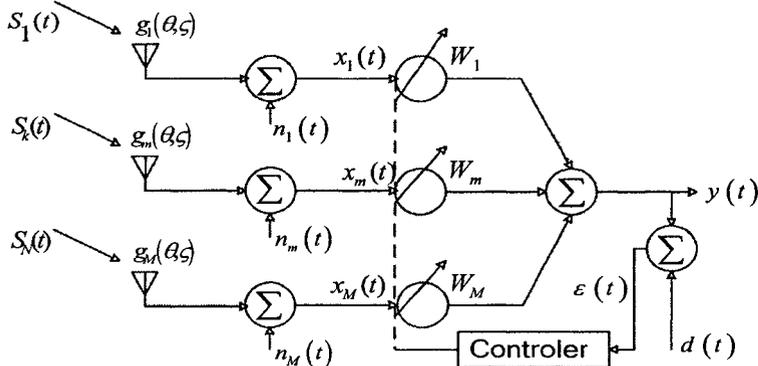


Figure 3.1: Delay and Sum Beamformer

As a result, the nulls of the radiation pattern are placed to the interference while the peak of the radiation pattern is positioned towards the user, which amplifies its signal prior to its arrival at the radio.

In other words, *ABA* systems have the ability to null out the interfering signal (uncorrelated) and steer the main beam in the direction of the *SOI* [PEN02]. A detailed schematic model of the functionality of this system is illustrated in Figure 2.9.

Based on computing the *DOA* of the signal by a *DOA* algorithm which is beyond the scope of this work (for more of *DOA* algorithms, please refer to [GOD97b], [GOD04]), the *AB* algorithms implemented in the digital signal processor adapt the complex weights, leading the *SA* system to give his main beam towards *SOI*, and nulls towards *SNOI*.

Since the successful application of *SA* technology not only depends on the knowledge of the propagation channel, however, but also on the choice of the *AB* algorithm applied at the base station [SHI98], *SA* concept has been extensively studied and most of the research activities have been dedicated to the algorithm development, theoretical and simulation analysis.

In addition to that the choice of an *SA* system and algorithm today is highly dependant on the air interface and its parameters [BOU00]. Among the most critical parameters are the multiple access method, the type of duplexing, pilot availability, channel, modulation, diversity, and frame structure.

Besides the compatibility with the air, the level of *Integrated Circuit (IC)* technology on *DSP*, *FPGAs* and *Application Specific Integrated Circuits (ASICs)*, can also be limiting factors for implementation of *SA* algorithms.

For all these reasons, the main contribution of this thesis takes on searching for the best adaptive algorithm for *beamforming*, allowing the antenna to give his main beam towards the *SOI* and null towards the interferer or the *SNOI*.

3.2 Signal model

Adaptive signal processing involves the manipulation of signals induced onto the elements of an array [CHR00]. The *AB* techniques exist that can yield multiple, simultaneously available beams. The beams can be made to have high gain and low side lobes, or controlled beamwidth.

The *AB* techniques dynamically adjust the array pattern to optimize some characteristics of the received signal. In beam scanning, a single main beam of an array is steered and the direction can be varied either continuously or in the small discrete

steps. The *ABA* system uses *AB* techniques in order to reject interfering signals having a *DOA* different from that of the desired user.

A signal model used for array processing is presented in this section. For more extended and in-depth details, the reader is referred to [GOD97b], [JIN00], and [GOD04].

Consider a reverse link (uplink) from K mobile stations to a base station with multiple antennas. Let the baseband traffic transmission (stream) from the k th mobile station be $x_{(k)}(t) = \sum_{n=-\infty}^{\infty} A_{(k)} s_{(k)}[n] d_{(k)}(t - nT; n)$, where T represents the symbol interval; $s_{(k)}[n] \in S$ (S is the symbol alphabet; $S \triangleq \{\pm 1\}$ for *Binary Phase Shift keying (BPSK)* case and $S \triangleq \{\pm 1/\sqrt{2} \pm j/\sqrt{2}\}$ for *Quadrature Phase Shift Keying (QPSK)* case, the n th symbol; and $d_{(k)}(t; n)$ the spreading waveform at the n th symbol of the k th transmission.

The spreading waveform is given by $d_{(k)}(t; n) = \sum_{\ell=0}^{N-1} c_{(k)}[\ell; n] \varphi(t - \ell T_c)$, where T_c is the chip interval; $N = T/T_c$ the processing gain; $c_{(k)}[\ell; n]$ the ℓ th element of the spreading sequence for the n th symbol of the k th transmission; and $\varphi(t)$ is the chip waveform of unit energy.

Assume that the base station is equipped with L antenna elements and $\mathbf{a}(\theta)$ denotes the array response vector to the signal with angle of arrival (*AOA*) θ . If the angle spread of the nominal *AOA* θ , denoted by $\Delta\theta$, is sufficiently small, the channel vector $\mathbf{h}_{(k)}$ can be written as $\mathbf{h} \cong \alpha \mathbf{a}(\theta)$. Notice that the spatial signature vector can be

characterized by a single nominal AOA θ and a single attenuation factor α under the local scatter model.

In general α can be considered as a random variable. For *Rayleigh* fading channels α would be a complex Gaussian random variable. For M multipaths, there would be M spatial signature vectors corresponding to each path.

In agreement with the current 3G CDMA transmission schemes, the pilot channel is orthogonally multiplexed with the traffic channel.

First, let $M = 1$. Therefore, the baseband received signal vector from $m_{(k)}(t)$ can be written as

$$\mathbf{r}_{(k)}(t - \tau_{(k)}) = \sum_{n=-\infty}^{\infty} A_{(k)} \mathbf{h}_{(k)} s_{(k)}[n] d_{(k)}(t - \tau_{(k)} - nT; n), \quad 3.1$$

where $\tau_{(k)}$ is the transmission delay. Instantaneously, the pilot transmission has been received. The received k th pilot transmission is given by

$$\mathbf{p}_{(k)}(t - \tau_{(k)}) = \sum_{n=-\infty}^{\infty} \bar{A}_{(k)} \mathbf{h}_{(k)} \bar{s}_{(k)}[n] \bar{d}_{(k)}(t - \tau_{(k)} - nT; n), \quad 3.2$$

where $\bar{A}_{(k)}$ is the amplitude of the pilot transmission; $\bar{s}_{(k)}[n]$ the pilot symbol; and $\bar{d}_{(k)}(t; n)$ the spreading wave form of the k th pilot transmission. For simplicity, the pilot symbol, $\bar{s}_{(k)}[n]$ is assumed to be fixed, i.e., $\bar{s}_{(k)}[n] = \bar{s}$ and the spreading waveform is $\bar{d}_{(k)}(t; n) = \sum_{\ell=0}^{N-1} \bar{c}_{(k)}[\ell; n] \varphi(t - \ell T_c)$, where $\bar{c}_{(k)}[\ell; n]$ is the spreading sequence for the k th pilot transmission. In general, the amplitude of the traffic transmission is greater

than that of the pilot transmission, i.e. $A_{(k)} > \bar{A}_{(k)}$. The received signal vector at a base station is written as

$$\mathbf{y}(t) = \sum_{k=1}^K \left(\mathbf{r}_{(k)}(t - \tau_{(k)}) + \mathbf{p}_{(k)}(t - \tau_{(k)}) \right) + \mathbf{n}(t) \quad 3.3$$

where $\mathbf{n}(t)$ is the zero-mean background noise with $E[\mathbf{n}(t)\mathbf{n}^H(t)] = (N_0/2)\mathbf{I}\delta(t - \tau)$.

Let $\mathbf{y}[n]$ denote the sample vector sequence of $\mathbf{y}(t)$ from analog-to-digital converters following the bank of the L matched filters for the chip waveform $\varphi(t)$.

The sampled matched filter outputs can be written as $\mathbf{y}[\ell] = \int_{-\infty}^{+\infty} \mathbf{y}(t - \tau)\varphi(\tau) d\tau \Big|_{t = \ell T_c + \tau_{(i)}}$. Assuming despanders for the first transmissions,

the despread signals for the desired traffic and pilot transmissions are written as [JIN00]

$$\mathbf{r}_{(i)}[n] = \frac{1}{N} \sum_{\ell=0}^{N-1} \mathbf{y}[nN + \ell] \mathbf{c}_{(i)}^*[\ell; n] = A_{(i)} \mathbf{h}_{(i)} \mathbf{s}_{(i)}[n] + \mathbf{i}_{r,(i)}[n] \quad 3.4$$

and

$$\mathbf{p}_{(i)}[n] = \frac{1}{N} \sum_{\ell=0}^{N-1} \mathbf{y}[nN + \ell] \bar{\mathbf{c}}_{(i)}^*[\ell; n] = \bar{A}_{(i)} \mathbf{h}_{(i)} \bar{\mathbf{s}} + \mathbf{i}_{p,(i)}[n], \quad 3.5$$

where $\mathbf{i}_{r,(i)}[n]$ and $\mathbf{i}_{p,(i)}[n]$ are *Interference Plus Noise (IPN)* vector sequences in the despread traffic and pilot transmissions, respectively [JIN00]. For multipath fading vector channels of M paths, the received signal vector is written as

$$\mathbf{y}(t) = \sum_{k=1}^K \sum_{m=1}^M \left(\mathbf{r}_{(k),m}(t - \tau_{(k),m}) + \mathbf{p}_{(k),m}(t - \tau_{(k),m}) \right) + \mathbf{n}(t) \quad 3.6$$

where $\mathbf{r}_{(k),m}(t) [\mathbf{p}_{(k),m}(t)]$ is the received signal vector due to the k th traffic (resp., pilot) transmission in the m th path of the channel and $\tau_{(k),m}$ is the corresponding transmission delay. To deal with the multipath components independently after despreading, we assume that the multipath components are resolvable, i.e., $\min|\tau_{(k),m} - \tau_{(k),m'}| > T_c$ for $m \neq m'$ and all k .

In each path, the channel vector is given by $\mathbf{h}_{(k),m}$. Assume that the propagation delays of the desired transmission are known. For the m th multipath component, the sampled matched filter outputs are given by $\mathbf{y}_m[\ell] = \int_{-\infty}^{+\infty} \mathbf{y}(t - \tau) \varphi(\tau) d\tau \Big|_{t = \ell T_c + \tau_{(1),m}}$.

Through the despreading operation, we can get despread traffic and pilot vectors of each path. The despread traffic of the m th path is written as

$$\mathbf{r}_{(1),m}[n] = A_{(1)} \mathbf{h}_{(1),m} s_{(1)}[n] + \mathbf{i}_{r,(1),m}[n] \quad 3.7$$

where $\mathbf{i}_{r,(1),m}[n]$ is the *IPN* vector in the despread traffic vector sequence of the m th path. In addition, the despread pilot vector of the m th path is written as

$$\mathbf{p}_{(1),m}[n] = \bar{A}_{(1)} \mathbf{h}_{(1),m} \bar{s} + \mathbf{i}_{p,(1),m}[n] \quad 3.8$$

where $\mathbf{i}_{p,(1),m}[n]$ denotes the *IPN* in the despread pilot vector sequence. The subscript (1) to denote the first transmission will be omitted hereafter for notational convenience except when explicitly needed. Furthermore, without loss of generality, we assume that $A = 1$ and $\bar{A} = \xi$. We define the *PTR*, η as $\eta = (\bar{A}^2 / A^2) = \xi^2$. Thus, 3.7 and 3.8 are rewritten as

$$\mathbf{r}_m[n] = \mathbf{h}_m s[n] + \mathbf{i}_{r,m}[n] \quad \text{and} \quad \mathbf{p}_m[n] = \xi \mathbf{h}_m \bar{s} + \mathbf{i}_{p,m}[n] \quad 3.9$$

respectively.

In [NAG96], the two-dimensional *Rake* (*2D-Rake*) combiner, wherein the *beamforming* weights are computed according to the approach presented in [SUA93], has been proposed for a base station equipped with multiple antennas. Recently, the approaches in [SUA93] have been modified to account for the pilot channel presence, wherein the processing is, instead, performed on the received signal after despreading [JIN00].

Moreover, in the presence of a training sequence, an interesting pool of adaptive algorithms can be used to compute the *beamforming* weights for the *2D-Rake* receiver [HAY96]. *2D-Rake* combiner can provide the estimate of the symbol $s_l[n]$ using the M vectors $L \times 1$ given in (3.7).

In addition, if the *IPN* vectors are temporally uncorrelated and possibly spatially correlated [JIN00],

$$\text{i.e. } E\{\bar{\mathbf{i}}_l[n] \bar{\mathbf{i}}_l^H[n]\} = \mathbf{R}_{i,r,m} \delta_{m,l},$$

where $\bar{\mathbf{i}}_l[n] \triangleq [\mathbf{i}_{r,1}^T[n] \quad \mathbf{i}_{r,2}^T[n] \quad \cdots \quad \mathbf{i}_{r,M}^T[n]]^T$, then, the maximum SINR (MSINR) symbol estimate (decision variable) is given by [JIN00]

$$\hat{s}[n] = \sum_{m=1}^M \mathbf{w}_m^H \mathbf{r}_m[n] \quad 3.10$$

where

$$\mathbf{w}_m \triangleq [w_{m,1} \quad w_{m,2} \quad \cdots \quad w_{m,L}]^T = \mathbf{R}_{i,r,m}^{-1} \mathbf{h}_m \quad 3.11$$

In the following subsection, we will review the approaches to compute the weight vectors \mathbf{w}_m that are used to combine the dispread signals from L antennas [NAG96], [SUA93] and [JIN00].

It is worth mentioning that in the code-filtering approach [NAG96], [SUA93]; both the received signal vector $\mathbf{y}_m[\ell]$ and the dispread signal vector $\mathbf{r}_m[n]$ are used. Since the received signal vector $\mathbf{y}_m[\ell]$ is used, the computation at the chip rate is required. On the other hand, the approaches in [JIN00] only use the dispread signal vector $\mathbf{r}_m[n]$ to compute the weight vectors. It implies that the approaches in [JIN00] have an advantage over the code-filtering approach in the computation.

3.3 Non adaptive pilot-channel aided beamforming techniques

3.3.1 Direct Approach (MRC)

Pilot channel-aided channel vector estimation is possible because the pilot symbol in the pilot channel is known. A simple method is the averaging technique to estimate the channel vector. The estimate of the channel vector associated with the m th path is given by

$$\hat{\mathbf{h}}_m = \frac{1}{J\xi} \sum_{n=0}^{J-1} \mathbf{p}_m[n] \bar{s}^* \quad 3.12$$

where J is the number of sample vectors for averaging. Alternatively, the estimate of the vector \mathbf{h}_m in 3.12 can be recursively computed using *Wiener LMS* algorithm for time varying channels [LIN01]. *Wiener LMS* considers and efficiently incorporates the channel dynamics into the filtering/smoothing/prediction process. It was successfully

used to estimate and track time varying channel in *WCDMA (Wide band CDMA)* and *CDMA2000* systems [OJA05]. This approach is also called *Maximum Ratio Combining (MRC)*, wherein the weight vector is made equal to the channel vector.

$$\mathbf{w}_m = [w_{m,1} \quad w_{m,2} \quad \cdots \quad w_{m,L}]^T = \hat{\mathbf{h}}_m \quad 3.13$$

3.3.2 Direct Approach (DMI)

In the eigen-decomposition approach, since the channel vector estimate is computed from the eigenvector, the power algorithm or steepest ascent algorithm can be used [GOL83], [OJA83].

In addition to the estimates of the channel vectors, estimates of the covariance matrices of the *IPN* vectors are required. Let us define the two covariance matrices as $\mathbf{R}_{r,m} \triangleq E[\mathbf{r}_m[n]\mathbf{r}_m^H[n]]$ and $\mathbf{R}_{p,m} \triangleq E[\mathbf{p}_m[n]\mathbf{p}_m^H[n]]$. It can be shown that [JIN00] $\mathbf{R}_{r,m} = \mathbf{h}_m\mathbf{h}_m^H + \mathbf{R}_{i,r,m}$ and $\mathbf{R}_{p,m} = \eta\mathbf{h}_m\mathbf{h}_m^H + \mathbf{R}_{i,p,m}$, where $\mathbf{R}_{i,r,m} \triangleq E[\mathbf{i}_{r,m}[n]\mathbf{i}_{r,m}^H[n]]$ and $\mathbf{R}_{i,p,m} \triangleq E[\mathbf{i}_{p,m}[n]\mathbf{i}_{p,m}^H[n]]$. It has also been observed that $\mathbf{R}_{i,r,m} = \mathbf{R}_{i,p,m} = \mathbf{R}_{i,m}$ [JIN00].

The estimates of $\mathbf{R}_{i,r,m}$ and $\mathbf{R}_{i,p,m}$ are obtained by

$$\hat{\mathbf{R}}_{r,m} = \frac{1}{J} \sum_{n=0}^{J-1} \mathbf{r}_m[n]\mathbf{r}_m^H[n] \quad \text{and} \quad \hat{\mathbf{R}}_{p,m} = \frac{1}{J} \sum_{n=0}^{J-1} \mathbf{p}_m[n]\mathbf{p}_m^H[n] \quad 3.14$$

The covariance matrices of the *IPN* vectors can be estimated as

$$\hat{\mathbf{R}}_{i,m} = \frac{1}{2} (\hat{\mathbf{R}}_{p,m} + \hat{\mathbf{R}}_{r,m} - (1+\eta)\hat{\mathbf{h}}_m\hat{\mathbf{h}}_m^H) \quad 3.15$$

The above estimates are not guaranteed to be positive definite. The weight vectors \mathbf{w}_m are now given by

$$\mathbf{w}_m = \hat{\mathbf{R}}_{\mathbf{t},m}^{-1} \hat{\mathbf{h}}_m \quad 3.16$$

from the estimates in 3.13 and 3.15. Notice that the above approach to provide the weight vector is a *Direct Matrix Inversion (DMI)* approach in [MON80]. In addition, there is no scalar ambiguity with this approach. This algorithm is later re-designed (SD-DMI), to reduce its complexity for a hardware implementation

3.3.3 Eigen-Decomposition Approach

In the presence of the pilot transmission, the eigen-decomposition approach in [MON80] is modified as follows. Using (13), the estimate of the channel vector \mathbf{h}_m is obtained by the eigen-decomposition [GOL83] of $\hat{\mathbf{R}}_{\mathbf{h},m} = \frac{1}{1-\gamma} (\hat{\mathbf{R}}_{\mathbf{r},m} - \hat{\mathbf{R}}_{\mathbf{p},m})$ [JIN00].

Let $\hat{\mathbf{e}}_{\mathbf{h},m}$ denote the eigenvector associated with the maximum eigenvalue, $\hat{\lambda}_{\mathbf{h},m}$, of the matrix $\hat{\mathbf{R}}_{\mathbf{h},m}$. Therefore, \mathbf{h}_m can be estimated as

$$\hat{\mathbf{h}}_m = \sqrt{\hat{\lambda}_{\mathbf{h},m}} \hat{\mathbf{e}}_{\mathbf{h},m} \quad 3.17$$

The estimate in 3.17 still has the phase ambiguity. Using the dispread pilot transmission $\mathbf{p}_m[n]$, the phase ambiguity can be resolved by letting $\hat{\phi}_m = \angle \left(\sum_{n=0}^{J-1} \mathbf{p}_m^H[n] \hat{\mathbf{e}}_{\mathbf{h},m} \right) - \angle(\bar{s})$, where $\angle(Ae^{j\theta}) = \theta$, so that the estimate of the vector \mathbf{h}_m can be written as

$$\hat{\mathbf{h}}_m = e^{-j\hat{\phi}_m} \sqrt{\hat{\lambda}_{\mathbf{h},m}} \hat{\mathbf{e}}_{\mathbf{h},m} \quad 3.18$$

In order to compute the weight vectors \mathbf{w}_m , the covariance matrices $\mathbf{R}_{\mathbf{t},m}$ shall be estimated. From 3.14, the estimate of the covariance matrix $\mathbf{R}_{\mathbf{t},m}$ is now given by

$$\hat{\mathbf{R}}_{i,m} = \frac{1}{1-\eta} (\hat{\mathbf{R}}_{p,m} - \eta \hat{\mathbf{R}}_{r,m}) \quad 3.19$$

It is worth to mention that the estimate $\hat{\mathbf{R}}_{i,m}$ in 3.19 can be replaced by the estimate in 3.15, while $\hat{\mathbf{h}}_m$ is given in 3.18.

3.3.4 Code Filtering Approach

The approach in [SUA93] operates on the received signal $\mathbf{y}_m[\ell]$ and the dispread signal $\mathbf{r}_m[n]$. Note that since the pilot transmission is available, the scalar ambiguity can be resolved as well. The latter approach, referred to us as "Chip-EVD", can provide the optimal weight vectors.

Let $\check{\mathbf{e}}_{h,m}$ denote the eigenvector associated with the maximum eigenvalue, $\check{\lambda}_{h,m}$ of the generalized eigen decomposition of the matrix pencil $(\mathbf{R}_{y,m}, \mathbf{R}_{r,m})$, so that

$$\hat{\mathbf{h}}_m = \sqrt{\check{\lambda}_{h,m}} \check{\mathbf{e}}_{h,m} \quad 3.20$$

Here $\mathbf{R}_{y,m}$ is approximated by $\mathbf{R}_{y,m} = \frac{1}{JN} \sum_{\ell=1}^{JN} \mathbf{y}_m[\ell] \mathbf{y}_m^H[\ell]$ and $\mathbf{R}_{r,m}$ by $\hat{\mathbf{R}}_{r,m}$.

Note that the complexity burden of the Chip-EVD technique stems mainly from the computation of $\hat{\mathbf{R}}_{y,m}$. The estimate in 3.20 still has a phase ambiguity. Using the dispread pilot transmission $\mathbf{p}_m[n]$, the phase ambiguity can be resolved by letting

$\check{\phi}_m = \angle \left(\sum_{n=0}^{J-1} \mathbf{p}_m^H[n] \check{\mathbf{e}}_{h,m} \right) - \angle(\bar{s})$ so that the estimate of the vector \mathbf{h}_m can be written as

$$\check{\mathbf{h}}_m = e^{-j\check{\phi}_m} \sqrt{\check{\lambda}_{h,m}} \check{\mathbf{e}}_{h,m} \quad 3.21$$

The performances of the *Chip-EVD* approach are found to be affected by the processing gain being used instead of *PTR* as in the *EVD* approach [NAG96], [SUA93].

