

5-2020

Developing an Enhanced Adaptive Antenna Beamforming Algorithm for Telecommunication Applications

Mohammad Omar Tawfiq Abualhayja'a

Follow this and additional works at: https://scholarworks.uaeu.ac.ae/electric_theses



Part of the [Engineering Commons](#)

Recommended Citation

Abualhayja'a, Mohammad Omar Tawfiq, "Developing an Enhanced Adaptive Antenna Beamforming Algorithm for Telecommunication Applications" (2020). *Electrical Engineering Theses*. 9.
https://scholarworks.uaeu.ac.ae/electric_theses/9

This Thesis is brought to you for free and open access by the Electrical Engineering at Scholarworks@UAEU. It has been accepted for inclusion in Electrical Engineering Theses by an authorized administrator of Scholarworks@UAEU. For more information, please contact fadl.musa@uaeu.ac.ae.

United Arab Emirates University

College of Engineering

Department of Electrical Engineering

DEVELOPING AN ENHANCED ADAPTIVE ANTENNA
BEAMFORMING ALGORITHM FOR TELECOMMUNICATION
APPLICATIONS

Mohammad Omar Tawfiq Abualhayja'a

This thesis is submitted in partial fulfilment of the requirements for the degree of
Master of Science in Electrical Engineering

Under the Supervision of Dr. Mousa Hussein

May 2020

Declaration of Original Work

I, Mohammad Omar Tawfiq Abualhayja'a, the undersigned, a graduate student at the United Arab Emirates University (UAEU), and the author of this thesis entitled "*Developing an Enhanced Adaptive Antenna Beamforming Algorithm for Telecommunication Applications*", hereby, solemnly declare that this thesis is my own original research work that has been done and prepared by me under the supervision of Dr. Mousa Hussein, in the College of Engineering at UAEU. This work has not previously been presented or published, or formed the basis for the award of any academic degree, diploma or a similar title at this or any other university. Any materials borrowed from other sources (whether published or unpublished) and relied upon or included in my thesis have been properly cited and acknowledged in accordance with appropriate academic conventions. I further declare that there is no potential conflict of interest with respect to the research, data collection, authorship, presentation and/or publication of this thesis.

Student's Signature: محمد أبو الطيحاء Date: 18/06/2020

Copyright © 2020 Mohammad Omar Tawfiq Abualhayja'a
All Rights Reserved

Approval of the Master Thesis


This Master Thesis is approved by the following Examining Committee Members:

- 1) Advisor (Committee Chair): Mousa Hussein

Title: Associate Professor

Department of Electrical Engineering

College of Engineering

Signature 

Date June 23, 2020

- 2) Member: Imad Barhumi

Title: Associate Professor

Department of Electrical Engineering

College of Engineering

Signature 


Date June 23, 2020

- 4) Member (External Examiner): Josep M. Jornet

Title: Associate Professor

Department of Electrical and Computer Engineering

Institution: Northeastern University, Boston, USA

Signature 

Date June 23, 2020

This Master Thesis is accepted by:

Dean of the College of Engineering: Professor Sabah Alkass

Signature  _____ Date 7/7/2020 _____

Dean of the College of Graduate Studies: Professor Ali Al-Marzouqi

Signature  _____ Date 07/07/2020 _____

Copy ____ of ____

Abstract

As a key enabler for advanced wireless communication technologies, smart antennas have become an intense field of study. Smart antennas use adaptive beamforming algorithms which allow the antenna system to search for specific signals even in a background of noise and interference. Beamforming is a signal processing technique used to shape the antenna array pattern according to prescribed criteria.

In this thesis, a comparative study is presented for various adaptive antenna beamforming algorithms. Least mean square (LMS), normalized least mean square (NLMS), recursive least square (RLS) and sample matrix inversion (SMI) algorithms are studied and analyzed. The study also considers some possible adaptive filters combinations and variations, such as: LMS with SMI weights initialization, and combined NLMS filters with a variable mixing parameter. Furthermore, a new adaptive variable step-size normalized least mean square (VSS-NLMS) algorithm is proposed. Sparse adaptive algorithms, are also studied and analyzed, and two channel estimations sparse algorithms are applied to an adaptive beamformer, namely: proportionate normalized least-mean-square (PNLMS), and l_p norm PNLMS (LP-PNLMS) algorithms. Moreover, a variable step size has been applied to both of these algorithms for improved performance. These algorithms are simulated for antenna arrays with different geometries and sizes, and results are discussed in terms of their convergence speed, max side lobe level (SLL), null depths, steady state error and sensitivity to noise.

Simulation results confirm the superiority of the proposed VSS-NLMS algorithms over the standard NLMS without the need of using combined filters. Results also show an improved performance for the sparse algorithms after applying the proposed variable step size.

Keywords: Adaptive beamforming, antenna array, adaptive filters algorithms, sparse signal processing.

Title and Abstract (in Arabic)

تطوير خوارزمية محسنة لتكوين شعاع الهوائي التكيفي لتطبيقات الإتصالات

الملخص

أصبحت الهوائيات الذكية موضع اهتمام للبحث والدراسة باعتبارها عامل تمكين مفتاحي لتكنولوجيا الإتصالات اللاسلكية المتقدمة، تستعمل الهوائيات الذكية خوارزميات تكوين الشعاع التكيفي ما يسمح لأنظمة الهوائيات بالبحث عن إشارة معينة حتى في ظروف من تداخل الموجات والتشويش. تكوين الشعاع التكيفي هي تقنية معالجة اشارات تستعمل لتشكيل الطيف الراديوي لمصفوفة الهوائيات وفقاً لمعايير ومقاييس محددة.

في هذه الرسالة تم تقديم دراسة لمقارنة مختلف خوارزميات تكوين الشعاع التكيفي حيث تم دراسة وتحليل خوارزمية أقل مربع متوسط (LMS)، وأقل مربع متوسط المعيار (NLMS)، والمربعات الصغرى المتكررة (RLS)، وانعكاس المصفوفة النموذجي (SMI). الدراسة تناولت أيضاً طرق لدمج أو تغيير المرشحات (الفلاتر) التكيفية، مثل: أقل مربع متوسط (LMS) وابتداء الأوزان من خلال انعكاس المصفوفة النموذجي (SMI)، ودمج مرشحات أقل مربع متوسط معيار (NLMS) باستخدام معامل دمج متغير. إضافة إلى تقديم خوارزمية جديدة هي أقل مربع متوسط المعيار (NLMS) ذات حجم الخطوة المتغير الجديدة (VSS-NLMS). أيضاً تم دراسة وتحليل الخوارزميات التكيفية التناثرية، وتم استخدام اثنتان من خوارزميات تخمين القنوات التكيفية التناثرية على مكون الشعاع التكيفي، هي خوارزمية أقل مربع متوسط المعيار المتناسبة (PNLMS)، وخوارزمية I_p المعدل - أقل مربع متوسط المعيار المتناسبة (LP-PNLMS). بالإضافة إلى ذلك تم تطبيق حجم الخطوة المتغير على كلتا الخوارزميتين لتحسين الأداء.

تم محاكاة هذه الخوارزميات باستخدامها على مصفوفات هوائيات ذات أشكال وأحجام مختلفة، وتم مناقشة النتائج تحت اعتبارات سرعة الاستجابة، وأقصى مستوى للفص الجانبي، وعمق القمع، ونسبة خطأ الحالة المستقرة، ومقدار التأثير بالتشويش. وقد أظهرت نتائج المحاكاة تفوق الخوارزمية الجديدة (VSS-NLMS) على (NLMS) الأساسية دون الحاجة لاستعمال مرشحات مدمجة. كذلك تظهر النتائج تحسن أداء الخوارزميات التكيفية التناثرية عند استخدام حجم الخطوة المتغير.

مفاهيم البحث الرئيسية: تكوين الشعاع التكيفي، مصفوفة الهوائيات، خوارزميات المرشحات التكيفية، تحليل الاشارات التناثري.

Acknowledgements

All thanks and praise to Almighty Allah for all the graces He granted to me along my life.

I cannot express my gratitude and appreciation toward my advisor Dr. Mousa Hussein for his continuous support, valuable supervision and useful advices, without which much of this work would not have been possible. Further, thanks also go to all the committee members for their efforts.

Moreover, I would like to thank my parents, family and friends for their unconditional support and encouragement.