As the processing gain approaches unity, the *Chip-EVD* approach suffers from performance degradation.

3.4 Adaptive pilot-channel aided beamforming techniques

The literature on applying the adaptive algorithm to compute the weight vectors is abundant [HAY96]. Recently, an attractive *LMS* based adaptive technique based on incorporating partial information on the channel is suggested [WEI02], the adaptive scheme falls under the variable-step-size *LMS* family. From the other side, unlike *LMS* based methods, *RLS* technique, yet more complex, is not sensitive to the eigen spread of the covariance matrix of the received signal.

Eigen spread is more pronounced in case of near far situation and time varying channel conditions [CAI00]. Adaptive techniques can be applied using pilot transmission as a training sequence. The weight vector is computed according to the following *MMSE* criterion

$$\mathbf{w}_m = \arg \min_{\mathbf{w}} E \left\{ \left| \mathbf{w}^H \mathbf{p}_m - \bar{s} \right|^2 \right\} \quad 3.22$$

The processing is performed after despreading which maintain the low implementation complexity. As the pilot strength may be as low as few *dB*'s compared to the traffic transmission power, some adaptive algorithms (eg *LMS* and *RLS*) may suffer from the high interference level due to traffic transmission. A two stage approach [CAI00] recently applied for multiuser detection can be used.

In [CAI00], in contrast to the decision directed approach, which suffers from losing tracking time varying channel after deep fades, a two stage strategy is adopted wherein no data is needed. The first stage is implemented in a blind mode while the

second stage uses the pre-detected symbols as a training sequence. In this work a two-stage approach is performed. However, the blind stage is replaced by a non-blind adaptive scheme using despread pilot transmission.

The second stage considers the detected traffic transmission from the first stage to compute the weight vector of the second stage. At low *PTR* ratio, as low as -12dB, the two-stage approach provides performance improvement over the single stage one.

3.4.1 Least Mean Squares (LMS)

It is a commonly used algorithm. The *LMS* algorithm is a stochastic gradient algorithm, which consists of two basic processes: the first one is Filtering which involves (a) computing the output of a transversal filter produced by a set of tap inputs, and (b) generating an estimation error by comparing this output to a desired response; the second process is adaptation which involves the automatic adjustment of the tap weights of the filter in accordance with the estimation error [HAY96]. For more about this algorithm the reader is referred to [GOD97b], [GOD04], [HAY96], [MOO99], and [SLO93].

The *LMS* algorithm can be summarized as follows:

1. The filter output is computed as

$$y_m[n] = \hat{\mathbf{w}}_m^H[n] \mathbf{P}_m[n] \quad 3.23$$

2. Estimation error is computed as

$$\mathbf{e}_m[n] = y_m[n] - \bar{s}[n] \quad 3.24$$

3. Weight adaptation

$$\hat{\mathbf{w}}_m[n+1] = \hat{\mathbf{w}}_m[n] + \mu \mathbf{P}_m[n] \mathbf{e}_m^*[n] \quad 3.25$$

Where H denotes the transpose, and $*$ denotes the conjugate. $\mathbf{P}_m[n]$ is the input signal, $\bar{s}[n]$ is the reference signal, n is the number of iterations, L is the number of taps or receiving antennas, μ is the adaptation step size, between 0 and $\frac{2}{L-Input_power}$ where $L-Input_power$ is the sum of the mean-square values of all the tap inputs in the filter:

$$L-Input_power = \sum_{k=0}^{L-1} E[|\mathbf{P}_m(n-k)|^2] \quad 3.26$$

3.4.2 Normalized Least Mean Squares (NLMS)

The *NLMS* algorithm is a companion of the ordinary *LMS* algorithm [HAY96]. It has been introduced to resolve certain problem concerning the input vector in the *LMS* algorithm. That problem can be summarized as follows: As in the standard *LMS* algorithm, the correction $\mu\mathbf{P}_m[n]e^*[n]$ applied to the tap weight vector $\hat{\mathbf{w}}_m[n]$ at iteration $[n+1]$ is directly proportional to the tap-input vector $\mathbf{P}_m[n]$.

Therefore, when $\mathbf{P}_m[n]$ is large, the *LMS* algorithm experiences a gradient noise amplification problem. So the correction applied to the tap-weight vector $\hat{\mathbf{w}}_m[n]$ at iteration $[n+1]$ is “normalized” with respect to the squared Euclidean norm of the tap-input vector $\mathbf{P}_m[n]$ at iteration $[n]$, hence the term “normalized”. For more about this algorithm, the reader is referred to [GOD04], [HAY96], [MOO99], and [SLO93]. This algorithm can be summarized as follows:

The *NLMS* algorithm can be summarized as follows:

1. The filter output is computed as

$$y_m[n] = \hat{\mathbf{w}}_m^H[n] \mathbf{P}_m[n] \quad 3.27$$

2. Estimation error is computed as

$$\mathbf{e}_m[n] = y_m[n] - \bar{s}[n] \quad 3.28$$

3. The weight adaptation is computed as

$$\hat{\mathbf{w}}_m[n+1] = \hat{\mathbf{w}}_m[n] + \frac{\tilde{\mu}}{a + \|\mathbf{P}_m[n]\|^2} \mathbf{P}_m[n] e^*[n] \quad 3.29$$

Where H denotes the transpose, and $*$ denotes the conjugate. $\mathbf{P}_m[n]$ is the input signal, $\bar{s}[n]$ is the reference signal, n is the number of iterations, L is the number of taps or receiving antennas, μ is the adaptation step size, and a is a small constant.

3.4.3 Noise Constrained Least Mean Squares (NC-LMS)

NC-LMS is a type of variable step-size *LMS* (*VS-LMS*) algorithm where the step-size rule arises naturally from the constraints. *VS-LMS* algorithms are a popular modification of requirements of large step-size to maximize convergence rate and small step-size to minimize misadjustment.

What is particular about this algorithm is that it exploits knowledge of channel noise variance for identification and tracking of *Finite Impulse Response (FIR)* channels. So the knowledge of the noise variance might be useful in selecting search directions and/or step-size in an adaptive algorithm. This algorithm has not been applied to beamforming yet, in the aforementioned papers.

We chose this algorithm to evaluate its complexity and performance comparing to other adaptive algorithms. For more about this algorithm the reader is referred to [YON01].

The *NC-LMS* algorithm can be summarized as follows:

1. The step size adaptation is computed as

$$\alpha_m[n'] = \alpha(1 + \gamma\lambda_m[n']) \quad 3.30$$

$$\lambda_m[n'+1] = \lambda_m[n'] + \beta \left(\frac{1}{2} (\varepsilon_m^2[n'] - \sigma_m^2) - \lambda_m[n'] \right) \quad 3.31$$

2. The estimation error is computed as

$$\varepsilon_m[n'] = \xi \bar{s}[n'] - \mathbf{w}_m^H[n'] \mathbf{p}_m[n'] \quad 3.32$$

3. The weight adaptation is computed as

$$\mathbf{w}_m[n'+1] = \mathbf{w}_m[n'] + \alpha_m[n'] \varepsilon_m[n'] \mathbf{p}_m[n'] \quad 3.33$$

where $\bar{s}[n']$ is the reference pilot signal, $\mathbf{p}_m[n']$ is the input signal,

$\sigma_m = \frac{1}{\sqrt{L}} E(\|i_{p,m}^H[n'] i_{p,m}[n']\|)$ is the average *IPN* variance across the array, $\varepsilon_m[n']$ is the estimation error and $\mathbf{w}_m[n']$ is the weight vector. α , β and γ are three constants, $\alpha_m[n']$ and $\lambda_m[n']$ are used for step-size adaptation.

3.4.4 Recursive Least Squares (RLS)

The *RLS* is an extension of the method of the least square, where the update estimate of the tap-weight vector is at iteration n upon the arrival of new data is computed. This algorithm may be viewed as special case of the *Kalman* filter [HAY96]. For more about this algorithm the reader is referred to [GOD97b], [GOD04], and [HAY96].

The *RLS* algorithm can be summarized as follows:

1. Initialization

$$\mathbf{P}(0) = \delta^{-1}\mathbf{I} \quad 3.34$$

2. The gain vector is computed as

$$\mathbf{K}[n] = \frac{\lambda^{-1}\mathbf{P}[n-1]\mathbf{u}[n]}{1 + \lambda^{-1}\mathbf{u}^H[n]\mathbf{P}[n-1]\mathbf{u}[n]} \quad 3.35$$

3. The estimation error is computed as

$$\boldsymbol{\alpha}[n] = \mathbf{d}[n] - \mathbf{w}^H[n-1]\mathbf{u}[n] \quad 3.36$$

4. The weight adaptation is computed as

$$\mathbf{w}[n] = \mathbf{w}[n-1] + \mathbf{k}[n]\boldsymbol{\alpha}^*[n] \quad 3.37$$

5. The Correlation matrix adaptation is computed

$$\mathbf{p}[n] = \lambda^{-1}\mathbf{p}[n-1] - \lambda^{-1}\mathbf{k}[n]\mathbf{u}^H[n]\mathbf{p}[n-1] \quad 3.38$$

Where δ is a small positive number, \mathbf{I} is the $L \times L$ identity matrix, where H is the number of taps. λ is the forgetting factor, between 0 and 1, $\mathbf{u}[n]$ is the input signal, H denotes the transpose, and $*$ denotes the conjugate, $\mathbf{d}[n]$ is the reference signal and $[n]$ is the number of iterations.

3.4.5 Set Membership Identification (SMI)

SMI theory is extended to the more general problem of linear-in-parameters filtering by defining a set-membership specification, as opposed to a bounded noise assumption. It is one of *Optimal Bounding Ellipsoids (OBE)* algorithms.

It is known as the set-membership *NLMS* algorithm, it has an optimized adaptive step size and it has never been applied to beamforming. For more about this algorithm, please refer to [GOL98].

The *SMI* algorithm can be summarized as follows:

1. The error estimation can be computed as

$$\delta[n] = \mathbf{d}[n] - \hat{\boldsymbol{\theta}}^H[n-1] \mathbf{x}[n] \quad 3.39$$

2. The weight adaptation is computed as

$$\boldsymbol{\theta}[n] = \boldsymbol{\theta}[n-1] + \alpha[n] \frac{\delta^*[n] \mathbf{x}[n]}{\mathbf{x}^H[n] \mathbf{x}[n]} \quad 3.40$$

3. The adaptation of step-size matrices can be computed as

$$\alpha[n] = \begin{cases} 1 - \frac{\gamma}{|\delta[n]|}, & \text{if } |\delta[n]| > 1 \\ 0, & \text{else} \end{cases} \quad 3.41$$

Where $d[n]$ is the reference signal or the desired response, $\mathbf{x}[n]$ is the input signal, H denotes the transpose, and $*$ denotes the conjugate, $[n]$ is the number of iterations, $\alpha[n]$ is the function that adapts the step-size, and γ is the only variable parameter in this algorithm (between 0 and 1).

3.5 Simulations results

In order to compare the performance of the beamforming approaches described above, consider the following system parameters and channel conditions. A base station is equipped with an *Uniform Linear Array (ULA)*. The *ULA* consists of half wavelength spaced L antennas. The angles of arrivals are uniformly distributed within $[0 \ \theta_{\max}]$, where $\theta_{\max} = 120^\circ$. The array response vector $\mathbf{a}(\theta)$ is modeled as

$$\mathbf{a}(\theta) = [1, \exp(j\pi \sin(\theta)), \dots, \exp(j(m-1)\pi \sin(\theta)), \dots, \exp(j(L-1)\pi \sin(\theta))]^T$$

. These settings, along with good antenna spacing, result in $E[\mathbf{a}(\theta) \mathbf{a}^H(\theta)] \simeq \mathbf{I}_{L \times L}$

[SUA93]. Two orthogonal codes are used for the traffic and pilot transmissions. Considering a processing gain of 8, the orthogonal codes for the traffic and the pilot

transmissions are $\{1, -1, 1, -1, 1, -1, 1, -1\}$ and $\{1, 1, 1, 1, 1, 1, 1, 1\}$ respectively. The different mobile stations are identified by their assigned unique random spreading (scrambling) sequences. Therefore the spreading sequences for each mobile can be viewed as the multiplication of the orthogonal codes and the random spreading sequences. For the unite energy pulse shaping filter, the root raised cosine with a roll off factor of 0.22 is used.

The channel is modeled as a 3 path time varying *Raleigh* fading process. The paths relative delays are $[0, 1.1, 3.19]\mu s$ with the respective relative powers of $[0, -3, -9]dB$. The absolute delays are randomly generated so that the maximum delay does not exceed the round trip delay within the base station coverage range of $5km$ cell radius. The *SINR* for the additive white Gaussian noise is set to 10 dB.

Other system parameters include the carrier frequency of 2GHz, the chip rate of $3.84Mcps$, and the mobile speeds of $60km/h$ -important for time-varying fading generation. The figures are representative of *WCDMA* system parameters. Unless otherwise stated the number of antennas is 4, the *PTR* is set to -6dB and the number of users is 5.

The transmission is packetized as follows; a normalized slot size of 2560 chip is considered as a basic packet, then frames consisting of 15 slots are constructed (worth 38400 chips). For the different approaches, we consider that the number of samples available for the parameters update is 2 slots worth of samples ($2 \times 2560/8 = 2 \times 320$ samples) after despreading. Below we analyze the effect of *PTR*, the array's size and number of users on the different approaches' performance.

3.5.1 PTR Effect

The effect of PTR is analyzed by sweeping the PTR over the range of $[-12\ 0] dB$. It has been found that the EVD approach exhibits performance degradation as the PTR increases [8]. Figure 3.2 below reveals the same effect. As for the DMI approach, a BER degradation in the low PTR region is observed.

It has been shown that the degradation could be higher due to channel estimation error using despread pilot transmission averaging [JIN00]. We observed, through simulations, that averaging over two slots provides good channel vector estimate. With the $Chip-EVD$ approach, the performance is as good as the MRC approach. This relatively expected poor performance is not affected by the level of PTR , However, it is due to the low processing gain being used (here 8). Notice that the adaptive approaches based on RLS are performing equally well at high PTR . Interestingly, at low PTR level the two stage RLS ($TS-RLS$) technique provides a relatively better BER performance.

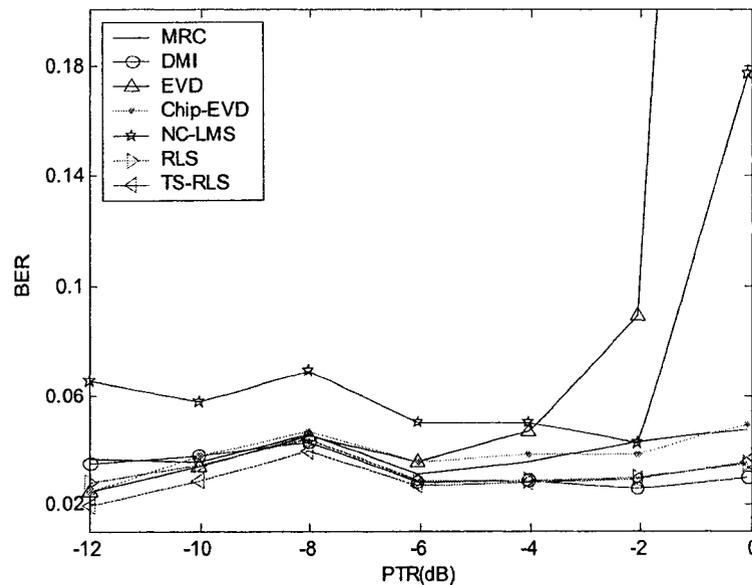


Figure 3.2: PTR effect.

3.5.2 Number of Antenna Effect

To analyze the effect of the number of antennas on the different approaches' performances, we set the *PTR* level to -6dB and maintain the number of users to 5. The results in Figure 3.2 reveals that all the beamforming approaches benefit from the increase in the number of antennas. It has been shown in [JIN00] that the *DMI* and *EVD* approaches' performances do not get improved after lets say 4 antennas.

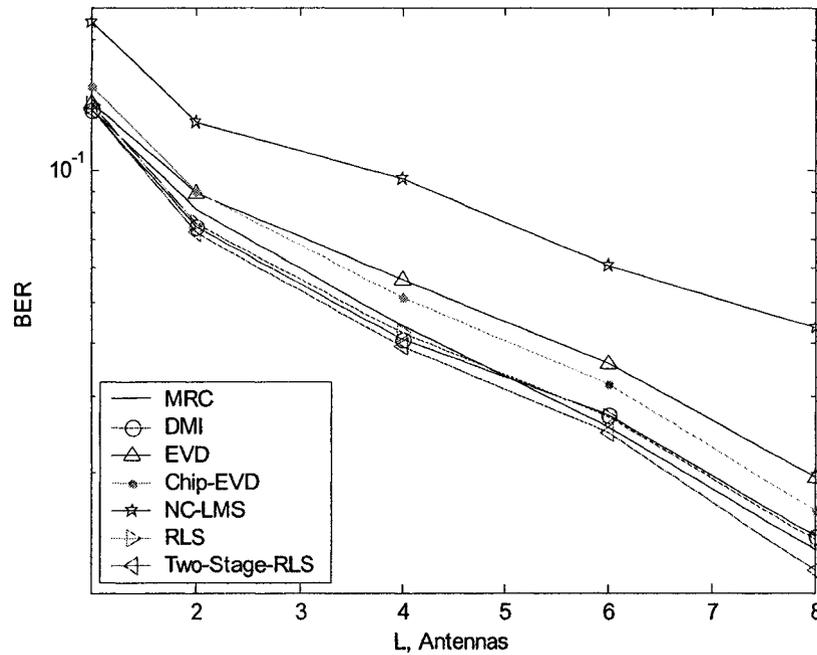


Figure 3.3: Number of antenna effect.

This observation is attributed to the errors in the estimation of the channel vector and *IPN* covariance matrix. Utilizing an averaging window of 2 slots in our simulation helps to overcome such problems. The relative performances are kept the same as in the previous subsection. It is worth to notice the poor performance of the *EVD* as compared to other schemes. This is attributed to the relatively high *PTR* level (-6dB) being used. The *Chip-EVD* approach suffers from the performance degradation due to possible

relatively low processing gain. Once again the two-stage *RLS* technique seems to outperform all other methods. However, nor substantial improvement is relatively recorded, neither for the single stage *RLS* technique nor for the *DMI*.

3.5.3 Number of Users Effect

The last experiment involves the effect of *MAI*. With a *PTR* level to $-6dB$ and array size of 4 antennas. The results (see Figure 3.4) show that the adaptive two-stage approach performs better than all the remaining approaches at different system loads (capacity, number of users). Compared to the *EVD* and *Chip-EVD* approaches, the two-stage *RLS* technique provides a relatively substantial capacity increase. However, *DMI* and even *MRC* are found to compete with almost similar performances.

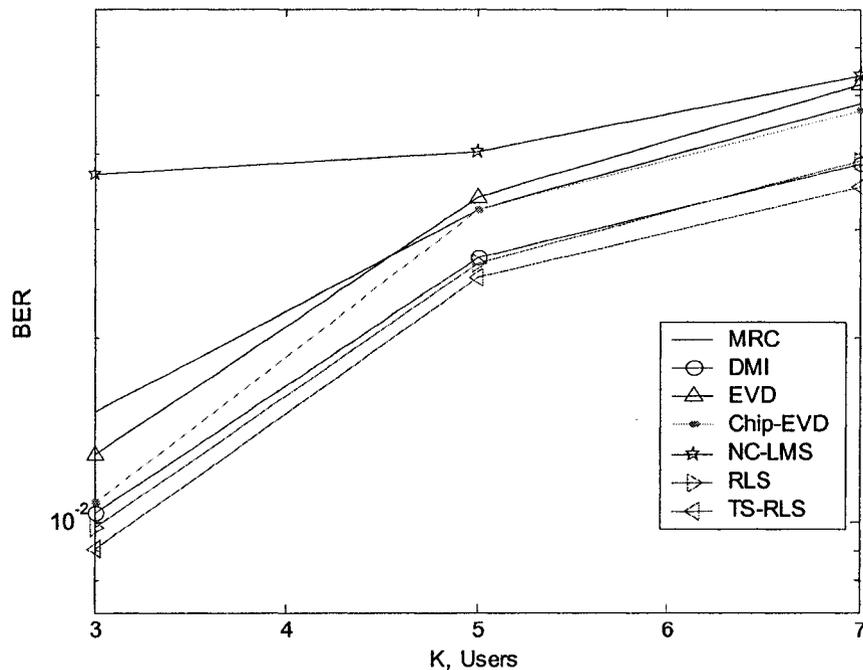


Figure 3.4: Number of users effect.

3.6 Summary

In *beamforming*, both the amplitude and phase of each antenna element are controlled. Combined amplitude and phase control can be used to adjust side lobe levels and steer nulls better than can be achieved by phase control alone. So as a result, the beamformer for a radio transmitter applies the complex weights to the transmit signal, shifts the phase and sets the amplitude for each element of the antenna array.

Beamforming techniques, adaptive and non adaptive approaches, have been discussed in this chapter. Simulations results have been presented based on a *DS-CDMA* platform, and for different parameters. The purpose of this evaluation study is to propose an algorithm which is favorable for hardware implementation.

Thus, the global target is to achieve a performance/complexity tradeoff; however when an efficient algorithm is presented such as the *DMI* a challenge in the corresponding architecture is about to begin. As we have mentioned before, that the algorithm although his efficiency and accuracy remains invaluable unless someone succeeds to implement it at a moderate cost.

Therefore it is important to look for a performance/complexity tradeoff when searching for the best candidate for *SAs*, however it's more important to drive up the design level by implementing a complex algorithm, and most importantly is to redesign that complex algorithm so that non-complex low power architecture can be revealed.

These hardware challenges take on the next chapter, where different implementations methods and techniques for different *beamforming* algorithms are presented and discussed.

Chapter 4

FPGA Implementation of Beamforming Techniques

4.1 Introduction

According to [GAM05], wireless infrastructure revenue continues to experience phenomenal growth, increasing from approximately \$27 billion in 2003 to an estimated \$35 billion in 2004. Industry analysts are predicting that 2004 will be the peak revenue year, as forecasts show the revenue figure dropping back to \$27 billion in 2005, eventually settling in to the \$10-\$15 billion range by the end of the decade. This revenue decline is driven both by lower prices as well as a drop in base station deployments, from nearly 500,000 stations in 2004 to less than 200,000 in 2010.

To begin addressing the challenge, wireless *BS* designs are shifting from *ASIC* technology to readily available off-the-shelf components such as *FPGAs* which are increasingly being used for signal processing applications. This shift is driven both by declining annual base station unit volumes as well as *FPGA* technology improvements that increase processing power and enable a much lower cost per channel [GAM05].

FPGAs are increasingly being used for signal processing applications. They provide the necessary performance and flexibility to tackle many of today's most challenging *DSP* applications, from *MIMO* digital communications systems to *H.264* encoding to a high definition broadcast system. Within such systems, *FPGAs* are ideally suited high-performance tasks traditionally served by *ASICs* and *ASSP* (*Application Specified System Processor*) [TAH05].

The migration to *FPGAs* is not just an attempt to reduce costs and create a common platform to achieve commoditization, but it also being driven by time-to-market pressures, along with the need to make in-field upgrades of *BS* deployments [GAM05].

Hence, among the major challenges is the ones associated with the implementation of future wireless communication systems (e.g., beamforming receiver). The design of low-complexity beamforming algorithms and corresponding *VLSI* (*Very Large Scale Integration*) architectures (or re-designing existing algorithms so that massive parallelism and bit level arithmetic present in these algorithms can be revealed and efficiently implemented in *VLSI* architecture) constitutes an interesting example.

In this chapter we present an *FPGA* implementation of three *beamforming* algorithms: *Maximum Ratio Combining (MRC)*, *Noise Constrained Least Mean Squares (NC-LMS)* and *Direct Matrix Inversion (DMI)*. The first two methods were implemented using rapid prototyping techniques while the third one is implemented using hand coded *VHDL* (*VHSIC Hardware Description Language*). Fixed point data representation and arithmetic are investigated to evaluate the effect of finite word length. The targeted *FPGA* devices are Virtex-II, Virtex-II Pro and Virtex-IV families of Xilinx.

4.2 Implementation of Beamforming algorithms

Considering the hardware implementation for *beamforming* algorithms, few past works have been reported. In [POW93], a reduced sampling rate *beamforming* technique for the reception of acoustic data transmitted through the water has been implemented on a *DSP* from Texas Instrument. In [NUT02], the authors present an *FPGA* implementation using *Xilinx* devices and calibration methodologies for a digital *beamforming* system consisting of eight channels and operating in *L* band (1.8 to 2GHz). A comparison of efficient broadband *beamforming* architectures is presented in [WEI02]; however no comparison on hardware resource is reported.

A low power *ASIC* implementation of 2Mbps antenna-rake combiner supporting both *Maximum Ratio Combining (MRC)* and *Least Mean Square (LMS)* filter is presented in [TAR05] where a 107 MHz clock frequency and 550mW as highest rate power are achieved. In [CES05] a 1.7 MSPS (megasamples per second) has been achieved in an implementation of a matrix inversion on a Virtex4XC4VSX55 for *beamforming* using the *RLS* approximation with *QR* decomposition of the input data matrix.

4.3 FPGA implementation of MRC and NC-LMS

4.3.1 Complexity analysis

MRC is chosen because it has the lowest complexity in hardware terms among the evaluated algorithms. It is considered as a reference for most of the *beamforming* algorithms.

NC-LMS is chosen because it represents an intermediate complexity between *MRC* and *DMI* or *RLS*. It is an adaptive algorithm that hasn't been applied to *beamforming* yet in the aforementioned papers.

We chose this algorithm to evaluate, because (i) it belongs to the *LMS* family and (ii) to evaluate its complexity comparing to the *MRC*.

The number of arithmetic operations per iterations for both algorithms is given in Table 4.1.

Table 4.1: Number of arithmetic operations per iterations per antenna and per user for *MRC* and *NC-LMS*.

	<i>Adders</i>	<i>Subtracts</i>	<i>Delays</i>	<i>Multipliers</i>
<i>MRC</i>	4	2	2	8
<i>NC-LMS</i>	8	7	10	19

From the required number of arithmetic operations per iteration, we can see that the *NC-LMS* is about 2.5 more arithmetically complex than the *MRC*, this arithmetic complexity reflects directly the hardware complexity and the amount of resources required (*Silicon*) to implement the algorithm, and later on it plays a decisive role to choose the corresponding *FPGA*.

4.3.2 Proposed architectures

Based on the equations presented in Chapter 3, (3.12, 3.13) for the *MRC* and (3.30-3.33) for the *NC-LMS*, the proposed architectures are illustrated in Figure 4.1, the hexagon presents a complex arithmetic block.

System Generator maps system parameters defined in Simulink (e.g. as mask variables in *Xilinx Blockset* blocks), into entities and architectures, ports, signals, and attributes in a hardware realization. In addition, *System Generator* automatically produces command files for *FPGA* synthesis, *HDL* simulation, and implementation tools, so that the user can work entirely in graphical environments in going from system specification to hardware realization. For more about this tool, the reader is referred to [XIL05].

The other importance with our design is the balancing computation with *I/O* (*Input/Output*). Since a special-purpose system typically receives data and outputs results through an attached host, *I/O* considerations influence overall system. The proposed implementations have very good concurrency and good communication by using fast components. Besides the degree of concurrency is determined by the underlying algorithm whether *MRC* or *NC-LMS*.

The architecture of *MRC* and *NC-LMS* are mounted into Simulink blocks (Figure 4.2, 4.3 respectively).

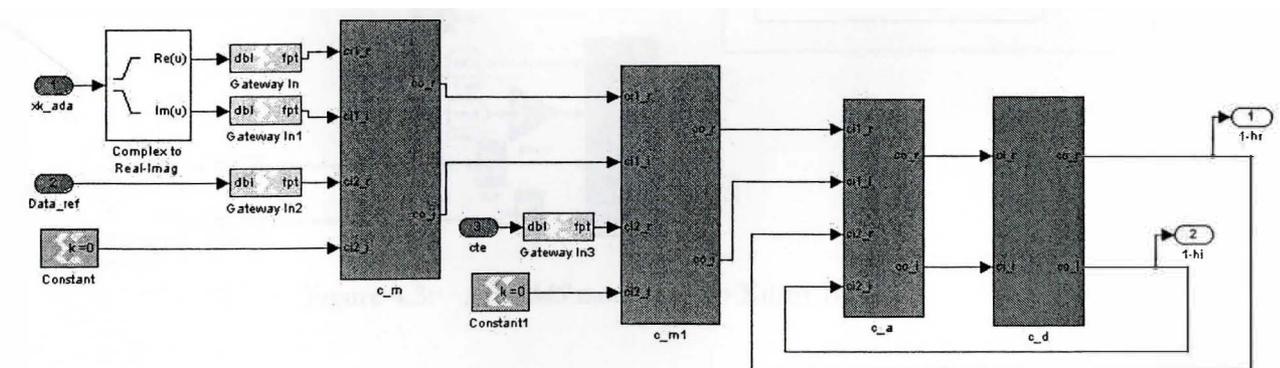


Figure 4.2: *MRC* mounted into Xilinx Blocks.

Both architectures have been synthesized using real data which has been generated from a *DS-CDMA* platform. The data has been converted to digital using the

Gateway In and *Gateway Out* blocks which provide an interface to the *Xilinx Blockset* in *Simulink*. *Xilinx Gateway* blocks handle the type conversions, since *MATLAB* uses double-precision floating-point and the *Xilinx* portion of the design uses fixed-point precision. *Xilinx Gateway* blocks handle the type conversions.

They play the role of *Analogic to Digital (A/D)* or *Digital to Analogic (D/A)* converters. The *Xilinx Gateway In* block represents an input port into the *FPGA*, while the *Gateway Out* block is an output port from the *FPGA*.

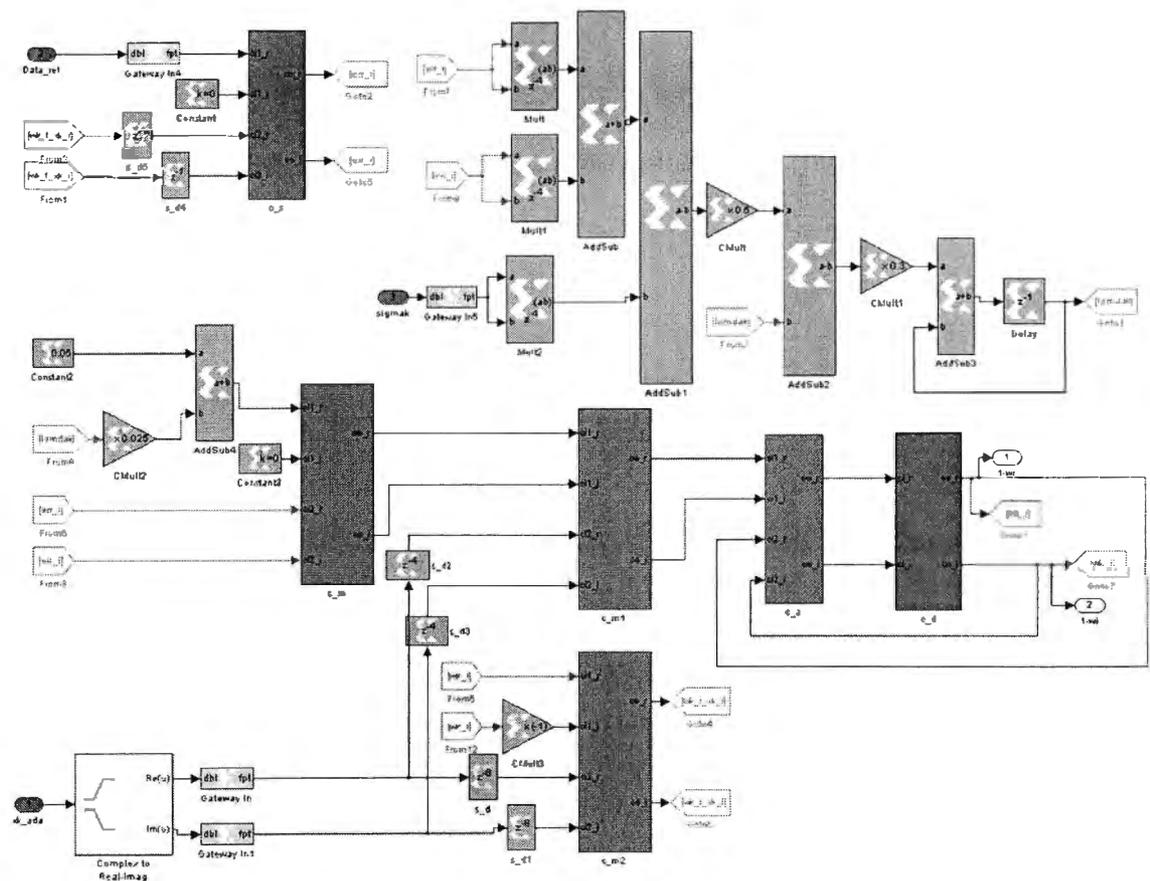


Figure 4.3: *NC-LMS* mounted into Xilinx Blocks.

It should be noted that we have a latency of 15 cycles and 10 cycles for *NC-LMS* and *MRC* respectively. The achieved clock frequency in both approaches is acceptable. For the *MRC* we have a minimum period of 10ns and a maximum frequency of 95 MHz, and

for the *NC-LMS* the minimum period and the maximum frequency are 30 ns and 33 MHz respectively.

4.3.4 Hardware resources

The hardware resources estimate include numbers of slices, lookup tables, memory blocks (BRAM), embedded multipliers, *I/O* blocks, flip flops (FF), and tristate buffers (TBUF). These estimates make it easy to determine how design choices affect hardware requirements. It is so clear and evident that the amount of *FPGA* hardware used is directly related to the data width, so it is best, for reasons of cost, circuit speed, and power dissipation to request only the precision required for a particular application. Table 4.2 and Table 4.3 show the hardware resources of one and four antennas respectively.

Table 4.2: Hardware resources for one antenna.

Maximum Ratio Combining MRC				
	(32, 16)	(24, 16)	(18, 10)	(16, 12)
Slices	1006	758	348	310
FFs	1328	1184	468	416
BRAMs	0	0	0	0
LUTs	1752	1272	360	320
IOBs	128	144	72	64
Embedded Mults	32	32	8	8
TBUFs	0	0	0	0
Noise Constrained Least Mean Squares NC-LMS				
Slices	2529	2019	1191	890
FFs	3138	2346	1242	1104
BRAMs	0	0	0	0
LUTs	4540	3534	1445	971
IOBs	128	144	72	64
Embedded Mults	60	60	15	15
TBUFs	0	0	0	0

The notation (32, 24)-bit indicates that the fixed point arithmetic operations represent 32-bit signed two's complement value, with 24 fractional value. It should be

noted that for a given bit representation e.g. (32, 24)-bit the amount of hardware for the *NC-LMS* is 2.5 times the amount of hardware for *MRC*, which shows the real complexity of the *NC-LMS*.

Table 4.3: Hardware resources for four antennas.

<i>Maximum Ratio Combining MRC</i>				
	<i>(32, 16)</i>	<i>(24, 16)</i>	<i>(18, 10)</i>	<i>(16, 12)</i>
Slices	4024	3032	1392	1240
FFs	5312	4736	1872	1664
BRAMs	0	0	0	0
LUTs	7008	5088	1440	1280
IOBs	512	576	288	256
Embedded Mults	128	128	32	32
TBUFs	0	0	0	0
<i>Noise Constrained Least Mean Squares NC-LMS</i>				
Slices	10116	8076	4764	3560
FFs	12552	9384	4968	4416
BRAMs	0	0	0	0
LUTs	18160	14136	5780	3884
IOBs	512	576	288	256
Embedded Mults	240	240	60	60
TBUFs	0	0	0	0

Another comparison between *MRC* and *NC-LMS* compare to percentage of number of slices and multipliers for a given *FPGA*. For this comparison, we chose three *FPGAs*, Virtex-II Pro XC2VP30 and XC2VP100, and Virtex-4 XC4VSX55. The results are shown in Table 4.4.

Table 4.4: Percentage of slices S and multipliers M for four antennas.

		<i>Maximum Ratio Combining MRC</i>			
		<i>(32, 16)</i>	<i>(24, 16)</i>	<i>(18, 10)</i>	<i>(16, 12)</i>
VP30	% S	29.4	22.1	10.1	9.0
	% M	N/A	N/A	47	47
VP100	% S	9.1	6.8	3.1	2.8
	% M	57.6	57.6	14.4	14.4
VSX55	% S	16.4	12.3	5.7	5.0
	% M	50	50	12.5	12.5
		<i>Noise Constrained Least Mean Squares NC-LMS</i>			
VP30	% S	73.9	58.9	34.7	26.0
	% M	N/A	N/A	88.2	88.2
VP100	% S	22.9	18.3	10.8	8.1
	% M	N/A	N/A	27	27
VSX55	% S	41.16	32.8	19.3	14.5
	% M	93.8	93.8	23.4	23.4

For a given bit-representation, compare to the percentage of slices the *MRC* barely scratches the surface of what can be done for the three chosen *FPGAs*, the *NC-LMS* shows low percentage but not as much as *MRC* for the same bit-wordlength. Comparing to the percentage of multipliers, we can see the *NC-LMS* needs almost the double amount of multipliers needed by *MRC* in the three chosen *FPGAs* (*N/A* marked show an overflow) and three times more of slices.

4.3.5 Quantization study

In the quantization effect study and evaluation (see Figure 4.4) the performance is evaluated based on the Loss in *dB* using the following equation:

$$\text{Loss} = 20 \log_{10} \left[\frac{1}{KM} \sum_{k=1}^K \sum_{m=1}^M E \left(\frac{\mathbf{w}_{(k),m}^H \hat{\mathbf{w}}_{(k),m}}{\|\mathbf{w}_{(k),m}\| \|\hat{\mathbf{w}}_{(k),m}\|} \right) \right] \text{dB} \quad 4.1$$

where $\mathbf{w}_{(k),m}^H$ and $\hat{\mathbf{w}}_{(k),m}$ are the result weight vector in floating point and fixed point respectively.

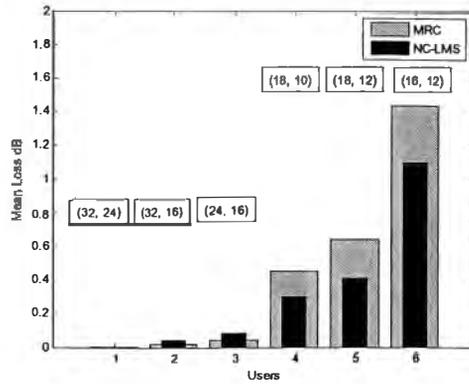


Figure 4.4: Quantization study results for *MRC* and *NC-LMS* using $N=8$, $K=5$ users, $PTR=0$ dB, mobile speeds of 60Km/h, carrier frequency 2GHz and chip rate of 1.25Mchip/s, and *Simulink* blocks.

It should be noted that both algorithms operate well for a (32, 24)-bit representation and, both show good performance for (32, 16)-bit representation. For an (18,12)-bit representation *MRC* shows a poor performance (Loss=-0.7 dB) while the *NC-LMS* shows a better performance (-0.45 dB).

4.4 FPGA design and implementation of DMI

4.4.1 Redesign of DMI

The *DMI* technique in the subsection 3.3.2 requires an inherent matrix inverse operation (3.16) which is generally considered as an expensive arithmetic operation that requires high word length for fixed point data representation and arithmetic. Besides, the time varying nature of the wireless channel requires performing such operation, if possible, at a symbol rate.

To overcome such problems we redesigned the *DMI* technique using the steepest descent method (*SD-DMI*) to (i) render the inherent matrix inversion problem

computationally feasible through an iterative method and to (ii) ensure good tracking properties for time varying channels.

A *VLSI* architecture targeted on *FPGAs* is presented. The goal is to maximize the number of users on a chip (or device in case of *FPGA*). Maximizing the number of users makes it possible to increase the capacity of the base station equipped with multiple antennas.

In [HOQ05] a study for the maximum number of users on a device is reported based on an *FPGA* implementation of a *Multuser Detector (MUD)*, however no work was done to maximum the number of users on a chip for *beamforming* or *SAs*.

Consider a window of width J samples. The value of the window's width will depend on the time varying channel. As the channel variations go higher, the width value goes lower. Applying gradient descent technique to the *DMI* technique (*SD-DMI*) will require the following operations:

1. The estimate of the channel vector $\hat{\mathbf{h}}_m[n]$ and the channel covariance matrix $\hat{\mathbf{R}}_h[n]$ associated with the m th path are recursively computed as

$$\hat{\mathbf{h}}_m[n] = \frac{1}{J\xi} \left((J-1)\hat{\mathbf{h}}_m[n-1] + \mathbf{p}_m[n]\bar{s}^* - \mathbf{p}_m[n-J]\bar{s}^* \right) \quad 4.2$$

$$\hat{\mathbf{R}}_h[n] = \hat{\mathbf{h}}_m[n]\hat{\mathbf{h}}_m^H[n] \quad 4.3$$

2. The covariance matrices $\mathbf{R}_{r,m}$ and $\mathbf{R}_{p,m}$ can, similarly, be computed as

$$\hat{\mathbf{R}}_{r,m}[n] = \frac{1}{J} \left((J-1)\hat{\mathbf{R}}_{r,m}[n-1] + \mathbf{r}_m[n]\mathbf{r}_m^H[n] - \mathbf{r}_m[n-J]\mathbf{r}_m^H[n-J] \right) \quad 4.4$$

and

$$\hat{\mathbf{R}}_{\mathbf{p},m}[n] = \frac{1}{J} \left((J-1) \hat{\mathbf{R}}_{\mathbf{p},m}[n-1] + \mathbf{p}_m[n] \mathbf{p}_m^H[n] - \mathbf{p}_m[n-J] \mathbf{p}_m^H[n-J] \right) \quad 4.5$$

3. The covariance matrices of the *IPN* vectors can be estimated as

$$\hat{\mathbf{R}}_{\mathbf{i},m}[n] = \frac{1}{2} \left(\hat{\mathbf{R}}_{\mathbf{p},m}[n] + \hat{\mathbf{R}}_{\mathbf{r},m}[n] - (1+\eta) \hat{\mathbf{R}}_{\mathbf{h}}[n] \right) \quad 4.6$$

4. Using gradient descent techniques the weight vectors $\mathbf{w}_m[n]$ are now given by

$$\mathbf{e}_m[n] = \hat{\mathbf{R}}_{\mathbf{i},m}[n] \mathbf{w}_m[n-1] - \hat{\mathbf{h}}_m[n] \quad 4.7$$

$$\mathbf{w}_m[n] = \mathbf{w}_m[n-1] - \mu \mathbf{e}_m[n] \quad 4.8$$

5. An estimate of the required symbol $s[n]$ will be given by

$$\hat{s}[n] = \text{sign} \left(\sum_{m=1}^M \mathbf{w}_m^H[n] \mathbf{r}_m[n] \right) \quad 4.9$$

4.4.2 Complexity analysis

We can briefly conclude from the simulation results that *SD-DMI* represents an attractive approach. Compared to the two-stage *RLS*, one can easily notice that the redesigned *DMI* approach *SD-DMI* offers relatively lower computations complexity (see Table 4.5).

Moreover the recursive nature renders its attractive for fast time varying channels. Besides, it is worth mentioning that the *RLS* algorithm represents potential divergence problems when implemented using fixed point data representation and arithmetic.

Unless efficient techniques like square QR decomposition is used, long wordlengths are recommended. For comparison, Table 4.5 depicts the number of real additions and multiplications per single pass weight update (one pilot sample) required per user for L antennas.

Table 4.5: Arithmetic Complexity of the SD-DMI (proposed) versus RLS , per antenna and per user.