Dedication

To my beloved parents and family

Table of Contents

Title.....	i
Declaration of Original Work.....	ii
Copyright.....	iii
Approval of the Master Thesis	iv
Abstract	vi
Title and Abstract (in Arabic).....	vii
Acknowledgements.....	ix
Dedication	x
Table of Contents.....	xi
List of Tables.....	xiii
List of Figures.....	xiv
List of Abbreviations	xvi
Chapter 1: Introduction	1
1.1 Statement of the Problem.....	1
1.2 Overview.....	2
1.3 Research Objectives.....	3
1.4 Thesis Organization.....	4
Chapter 2: Literature Review	5
2.1 Adaptive Algorithms.....	5
2.2 Sparsity Aware Adaptive Algorithms.....	7
Chapter 3: Technologies and Methods	8
3.1 Introduction to Antenna Array	8
3.2 Phased Array and Adaptive Beamforming	11
3.3 Adaptive Algorithms.....	13
3.3.1 Introduction to Adaptive Filtering and Weiner-Hopf Equations.....	13
3.3.2 Least Mean Square Algorithm.....	16
3.3.3 Normalized Least Mean Square Algorithm.....	17
3.3.4 Recursive Least Square Algorithm	18
3.3.5 Sample Matrix Inversion Algorithm.....	21
3.3.6 Combination of Two NLMS Filters with Variable Mixing Parameter	23
3.3.7 New Variable Step-Size NLMS Algorithm.....	25
3.4 Sparse Adaptive Signal Processing and Zero-Attracting Algorithms.....	29

3.4.1 Panelized LMS-Based Algorithms	29
3.4.2 Proportionate Normalized Least Mean Square Algorithm	30
3.4.3 LP-PNLMS Algorithm.....	32
3.4.4 Variable Step-Size PNLMS/LP-PNLMS Algorithms.....	33
Chapter 4: Results and Discussions	35
4.1 Linear Array for Non-Sparse Algorithms	35
4.2 Rectangular Array for Non-Sparse Algorithms.....	46
4.3 Sparsity Aware Adaptive Algorithms.....	51
Chapter 5: Conclusion and Future Work	56
References	58

List of Tables

Table 3.1: Summary of LMS algorithm.....	17
Table 3.2: Summary of NLMS algorithm.....	18
Table 3.3: Summary of RLS algorithm.....	21
Table 3.4: Summary of SMI algorithm.....	23
Table 3.5: Summary of combined NLMS filters with variable mixing parameter.....	25
Table 3.6: Summary of the new VSSNLMS algorithm.....	28
Table 3.7: Summary of the PNLMS algorithm.....	32
Table 3.8: Summary of the LP-PNLMS algorithm.....	33

List of Figures

Figure 1.1: Adaptive array pattern.....	3
Figure 3.1: Omni directional antenna radiation pattern	8
Figure 3.2: Uniform linear array.....	11
Figure 3.3: Adaptive antenna array system.....	12
Figure 3.4: Adaptive filter block diagram.....	14
Figure 3.5: Combination of two NLMS filters with variable mixing parameter.....	24
Figure 3.6: $\mu(k)$ for different α values.....	26
Figure 3.7: $g(k)$ for different values of m with ($\beta = 0.4, \gamma = 0.8$ and $\alpha = 5$).....	28
Figure 4.1: Linear array Normalized gain for 8 elements antenna array (SINR=30)	36
Figure 4.2: Linear array Normalized gain for 16 elements antenna array (SINR=30)	36
Figure 4.3: Linear array Normalized gain for 21 elements antenna array (SINR=30)	37
Figure 4.4: MSE versus iterations for LMS with 8, 16 and 21 elements linear arrays (SINR=30)	38
Figure 4.5: MSE versus iterations for NLMS with 8, 16 and 21 elements linear arrays (SINR=30)	38
Figure 4.6: MSE versus iterations for RLS with 8, 16 and 21 elements linear arrays (SINR=30)	39
Figure 4.7: MSE versus iterations for LMS/SMI with 8, 16 and 21 elements linear arrays (SINR=30)	39
Figure 4.8: MSE versus iterations for Combined NLMS with 8, 16 and 21 elements linear arrays (SINR=30)	40
Figure 4.9: MSE versus iterations for VSS-NLMS with 8, 16 and 21 elements linear arrays (SINR=30)	40
Figure 4.10: MSE versus iterations for SMI with 8, 16 and 21 elements linear arrays (SINR=30).....	41
Figure 4.11: Linear array Normalized gain for 8 elements antenna array (SINR=10).....	42
Figure 4.12: Linear array Normalized gain for 16 elements antenna array (SINR=10).....	42
Figure 4.13: Linear array Normalized gain for 21 elements antenna array (SINR=10).....	43
Figure 4.14: MSE versus iterations for LMS with 8, 16 and 21 elements linear arrays (SINR=10).....	43
Figure 4.15: MSE versus iterations for NLMS with 8, 16 and 21 elements linear arrays (SINR=10).....	44

Figure 4.16: MSE versus iterations for RLS with 8, 16 and 21 elements linear arrays (SINR=10)	44
Figure 4.17: MSE versus iterations for SMI with 8, 16 and 21 elements linear arrays (SINR=10)	45
Figure 4.18: MSE versus iterations for LMS/SMI with 8, 16 and 21 elements linear arrays (SINR=10)	45
Figure 4.19: MSE versus iterations for combined NLMS with 8, 16 and 21 elements linear arrays (SINR=10)	46
Figure 4.20: MSE versus iterations for VSS-NLMS with 8, 16 and 21 elements linear arrays (SINR=10)	46
Figure 4.21: Normalized gain for 8×8 rectangular antenna array	47
Figure 4.22: Normalized gain for 16×16 rectangular antenna array	48
Figure 4.23: MSE versus iterations for LMS for different sizes rectangular arrays	48
Figure 4.24: MSE versus iterations for NLMS for different sizes rectangular arrays	49
Figure 4.25: MSE versus iterations for RLS for different sizes rectangular arrays	49
Figure 4.26: MSE versus iterations for SMI for different sizes rectangular arrays	50
Figure 4.27: MSE versus iterations for LMS/SMI for different sizes rectangular arrays	50
Figure 4.28: MSE versus iterations for combined NLMS for different sizes rectangular arrays	51
Figure 4.29: MSE versus iterations for VSS-NLMS for different sizes rectangular arrays	51
Figure 4.30: Sparse algorithms normalized gain for 8 elements antenna array	52
Figure 4.31: Sparse algorithms normalized gain for 16 elements antenna array	53
Figure 4.32: Sparse algorithms normalized gain for 21 elements antenna array	53
Figure 4.33: MSE versus iterations for PNLMS with 8, 16 and 21 elements	54
Figure 4.34: MSE versus iterations for VSS-PNLMS with 8, 16 and 21 elements	54
Figure 4.35: MSE versus iterations for Lp-PNLMS with 8, 16 and 21 elements	55
Figure 4.36: MSE versus iterations for VSS-Lp-PNLMS with 8, 16 and 21 elements	55

List of Abbreviations

3GPP	3 rd Generation Partnership Project
D2D	Device to Device
FIR	Finite Impulse Response
IoT	Internet of Things
LMS	Least Mean Square
LP-PNLMS	L _p Norm Constrained PNLMS
M2M	Machine to Machine
M-MIMO	Massive-Multiple Input Multiple Output Antenna
NLMS	Normalized Least Mean Square
PNLMS	Proportionate Normalized Least Mean Square
RF	Radio Frequency
RLS	Recursive Least Square
SINR	Signal to Interference Noise Ratio
SMI	Sample Matrix Inversion
SNR	Signal to Noise Ratio
SLL	Side Lobe Level
VSS	Variable Step Size
ZA	Zero Attractor
ZA-LMS	Zero Attractor Least Mean Square
5G	Fifth Generation

Chapter 1: Introduction

1.1 Statement of the Problem

Mobile operators around the world started investing in the deployment of the fifth generation (5G) solutions laying the foundation for smart city development, although it is still in the planning stages. This is leading to an explosive growing demand for high mobile data rates, reduced end-to-end latencies, and connectivity across a diversity of new applications such as the internet of things (IoT), massive machine type communication, etc. Massive multiple input multiple output (M-MIMO) antennas, beamforming, millimeter-wave communications, dense small cell deployment, device to device (D2D), and machine to machine (M2M) communications, are critical research areas which will have the greatest impact on progressing mobile networks [1].

5G mobile network would utilize the huge spectrum in the millimeter wave bands to which will reflect on systems capacity and performance. Recently, the frequency bands above 24 GHz has been discussed in the 3GPP as the carrier at (5G) mobile networks. However, using the millimeter Wave bands has many challenges compared to the existing systems, in terms of high propagation loss, directivity, and sensitivity to blockage [2,3]. In order to fulfil the requirements of involving much higher frequencies and higher order modulation schemes, the power utilization need to be maximized by focusing the radio frequency (RF) resources where they are most needed. At the same time, to eliminate any source of interference or improve the signal to interference noise ratio (SINR), higher gain antennas with narrower beam patterns pointing to the receiver are needed.

Adaptive beamforming is a key-enabling technology for exploiting the millimeter wave bands, by directing the narrow beam pattern towards the desired direction and forming nulls towards the interferer directions. Adaptive beamforming provides improved coverage and maintain continuous signal or user tracing. Another advantage of the beamforming, that smaller cells can be created more efficiently, where the power on the cell boundary can be managed, so there is less interference and spillover across cell edges. This leads to increase system capacity for the existing mobile communications systems by maximizing the reuse factor. This study will help to design an enhanced smart antenna system that can improve the performance of wireless communication systems and overcome the challenges of using higher frequency bands. This can be accomplished by developing an enhanced adaptive beamforming technique.

1.2 Overview

As shown in Figure 1.1 [4], adaptive beamforming is a signal processing approach that spatially filters the antenna array input by steering the array main beam toward the desired signal and forming nulls at the directions of interference. Basically, the adaptive beamforming is based on adaptive filter techniques that multiply the received signal by a complex weight vector to adjust the magnitude and phase of the signal in order to iteratively drive the output to a desired value.

There are several types of adaptive algorithms which are typically characterized in terms of their convergence properties, steady state error and computational complexity, such as the least mean squares algorithm (LMS) and its variances, sample matrix inversion (SMI) algorithm, recursive least squares (RLS) algorithm, and many other algorithms. These algorithms can be viewed as

approximations of the Wiener filter, which tries to minimize the mean square error of the output signal. In general, the adaptive filter consists of digital filter with adjustable tap coefficients or weights and adaptive algorithm. An antenna array with N elements is considered as an FIR filter with N tap coefficients represent the array weights, here the Wiener filter solves the Weiner-Hoff equation for the optimum weights that minimize the mean-square value of the estimation error.