Method	RLS	SD-DMI
Additions	$O(4L^3 + 12L^2 + 12L - 4)$	$O(22L^2 + 6L)$
Multiplications	$O(4L^3 + 8L^2 + 12L)$	$O(17L^2 + 4L)$

4.4.3 Quantization study

Prior to any implementation, fixed point representation and arithmetic effect should be studied. Toward this end, this section is devoted for fixed point simulation wherein it is noticed through trial and error that at least 13 bits are required. It will be interesting, in the light of the signal model (3.1)-(3.33), to develop a closed form formula to first calculate the amount of precision \wp (in bits) required by A/D converter as follows.

$$\wp(\text{in bits}) = \left\lceil \log_2 \left(K \left(\sqrt{\frac{N_p}{N_T}} + \sqrt{\eta} \right) \sum_{m=1}^M 4\sigma_{h_m} \frac{\sqrt{N_T}}{\sqrt{2}} + 4 \cdot 10^{\frac{\Gamma}{20}} \frac{\sqrt{N_T}}{\sqrt{2}} \right) \right\rceil + \Delta \quad 4.10$$

where K is the total number of users, N_p and N_T are the pilot and traffic spreading factors. The pilot-to-traffic power ratio is $\eta \triangleq \bar{A}_{(k)}^2 / A_{(k)}^2$ and the channel attenuations $h_m(t)$ for the m th path are complex Gaussian random variables with variance $\sigma_{h_m}^2$. The SNR is denoted by Γ (in dB).

The factor 4 accounts for the fact that with a probability of 0.99 that the received signal sample will be within the range. Δ bits for additional precision are added with one bit for the sign bit. Equation (4.10) gives precision in the range of 8-12bits, for almost all possible (feasible) combinations of the parameters stated above, which is possible with current A/D converters. For illustration consider $N_p = 256$, $N_r = 16$ (for 64kb/s data rate), $\eta = 0.1$, $M = 3$, $\sigma_{h_n}^2 \in \{0, -3, -9\}$ dB, $\Gamma = 12$ dB and $\Delta = 1$ then $\phi \approx 12$ bits.

Through simulations it is observed that the word length of 13 bit or higher will be required to maintain BER performances close to those in floating point arithmetic (see Figure 4.5). For $FPGA$ implementation, we consider a word length of 16 bits for fixed point data representation and arithmetic.

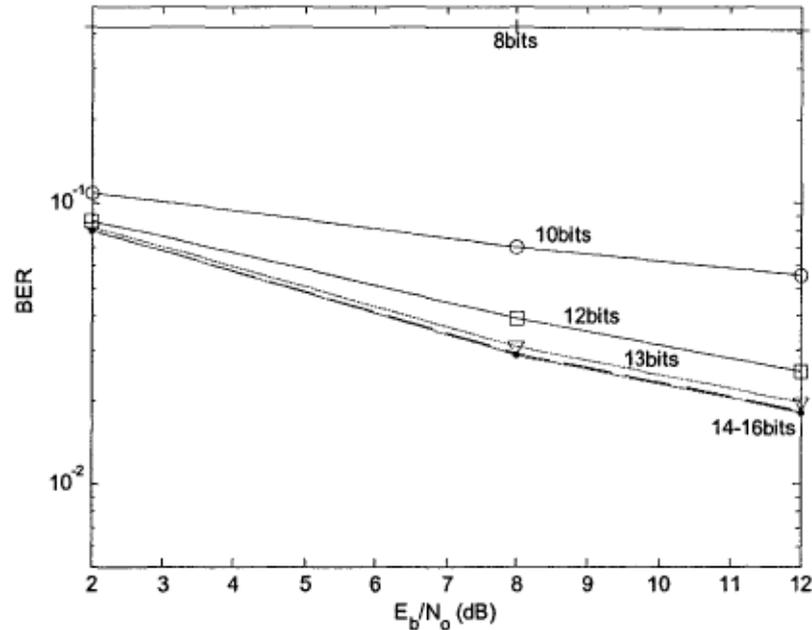


Figure 4.5: Fixed point results of the SD-DMI technique (4.2-4.9). $N = 16$ (Processing gain), $K = 10$ users, PTR=-6dB.

4.4.4 Proposed architecture

To achieve real-time implementation, either *DSP* processors or *VLSI* architectures could be applied. However, as real-time implementation is concerned, *System On Chip (SOC)* architectures offers more parallelism, more compact size and lower power consumption than general *DSP* processors [GUO06].

Considering the hardware implementation of matrix inversions few works have been reported. In [JEN85] a matrix inversion algorithm has been implemented in a limited-precision adaptive antenna array processor where the matrix inversion was performed in a Multibus processor (Marinco APB3024M), using digitized baseband samples from the antennas. The actual device accumulator used a 32-bit mantissa, and for this reason the *LU* decomposition of matrix inversion had been selected.

A high speed implementation of two matrix inversion algorithms, for general and symmetrical matrices in orthogonal systolic architectures, has been presented in [PAP88]. A direct *Cholesky* decomposition and iterative (Newton's iterative) matrix inversion methods have been implemented using *Altera's APEX FPGA* in [YLI04].

In [ECH05] a scalable pipelined complex valued matrix inversion architecture that performs a QR factorization and a triangular matrix inversion, has been proposed and implemented on a *FPGA* from *Xilinx*, where the hardware implementation can be used as a core processor in a real-time smart antenna system and in a wide variety of implementations of *beamforming* and *MIMO* algorithms as well.

In [SAL05] three matrix inversion implementations based on *Cholesky* decomposition in fixed-point are presented. An *FPGA* implementation of matrix

inversion using *QRD-RLS* algorithm is presented in [KAR05], where a throughput of 0.13 M updates per second was achieved on a Virtex4 *FPGA* running at 115 MHz.

The architecture must be able to respect real-time constraints therefore, an efficient implementation is essential to reduce the critical path delay, power and area of wireless receiver. To explore the proposed architecture, we elaborate the tasks as depicted in Table 4.6.

Table 4.6: Time steps derivation from the SD-DMI scheduling of equations (4.2 – 4.9)

STEPS	#EQ.	BLOCK	OPERATIONS	LATENCY	STEP LATENCY
1	(22)	b1	$\hat{\mathbf{h}}_m[n]$	L	$L+4$
	(25)	b2	$\mathbf{p}_m[n]\mathbf{p}_m^H[n]$	$L+4$	
	(24)	b2	$\mathbf{r}_m[n]\mathbf{r}_m^H[n]$	$L+4$	
2	(25)	b3	$\hat{\mathbf{R}}_{p,m}[n]$	L	$L+4$
	(24)	b3	$\hat{\mathbf{R}}_{r,m}[n]$	L	
	(23)	b2	$\hat{\mathbf{R}}_h[n]$	$L+4$	
3	(26)	b4	$\hat{\mathbf{R}}_{t,m}[n]$	L	L
4	(27)	b5	$\hat{\mathbf{R}}_{t,m}[n]\mathbf{w}_m[n-1]$	$2L+3$	$2L+4$
	(28)	b6	$\mathbf{w}_m[n]$	1	
5	(29)	b7	$\hat{s}[n]$	$L+4$	$L+4$
				Total Latency	$6L+16$
				Throughput	$(2L+4)F_{\text{clk}}$

With a timing- and data-dependency analysis, the top level block diagram for the *SD-DMI* algorithm is shown in Figure 4.6. This block diagram is a general architecture representing the data dependency and the data flow between different sub-blocks representing the arithmetic operations of (4.2)-(4.9) of the *SD-DMI*.

The architecture demonstrates parallelism and reduced redundancy. For example, to obtain the result of Step 1 we need b_1 and b_2 twice. The data path is balanced and facilitates the pipelining in multiple subblocks for high-speed implementation. In addition to that the developed architecture should be reconfigurable to several baseband processing *UMTS (Universal Mobile Telecommunication Systems)* systems, thus it can be reconfigured by respecting all systems constraints (hardware, algorithm, speed ...).

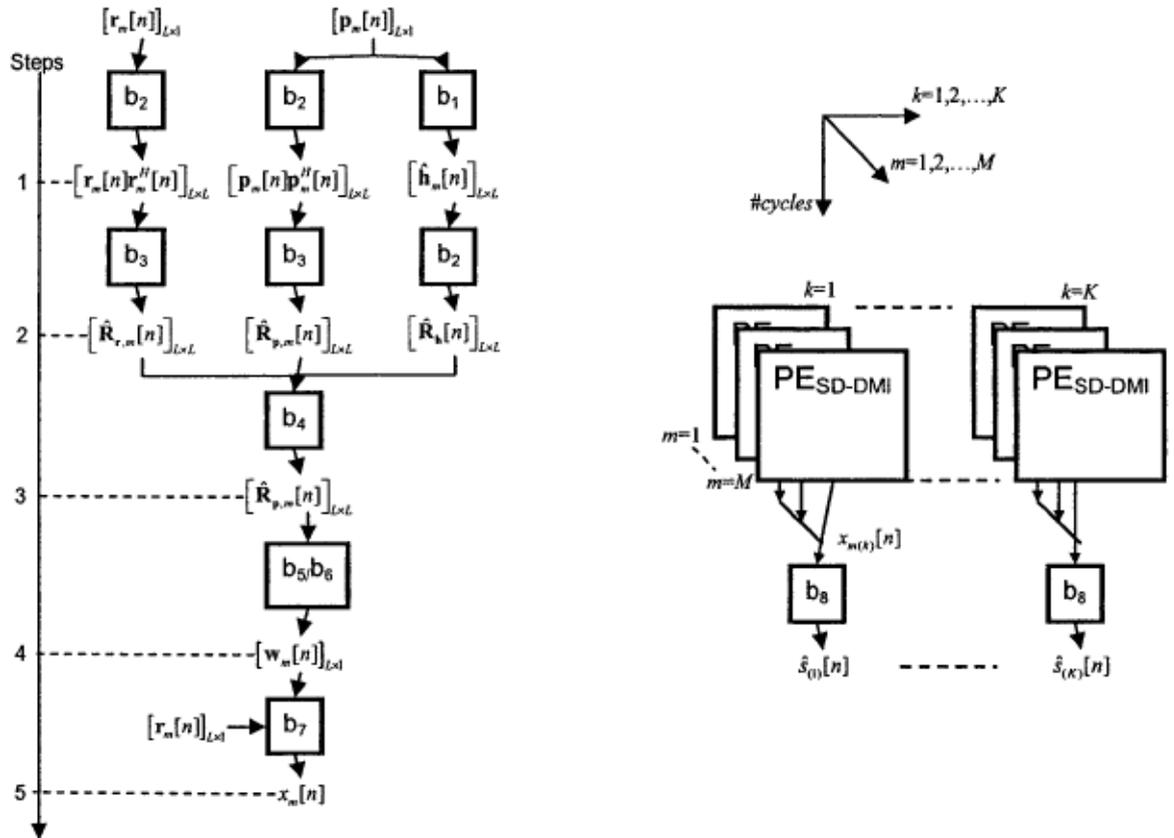


Figure 4.6: Proposed block diagram of the proposed *VLSI* architecture for the *SD-DMI* with the corresponding data flow.

Each block diagram shown is for one user, $k=1$ (multiple antennas are mounted inside the one block diagram), and one channel path, $m=1$. The architecture can be easily replicated for multiple users and multiple paths. The *VLSI* architecture shown in Figure

4.6 is verified through end-to-end VHDL simulations along with actual hardware implementation to determine the device resources.

4.4.5 Implementation technique

Matrix inversion implementation's problem inversion has been an important challenge and consideration for most of the hardware engineers. Engineers have been drawn to design a low complexity solution for this problem since it is considered as one of the main cores in most of the signal processing applications.

The choice of suitable hardware architecture for the implementation of equations (4.2-4.9) depends on the system specifications and on the available hardware resources [BUR06]. The role of bit-wordlength has an important effect on the hardware resources to execute with precision required the implemented algorithm. Hence, we assume an implementation on 16-bits wordlength architectures as we have shown them to have sufficient fixed-point range.

All *Process Elements (PEs)* mainly are implemented for complex numbers, however in some *PEs* as in blocks b3 and b7 there are no multiplications between two complex numbers so the architecture can be assumed to be replicated for the imaginary part. The complex multiplier is derived from [XIL05] and the equations in [SHA98]. The complex multiplier contents three real multipliers in a pipelined structure with a latency of 4 cycles and a throughput of a result each clock cycle F_{clk} [XIL05].

The architecture is coded in *VHDL* using Xilinx ISE which is invoked for gate level synthesis. ISE Foundation integrates architecture in a complete logic design environment for all leading Xilinx *FPGA* and *CPLD (Complex Programmable Logic Device)* products, for more about this tool; the reader is referred to [XIL05].

The targeted *FPGA* components are Virtex-II and Virtex-II Pro families of Xilinx. The bottlenecks in our design are the matrix-vector product in (4.7), the vector-vector product in (4.3, 4.4, 4.5, and 4.9). For these multiplications parallel and systolic architectures have been employed.

4.4.6 Mapping to VLSI architectures

High-level synthesis has become crucial for developing fast time-to-market delays for new *VLSI* implementations of signal processing algorithms in order to estimate the complexity which is an important consideration for real time implementations. However, the complexity in the hardware is reflected by the number of arithmetic operations, recursiveness, dataflow, and memory access.

Considering the fact that is unlikely to have more than four receiving antennas in the *BTS*, the analysis is made for four antennas ($L=4$).

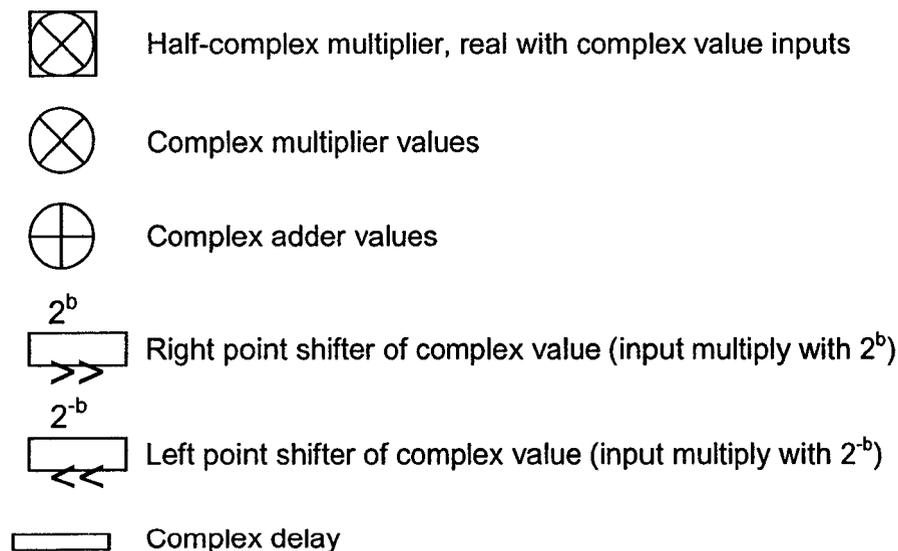


Figure 4.7: Legend of operators.

Equation (4.2) computes the channel vector. The block architecture with the corresponding process element PE for this equation is showed in Figure 4.8. The PE for this equation has two shifters, one complex addition and one complex subtraction. The number of PE in the block (b1) is equal to L . The computation of the channel vector requires a maximum of 20.680 ns.

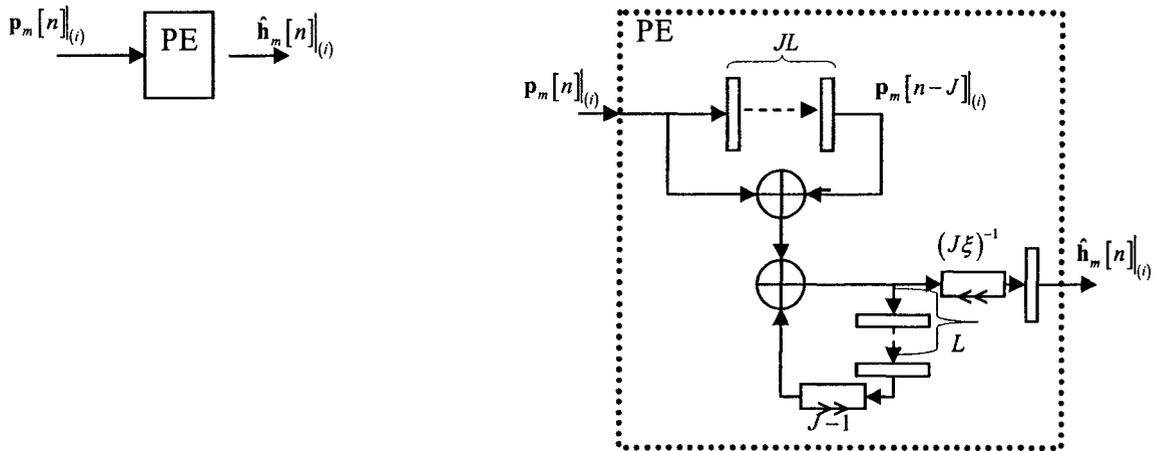


Figure 4.8: Block b1.

Block b2 is used for the vector-vector product where the dimensions of the two multiplied vectors are $L \times 1$ and $1 \times L$ yielding a $L \times L$ matrix. This block is used to calculate the partial covariance matrix for different input vectors as in (4.3, 4.4 and 4.5). The block architecture with the corresponding PE for this product is shown in Figure 4.9. The number of PE is equal to L^2 . The required time for computation of the matrix is 4.728 ns.

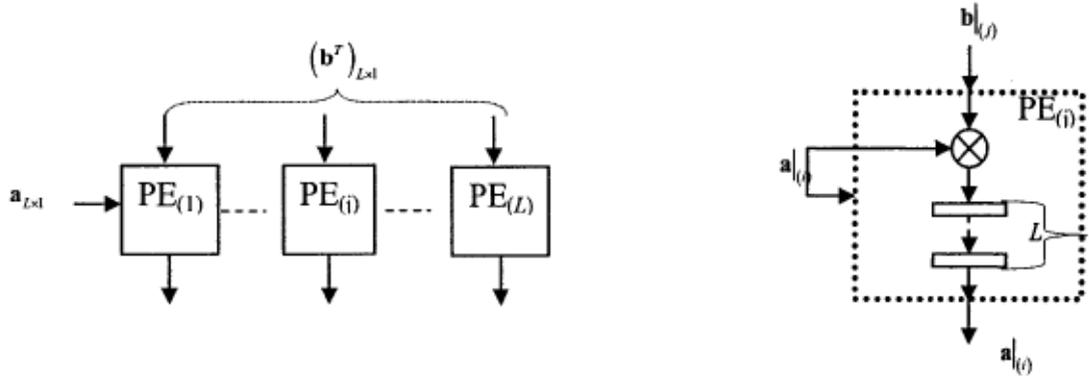


Figure 4.9: Block b2.

Equations (4.4) and (4.5) compute the covariance matrices for the pilot and data traffic respectively. The block architecture with the corresponding *PE* for this equation is showed in Figure 4.10. The same architecture can be applied for the pilot covariance matrix. The *PE* for this equation has one complex addition and one complex subtraction and two shifting operations. The number of delay elements can be determined based on the number of samples J . In addition to that, this *PE* has a $L \times L$ matrix input which has been computed by a separate block (b2). The number of the *PEs* in the block (b3) is L^2 . The computation of the covariance matrices of the pilot and the traffic requires of 11.120 ns each.

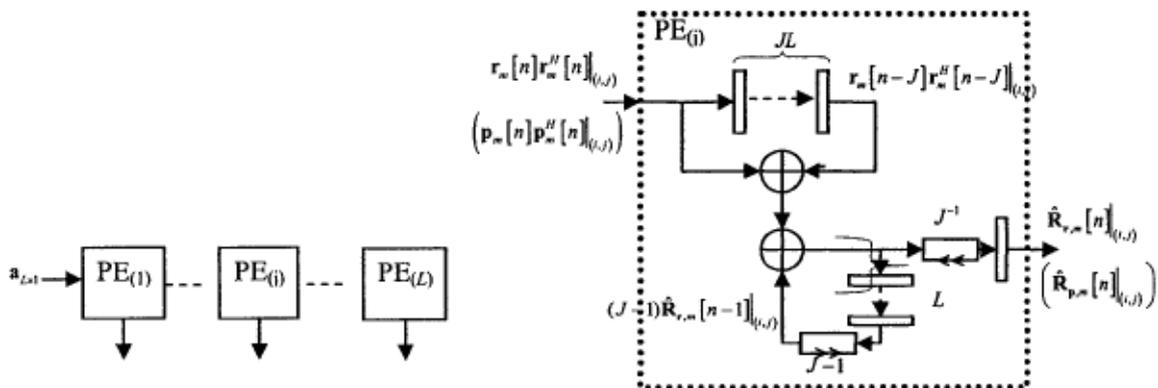


Figure 4.10: Block b3.

Equation (4.6) computes the covariance matrices of the *IPN* vectors. The block architecture with the corresponding *PE* for this equation is showed in Figure 4.11. The *PE* for this equation has two real shifting operations, one complex addition and one complex subtraction. The number of *PE* in the block (b4) is equal to L^2 . The computation of the $\hat{\mathbf{R}}_{i,m}[n]$ matrix requires 17.057 ns.

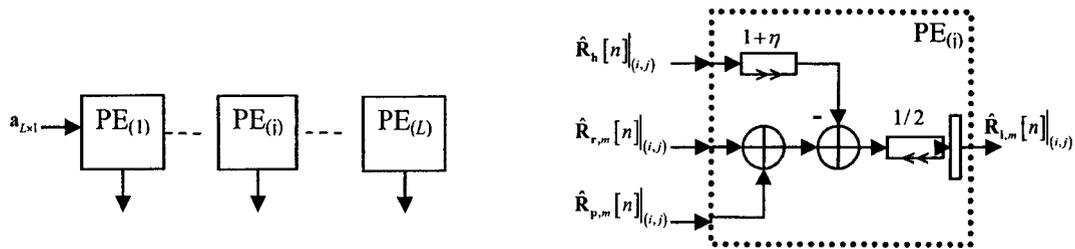


Figure 4.11: Block b4.

Equation (4.7) and (4.8) computes the error $\mathbf{e}_m[n]$ and the weight vector $\mathbf{w}_m[n]$. For the matrix-vector product $\hat{\mathbf{R}}_{i,m}[n]\mathbf{w}_m[n-1]$ a systolic architecture has been deployed (see Figure 4.12). The matrix-vector product is identified as block b5. For a $L \times L$ matrix and $L \times 1$ vector, the result is a $L \times 1$ vector. To compute the resulting vector $2L+1$ clock tops are needed. The number of *PE* in the block (b5) is equal to L . This architecture derives from the architecture presented in [QUI89]. There are many optimizations for the matrix multiplications in [QUI89] and [PAR99]. These optimizations can be used for area-time trade-offs architectures resulting in less power dissipation. This architecture has L complex-valued *Multiplier-Accumulator (MACs)* where each *MAC* has one complex multiplication, one complex addition and one complex register or delay element. The required time for computation of the vector is 9.15 ns.

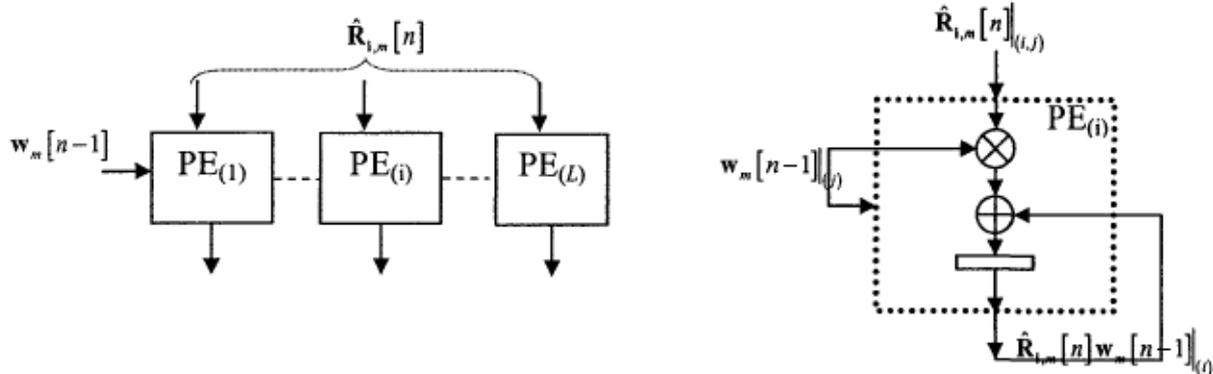


Figure 4.12: Block b5.

The block architecture with the corresponding *PE* for computing the weight vector is identified as block b6. The number of *PEs* in this block is equal to L (see Figure 4.13). This architecture has one shifting operation, one complex addition, one complex subtraction, one complex delay element. The required time for computation of the vector is 7.598 ns.

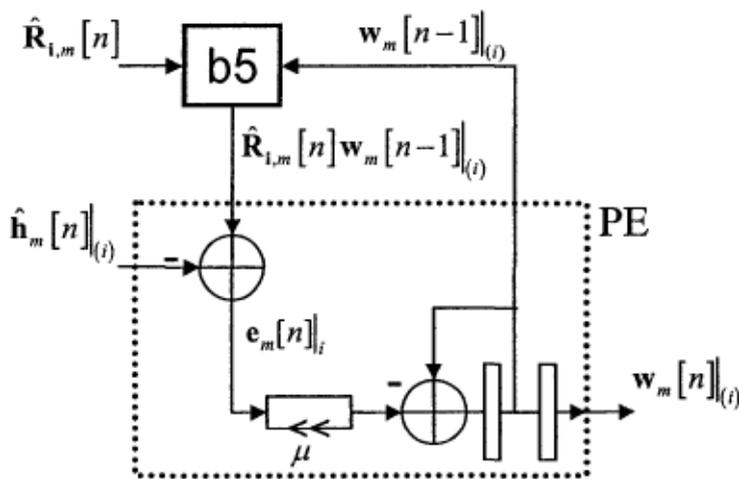


Figure 4.13: Block b6.

Equation (4.9) computes the estimate of the symbol for a certain path $m = \{1, \dots, M\}$. This is a vector-vector product where the dimensions of the two multiplied vectors are $1 \times L$ and $L \times 1$ the result is a scalar.

The architecture is shown in Figure 4.14 where one complex *MAC* is been used. This architecture is identified as block b7. The result is computed at the $L+2$ clock top. The computation of the estimate requires 4 ns.

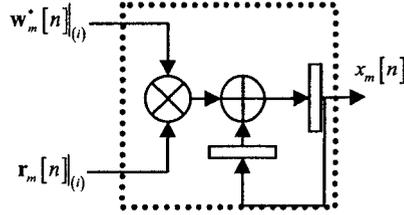


Figure 4.14: Block b7.

At last, all estimates for all paths are summed, which yields the estimate of the required symbol $\hat{s}[n]$.

4.4.7 Hardware resources

The hardware resources of the different blocks are shown in Table 4.7. The estimates include numbers of slices, lookup tables, flip-flops, memory blocks and multipliers. These estimates make it easy to determine how design choices affect hardware requirements. Block b2 requires 36 real multipliers (for four antennas), with a considerable number of slices. This block can be considered as bottlenecks in our design. Block b2 and b7 require the least computational time, though they are the fastest among all the others in the block diagram.

Table 4.7: Hardware resources for different blocks in the architecture for $L=4$, $J=10$ and $M=3$.

Block	b1	b2	b3	b4	b5	b6	b7
Slices	221	700	820	128	764	58	191
MULT. 18X18	0	12	0	0	12	0	3
Path Delay-max (ns)	6.96	4.76	5.27	5.88	4.76	3.40	4.76
Clock Freq. (MHz)	144	210	190	210	210	294	210

The computational time of b2 and b7 is the computational time to perform a multiplication. The largest computational time is for the computation of the weight vector (block b5 and b6) which is considered as the computational time for each task since the architecture is fully pipelined and the critical path is constrained to be set to less or equal to 20 ns (it is rather our design choice). All blocks have the same throughput, with quiet similar latencies except for block b4 and b6. In some blocks the latency depends on the number of antenna elements L as in b5 and b7 and the number of delay elements J as in b1 and b3.

The total latency (in cycles) for the first estimate is given by the following formula

$$T_{Latency} = 6L + 16 \quad 4.11$$

All the necessary variables to compute the mathematical expressions of the *SD-DMI* algorithm are available at the proper time at the proper block at the proper *PE*. Thus the data flow through the *VLSI* network is correct, though the *VLSI* architecture executes the algorithm correctly. It should be noted that the total architecture respects the total latency calculated using (4.11).

4.4.8 Tradeoffs for maximum number of users

Depending on the objective of the implementation, a trade-off can be chosen. There are many trade-offs in the *VLSI* architectures providing critical insights into implementation issues that may arise during the product development process. These optimizations or trade-offs make the proposed architecture more suitable for real-time implementation.

In order to achieve high throughput operations, pipelined and parallel processing are often used in the implementation of digital signal processing algorithms. Time-Constraints are respected since the architecture is fully pipelined. For area-constraints, the bottleneck lies in the area which multipliers and slices occupy. The number of slices and multipliers in the *FPGA* play a decisive role for determining the number of users a base station (equipped with 4 antenna elements) can serve.

In our application, the objective is to attain a maximum number of users in the cell based on the real time requirements. Therefore the area-constraints are closer to be a slices/multipliers trade-off. The following equations help finding the approximate maximum number of users.

Since the computational time of block b2 is twice the computational time of block b1, which are considered in the same task, block b2 (hardware) is used once. Hence the total number of slices for the proposed architecture is given by

$$S_{Total} \triangleq S_{b1} + 2S_{b2} + 2S_{b3} + S_{b4} + S_{b5} + S_{b6} + S_{b7} = 1453 \quad 4.12$$

And the total number of multipliers for the proposed architecture is given by

$$M_{Total} \triangleq M_{b1} + 2M_{b2} + 2M_{b3} + M_{b4} + M_{b5} + M_{b6} + M_{b7} = 36 \quad 4.13$$

The critical path is assumed to be less than 20 ns, though the maximum achieved throughput is $F_{max} = 50$ MHz. If the targeted throughput of the user is equal to $F = 2$ Mb/s, then we define the throughput factor F_f as

$$F_f \triangleq \frac{F_{max}}{F} = \frac{50}{2} = 25 \quad 4.14$$

If the channel has 3 paths, then we define the reuse factor R_f as

$$R_f \triangleq \frac{F_f}{M} = \frac{25}{3} \approx 8 \quad 4.15$$

The number of users can be determined based on two constraints, the number of slices and the number of multipliers; hence we define the *Maximum Core Unit (MCU)*:

$$MCU_{Slices} = \frac{\text{Number of slices available in a device}}{\text{Total required number of slices}} \quad 4.16$$

$$MCU_{Multipliers} = \frac{\text{Number of multipliers available in a device}}{\text{Total required number of multipliers}} \quad 4.17$$

Therefore the number of users is given by:

$$K_{\max} = \min(R_f \times MCU_{Slices}, R_f \times MCU_{Multipliers}) \quad 4.18$$

From Table 4.8, we can see that the maximum number of users we can attain, for the chosen Xilinx *FPGA* devices, is 90 (for Virtex2P XC2VP50). With a 2Mb/s for each user, the maximum number of users attained is a pretty goal for the DS-CDMA communication system.

Therefore a cheaper *FPGA* with/or a lower bit rate, allows for a large augmentation in the number of users to be served by the base station; on the other hand, for a state-of-the-art *FPGA*, such as the Xilinx Virtex IV and Virtex V the number of users to be served by the base station increases however a cost penalty comes along.

Table 4.8: Maximum number of users K^{MAX} over different Xilinx FPGA devices in the case of $L=4$, $J=10$ and $M=3$.

MAX. NUMBER OF USERS	VIRTEX 2				VIRTEX 2 PRO			
	XC2V80	XC2V1000	XC2V3000	XC2V8000	XC2VP2	XC2VP4	XC2VP30	XC2VP50
K^{MAX}	2	15	37	65	4	10	53	90

4.5 Summary

In this chapter, *FPGA* implementations of *beamforming* receiver based on *MRC* and *NC-LMS* for *DS-CDMA* System have been presented and studied.

However, due to the adaptive nature of *NC-LMS* the loss in computing the weight vectors is lower compared to *MRC* which in the overall makes *NC-LMS* attractive in a low bit-wordlength, but due to the low complexity of *MRC*, it is more suitable for *FPGA* implementation than the *NC-LMS*. Pipelining and parallelism can be introduced in both architectures to allow for higher clock frequencies though higher throughputs.

Moreover, for high performance, robust transmission schemes require the implementation of complex signal processing algorithms such as the *DMI*. An *FPGA* implementation for the *DMI* has been presented, after resolving the matrix inversion using a gradient descent method (*SD-DMI*). Performance analysis for *beamforming* techniques and simulations in fixed-point are carried out, and hardware resources are shown.

The algorithm is massively parallel and is suitable for a fixed-point *VLSI* implementation. We have demonstrated that there is no loss in the algorithm performance due to this redesign of the algorithm. Results showed that we can attain a sufficient number of users to be served in a cell, using complex algorithm and at a modest cost.

The *FPGA* implementation can be used as a core processor in a real-time *SA* system and in a wide variety of *MIMO* algorithms. Furthermore the *FPGA* implementation can easily be ported to an *ASIC* implementation.

Chapter 5

Conclusion

As the industry transitions from a high-growth phase to a more mature state, cost pressures will increasingly mount in all facets of the infrastructure, including the wireless base station. Theoretically, this cost should be correlated with the augmentation in the performance of the base station, in other words, next-generation base station deployments must conquer the challenge of continually reducing cost (as measured by cost per channel) while adding functionality to support new services, protocols, and changing subscriber usage patterns.

However can we maintain a moderate cost when launching the next-generation base station?

SA technology can significantly improve wireless system performance and economics for a range of potential users. It enables operators of *Personal Communication Service (PCS)*, cellular, and *Wireless Local Loop (WLL)* networks to realize significant increase in signal quality, capacity, and coverage.

Nevertheless, the success of the *SA* systems relies on two considerations that have been often overlooked when investigating *SA* technologies: first, the *SAs* features need to be considered early in the design phase of future wireless systems which is called the top-down feasibility; second a realistic performance evaluation of *SA* techniques need to be performed according to the critical parameters associated with future systems requirements and this is called bottom-up feasibility.

The first consideration was the objective of Chapter 2 of this work, where a global overview of *SA* systems was studied and discussed. The second consideration consist the main core of Chapter 3, where we focused on the research of the best *beamforming* algorithm, leading the *SA* system to give his main beam towards *SOI*, and nulls towards *SNOI*, in any wireless communication environment.

However, based on the simulations results, we have found that the *SD-DMI* is our best candidate for *beamforming*; it is a very efficient and powerful algorithm. Hence, most of signal processing engineers recommend this algorithm to the industry due to its high performance; on the other hand, hardware engineers are trying to avoid it as much as they can, or if not, worst case scenario, be looking for new techniques to implement it, ensuring no loss or degradation in the performance.

As the famous quote about challenge for Tommy LASORDA got our mind “the difference between the impossible and the possible lies in person’s determination”, we, hardware and signal processing engineers have accepted the challenge of the implementation of the *DMI*.

We have redesigned the algorithm, so that massive parallelism and pipelining and bit level arithmetic can be revealed. The redesign of the *DMI* has made it more powerful especially ensuring tracking properties for time varying channels. We have

proposed an efficient *FPGA* implementation, and estimated the maximum number of users a base station (equipped with four antenna elements) can serve. This is done using Xilinx ISE, or *VHDL* manual coding, the good thing about this software is that it allow the designer to verify each level of the design, so each level separately can be identified and checked, plus the properties in placing and routing, assigning package pins, area and time constraints.

We have implemented also two other *beamforming* algorithms, *MRC* and *NC-LMS*, using rapid prototyping methodology. These two techniques represent an intermediate complexity comparing to the *SD-DMI*. This rapid prototyping technique is fast in design, “just connecting blocks – no *VHDL* coding”, however it suffers when it comes to timing, scheduling and routing.

In conclusion, the manual *VHDL* coding stays the reference for deeply hardware designing and implementation, it allows the designer to interact with every stage of the design, while the rapid prototyping technique, in our own perspective point of view, serves for fast testing of the design, because the design even it is coded in a known *HDL*, it remains weird and a foreign language to the designer. Once the design is verified with rapid prototyping technique, the designer can start working deeply with *ISE*.

For future work and challenge, there is a large possibility for extending the work of this thesis by:

- Developing an algorithm, more powerful than the *DMI* and the *TS-RLS* ($BER < 0.02$).
- Developing VLSI architectures with low power and low complexity for the developed algorithms. These architectures play the role of a core

processor in a real-time smart antenna system and in a wide variety of implementations of *beamforming* and *MIMO* algorithms as well.

- Implementing the developed architectures on *FPGAs* and *ASICs*.
- Developing a *SOC* based on *SA*.

References

- [KRA88] J. D. Kraus, *ANTENNAS*: McGraw-Hill, 1988.
- [BOL06] Ivo Bolsens, "Programming modern FPGA platforms", *IEEE 17th International Conference on Application-specific Systems, Architectures and Processors*, Steamboat Springs, Colorado, September 11-13, 2006.
- [BHO01] A. U. Bhobe and P. L. Perini, "An Overview of Smart Antenna Technology For Wireless Communication," *IEEE Proceedings of Aerospace Conference*, Volume 2, 10-17 March 2001 Page(s):2/875 - 2/883 vol.2.
- [BOU00] A. O. Boukalov and S. G. Haggman, "System aspects of smart-antenna technology in cellular wireless communications-an overview," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, pp. 919, 2000.
- [BEL02a] S. Bellofiore, C. A. Balanis, J. Foutz, and A. S. Spanias, "Smart-antenna systems for mobile communication networks. Part 1. Overview and antenna design," *Antennas and Propagation Magazine, IEEE*, vol. 44, pp. 145, 2002.
- [IECOE] International Engineering Consortium, Online Education, Web ProForum Tutorials, "Smart Antenna Systems", http://www.iec.org/online/tutorials/smart_ant/
- [ALE04] A. Alexiou and M. Haardt, "Smart antenna technologies for future wireless systems: trends and challenges," *Communications Magazine, IEEE*, vol. 42, pp. 90, 2004.
- [ZHA01] M. Zhai and Y. Liu, "An overview of spatial channel models used in smart antenna system analysis", *International Conferences on*

Info-tech and Info-net.ICII 2001 – Beijing, Volume 2, 29 Oct.-1 Nov. 2001 Page(s):542 - 548 vol.2.

- [SHI97] J. Shiann-Shiun, O. Garret, and X. Guanghan, "Experimental evaluation of smart antenna system performance for capacity improvement," *IEEE Global Telecommunications Conference, GLOBECOM '97*, Volume 1, 3-8 Nov. 1997 Page(s):369 – 373.
- [CHR00] M. Chryssomallis, "Smart antennas," *Antennas and Propagation Magazine, IEEE*, vol. 42, pp. 129, 2000.
- [SEU99] C. Seungwon, S. Donghee, and T. K. Sarkar, "A comparison of tracking-beam arrays and switching-beam arrays operating in a CDMA mobile communication channel," *Antennas and Propagation Magazine, IEEE*, vol. 41, pp. 10, 1999.
- [LEB99] j. Joseph C. Leberti, Theodore C. Liberti, JR., Theodore S. Rappaport, *Smart antennas for Wireless Coomunications*: Prentice Hall, 1999.
- [ROY97] R. H. Roy, "An overview of smart antenna technology and its application to wireless communication sybstems," *IEEE International Conference on Personal Wireless Communications*, 17-19 Dec. 1997 Page(s):234 – 238.
- [GOD97a] L. C. Godara, "Applications of antenna arrays to mobile communications. I. Performance improvement, feasibility, and system considerations," *Proceedings of the IEEE*, vol. 85, pp. 1031, 1997.
- [GOD97b] L. C. Godara, "Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations," *Proceedings of the IEEE*, vol. 85, pp. 1195, 1997.
- [JIN05] H. Jin and A. Acampora, "A Reservation-Based Media Access Control (MAC) Protocol Design for Cellular Systems Using Smart Antennas— Part I. Flat Fading," *IEEE Transactions on Wireless Communications*, vol. 4, pp. 792, 2005.
- [KAI05] T. Kaiser, "When will smart antennas be ready for the market? Part

- I," *Signal Processing Magazine, IEEE*, vol. 22, pp. 87, 2005.
- [OHI02] T. Ohira, "Analog smart antennas: an overview," *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Volume 4, 15-18 Sept. 2002 Page(s):1502 - 1506 vol.4.
- [PEN02] J. Peng and Z. Duan, "The research of adaptive algorithm of smart antenna arrays in the CDMA system," Proceedings of the 4th World Congress on Intelligent Control and Automation, Volume 2, 10-14 June 2002 Page(s):838 - 841 vol.2
- [GOD04] L. C. Godara, *SMART ANTENNAS*: CRC PRESS, 2004.
- [SHI98] J. Shiann-Shiun, G. T. Okamoto, X. Guanghan, L. Hsin-Piao, and W. J. Vogel, "Experimental evaluation of smart antenna system performance for wireless communications," *IEEE Transactions on Antennas and Propagation*, vol. 46, pp. 749, 1998.
- [JIN00] C. Jinho, "Pilot channel-aided techniques to compute the beamforming vector for CDMA systems with antenna array," *IEEE Transactions on Vehicular Technology*, , vol. 49, pp. 1760, 2000.
- [NAG96] A.F. Naguib, "Adaptive antennas for CDMA wireless networks," PhD. Dissertation, Stanford University, Stanford, CA, 1996.
- [SUA93] B. Suard, A. Naguib, G. Xu, and A. Paulraj, "Performance Analysis of CDMA Mobile Communication Systems Using Antenna Arrays," in *Proc. Int Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Mineapolis, MN, April 1993, pp. 153-156
- [HAY96] S. Haykin, *Adaptive Filter Theory*, Third ed: Prentice Hall, 1996.
- [LIN01] L. Lindbom, M. Sternad, and A. Ahlen, "Tracking of time-varying mobile radio channels .1. The Wiener LMS algorithm," *IEEE Transactions on Communications*, vol. 49, pp. 2207-2217, 2001.
- [OUA05] M. Ahmed Ouameur and D. Massicotte, "Multiuser Wiener LMS for Time Varying Multipath Channel Estimation and Tracking in DS-CDMA Systems," submitted to *EURASIP Journal on Applied*

- Signal Processing, Special Issues on Reliable Communication over Rapidly Time-Varying Channels, June 1st, 2005.
- [GOL83] G.H. Golub and C. F. Van Loan, *Matrix Computations*, Baltimore, MD: the John Hopkins Univ. Press, 1983.
- [OJA83] E. Oja, *Subspace Methods of Pattern Recognition*. New York: Wiley, 1983.
- [WEI02] S. Weiss and I. K. Proudler, "Comparing efficient broadband beamforming architectures and their performance trade-offs", *14th International Conference on Digital Signal Processing, DSP Volume 1*, 2002 Page(s):417 - 423 vol.1.
- [CAI00] G. Caire, "Two-stage nondata-aided adaptive linear receivers for DS/CDMA," *IEEE Transactions on Communications*, vol. 48, pp. 1712-1724, 2000.
- [SLO93] D. T. M. Slock, "On the convergence behavior of the LMS and the normalized LMS algorithms," *IEEE Transactions on Signal Processing*, , vol. 41, pp. 2811, 1993.
- [MOO99] Wynn. C. Stirling. Todd K. Moon, "Mathematical Methods and Algorithms for Signal Processing", *Prentice Hall*, 1999.
- [YON01] W. Yongbin, S. B. Gelfand, and J. V. Krogmeier, "Noise-constrained least mean squares algorithm," *IEEE Transactions on Signal Processing*, vol. 49, pp. 1961, 2001.
- [GOL98] S. Gollamudi, S. Nagaraj, S. Kapoor, and H. Yih-Fang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Processing Letters*, vol. 5, pp. 111, 1998.
- [GAM05] D. Gamba, "Using FPGAs in Wireless Base Station Designs," in *DSP Magazine*, October ed, 2005, pp. 20-22.
- [POW93] D. G. Powell and A. G. J. Holt, "Reduced sampling rate beamforming technique and its hardware implementation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, pp. 209-215, 1993.