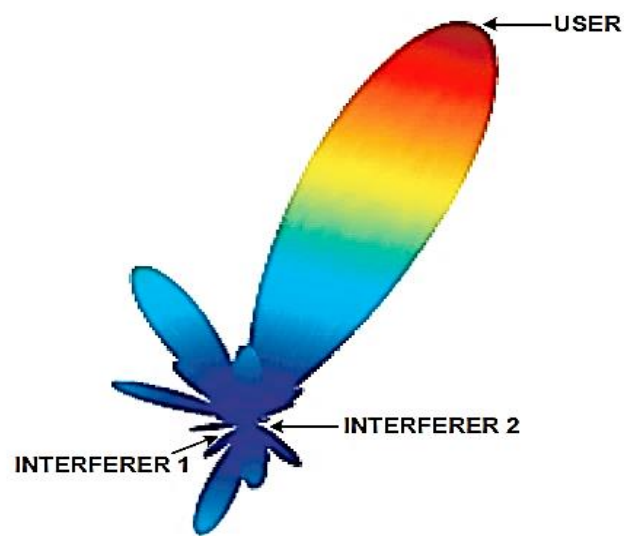


Figure 1.1: Adaptive array pattern [4].

1.3 Research Objectives

This research aims to provide a comprehensive study for the adaptive algorithms, compare their feature characteristics by applying such algorithms and their possible combinations to different geometries antenna arrays with different number of elements, analyze and synthesize radiation patterns for each adaptive algorithm, and study characteristics of each adaptive algorithm in terms of convergence speed, steady state error, sensitivity to the noise and computational complexity.

Also, it is the goal of this work to develop an enhanced adaptive beamforming

algorithm based on the analysis of characteristic features of the adaptive algorithms, and apply the enhanced algorithm to different antenna arrays, and study the impact of the developed algorithm on different wireless communication environments.

1.4 Thesis Organization

The thesis is organized as follows:

Chapter 2 (Literature Review): It gives a brief background about beamforming and adaptive algorithms; it also goes through the previous researches related to the adaptive algorithms.

Chapter 3 (Technologies and Methods): This chapter introduces the antenna arrays and the adaptive beamforming algorithms starting with the standard recursive parameter estimation algorithms, and its possible enhanced combinations and variations. Then, adaptive sparsity aware algorithms are introduced. Enhanced performance adaptive algorithms are also proposed.

Chapter 4 (simulation results and discussions): In this chapter adaptive algorithms are applied to different geometries antenna arrays with different number of elements antenna arrays, the resulting radiation patterns for each adaptive algorithm is compared, and characteristics of each adaptive algorithm is studied in terms of convergence speed, steady state error, sensitivity to the noise and computational complexity.

Chapter 5 (Conclusion and Future Work): Chapter five concludes the thesis and proposes future work.

Chapter 2: Literature Review

Antenna engineering plays a major role in our lives, and there are different types of antennas that serve different applications. Some of these applications require radiation characteristics such as high gain pattern that cannot be achieved using single antenna element. However, antenna arrays can provide the desired characteristics in more efficient way [5]. Another advantage of antenna array that its radiation pattern can be shaped and controlled easily using beamforming techniques.

Antenna arrays and beamforming have been used in many applications such as radars, sonar imaging, communications, geophysical exploration, astrophysical exploration and biomedical applications [6].

Adaptive beamforming is an array processing technique that specially filters the received signals by controlling the antenna array pattern using an adaptive algorithm. Among the various adaptive algorithms, the least mean squares (LMS) algorithm, the normalized least mean square (NLMS) algorithm, direct sample matrix inversion (SMI) algorithm, and recursive least squares (RLS) algorithm are very popular and widely used.

2.1 Adaptive Algorithms

With the recent exponential growth of the wireless communication demands and the extreme interest in the field of smart antennas, antenna arrays and beamforming have been studied extensively and many different algorithms have been proposed to implement adaptive beamforming [7]. In the last several years, there have been massive research efforts dealing with the different adaptive filtering algorithms such as LMS, NLMS, SMI and RLS algorithms [8-10]. Each of these algorithms has

advantages and drawbacks in terms of algorithm complexity, convergence speed and the resulting radiation pattern characteristics such as sidelobe level [4, 6-10]. However, the performance of these algorithms has been improved by proposing combined algorithms that overcome the weaknesses of original algorithms and achieve better performance.

In [11] an enhanced adaptive beamforming using LMMN algorithm with SMI initialization was proposed, which shows more stability and improved steady state error. In [12] a hybrid NLMS/RLS algorithm was developed, the performance of the new hybrid algorithm is compared with LMS, NLMS, RLS, SMI and SMI/LMS algorithms, and simulation results show that new hybrid algorithm achieved the best performance among these algorithms in terms of convergence speed, beamforming pattern stability, and sidelobe levels.

Since the step size is a significant factor for the convergence speed and stability of the LMS filters, combinations of LMS based filters with different step sizes using adaptive mixing parameter is proposed in [13-15]. These combined filters are showing improved results in channel estimation and echo cancelation applications without a costly increase in the computation complexity.

Variable step-size is another efficient way to speed up the convergence rates and ensure stable system performance, different variable step-size LMS algorithms was proposed and analyzed in [16,17]. Also, a novel variable step-size NLMS algorithm with improved convergence speed and steady state error was proposed in [18].

2.2 Sparsity Aware Adaptive Algorithms

Standard recursive parameter estimation algorithms such as LMS, RLS and SMI algorithms cannot exploit the sparsity characteristic, resulting in poor performance especially for cases dealing with sparse signals. Many studies have been performed on the design and analysis of sparsity aware adaptive algorithms. Multiple sparse LMS based algorithms were developed for channel estimation and system identification, by applying different zero attractor penalty terms to the original cost function in [19-21]. Constraint NLMS algorithms were proposed for sparse adaptive array beamforming control applications in [22,23]. Also, a proportionate normalized least mean square (PNLMS) algorithm was proposed as sparse algorithm. The PNLMS algorithm performance was enhanced using variable step size in [24], and using different zero-attractor penalty terms as in [25-27].

Chapter 3: Technologies and Methods

3.1 Introduction to Antenna Array

The Antenna radiation pattern is a graphical representation of the antenna radiation characteristics, usually in the far-field region and commonly normalized to the maximum value, as a function of spherical coordinates. These radiation characteristics include power flux density, radiation intensity, field strength, directivity, and polarization [5,27]. In general, there are three types of radiation patterns [4]:

- Isotropic: in which the antenna radiates equally in all directions.
- Omni-directional: in which the radiation is non-directive in one plane while it is directive in another orthogonal plane, as shown in Figure 3.1 [5].
- Directional: in which the radiated power is concentrated in a specific direction.

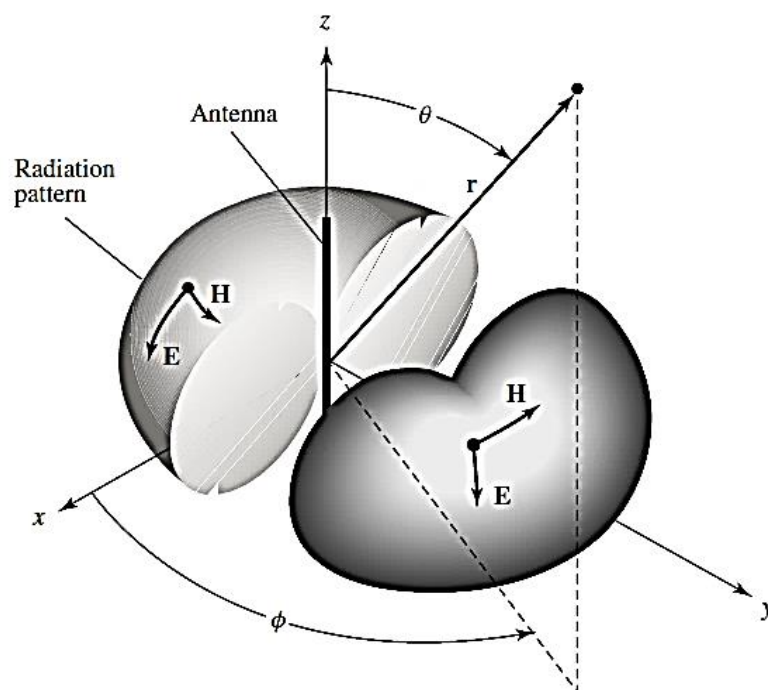


Figure 3.1: Omni directional antenna radiation pattern [5].

Among the radiation parameters that characterize antennas, directivity and gain are the most effective parameter that expresses its directional properties and performance characteristics. Antenna directivity is defined as the ratio of the radiation intensity in a given direction from the antenna to the radiation intensity averaged over all directions [5]. The directivity can be defined as follows:

$$D(\theta, \phi) = \frac{U(\theta, \phi)}{U_0} = \frac{4\pi U(\theta, \phi)}{P_{rad}} \quad (1)$$

where U is the radiation intensity, U_0 is the radiation intensity of an isotropic source, and P_{rad} is the total radiated power.

Antenna gain is defined as the ability of antenna to direct energy in particular direction, taking into account the mismatch and polarization losses [5]. The gain is related to the directivity as follows:

$$G = e_{rad} D \quad (2)$$

where e_{rad} is the radiation efficiency.

Antenna gain also reflected on its beamwidth, where a high gain corresponds to a narrower beamwidth, hence improved power utilization and fewer opportunities to receive interference. Oppositely, the antenna with the low-gain has the a higher chance to receive interference due to its wide beamwidth.

The design of high gain antennas became a critical aspect in the modern wireless communication systems, which tends to use higher frequency bands resulting in higher attenuation. To compensate the additional attenuation higher-gain antennas are required which can be achieved by increasing the electrical size of the antenna. However, there is another efficient way to achieve high gain characteristics without

increasing the antenna dimension by using antenna arrays, multiple antenna elements assembled in a geometrical and electrical configuration [4,5].