- [NUT02] T. W. Nuteson, J. E. Stocker, J. S. Clark, D. S. Haque, and G. S. Mitchell, "Performance characterization of FPGA techniques for calibration and beamforming in smart antenna applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, pp. 3043-3051, 2002.
- [TAR05] A. Tarighat, E. Grayver, A. Eltawil, J. F. Frigon, G. Poberezhskiy, Z. Hanli, and B. Daneshrad, "A low-power ASIC implementation of 2Mbps antenna-rake combiner for WCDMA with MRC and LMS capabilities," *Proceedings of the IEEE Custom Integrated Circuits Conference*, 18-21 Sept. 2005 Page(s):69 – 72.
- [CES05] T. Cesear and R. Uribe, "Implementing Matrix Inversions in Fixed-Point Hardware," in *DSP Magazine*, October ed, 2005, pp. 32-35.
- [XIL05] Xilinx IP center, Logic Core, Complex multiplier Core v2.1, April 2005, http://www.xilinx.com/bvdocs/ipcenter/data_sheet/comp.pdf
- [HOQ05] Quoc-Thai Ho, Daniel Massicotte, and Adel-Omar Dahmane, "FPGA Implementation of an MUD Based on Cascade Filters for a WCDMA System," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 52919, 12 pages, 2006.
- [GUO06] Yuanbin Guo, Jianzhong(Charlie) Zhang, Dennis McCain, and Joseph R. Cavallaro, "An Efficient Circulant MIMO Equalizer for CDMA Downlink: Algorithm and VLSI Architecture," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 57134, 18 pages, 2006.
- [JEN85] R. Jenkins, "Implementing a matrix-inversion algorithm in a limited-precision adaptive antenna array processor," *Antennas and Propagation Society International Symposium*, Volume 23, Jun 1985 Page(s):289 - 292
- [PAP88] G. M. Papadourakis and H. Andre, "High speed implementation of matrix inversion algorithms in orthogonal systolic architectures", *IEEE Conference Proceedings Southeastcon*, 11-13 April 1988

- Page(s):200 – 204.
- [YLI04] M. Ylinen, A. Burian, and J. Takala, "Direct versus iterative methods for fixed-point implementation of matrix inversion," *International Symposium on Circuits and Systems ISCAS* Volume 3, 23-26 May 2004 Page(s): III - 225-8 Vol.3.
- [ECH05] F. Echman and V. Owall, "A scalable pipelined complex valued matrix inversion architecture" *IEEE International Symposium on Circuits and Systems, ISCAS*, 23-26 May 2005 Page(s):4489 - 4492 Vol. 5.
- [SAL05] P. Salmela, A. Happonen, A. Burian, and J. Takala, "Several approaches to fixed-point implementation of matrix inversion" *International Symposium on Signals, Circuits and System, ISSCS* 2005, Volume 2, 14-15 July 2005 Page(s):497 – 500.
- [KAR05] M. Karkooti, J. R. Cavallaro, and C. Dick, "FPGA Implementation of Matrix Inversion Using QRD-RLS Algorithm", *Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, October 28 - November 1, 2005 Page(s):1625 – 1629.
- [BUR06] A. Burg, S. Haene, D. Perels, P. Luethi, N. Felber, and W. Fichtner, "Algorithm and VLSI architecture for linear MMSE detection in MIMO-OFDM systems" *IEEE International Symposium on Circuits and Systems, ISCAS 2006*, 21-24 May 2006 Page(s):4 pp.
- [SHA98] N. R. Shanbhag, "Algorithms Transformation Techniques for Low-Power Wireless VLSI Systems Design," *International Journal of Wireless Information Networks*, vol. 5, pp. 147-171, 1998.
- [QUI89] P. Quinton. Y.Robert, *Algorithmes et architectures systoliques*: MASSON, 1989.
- [PAR99] K. K. PARHI, *VLSI Digital Signal Processing System*: WILEY, 1999.

- [TAH05] O. Tahernia, "High-Performance DSP-Vision, Leadership, Commitment", *DSP Magazine*, October ed, 2005, pp. 4.
- [MON80] R.A. Monzingo and T.W. Miller, *Introduction to Adaptive Arrays*, New York: Wiley, 1980.

Appendix A

Results with a Lower Complexity Platform

A.1 Platform characteristics

In order to study the performance of the adaptive algorithms, in a wireless communication system, (Transmission, reception, noise ...), we have developed a low complexity platform based on CDMA protocol. The characteristics of the platform can be summarized as follows:

- One transmitting antenna, multiple receiving antennas (SIMO), in default, the number of receiving antennas is equal to 4.
- Variable number of users, in default it is set to 8.
- SINR is varying from zero dB to 16 dB with a step of four.
- The Process gain vector or the OVSF (Orthogonal Variable Spreading Factor), in default it is set to 16.
- Number of frames = 40;
- Number of slots per frame is 15.
- The slot length chip is 256000.

- Time varying channel, the variation rate depends on the speed of the user. The channel change every 15 slots if the speed of the user is less than 5 Km/h, it changes every 7 slots if the speed is between 5 and 20 Km/h, it changes every 4 slots if the speed is between 20 and 60 Km/h, it changes every 2 slots if the speed is between 60 and 90 Km/h and changes every slots if the speed is greater than 90 Km/h.
- Generating long code, at each time the channel changes, the system regenerates the code.
- The percentage of the error in the channel is about 5%.

A.2 Maximum Ratio Combining

For the default parameters of the platform, the result of this algorithm is illustrated in Fig 1. It can be seen the BER decreases as the SINR increases. The BER is about $10^{-3.7}$ for an SINR of 16 dB.

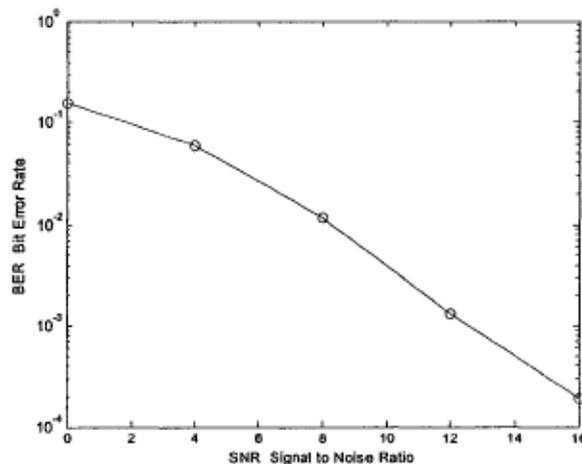


Figure 1: BER vs. SINR for the MRC

A.3 Direct Matrix Inversion

For the default parameters of the platform, the result of this algorithm is illustrated in Fig 2. It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-3.7}$ for an SINR of 16 dB.

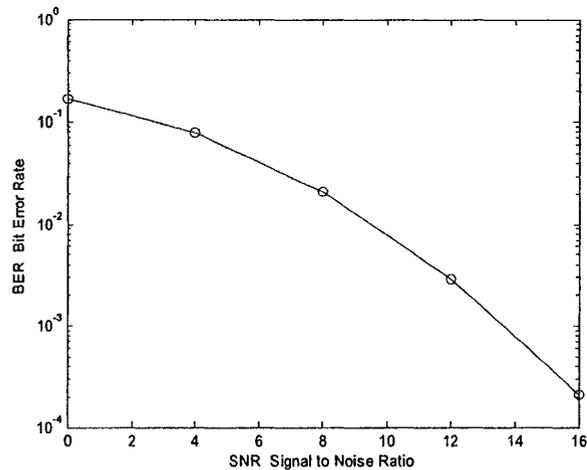


Figure 2: BER vs. SINR for the DMI

A.3 Least Mean Squares

The first scenario of simulations for this algorithm is to display the bit error rate BER versus the SINR for different values of μ (see Fig 3). It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-3.8}$ for an SINR of 16 dB. We have taken many values for the step-size μ , the optimal μ is one that gives the best result at $\text{BER} = 10^{-2}$ and $\text{SINR} = 5\text{dB}$. It can be seen that the optimal μ is 0.05.

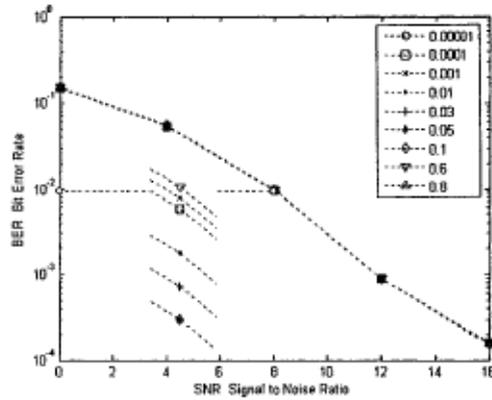


Figure 3: BER vs. SINR for the LMS

The second scenario of simulations for this algorithm is to display the MSE versus the number of iterations for different values of μ (see Fig 4). This is done by fixing the SINR to 10 dB, and the remaining parameters of the platform to their default values. We have taken three values of the step-size μ . It can be seen that as μ increases, more it converges, but on the other hand the MSE increases. Besides as the number of iterations increases the MSE decreases.

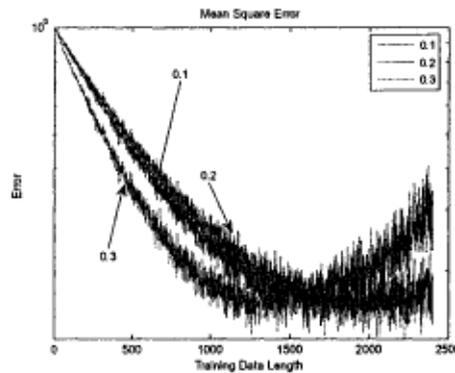


Figure 4: MSE vs number of iterations for the LMS

The third scenario of simulations for this algorithm is to display the BER versus the number of training bits for different values of μ (see Fig 5). This is done by setting the SINR to 10dB and the remaining value of the platform to their default values. For this simulation we have taken two values of the step-size μ . We see that as the number

of iterations increases, the BER decreases ($\sim 10^{-3.5}$).

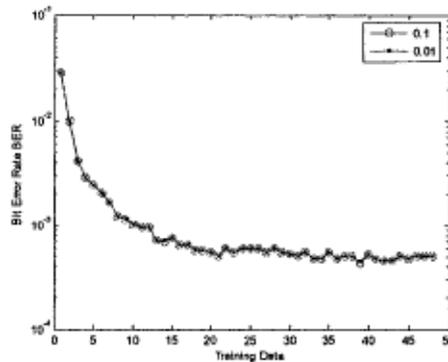


Figure 5: BER vs. number of training data for the LMS

A.3 Normalized Least Mean Squares

The first scenario of simulations for this algorithm is to display the bit error rate BER versus the SINR for different values of μ (see Fig 6). It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-3.8}$ for an SINR of 16 dB. We have taken many values for the step-size $\bar{\mu}$, the optimal μ is one that gives the best result at BER = 10^{-2} and SINR = 5dB. It can be seen that the optimal $\bar{\mu}$ is 0.03.

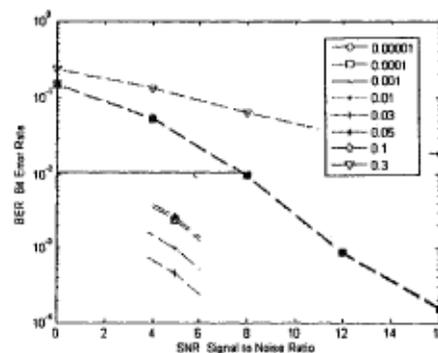


Figure 6: BER vs. SINR for the NLMS

The second scenario of simulations for this algorithm is to display the MSE versus the number of iterations for different values of μ (see Fig 7). This is done by

fixing the SINR to 10 dB, and the remaining parameters of the platform to their default values. We have taken three values of the step-size μ . It can be seen that as $\bar{\mu}$ increases, more it converges, but on the other hand the MSE increases. Besides as the number of iterations increases the MSE decreases.

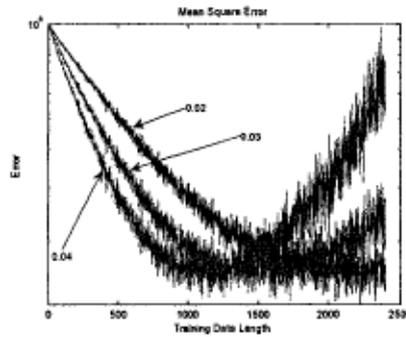


Figure 7: MSE vs. number of iterations for the NLMS

The third scenario of simulations for this algorithm is to display the BER versus the number of training bits for different values of μ (see Fig 8). This is done by setting the SINR to 10dB and the remaining value of the platform to their default values. For this simulation we have taken two values of the step-size $\bar{\mu}$. We see that as the number of iterations increases, the BER decreases ($\sim 10^{-3.5}$).

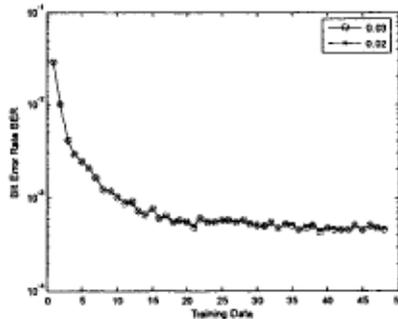


Figure 8: BER vs. number of training data for the LMS

A.4 Noise Constrained Least Mean Squares

The first scenario of simulations for this algorithm is to display the bit error rate

BER versus the SINR for different values of α , β and γ (see Fig 9). It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-3.8}$ for an SINR of 16 dB. We have fixed the α constant to the optimal step-size of LMS, 0.05, and we have taken many values for β and γ , the optimal values for β and γ are the ones that give the best result at BER = 10^{-2} and SINR = 5dB. It can be seen that the optimal β and γ are 0.3 and 0.5 respectively.

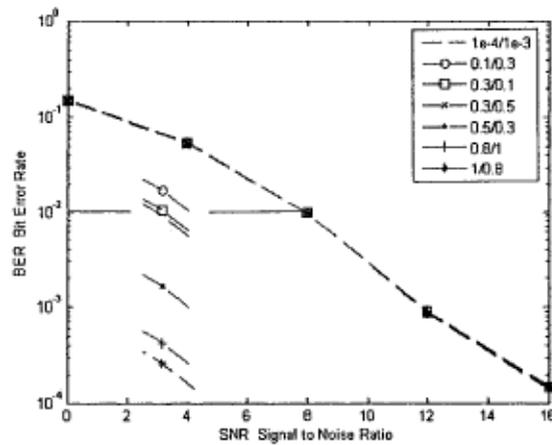


Figure 9: BER vs. SINR for the NCLMS

The second scenario of simulations for this algorithm is to display the MSE versus the number of iterations for different values of α , β and γ (see Fig 10). This is done by fixing the SINR to 10 dB, and the remaining parameters of the platform to their default values. We have taken sixth different values of the constants β and γ . α is set to 0.05. It can be seen that as β and γ increases, it ensure more convergence, but on the other hand the MSE increases 4 and 4.5 are the best values for β and γ respectively. Besides as the number of iterations increases the MSE decreases.

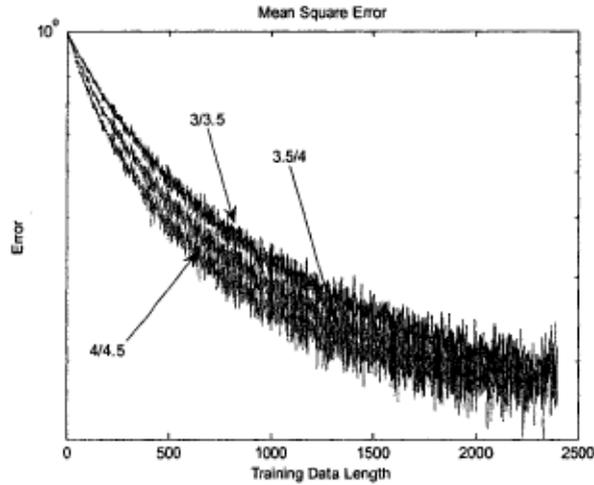


Figure 10: MSE vs. number of iterations for the NCLMS

The third scenario of simulations for this algorithm is to display the BER versus the number of training bits for different values of α , β and γ (see Fig 11). This is done by setting the SINR to 10dB and the remaining value of the platform to their default values. For this simulation we have taken two values for β and γ each. We see that as the number of iterations increases, the BER decreases ($\sim 10^{-3.6}$).

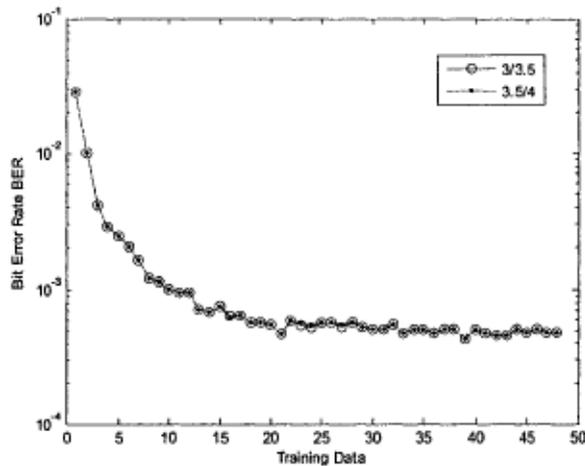


Figure 11: BER vs. number of iterations for the LMS

A.5 Recursive Least Squares

The first scenario of simulations for this algorithm is to display the bit error rate BER versus the SINR for different values of the constant δ (see Fig 12). It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-4.8}$ for an SINR of 16 dB. We have fixed the forgetting factor to 0.98, and we have taken many values for the constant δ . The optimal δ is the one that gives the best result at BER = 10^{-2} and SINR = 5dB. It can be seen that the optimal δ is $1e^{-2}$.

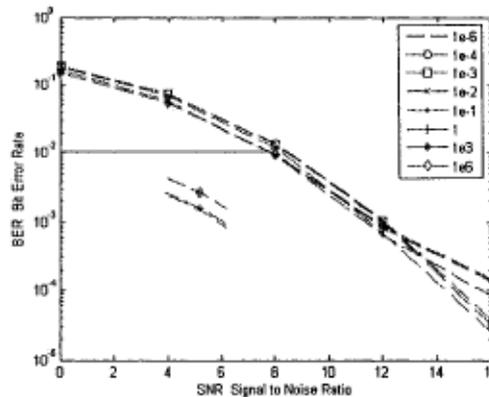


Figure 12: BER vs. SINR for the RLS

The second scenario of simulations for this algorithm is to display the MSE versus the number of iterations for different values of the constant δ (see Fig 13). This is done by fixing the SINR to 10 dB, and the remaining parameters of the platform to their default values. We have taken four values of the constant δ . It can be seen that when δ is set to $1e^{-2}$, it gives the best performance and at the same time the MSE remain constant. Besides as the number of iterations increases the MSE decreases.

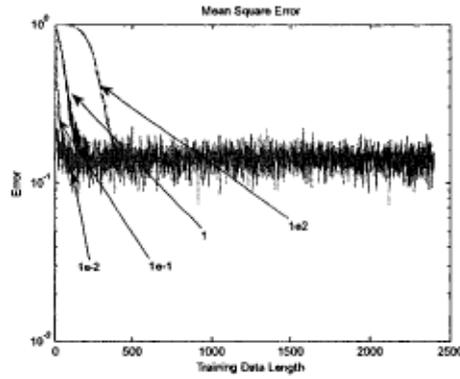


Figure 13: MSE vs. number of iterations for the RLS

The third scenario of simulations for this algorithm is to display the BER versus the number of training bits for different values of the constant δ (see Fig 14). This is done by setting the SINR to 10dB and the remaining value of the platform to their default values. For this simulation we have taken two values for the constant δ . We see that for δ equal 1e-1 the BER is much more less than it for $\delta = 1e-2$. Thus, as the number of iterations increases, the BER decreases ($\sim 10^{-3.8}$).

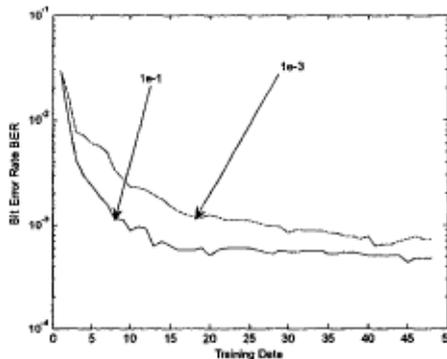


Figure 14: BER vs. number of training data for the RLS

A.6 Set Membership Identification

The first scenario of simulations for this algorithm is to display the bit error rate BER versus the SINR for different values of the constant γ (see Fig 15). It can be seen the BER diminishes as the SINR increases. The BER is about $10^{-4.5}$ for an SINR of 16

dB. We have taken many values for the constant γ between 0 and 1, the optimal γ is the one that gives the best result at $\text{BER} = 10^{-2}$ and $\text{SINR} = 5\text{dB}$. It can be seen that the optimal γ is 0.8.

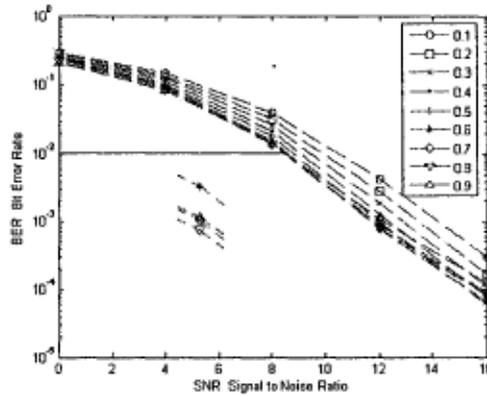


Figure 15: BER vs. SINR for the SMI

The second scenario of simulations for this algorithm is to display the MSE versus the number of iterations for different values of the constant γ (see Fig 16). This is done by fixing the SINR to 10 dB, and the remaining parameters of the platform to their default values. We have taken four values of the constant δ . It can be seen that when as γ decreases, the MSE decreases too, as well as the number of iterations increases the MSE decreases.