The antenna array pattern can be shaped by controlling the geometrical configuration, the excitation amplitude and phase of single element, distance between array elements and finally the pattern of single element; however, the geometry of the array and distance between elements is difficult to be changed [5].

Total field of the antenna array can be represented as follows:

$$E_{total} = E_{single\ element} \cdot Array\ Factor \quad (3)$$

Consider an N elements uniform linear antenna array with an elements spacing d and an excitation phase β , as shown in Figure 3.2 [5], then the array factor can be defined as:

$$AF = \sum_{n=1}^N e^{j(n-1)\Psi} \quad (4)$$

where $\Psi = kd \cos \theta + \beta$.

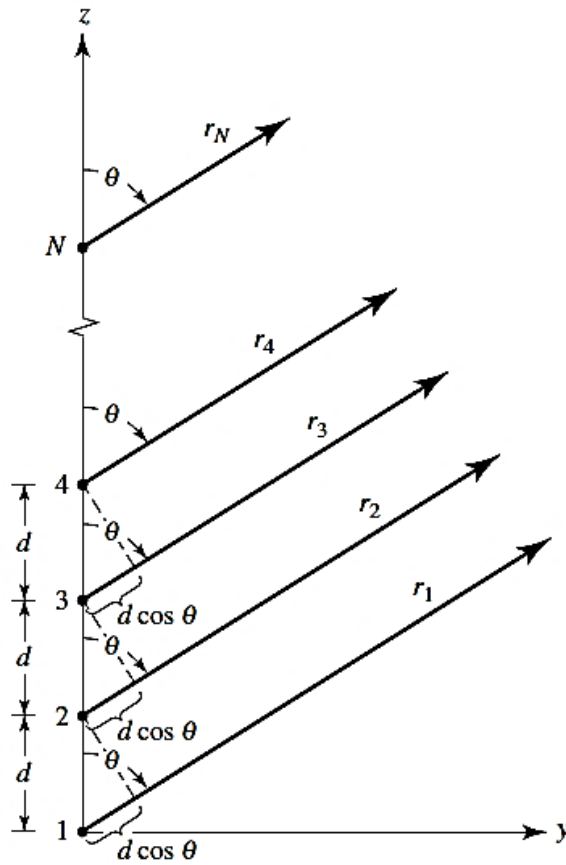


Figure 3.2: Uniform linear array [5].

3.2 Phased Array and Adaptive Beamforming

The idea behind a phased array is to control the array pattern by applying complex weights to the input signal as shown in Figure 3.3. Consider a model of a linear antenna array composed of N uniformly distributed isotropic antenna elements. Assume the input to the array consists of one desired signal $s(t)$ and M interference sources each having narrow band signal given by $I_a(t)$, in addition to white additive noise $N(t)$. The electric field of the plane wave can be expressed as:

$$E = e^{jk \cdot r} \quad (5)$$

where k is the wave vector, and r is the position vector of antenna elements.

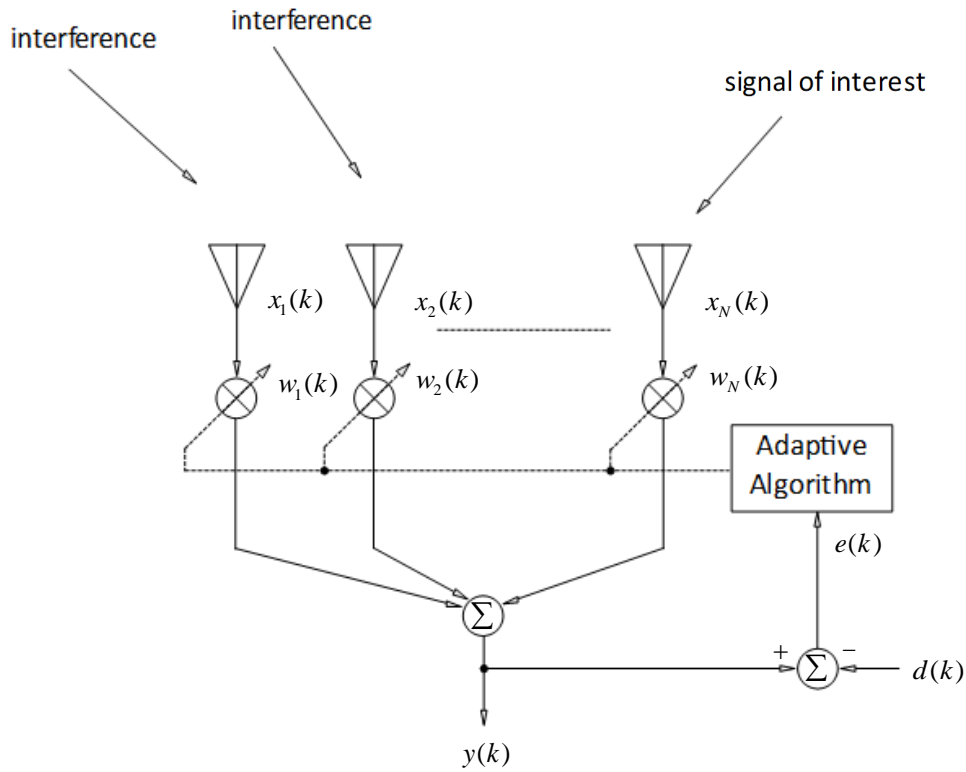


Figure 3.3: Adaptive antenna array system.

Define $v(t)$ the steering vector as:

$$v(k) = \begin{bmatrix} e^{-jk.r_1} \\ e^{-jk.r_2} \\ \vdots \\ e^{-jk.r_N} \end{bmatrix} \quad (6)$$

Now the received signal $x(t)$ at each antenna element consists of the summation of the signal of interest and interference signals each multiplied by its steering vector in addition to the white additive noise and can be defined as follows:

$$x(t) = s(t) v(k_s) + N(t) + \sum I_a(t) v(k_a) \quad (7)$$

The output of the array will be the summation of the input of each element multiplied by its weight [5], which can be given by the following equation:

$$y(k) = w^H(k) x(k) \quad (8)$$

where $w(k)$ is the array weight vector and $x(t)$ is the received signal vector.

The problem that adaptive beamforming addresses is how to adjust the array weights in order to drive the array output $y(k)$ to the desired output $d(k)$, accordingly an estimation error $e(k)$ can be defined as:

$$e(k) = d(k) - w^H x(k) \quad (9)$$

Adaptive algorithms are basically used to minimize the resulting error statistically [29], which is to solve:

$$\text{Min} E [e(k) e^*(k)] \quad (10)$$

with $E[\cdot]$ representing the expectation operator.

3.3 Adaptive Algorithms

The linear adaptive filter basically consists of two processes, the filter process where the output is produced as a response to the input sequence, and the adaptive process or algorithm which adjusts the set of parameters used in the filtering process.

3.3.1 Introduction to Adaptive Filtering and Wiener-Hopf Equations

Consider the block diagram of Figure 3.4, assuming a random input process $x(n)$, the goal behind the adaptive algorithms is to find the optimum filter weights $w(n)$ that drives the output $y(n) = \sum_{k=0}^{\infty} w(k) x(n-k)$ to a desired value $d(n)$, resulting in an estimation error $e(n) = d(n) - y(n)$.

Adaptive algorithms are basically approaches used to statically minimize the cost function or the performance index represented by the mean-square value of the estimation error, or the expectation of the absolute value of the expectation error [29].

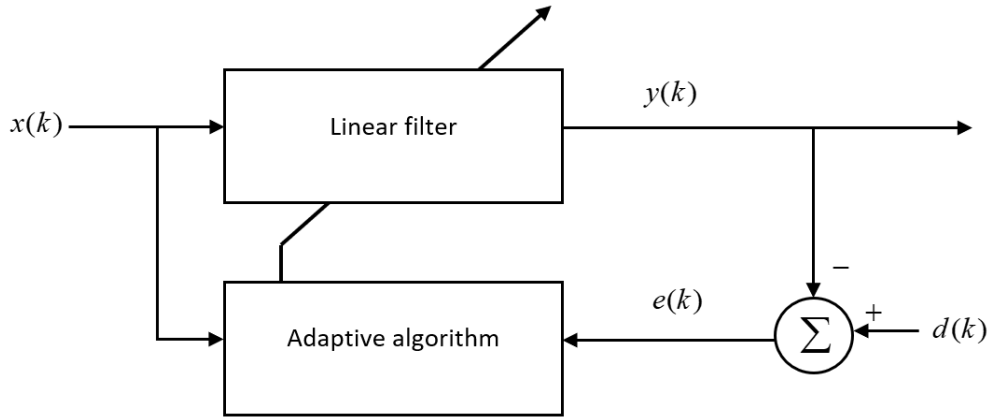


Figure 3.4: Adaptive filter block diagram.

Now consider the below cost function:

$$J(n) = E[e(n) e^*(n)] \quad (11)$$

To get the minimum value of the cost function as a function of the weight vector $w(n)$, take the gradient vector ∇J and set it to zero, where the k^{th} element of the weight vector is defined as $w_k(n) = a_k + jb_k$ [29]. Hence, the gradient of the cost function with respect to $w(n)$ is defined as follows,

$$\begin{aligned} \nabla J &= E \left[\frac{\partial e(n)}{\partial a_k} e^*(n) + \frac{\partial e^*(n)}{\partial a_k} e(n) + \frac{\partial e(n)}{\partial b_k} j e^*(n) + \frac{\partial e^*(n)}{\partial b_k} j e(n) \right] \\ &= -2 E[e^*(n) x(n-k)] \\ &= -2 E \left[\left(d^*(n) - \sum_{j=0}^{\infty} (w(j) x^*(n-j)) \right) x(n-k) \right] \end{aligned} \quad (12)$$

setting $\nabla J = 0$ to find the wight vector value that minimizes of the cost function,

$$E[d^*(n) x(n-k)] - \sum_{j=0}^{\infty} w(j) E[x^*(n-j) x(n-k)] = 0 \quad (13)$$

rearrange, to get the Wiener-Hopf Equations:

$$E[d^*(n) x(n-k)] = \sum_{j=0}^{\infty} w(j) E[x^*(n-j) x(n-k)] \quad (14)$$

The left side of the equation $E[d^*(n) x(n-k)]$ which represents the cross correlation between the filter input $x(n-k)$ and the desired output $d(n)$ can be defined as $p(-k)$, while the right side $E[x^*(n-j) x(n-k)]$ the autocorrelation for the input vector for a lag $j-k$ is defined as $r(j-k)$ [29].