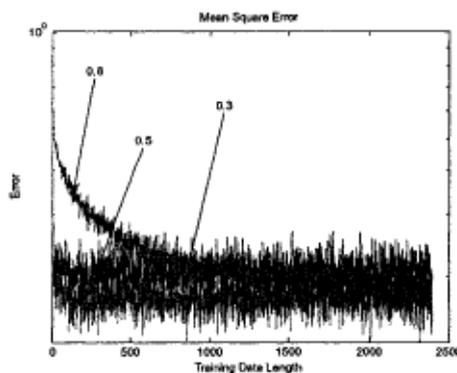


Figure 16: MSE vs. number of iterations for the LMS

The third scenario of simulations for this algorithm is to display the BER versus the number of training bits for different values of the constant γ (see Fig 17). This is done by setting the SINR to 10dB and the remaining value of the platform to their default values. For this simulation we have taken four different values for the constant γ . We see that for as γ increases the BER decreases as well as the number of iterations increases, the BER decreases too ($\sim 10^{-3.6}$).

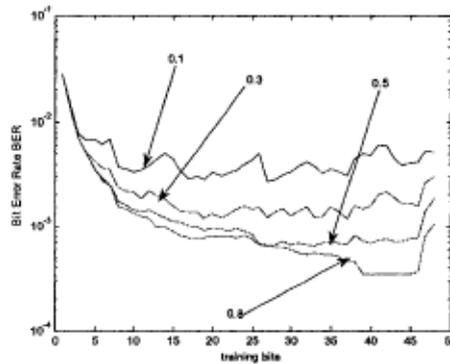


Figure 17: BER vs. number of training data for the LMS

A.7 Comparison

A.7.1 First scenario

The first performance comparison is for the default parameters of the platform, and taking the best parameters in each algorithm, simulated separately. The result is illustrated in Figure 18. We look for the algorithm that cut first or passes in a closely by the point (SNR=5dB, BER=10⁻²), for this simulation the best algorithm is the RLS.

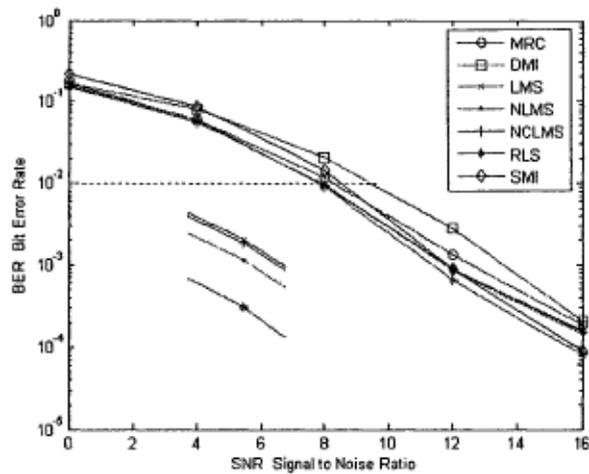


Figure 18: Default parameters

A.7.2 Second scenario

The second performance comparison is in terms of the BER taking the default parameters of the platform, just changing the speed of the user to 100 Km/h, and taking the best parameters in each algorithm, simulated separately. The result is illustrated in Figure 19. We look for the algorithm that cut first or passes in a closely by the point (SNR=5dB, BER=10⁻²), for this simulation the best algorithm is the RLS.

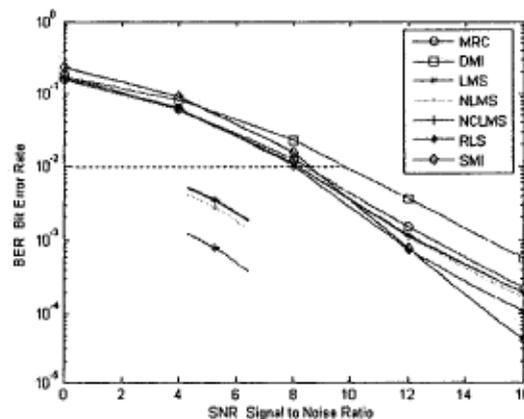


Figure 19: Speed = 100 Km/h

A.7.3 Third scenario

The third performance comparison is in terms of the BER taking the default parameters of the platform, just changing the number of users to 16, and taking the best parameters in each algorithm, simulated separately. The result is illustrated in Figure 20. We look for the algorithm that cut first or passes in a closely by the point (SNR=5dB, BER=10⁻²), for this simulation the best algorithm is the RLS.

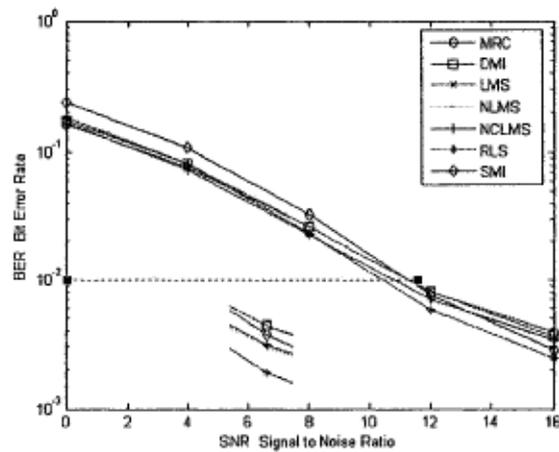


Figure 20: Number of users = 16

A.7.4 Fourth scenario

The fourth performance comparison is in terms of the BER taking the default parameters of the platform, just changing the number of users to 50, and the process gain to 64, and taking the best parameters in each algorithm, simulated separately. The result is illustrated in Fig 21. We look for the algorithm that cut first or passes in a closely by the point (SNR=5dB, BER=10⁻²), for this simulation the best algorithm is the DMI.

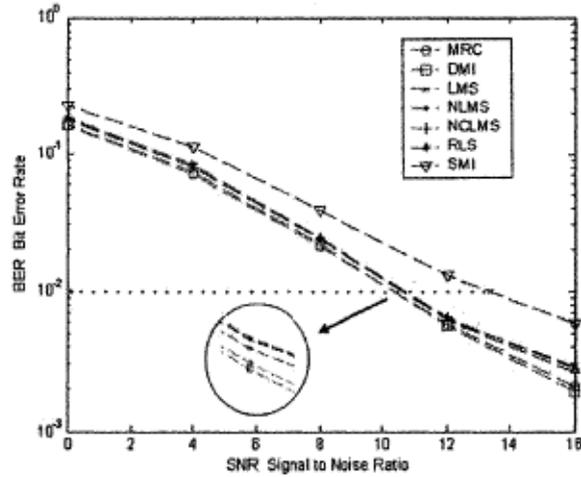


Figure 21: Number of users = 50, OVFS = 64

A.7.5 Fifth scenario

The fifth performance comparison is in terms of the MSE taking the default parameters of the platform, and taking the best parameters in each algorithm (giving the best performance in MSE simulations). The result is illustrated in Fig 22. We can see that the best algorithm for this simulation is the RLS, the SMI comes second.

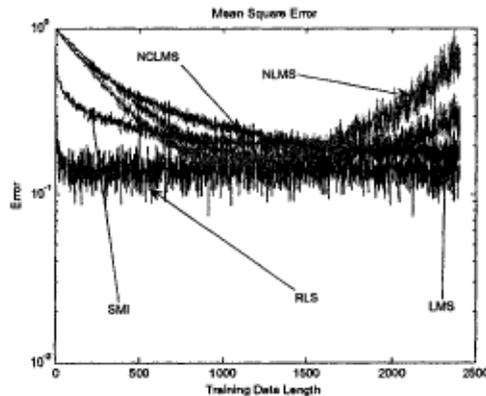


Figure 22: MSE vs number of iterations, default parameters

A.7.6 Sixth scenario

The sixth and the last performance comparison is in terms of the BER vs. the

number of training data, taking the default parameters of the platform, and taking the best parameters in each algorithm (giving the best performance in BER vs. the number of training data). The result is illustrated in Fig 23. We can see that the best algorithm for this simulation is the RLS, and the NLMS comes second.

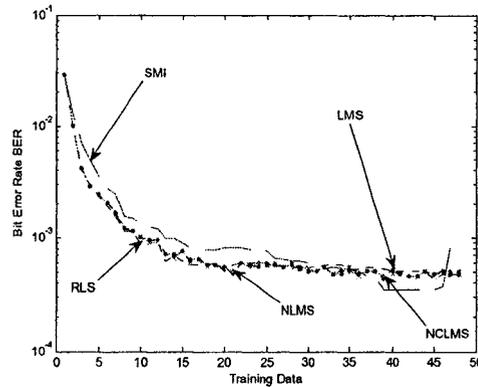


Figure 23: BER vs. number of training data, default parameters.

A.7 Conclusion

The DMI algorithm converges more rapidly than the LMS but it is more computationally complex. It requires a reference signal, and a matrix inversion.

The LMS algorithm requires knowledge of the desired signal. Considering the performance of the algorithm, **it always converges**. Two significant features about this algorithm are its **simplicity of implementation** and its **model-independent** and therefore his **robust performance**. It does not require measurements of the pertinent correlation functions, nor does it require a matrix inversion. Indeed, it is the simplicity of this algorithm that has made it the standard against which other adaptive filtering algorithms are benchmarked. In addition to that, we found that two factors affect its convergence behavior, the adaptation step size μ , and the eigenvalues of the correlation matrix R of the tap-input vector. So when a small number is assigned to μ , the

adaptation is slow which is equivalent to the LMS Algorithm having a long “memory”, correspondingly, the excess mean-squared error after adaptation is small, on the average because of the large amount of data used by the algorithm to estimate the gradient vector. On the other hand, when μ is large, the adaptation is relatively fast, but at the expense of an increase in the average excess mean-squared error after adaptation, so less data enter the estimation, hence a degraded estimation error performance. The speed of convergence of the mean-squared error is affected by a spread of the eigenvalues of \mathbf{R} to a lesser extent than the convergence of $\hat{E}[\hat{W}(n)]$. In addition to that this algorithm converges slowly if the eigenvector spread of R_{xx} is large. His major drawback is its relatively **slow rate of convergence**, and moreover it requires a reference signal (desired response). This is can be done in a digital system by transmitting a training sequence that is known to the receiver, or using the spreading code in the case.

Appendix B

Résumé en Français

B.1 Introduction

Ces dernières années ont été le témoin d'une évolution sans précédent du marché de télécommunication mobile, au niveau du portable, de la station de base (*BS*) et au niveau des services fournis aux consommateurs (courriel électronique, messages textes, multimédia, vidéo, conférences, etc.). Pour faire face à l'augmentation prévisible du nombre d'utilisateurs d'une part et à l'augmentation des débits de transmission d'autre part, les futurs réseaux de communications devront mettre en oeuvre des techniques de plus en plus évoluées et sophistiquées. Malheureusement, ces nouvelles techniques sont directement reliées et proportionnelles à l'élévation du coût du matériel ou de l'infrastructure du réseau utilisé, ce qui apporte une augmentation élevée du coût pour l'utilisateur et le consommateur.

Plusieurs approches à différents niveaux de coûts sont estimées, et l'une d'entre elles consiste à combiner les signaux reçus par les éléments d'une antenne réseau (plusieurs antennes connectées entre eux). Cette approche, de traitement de l'information, fait références aux systèmes utilisant des antennes intelligentes. Un des

principaux avantages de ces systèmes réside dans l'augmentation potentielle du nombre d'utilisateurs dans le réseau cellulaire d'une part, et l'accroissement de l'éventail des services offerts par le système cellulaire d'une autre part.

L'intérêt de ces systèmes est leur capacité à réagir automatiquement à un environnement complexe dont l'interférence est connue à priori. Ils ont le potentiel de réduire les interférences inhérentes aux multi-trajets, de valoriser le rapport signal à bruit (*SINR*), et d'introduire la réutilisation de fréquences dans un environnement confiné. De plus, ils permettent de réduire les niveaux des lobes secondaires existants dans la direction de l'interférence, tout en maintenant le lobe principal dans une direction utile. Cependant, en faisant circuler l'énergie directement entre la BS et le portable, une réduction des bruits ambiants est produite. Par conséquent, les interférences provenant d'autres usagers et obstacles sont éliminées. L'augmentation du nombre d'usagers et l'amélioration de la qualité du service offert représentent un atout pour les futurs systèmes sans fils de la troisième et quatrième génération.

Ces systèmes reposent sur des antennes réseau, des dispositifs pour calculer les angles d'arrivées et des outils numériques de synthèse. Ces derniers attribuent des poids aux éléments de l'antenne réseau afin d'optimiser le signal de sortie. Les méthodes d'optimisation utilisent des techniques de contrôle prédéfinies pour la formation des voies et l'annulation d'interférents.

Une antenne réseau adaptative peut être définie comme un réseau capable de modifier son diagramme de rayonnement. Cette modification est réalisée grâce à un algorithme performant implémenté et apte à répondre aux spécifications désirées. Plus spécifiquement, en considérant un système de communication à temps réel.

L'évolution du domaine de la microélectronique et les microsystèmes est caractérisée par la réduction des dimensions des circuits intégrés. Au cours des quatre dernières décennies, l'industrie des semiconducteurs n'a pas cessé d'améliorer ses produits grâce à l'augmentation de la densité d'intégration, de la vitesse de fonctionnement et de la diminution du coût de fabrication. En effet, le domaine des transmissions numériques connaît un essor considérable lié à l'évolution des moyens et des méthodes de conception des systèmes numériques (CAO, VHDL, FPGA, etc.). Les circuits programmables permettent de mettre au point des systèmes électroniques flexibles et des prototypes rapidement.

B.2 Problématique

Le canal de transmission radio-mobile est un des moyens de communication les plus variables et les plus incontrôlables. La modélisation de ce genre de canaux de communications est souvent très sophistiquée dues aux nombreux problèmes et paramètres que ce type de canal pose.

En parcourant un trajet entre l'émetteur et le récepteur, les ondes radioélectriques sont sujettes aux nombreuses irrégularités de morphologie, de caractéristiques électromagnétiques, de température, d'humidité du milieu traversé qui ont un effet de dégradation sur la qualité du signal. Pour cela, les transmissions hertziennes ont comme propriété de fluctuer en temps et en espace, souvent avec des variations très importantes dues à plusieurs phénomènes de propagation. Alors que le signal envoyé subira une dégradation ce qui permet de modifier les données émises de plusieurs manières, dont le besoin des systèmes sophistiqués à la réception pour reconstituer la « mésurande ». En d'autres termes, récupérer les données émises avec un taux d'erreurs *BER* très faible;

dont le besoin d'une technologie efficace comme les antennes intelligentes (Voir figure 1).

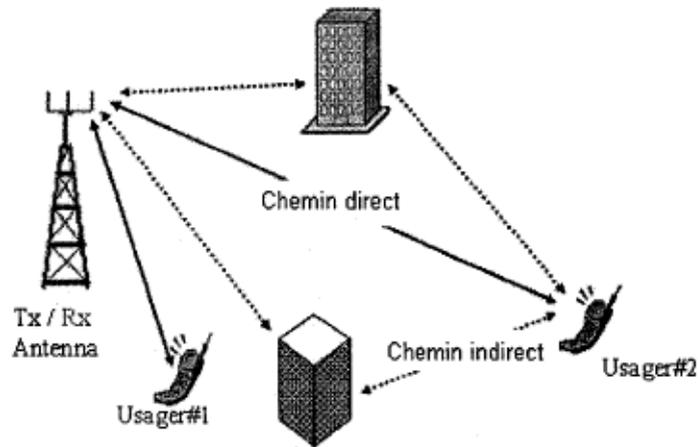


Figure 1: Exemple d'un environnement d'un réseau sans fil.

En bref, deux raisons principales pour l'introduction des *antennes intelligentes* : premièrement, la possibilité d'accroître la capacité du réseau cellulaire sans avoir à accroître le spectre lui même. Deuxièmement, résoudre les interférences imposées par le canal de transmission : les trajets multiples (*Multipath Fading*), l'effet Doppler pour le décalage de fréquence (*Doppler Shift*), interférences inter-symboles *ISI* (*Inter-Symbol Interference*), interférence co-canal (*CoChannel Interference*). L'une des grandes difficultés de recherche est d'atteindre des gains de performances adéquats tout en limitant l'augmentation de la complexité d'implémentations pratiques.

Le processus des antennes intelligentes appelé *Beamforming* est purement mathématique et joue un rôle vital dans les systèmes des antennes intelligentes. Deux types de *Beamforming* se présentent : Le *Beamforming* adaptative (*TBA*) et le *Beamforming* répartiteurs de faisceaux (*ABA*). Les études dans la littérature ont démontré l'avantage du premier par rapport au second, dans la poursuite des variations du canal et dans les environnements bruités.

Avec un modèle du signal, on remarque que la sortie du système *Beamforming* est fonction des poids complexes, alors la problématique réside dans:

- (i) Le choix d'un algorithme favorable qui calcule le poids optimal, afin que le système change la direction du faisceau (*Beam*) vers l'utilisateur d'intérêt en gardant la contribution des usagers interférents minimale.
- (ii) La proposition de l'architecture *VLSI* pour l'algorithme choisi, ceci devrait minimiser le coût tout en gardant les mêmes performances.

B.3 Objectifs de la recherche

Étant donné que la problématique se figure dans deux aspects physiquement différents, mais en terme d'application ces deux aspects sont très dépendants entre eux.

Le premier problème est un aspect logiciel (Software) et mathématique appelé «Traitement de signal», il comprend les méthodes mathématiques appliquées aux antennes intelligentes. Le deuxième problème est un aspect matériel (Hardware) appelé «Implémentation», il comprend les circuits intégrés (*DSP, FPGA, ASIC, ...*). Dans notre cas on utilise les composantes programmables *FPGA*.

L'objectif de notre étude est de trouver pour chaque aspect la meilleure combinaison dans le but de donner une solution optimale qui lui correspond.

Plus spécifiquement, si nous considérant le premier problème évoqué plus haut, notre objectif est d'insérer des méthodes d'antennes adaptatives dans une plateforme *DS-CDMA* multi antennes, et d'évaluer ces algorithmes mathématiques, c'est-à-dire d'examiner, en termes de performances (taux d'erreurs en fonction, de nombre d'utilisateurs, de nombre d'antennes et du trafic) et de complexité (nombre d'opérations arithmétiques par cycle), les algorithmes existants afin de faire ressortir ceux qui

répondent le mieux aux exigences de l'industrie de communication sans fil. Une prédiction du résultat attendu, la possibilité d'orienter le faisceau (*beam*) d'une antenne en appliquant un algorithme puissant et très performant, permet d'effectuer une vaste couverture et de suivre les déplacements d'un utilisateur (poursuite de l'utilisateur) à l'intérieur d'une même cellule en minimisant le bruit et les interférences, sans avoir recours à un mécanisme de rotation (mécanisme physique), ajouté à la possibilité d'obtenir un où plusieurs faisceaux ayant un gain important et une ouverture à mi-puissance étroite.

D'un autre côté, considérant le deuxième problème mentionné précédemment, notre objectif est de proposer des architectures *VLSI* pour les algorithmes choisis. Par la suite, d'implémenter ces algorithmes sur les composantes programmables *FPGA*, afin d'appliquer une analyse détaillée de quantification dans le but de déterminer la précision de l'algorithme, et d'estimer les ressources matérielles (nombre des additionneurs, multiplieurs, registres, soustracteurs, cellules logiques, tampons, etc.). Cette dernière s'appelle synthèse. Pour compléter l'objectif de recherche, une étude sur le nombre d'utilisateurs maximale qu'on peut atteindre dans la composante programmable sera appliquée.

B.4 Méthodologie préconisée

Pour simuler les algorithmes choisis, une plateforme *DS-CDMA* a été développée. Cette dernière est basée sur le système SIMO. Ses caractéristiques sont les suivantes :

- Les éléments du réseau linéaire d'antennes (*ULA*) sont placés d'une façon qu'on a une distance de $\lambda/2$ entre chacun. Les angles d'arrivées sont distribués uniformément dans un intervalle de $[0 \ \theta_{\max}]$ où $\theta_{\max} = 120^\circ$.

- Les données du pilot et du trafic utilisent deux codes orthogonaux. On considère un gain de 8, les codes utilisés pour le pilot et le trafic sont $\{1, -1, 1, -1, 1, -1, 1, -1\}$ et $\{1, 1, 1, 1, 1, 1, 1, 1\}$ respectivement.
- Le canal est modélisé par trois chemins, variant en fonction du temps (Rayleigh fading). Les délais de ces chemins sont $[0, 1.1, 3.19]\mu s$ ayant des puissances moyennes de $[0, -3, -9]dB$ respectivement. Le diamètre de la cellule est de 5 Km. Le rapport signal sur bruit est de 10 dB. La fréquence de la porteuse est de 2 GHz, « chip rate » de 3.84Mcps, la vitesse des usagers est 60 Km/h. Par défaut, quatre antennes sont utilisées. Le nombre des usagers est de cinq et le rapport de la puissance sur trafic est de -6 dB.
- La transmission est pactisé de la manière suivante : la taille de « slot » normalisée est de 2560 chip. Chaque trame contient 15 slot (38400 chip). On considère que le nombre des échantillons disponibles est de deux slots ce qui est équivalent à $2 \times 2560 / 8 = 2 \times 320$ données.

Les algorithmes disponibles dans la littérature sont les suivants:

- *Maximum Ratio Combining MRC*
- *Direct Matrix Inversion DMI*
- *Eigen Value Decomposition EVD*
- *Chip- Eigen Value Decomposition Chip-EVD*
- *Noise Constrained Least Mean Squares NC-LMS*
- *Recursive Least Squares RLS*
- *Two-Stage RLS.*
- *Etc.*

Les algorithmes implémentés sont : *MRC*, *DMI* et *NC-LMS*. Durant l'implémentation deux méthodes ont été utilisées. La première est reconnue par prototypage rapide. Cette méthode est descendante de la programmation orienté objet (OO). En effet, afin de simuler l'architecture conçue, il s'agit d'utiliser les blocks de Xilinx (additionneurs, multiplieurs, soustracteurs, registres, etc.) dans l'environnement Simulink et les connecter entre eux. Après simulation, il était possible d'estimer les ressources matérielles sur les composantes programmables *FPGA*. Cette technique permet de sauver le temps de conception, ou bien le « Time To Market ». Elle permet de générer automatiquement le code *VHDL* pour l'architecture proposée.

La deuxième méthode consiste à utiliser le codage classique du *VHDL* sous l'environnement du logiciel *ISE* de Xilinx. Cette méthode permet de visualiser les détails d'implémentation dans chaque phase de conception (placement et routage, timing, etc.). Cela permettre aux développeurs bien encadrer sa conception.