If a length M FIR filter is considered, the Wiener-Hopf equations become:

$$\sum_{j=0}^{M-1} w(j) r(j-k) = p(-k) \quad (15)$$

or in a matrix form:

$$Rw_{opt} = P$$

$$\begin{bmatrix} r(0) & r(1) & \dots & r(M-1) \\ r(1) & r(0) & \dots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(M-1) & \dots & \dots & r(0) \end{bmatrix} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(M-1) \end{bmatrix} = \begin{bmatrix} p(0) \\ p(-1) \\ \vdots \\ p(-(M-1)) \end{bmatrix} \quad (16)$$

then the optimum weight vector can be described as the follows:

$$w_{opt} = R^{-1} P \quad (17)$$

Hence, the minimum square error can be defined as:

$$J_{opt} = \sigma_d^2 - P^H R^{-1} P \quad (18)$$

where $\sigma_d^2 = w_{opt}^H R w_{opt}$, is the variance of the desired output.

3.3.2 Least Mean Square Algorithm

The method of steepest decent is an optimization technique that can be used to find the weight vector of the Weiner solution given the correlation matrix R and the cross-correlation vector P [29], where the optimum weight vector can be calculated iteratively as follows:

$$w(k+1) = w(k) - \frac{1}{2} \mu \nabla J(k) \quad (19)$$

where μ is the step size.

However, in reality it is not possible to get the optimum gradient vector $\nabla J(k)$ since it requires a prior knowledge of both the correlation matrix R and the cross-correlation vector P .

The LMS algorithm is a stochastic gradient algorithm, that replaces the correlation matrix R and the cross-correlation vector P in the steepest decent algorithm by their data driven approximation [29,30].

Let $\nabla J(k)$ denotes the approximate gradient vector

$$\hat{\nabla} J(n) = -2 \hat{P}(n) + 2 \hat{R}(n) w(n) \quad (20)$$

where $\hat{R}(n) = x(n) x^H(n)$ and $\hat{P}(n) = x(n) d(n)$, are the instantaneous estimate based on the input vector $x(n)$ for the correlation matrix R and the cross correlation vector P respectively [8,28].

Hence, the LMS algorithm updates filter taps or array weights iteratively [8-11,28], using the following equation:

$$w(k+1) = w(k) + \mu x(k) e^*(k) \quad (21)$$

It turns out that convergence of the filter is related to the step size μ , that is the step size μ should satisfy the following condition, $0 < \mu < \frac{2}{\lambda_{max}}$ where λ_{max} is the maximum eigenvalue of the input vector autocorrelation matrix [8,9,28].

The LMS computational simplicity, easy coding and robustness, are significant features make it one of the most used adaptive filtering algorithms [8-11,28,29]. A summary of the LMS algorithm including initial values, algorithm parameters and update equations are summarized in Table 3.1 below.

Table 3.1: Summary of LMS algorithm.

Initialization	$w(0) = 0$
Parameters	μ , the LMS step size
Update	$e(k) = d(k) - y(n)$ $w(k+1) = w(k) + \mu x(k) e^*(k)$

3.3.3 Normalized Least Mean Square Algorithm

The LMS algorithm weights update is driven by the input vector $x(k)$, as shown in Table 3.1, which raises the probability of having a gradient noise amplification problem in case of large $x(k)$ values. Moreover, the convergence of the LMS algorithm is relatively slow, hence, the normalized least mean square (NLMS) algorithm is proposed to overcome the gradient noise amplification problem and more importantly significantly increase the convergence rates.

Alternatively a modified update relation is given in equation (22). Compared to the LMS algorithm, the step size of the NLMS algorithm is time varying, since the

weights corrector term is normalized with respect to the norm of input vector weights update [12,29].

$$w(k+1) = w(k) + \frac{\mu_{NLMS}}{\|x(k)\|^2 + \alpha} x(k) e^*(k) \quad (22)$$

where $\|x(k)\|^2$ is the Euclidean norm of the input vector, μ_{NLMS} is the NLMS adaption constant with $1 < \mu_{NLMS} < 2$, and α is a small positive constant used to avoid division by zero [24]. NLMS algorithm is summarized in the below Table 3.2.

Table 3.2: Summary of NLMS algorithm.

Initialization	$w(0) = 0$
Parameters	μ_{NLMS} , the NLMS step size α , a small positive constant
Update	$e(k) = d(k) - y(n)$ $w(k+1) = w(k) + \frac{\mu_{NLMS}}{\ x(k)\ ^2 + \alpha} x(k) e^*(k)$

3.3.4 Recursive Least Square Algorithm

In the method of least squares, the optimum weights that drives the filter output to the desired value and minimizes the estimated error $\sum_{i=k_1}^{k_2} |e(i)|^2$ can be found by projecting the desired output vector on the column space of the input sequence matrix using the modified weighting vector [29].

Given an input vector $x(k)$, and a desired output value $d(k)$, error for M taps FIR filter is given by:

$$e(k) = d(k) - \sum_{n=0}^{M-1} w(n) x(k-n) \quad (23)$$

Now assume a data window between $k_1=0$, and $k_2= N-1$, (23) can be written as:

$$\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(N-1) \end{bmatrix} = \begin{bmatrix} d(0) \\ d(1) \\ \vdots \\ d(N-1) \end{bmatrix} - \begin{bmatrix} x(0) & x(-1) & \dots & r(M-1) \\ x(1) & x(0) & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ x(N-1) & \dots & \dots & r(N-M) \end{bmatrix} \begin{bmatrix} w(0) \\ w(1) \\ \vdots \\ w(M-1) \end{bmatrix}$$

$$e = d - Aw \quad (24)$$

where A is the input sequence matrix.

Hence, the cost function that consists of the sum of the error squares is given by:

$$\begin{aligned} \sum_{i=k_1}^{k_2} |e(i)|^2 &= \|e\|_2^2 = \|d - Aw\|_2^2 \\ &= (d - Aw)^T (d - Aw) \\ &= d^T d - w^T A^T d - d^T A w + w^T A^T A w \end{aligned} \quad (25)$$

To obtain the value of w that minimizes the cost function set $\nabla = 0$,

$$-2 A^T d + 2 A^T A w = 0 \quad (26)$$

then obtain the optimum weight $w^*(k)$,

$$\begin{aligned} w^*(k) &= (A^T A)^{-1} A^T d \\ &= A^\dagger d(k) \end{aligned} \quad (27)$$

where $A^\dagger = (A^T A)^{-1} A^T$ is the pseudoinverse of the input sequence matrix A .

Look at $A^T A$ as a data driven estimate of the autocorrelation matrix, where

$$A^T A \approx N R \quad (28)$$

similarly, $A^T d$ can be considered as an estimate of the cross correlation between the input vector and the desired output,

$$A^T d \approx N P \quad (29)$$

and by substituting in equation (26),

$$N R w^* = N P \quad (30)$$

The resulting equation looks similar to the Weiner-Hopf equation; thus, the least squares algorithm looks like an approximation to the Wiener filter.

The RLS algorithm is developed from the method of least squares using the matrix inversion lemma, so the updated weights vector $w(k)$ can be obtained from the old-squares estimate $w(k-1)$ without performing any matrix inversion calculations by utilizing the input vector sequence [4,29]. The RLS algorithm is firstly initiated by setting the weights vector $w(k)$ and the correlation matrix inverse $P(k)$ as follows:

$w(k) = 0$ and $P(0) = \delta^{-1} I$, where δ is a small positive constant.

The weights vector and the correlation matrix inverse are updated as follows:

$$w(k) = w(k-1) + g(k) \zeta^*(k) \quad (31)$$

$$P(k) = \lambda^{-1} P(k-1) - \lambda^{-1} g(k) x^H(k) P(k-1) \quad (32)$$

where $\zeta(k) = d(k) - w^H(k-1) x(k)$ is the prior estimated error and $g(k)$ is the gain vector which is defined as:

$$g(k) = \frac{\lambda^{-1} P(k-1) x(k)}{1 + \lambda^{-1} x^H(k) P(k-1) x(k)} \quad (33)$$

where λ is the forgetting factor, a positive constant less than 1.

Although the RLS rate of convergence is faster than the LMS algorithm, in terms of calculations complexity, the RLS algorithm is significantly costly, as shown in Table 3.3 the RLS needs many mathematical operations per iteration compared to the NLMS and LMS algorithms.

Table 3.3: Summary of RLS algorithm.

Initialization	$w(k) = 0$ and $P(0) = \delta^{-1} I$
Parameters	δ , a small positive constant λ , the forgetting factor
Update	$g(k) = \frac{\lambda^{-1} P(k-1) x(k)}{1 + \lambda^{-1} x^H(k) P(k-1) x(k)}$ $P(k) = \lambda^{-1} P(k-1) - \lambda^{-1} g(k) x^H(k) P(k-1)$ $\xi(k) = d(k) - w^H(k-1) x(k)$ $w(k) = w(k-1) + g(k) \xi^*(k)$

3.3.5 Sample Matrix Inversion Algorithm

In the LMS algorithm, the system goes through many iterations to drive the output toward the desired signal, and in case of rapidly changing signal characteristics, the system may not approach an acceptable convergence. A solution to this is to calculate the time average estimate of the correlation matrix by using a K-length block of data. This approach is called sample matrix inversion (SMI) [8-11].

By dividing the input data into k blocks, the array correlation matrix is defined as the following:

$$R_{xx} = \bar{X}_K(k) \bar{X}_K^H(k) \quad (34)$$

where $\bar{X}_K(k)$ is the k^{th} block of input vector ranging over K samples of data.

$$\bar{X}_K(k) = \begin{bmatrix} x_1(1+kK) & x_1(2+kK) & \dots & x_1(K+kK) \\ x_2(1+kK) & x_2(2+kK) & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ x_M(1+kK) & \dots & \dots & x_M(K+kK) \end{bmatrix} \quad (35)$$

The desired output vector can be also defined by:

$$d(k) = [d(1+kK) \quad d(2+kK) \quad \dots \quad d(K+kK)] \quad (36)$$

and the estimate of correlation vector by:

$$\bar{p}(k) = \frac{1}{K} d^*(k) \bar{X}_K(k) \quad (37)$$

The weights vector update equation is given by:

$$w(k) = \bar{R}_{xx}^{-1}(k) \bar{p}(k) \quad (38)$$

One drawback of the SMI algorithm that it is not sufficient for large number of antenna elements, but it can be used for weights initialization when combined with other algorithms as in [11].

Table 3.4 shows SMI algorithm parameters and update equations.

Table 3.4: Summary of SMI algorithm.

Parameters	k , the number of data blocks K , length of data block
Update	$\bar{p}(k) = \frac{1}{K} d^*(k) \bar{X}_K(k)$ $w(k) = \bar{R}_{xx}^{-1}(k) \bar{p}(k)$

3.3.6 Combination of Two NLMS Filters with Variable Mixing Parameter

As noted for both LMS and NLMS algorithms, convergence rates depend on the step size. On the other hand, there is a tradeoff between convergence speed and the ability of tracking the desired signal in a satisfactory manner. In order to increase convergence rate and ensure system robustness, several combined adaptive filters are proposed using an adaptive mixing parameter $\lambda(k)$ [13-15].

Consider a system of two combined NLMS filters with different adaption constants, μ_1 and μ_2 shown in Figure 3.5, then using the mixing parameter $\lambda(k)$ the combined output $y(k)$ is given by:

$$y(k) = \lambda(k) y_1(k) - (1-\lambda(k)) y_2(k) \quad (39)$$

As proposed in [14], $\lambda(k)$ is constrained to the interval [0,1] using an auxiliary variable $\alpha(k)$, where $\lambda(k) = \frac{1}{(1+e^{-\alpha(k)})}$, and $\alpha(k)$ is updated as follows:

$$\alpha(k+1) = \alpha(k) + \mu_\alpha e(k) (y_1(k) - y_2(k)) \lambda(k) (1-\lambda(k)) \quad (40)$$

where μ_α is the step size of updating the auxiliary variable $\alpha(k)$.

To ensure a continuous adaptation of the mixing parameter, $\alpha(k)$ is limited to the period between $[-\alpha^+, \alpha^+]$ [13,14].

The combined weight vector is defined as follows:

$$w(k) = \lambda(k) w_1(k) - [1 - \lambda(k)] w_2(k) \quad (41)$$

and each filter updates its weight vector using the NLMS algorithm:

$$w_i(k+1) = w_i(k) + \frac{\mu_i}{\|x(k)\|^2 + \alpha} x(k) e_i^*(k) \quad (42)$$

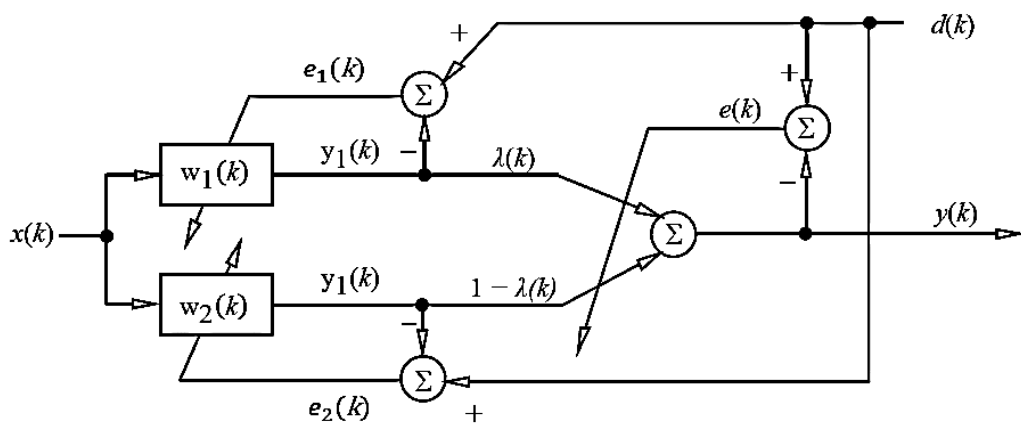


Figure 3.5: Combination of two NLMS filters with variable mixing parameter.

Combined NLMS algorithm is described in details on Table 3.5.

Table 3.5: Summary of combined NLMS filters with variable mixing parameter.

Initialization	$w_1(k)=0, w_2(k)=0, \lambda(0)=0.5, \alpha(0)=0$
Parameters	μ_i , the i^{th} filter step size $[-\alpha^+, \alpha^+]$, α constrains μ_α , the mixing parameter auxiliary variable step size
Update	$\alpha(k+1) = \alpha(k) + \mu_\alpha e(k) (y_1(k) - y_2(k)) \lambda(k) (1 - \lambda(k))$ $\lambda(k) = \frac{1}{(1 + e^{-\alpha(k)})}$ $e_i(k) = d(k) - y_i(k)$ $w_i(k+1) = w_i(k) + \frac{\mu_i}{\ x(k)\ ^2 + \alpha} x(k) e_i^*(k)$ $w(k) = \lambda(k) w_1(k) - [1 - \lambda(k)] w_2(k)$

3.3.7 New Variable Step-Size NLMS Algorithm

As stated in the previous section, convergence rates for both LMS and NLMS algorithms is related to the step size, and this arises a contradictory problem between the convergence speed and steady state error which can be solved by using combined filters with different step sizes. Another way to achieve faster convergence rates and ensure the system robustness at the same time, is to adapt a variable step-size where the step size adjustment is controlled by the error [17].

Many variable step-size algorithms have been built based on sigmoid function for LMS algorithm as proposed in [16], in which the step size is a function of error as shown below:

$$\mu(k) = \beta_\mu \left(\frac{1}{1 + e^{-\alpha|e(k)|}} - 0.5 \right) \quad (43)$$

where β and α are constants greater than zero.

As can be seen in Figure 3.6, which illustrates the relation between the step size $\mu(k)$ and the absolute error $|e(k)|$, the resulting value of the step size is large when the error has large value, which represents the early iterations this leads to a faster convergence. Later on, as the value of error decreases the step size also decreases to achieve smaller steady state error.

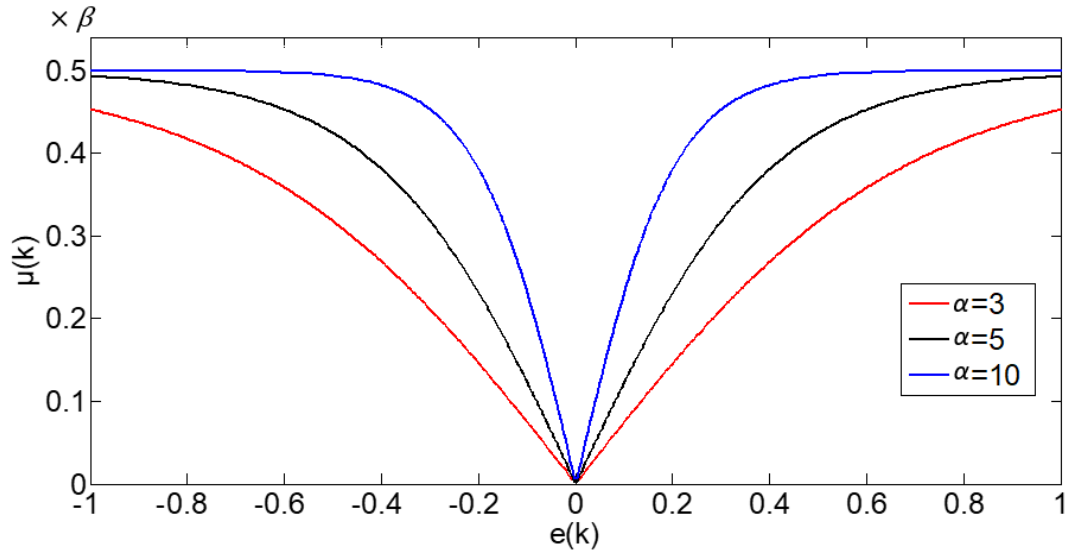


Figure 3.6: $\mu(k)$ for different α values.

Both values of β and α are important to achieve the optimal tradeoff between convergence speed and system robustness. The step size $\mu(k)$ is directly proportional to β : large β values may lead to unstable system, whereas small β will cause slower convergence. α value controls how sharp is the variation of the step size which could result a steady state error misadjustment [18,21].

Variable step-size sigmoid function can be generalized for the NLMS algorithm as proposed in [18]. The generalized sigmoid function is represented as:

$$\alpha_{GS}(k) = \beta \left(\frac{1}{1 + e^{-A(\sigma_e(k) - \sigma_n)}} - 0.5 \right) \quad (44)$$

where $\sigma_e^2 = E\{e^2(k)\}$ is the power of the error signal and $\sigma_n^2 = E\{n^2(k)\}$ is the noise power.