L'utilisation des différentes méthodes d'implémentation permet d'avoir un point de vue étendu et approfondi sur les résultats obtenus afin de bien conclure de façon objective.

B.5 Propositions

Tel que mentionné plus haut, trois algorithmes sont choisis pour l'implémentation. Le *MRC*, le choix de cet algorithme est fondé sur le fait que le *MRC* représente l'algorithme de base pour les antennes intelligentes. De la même façon, le *DMI* représente un défi pour la plupart des ingénieurs à cause de l'inverse de la matrice et son coût de l'implémentation. Finalement, l'algorithme *NC-LMS* se trouve à un niveau intermédiaire de complexité entre les deux précédents. Le *MRC* et le *NC-LMS* sont implémentes en utilisant la méthode de prototypage rapide (première méthode

proposée dans ce document). Le *DMI* est implémenté en utilisant la méthode de codage manuel (deuxième méthode).

La figure 2 montre les architectures proposées de *MRC* et de *NC-LMS*. Ces algorithmes sont montés d'une manière direct ou « straight-forward manner ». Le but est de faire une comparaison de complexité entre les deux, et d'appliquer le *NC-LMS* au *Beamforming*. À signaler que l'algorithme *NC-LMS* n'a pas été appliqué aux antennes intelligentes. Normalement, le *NC-LMS* est hors domaine d'application.

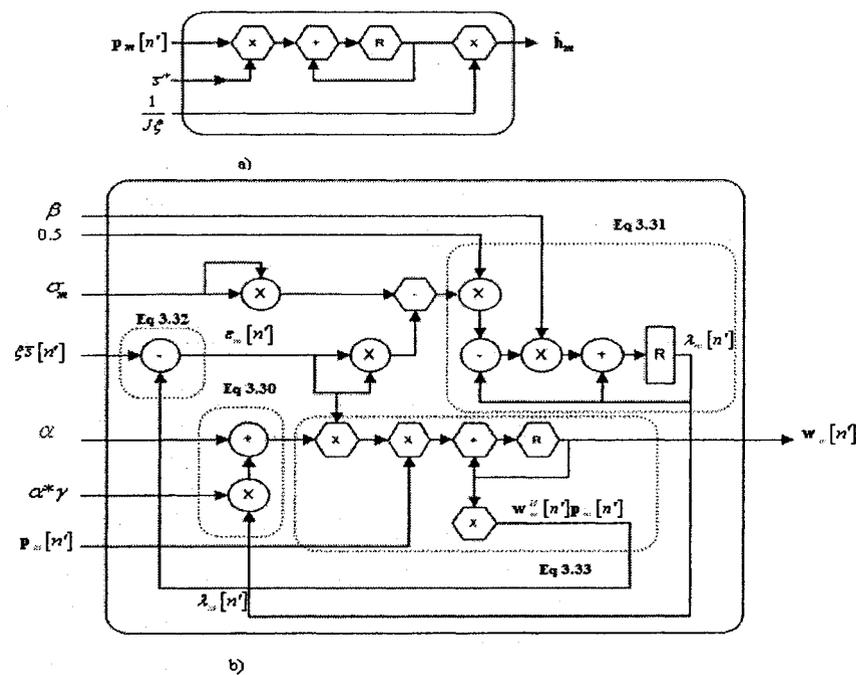


Figure 2: Implémentation proposée du *MRC* (a) et *NC-LMS* (b).

L'architecture générale proposée pour le *DMI* est visualisée par la figure 3. Cette architecture est composée de plusieurs blocks « pipelinés » et « parallélisés ». Dans l'architecture générale, des sous-architectures parallèles et systoliques ont été utilisées.

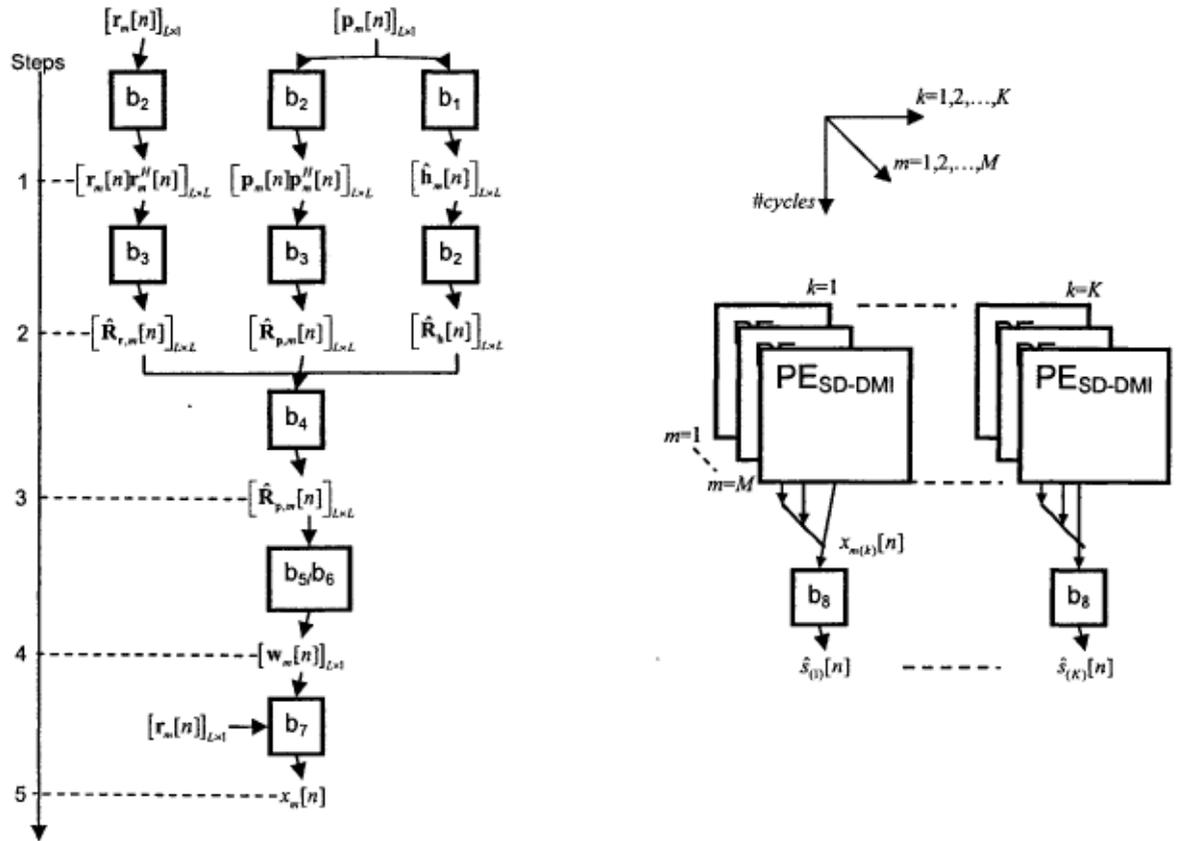


Figure 3: L'architecture générale proposée pour le SD-DMI.

B.6 Résultats

Dans les sections précédentes nous avons proposées des architectures pour résoudre la problématique évoquée ultérieurement. Par la suite, les résultats de l'implémentation de ces architectures seront discutés.

Deux types de résultats sont présentés : premièrement, études comparatives entre les différents algorithmes utilisés. Deuxièmement, la complexité des ces algorithmes et la quantification et l'implémentation seront évoquées.

B.6.1 Études comparatives

La première étude comparative est en fonction du rapport puissance sur trafic. D'après les résultats (figure 4), le *DMI* montre une dégradation réduite dans les régions du faible *PTR*. Par contre, cet algorithme donne une très haute performance dans les régions où le *PTR* est élevé.

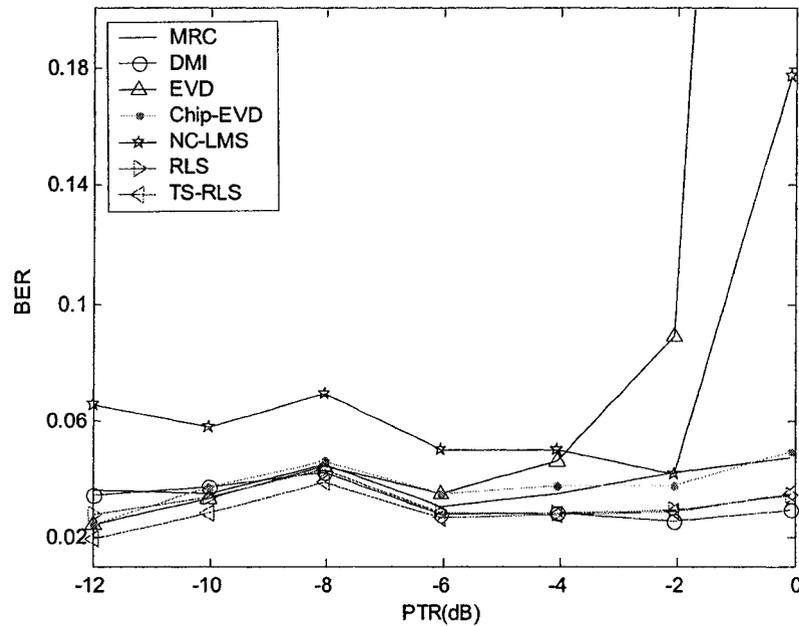


Figure 4: L'effet du changement du PTR.

La deuxième étude comparative est en fonction du nombre d'antennes (éléments) dans la station de base (réseau linéaire). Le *PTR* et le nombre d'utilisateurs sont fixés à -6 dB et à 5 respectivement. Le résultat obtenu montre une meilleure performance (convergence) dans l'algorithme *TS-RLS*. De la même façon le *DMI* donne une réponse similaire au *TS-RLS*. Le détail de cette constatation se trouve dans la figure suivante.

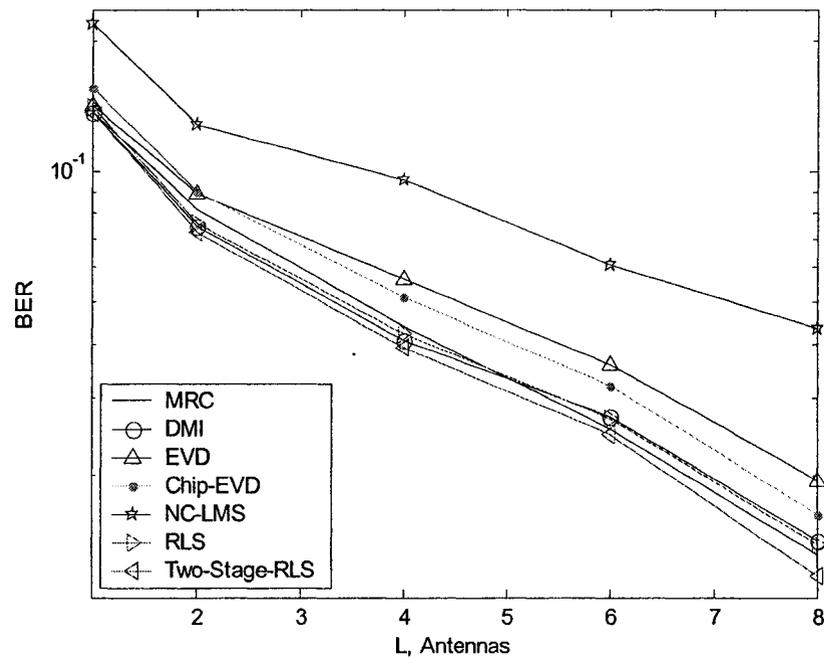


Figure 5: L'effet de changement du nombre d'antennes.

Pour avoir une comparaison scientifique objective, la troisième étude comparative est basée sur le nombre d'utilisateurs (figure 6). Cependant, le PTR et le nombre d'antennes sont fixés à -6 dB et 4 respectivement. Comparativement au deuxième essai, la performance dans le $TS-RLS$ est la même à une différence que les essais représentent deux environnements différents. Le DMI et le MRC ont une performance équivalente.

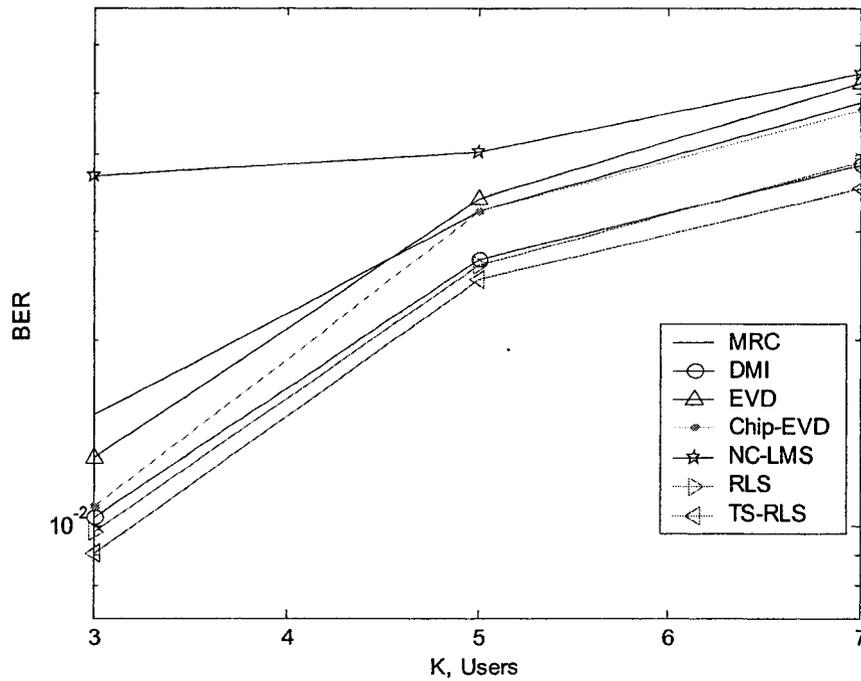


Figure 6: L'effet du changement du nombre d'utilisateurs.

En effet, selon les trois études précédentes en simulation, le *TS-RLS* donne une meilleure performance. Par contre, ce dernier est beaucoup plus complexe comparativement à l'algorithme *DMI*.

B.6.2 Études de complexités, de quantification et d'implémentation

À travers de cette section, nous allons réaliser l'étude de la complexité des algorithmes implémentés. Les algorithmes choisis pour l'étude de la complexité sont: *NC-LMS*, *MRC* et *DMI*.

Cette étude montre que le *NC-LMS* a une complexité de 2.5 plus que le *MRC*. De cette manière, le *NC-LMS* demande des ressources matérielles de 2.5 plus que le *MRC*. On a une latence (délai) de 15 et de 10 cycles pour le *NC-LMS* et le *MRC*

respectivement. Pour exécuter une opération, le MRC demande une période minimale de 10 ns et une fréquence maximale de 95 MHz. Similairement, le NC-LMS prend une période minimale de 30 ns et une fréquence maximale de 33 MHz. D'après les résultats (Tableau 1), on remarque qu'afin d'implémenter le NC-LMS avec 4 antennes et 5 usagers, nous devons fournir d'un FPGA qui est dispendieux (Virtex XC4VSX55).

Tableau 1: Pourcentage des « Slices » et des multiplieurs pour 4 antennes.

		<i>Maximum Ratio Combining MRC</i>			
		<i>32, 16¹</i>	<i>24, 16</i>	<i>18, 10</i>	<i>16, 12</i>
VP30	% S	29.4	22.1	10.1	9.0
	% M	N/A	N/A	47	47
VP100	% S	9.1	6.8	3.1	2.8
	% M	57.6	57.6	14.4	14.4
VSX55	% S	16.4	12.3	5.7	5.0
	% M	50	50	12.5	12.5
		<i>Noise Constrained Least Mean Squares NC-LMS</i>			
VP30	% S	73.9	58.9	34.7	26.0
	% M	N/A	N/A	88.2	88.2
VP100	% S	22.9	18.3	10.8	8.1
	% M	N/A	N/A	27	27
VSX55	% S	41.16	32.8	19.3	14.5
	% M	93.8	93.8	23.4	23.4

D'après l'étude de quantification, on observe que le MRC est plus sensible que le NC-LMS à cause de sa nature adaptative. Par contre, MRC est de 2.5 moins coûteux que l'autre.

Durant l'implémentation de l'algorithme DMI, l'inverse de la matrice a été résolu par la méthode itérative de descente du gradient. Nous avons remarqué que cette méthode n'est pas coûteuse, de plus avec le « re-design » plusieurs caractéristiques du DMI ont été découvertes. La résolution de l'inverse de la matrice par la méthode proposée, a beaucoup d'avantages surtout dans les environnements bruités et dans la poursuite des variations du canal mobile.

¹ X = (a, b), ou « a » représente la partie fractionnelle et « b » représente la partie décimale.

L'étude de quantification pour le DMI montre que 14 bits sont suffisants pour obtenir des performances adéquates. Afin d'avoir plus de précision, nous avons opté à une implémentation sur 16 bits.

Il a été démontré que l'architecture est bien parallélisée et pipelinée, avec une latence totale de : $T_{Latency} = 6L + 16$, où L est le nombre des antennes. Alors pour $L = 4$ la latence obtenue est de 40 cycles pour l'architecture total.

Finalement, une étude dans le but de maximiser le nombre d'utilisateurs sur la composante programmable *FPGA* a été réalisée. Pour se faire, plusieurs boucles d'optimisations (basant sur le nombre des « slices » et des multiplieurs) ont été adoptées. De cette manière, plusieurs blocks contiennent un grand nombre de multiplieurs ont été insérés dans l'architecture. Une méthode d'optimisation est de synthétiser les multiplieurs en *LUT*. Cette méthode réduit le nombre de multiplieur d'une façon évidente.

Le meilleur compromis nous permet d'atteindre un maximum de 90 usagers sur un VP 50 (Virtex II Pro) qui n'est pas assez coûteux.

Le résultat de l'optimisation donne une constatation claire, cette constatation stipule qu'on peut atteindre 90 usagers, en utilisant un algorithme complexe tel le *DMI* avec une fréquence (débit) de 2 *MHz*. Selon le tableau suivant le *DMI* « re-design » est une bonne solution pour les antennes intelligentes.

Tableau 2: Nombre maximal d'utilisateurs atteint dans une composante programmable pour $L=4$, $J=10$ et $M=3$.

MAX. NUMBER OF USERS	VIRTEX 2				VIRTEX 2 PRO			
	XC2V80	XC2V1000	XC2V3000	XC2V8000	XC2VP2	XC2VP4	XC2VP30	XC2VP50
K^{MAX}	2	15	37	65	4	10	53	90

B.7 Conclusion

À travers de ce travail, une recherche bibliographie a été élaborée. Dans cette dernière une série des algorithmes ont été ciblés dans le domaine des antennes intelligentes. La suite de cette conclusion se concentre sur deux aspects principaux visés par cette recherche: le premier consiste à montrer les accomplissements obtenus. Le deuxième stipule à donner le travaux futures afin d'avoir une continuité de ce projet.

Accomplissement : deux contributions originales on été élaborées durant le développement de cette recherche :

- 1) l'évaluation de plusieurs algorithmes appliqués aux antennes intelligentes.
- 2) l'implémentation de trois algorithmes trouvés dans la littérature (*MRC*, *NC-LMS*, *DMI*) sur les composantes programmables *FPGA*.

Ces contributions ont des points forts et originalités qui caractérisent le travail effectué.

Ces derniers sont les suivants:

- *La plateforme DS-CDMA assez développé*: La plateforme *DS-CDMA* développée comporte des caractéristiques et des conditions de simulations très proches de l'environnement de communication réelle, ce qui a aboutit à des vraies performances des algorithmes simulés.
- *Implémentation de NC-LMS* : L'intégration de l'algorithme *NC-LMS* aux antennes intelligentes. À noter que, cet algorithme n'était pas utilisé dans ce domaine.
- *Utilisation du prototypage rapide* : L'implémentation des deux algorithmes *MRC* et *NC-LMS* en utilisant la méthode de prototypage rapide. Cette méthode a permis de faire une étude approfondie sur l'outil et sur les différentes astuces d'implémentation.

- *Ingénierie inverse de l'algorithme DMI* : Le « re-design » de l'algorithme *DMI*, pour la résolution de l'inverse de la matrice, a donné beaucoup d'avantages dans la poursuite des usagers dans les environnements bruités. Ce re-design de l'algorithme a facilité le parallélisme et le pipeline pour augmenter le débit.
- *Utilisation du codage manuel* : L'implémentation du *DMI* en utilisant l'outil *ISE*, a permis de bien comprendre les différentes phases de l'intégration.
- *Estimation du nombre d'usagers* : Selon la recherche bibliographique, dans le domaine de l'antenne d'intelligent, l'estimation du nombre d'usagers n'était pas l'intérêt de chercheurs. Un point fort et original de cette recherche est l'estimation du nombre d'usagers dans la composante programmable en vue des systèmes d'antennes intelligentes, afin d'avoir un compromis entre la composante programmable et l'algorithme implémenté.

Travaux futurs : cette section donne un aperçu global sur des possibilités d'une extension du projet élaboré dans cette maîtrise.

- *Développement des algorithmes plus performants que le DMI et le TS-RLS (BER <0.02).*
- *Développement des architectures à faible consommation de puissance et de complexité, pour les algorithmes développés. Cette implémentation jouera le rôle d'un processeur « core » dans le circuit intégré.*
- *Implémentation des architectures pour les algorithmes développées au niveau les FPGAs et les ASICs.*
- *Développement d'un SOC (System On Chip) basant sur les systèmes des antennes intelligentes.*

B.8 Publications

- Elie H. Sarraf, Messaoud Ahmed-Ouameur, and Daniel Massicotte “FPGA Implementation of Beamforming Receiver Based on MRC and NC-LMS for DS-CDMA System”. IEEE International Conference on Application-specific Systems, Architectures and Processors, pp 114-117, Steamboat Springs, CO, September 11-13, 2006.
- Elie H. Sarraf, Messaoud Ahmed-Ouameur, and Daniel Massicotte “FPGA Design and Implementation of Direct Matrix Inversion Beamforming based on a Steepest Descent Method” (accepté à MWCAS 2007, Montréal 5-8 Août 2007).