The variable NLMS step-size is described as:

$$\mu_{GS}(k) = \mu_{NLMS}(k) \alpha_{GS}(k) \quad (45)$$

Inspired by the Proportionate Normalized Least-Mean-Square Algorithm [16] an additional step-size adjustment for each individual tap is proposed, this adjustment is based on the absolute approximate error of the tap weights. The new Variable Step-Size NLMS algorithm weights update will be calculated using the below equation:

$$w(k+1) = w(k) + \frac{\mu_{GS}(k) G(k)}{\|x(k)\|^2 + \alpha} x(k) e^*(k) \quad (46)$$

where $G(k)$ represents a diagonal matrix used to adjust individual tap step-size bases on the approximate error.

$$G(k) = \text{diag}(g_0(k), g_1(k), \dots, g_{M-1}(k)) \quad (47)$$

With $g_i(k)$ is represented as the following sigmoid function

$$g_i(k) = \beta_g \left(1 - e^{-\alpha |e_{a_i}(k)|^m} \right) + \gamma \quad (48)$$

where $e_{a_i}(k)$ is the approximate error for the i^{th} tap

$$e_{a_i}(k) = \frac{w_i(k) - w_i(k-1)}{w_i(k)} \quad (49)$$

and β , γ , α and m are positive values chosen to constrain $g_i(k)$ between [0.8, 1.2].

As can be seen in Figure 3.6, which shows how the value of $g(k)$ changes with the local error $e_a(k)$ for different values of m and the values of β , γ and α are selected to maintain the $g(k)$ between $[0.8, 1.2]$.

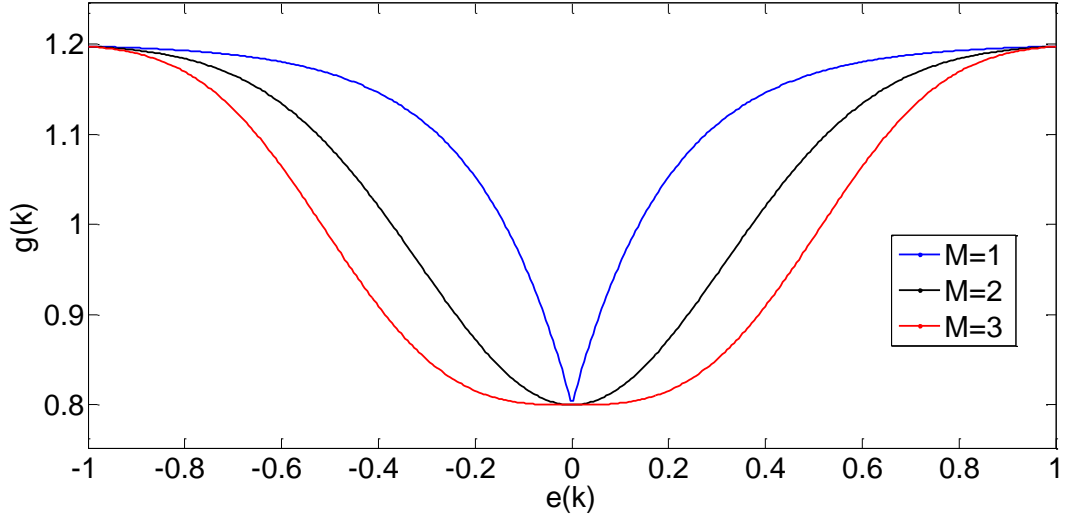


Figure 3.7: $g(k)$ for different values of m with $(\beta = 0.4, \gamma = 0.8$ and $\alpha = 5)$.

A detailed summary of the new VSSNLMS algorithm is provided in Table 3.6.

Table 3.6: Summary of the new VSSNLMS algorithm.

Initialization	$w(0) = 0, \mu(0) = \mu, e_{a_i}(0) = 1, g_i(0) = 1$
Parameters	μ the initial step-size β, γ, α and m are sigmoid function parameters.
Update	$\alpha_{GS}(k) = \beta \left(\frac{1}{1 + e^{-A(\sigma_e(k) - \sigma_n)}} - 0.5 \right)$ $\mu_{GS}(k) = \mu_{NLMS}(k) \alpha_{GS}(k)$ $e_{a_i}(k) = \frac{w_i(k) - w_i(k-1)}{w_i(k)}$ $g_i(k) = \beta_g \left(1 - e^{-\alpha e_{a_i}(k) ^m} \right) + \gamma$ $G(k) = \text{diag}(g_0(k), g_1(k), \dots, g_{M-1}(k))$ $w(k+1) = w(k) + \frac{\mu_{GS}(k) G(k)}{\ x(k)\ ^2 + \alpha} x(k) e^*(k)$

3.4 Sparse Adaptive Signal Processing and Zero-Attracting Algorithms

Large antenna arrays became a mandatory factor in order to realize the demand for higher capacity and improved performance. This attributes to the fact that big arrays are always restricted by the power consumption and processing complexity, and the adaptive algorithms such as LMS and cannot perform well in such systems. However, sparse signal processing techniques can fully utilize the sparse characteristics of the system, which enforces the filter weights towards sparsity by adding a penalty to the cost function to achieve better results in terms of performance and convergence especially for broadband systems. On the other hand, the development of sparse adaptive solutions will have a huge impact on energy conservation by reducing the ratio of active elements in the antenna array [19-23].

3.4.1 Panelized LMS-Based Algorithms

Recently, a lot of algorithms have been developed to exploit the sparse characteristics for various sparse systems. Among many sparse LMS based algorithms, the zero attracting LMS (ZA-LMS) algorithms which apply a penalty to the cost function show better steady-state performance than that of the standard LMS for sparse systems [20]. The ZA-LMS algorithms update taps weights with a zero-attractor on all filter taps that forces the inactive tap weights with values near to zero in to reach zero faster [19-21]. The standard LMS cost function can be represented as below:

$$J(k) = \frac{1}{2} e^2(k) \quad (50)$$

To develop a sparsity aware LMS algorithm, a penalty term is added to the original cost function. The modified cost function with l_1 penalty term can be represented as:

$$J_{l_1}(k) = \frac{1}{2} e^2(k) + \gamma_{l_1} \|w(k)\|_1 \quad (51)$$

The filter taps or array weights is updated iteratively based on steepest decent algorithm, using the following equation:

$$\begin{aligned} w(k+1) &= w(k) - \mu \frac{\partial J_1(k)}{\partial w(k)} \\ &= w(k) - \rho \operatorname{sgn} w(k) + \mu e(k) x(k) \end{aligned} \quad (52)$$

where $\rho = \mu \gamma$ and $\operatorname{sgn}(\cdot)$ is the sign function defined as the following:

$$\operatorname{sgn}(x) = \begin{cases} \frac{x}{|x|}, & x \neq 0 \\ 0, & x = 0 \end{cases} \quad (53)$$

Similarly, l_p penalty term can be also applied to the original cost function resulting an improved performance, where the modified cost function is shown below:

$$J_{l_p}(k) = \frac{1}{2} e^2(k) + \gamma_{l_p} \|w(k)\|_p \quad (54)$$

and the weights are updated as follows:

$$w(k+1) = w(k) + \mu e(k) x(k) - \rho_p \frac{(\|w(k)\|_p)^{1-p} \operatorname{sgn} w(k)}{\varepsilon_p + |w(k)|^{1-p}} \quad (55)$$

where ε_p is a value which bounds the penalty term.

3.4.2 Proportionate Normalized Least Mean Square Algorithm

The proportionate normalized least mean square algorithm was proposed as sparsity aware algorithm. In this algorithm, each tap weight is updated by assigning an

individual step size that is proportional to the previous weight estimation [25,26] using a gain diagonal matrix, the PNLMS updates its weights according to the following equation:

$$w(k+1) = w(k) + \mu_{PNLMS} \frac{e^*(k) G(k) x(k)}{x^T(k) G(k) x(k) + \delta_{PNLMS}} \quad (56)$$

where μ_{PNLMS} is the global PNLMS step size and $\delta_{PNLMS} = \frac{\sigma_x^2}{N}$ is a regularization parameter used to avoid division by zero.

The gain diagonal matrix $G(k)$ is given by:

$$G = \text{diag}(g_0(k), g_1(k), \dots, g_{M-1}(k)) \quad (57)$$

with

$$g_i(k) = \frac{\gamma_i(k)}{\sum_{i=0}^{N-1} \gamma_i(k)}, \quad 0 \leq i \leq N-1 \quad (58)$$

and $\gamma_i(k)$ is defined as the following:

$$\gamma_i(k) = \max \left[\rho_g \max[\delta_p, |w_0(k)|, |w_1(k)|, \dots, |w_{N-1}(k)|], |w_i(k)| \right] \quad (59)$$

where ρ_g and δ_p are positive constants used to ensure weights update continuity when the weights are initialized as zeros, with typical values $\rho_g = \frac{5}{M}$, and $\delta_p = 0.01$.

By assigning an individual step size to each element, the PNLMS algorithm achieves a very fast convergence rate at initial stages in case of highly sparse systems, but this rapid convergence is not maintained at later iterations [26]. The PNLMS

performance can be also improved by using a variable step size [24] or using a zero-attractor algorithm as in [25-27].

Table 3.7 provides a detailed summary for the PNLMS algorithm.

Table 3.7: Summary of the PNLMS algorithm.

Initialization	$w(0) = 0$
Parameters	μ_{PNLMS} the global PNLMS step-size δ_{PNLMS} is a regularization parameter ρ_g and δ_P are positive constants
Update	$e(k) = d(k) - y(k)$ $\gamma_i(k) = \max \left[\rho_g \max[\delta_P, w_0(k) , w_1(k) , \dots, w_{N-1}(k)], w_i(k) \right]$ $g_i(k) = \frac{\gamma_i(k)}{\sum_{i=0}^{N-1} \gamma_i(k)}$ $G(k) = \text{diag}(g_0(k), g_1(k), \dots, g_{M-1}(k))$ $w(k+1) = w(k) + \mu_{PNLMS} \frac{e^*(k) G(k) x(k)}{x^T(k) G(k) x(k) + \delta_{PNLMS}}$

3.4.3 LP-PNLMS Algorithm

The LP-PNLMS Algorithm was developed by applying a l_p penalty term to the PNLMS cost function, in order to utilize the sparsity awareness of the PNLMS and at the same time the benefits of ZA algorithms [27].

Now consider the following constrained optimization problem:

$$\min_{w(k+1)} \|w(k+1) - w(k)\|_{G^{-1}}^2 + \gamma_{lp} \|G^{-1} w(k+1)\|_p \quad (60)$$

subject to

$$d(k) - w(k+1) x(k) = 0 \quad (61)$$

Minimizing the above cost function using Lagrange multiplier to find the optimal weight resulting the below weights update equation:

$$w(k+1) = w(k) + \mu_{lp} \frac{e^*(k)G(k)x(k)}{x^T G(k)x(k) + \varepsilon_{lp}} - \rho_{lp} T(k) \quad (62)$$

with $\rho_{lp} = \mu_{lp} \gamma_{lp}$ and $T(k) = \|w(k)\|_p^{1-p} \text{sgn}(w(k)) \{|w(k)|^{1-p} + \varepsilon_p\}^{-1}$

where, $\varepsilon_{lp} = \frac{\delta_x^2}{N}$.

Table 3.8 provides a detailed summary for the LP-PNLMS algorithm.

Table 3.8: Summary of the LP-PNLMS algorithm.

Initialization	$w(0) = 0$
Parameters	μ_{lp} the LP-NLMS step-size ε_{lp} is a positive constant to avoid division by zero $\gamma_{lp}, \varepsilon_p$ are zero attractor term factors ρ_g and δ_p are PNLMS algorithm positive constants
Update	$g_i(k) = \frac{\gamma_i(k)}{\sum_{i=0}^{N-1} \gamma_i(k)}$ $G(k) = \text{diag}(g_0(k), g_1(k), \dots, g_{M-1}(k))$ $T(k) = \ w(k)\ _p^{1-p} \text{sgn}(w(k)) \{ w(k) ^{1-p} + \varepsilon_p\}^{-1}$ $w(k+1) = w(k) + \mu_{lp} \frac{e^*(k)G(k)x(k)}{x^T G(k)x(k) + \varepsilon_{lp}} - \rho_{lp} T(k)$

3.4.4 Variable Step-Size PNLMS/LP-PNLMS Algorithms

In order to achieve improved convergence rates and more stable steady state error, the sigmoid function variable step-size proposed in section 3.3.6 is also applied

to both PNLMS and LP-NLMS. Similar to the VSS-NLMS the variable step-size is applied for both VSSPNLMS and VSSLP-PNLMS respectively as follows:

$$\mu(k) = \mu_{PNLMS} \beta \left(\frac{1}{1 + e^{-A(\sigma_e(k) - \sigma_n)}} - 0.5 \right) \quad (63)$$

$$\mu(k) = \mu_{LP} \beta \left(\frac{1}{1 + e^{-A(\sigma_e(k) - \sigma_n)}} - 0.5 \right) \quad (64)$$

Chapter 4: Results and Discussions

In this chapter adaptive algorithms are applied to antenna arrays with different geometries and number of elements antenna arrays, antenna arrays patterns for each algorithm are simulated and compared with each other. Characteristics of each adaptive algorithm have been compared in terms of convergence speed, steady state error, sensitivity to the noise and computational complexity.

4.1 Linear Array for Non-Sparse Algorithms

In this section, a linear array is used to evaluate non sparsity aware algorithms with different number of elements. The array receives five narrowband signals, a desired signal and four interference signals from the azimuth of 35, 50, 10, -30 and -45 respectively, and the spacing between array elements is set to be $\lambda/2$.

The parameters of the adaptive algorithms are set as the following, the step size for both LMS and NLMS are 3×10^{-3} and 1.2 respectively, and the RLS forgetting factor is 0.9 and δ is 0.01, and the combined NLMS parameters are set as $\mu_1 = 0.9$, $\mu_2 = 1.7$ and $\mu_\alpha = 1$, where the parameters of the VSS-NLMS are selected as follows, $\mu = 1.9$, $\beta_g = 0.4$, $\alpha = 5$, $\gamma = 0.8$ and $m = 3$.

For an excellent signal to interference noise ratio (SINR) of 30 dB, Figures 4.1 to 4.3 show the normalized array gain for the LMS, NLMS, RLS, SMI, LMS with SMI weights initialization, combined NLMS and the proposed VSS-NLMS, using 8, 16 and 21 elements respectively.

It can be observed that the RLS, SMI and LMS with SMI weights initialization show the deepest nulls, on the other hand, they have the highest side lobe levels

(SSL's). The LMS, NLMS, combined NLMS and VSS-NLMS introduce lower SSL's, where the VSS-NLMS shows the lowest SSL's. Both NLMS and combined NLMS have deeper nulls than LMS and VSS-NLMS. However, the LMS and VSS-NLMS show an improved performance when using higher number of elements as in Figure 4.3.

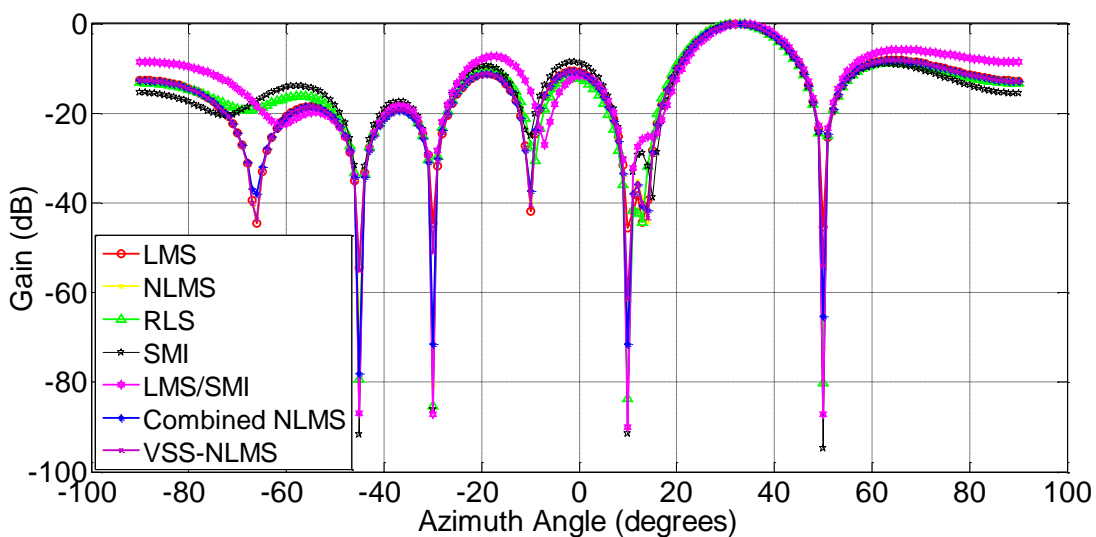


Figure 4.1: Linear array Normalized gain for 8 elements antenna array (SINR=30).

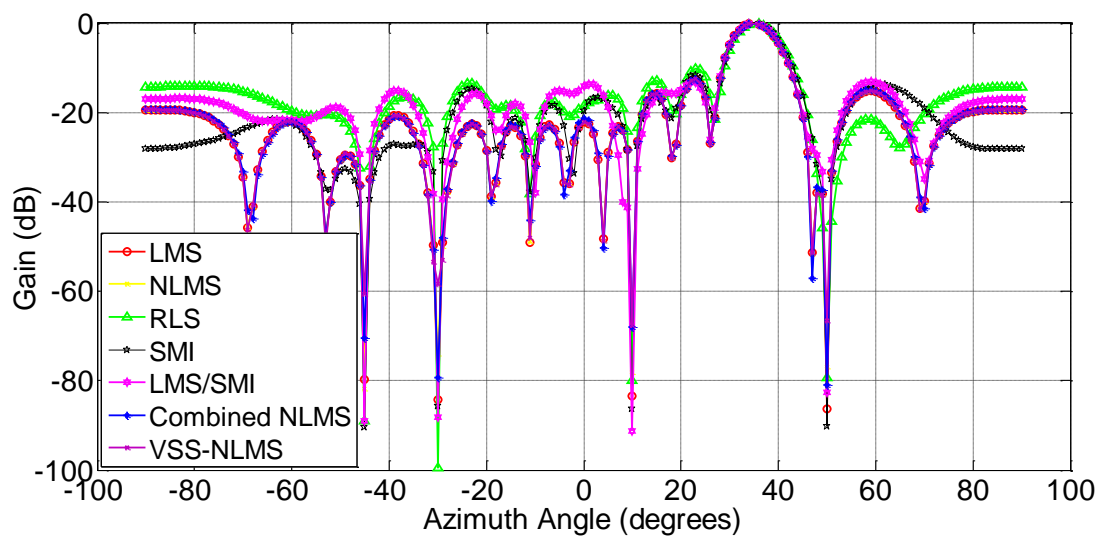


Figure 4.2: Linear array Normalized gain for 16 elements antenna array (SINR=30).

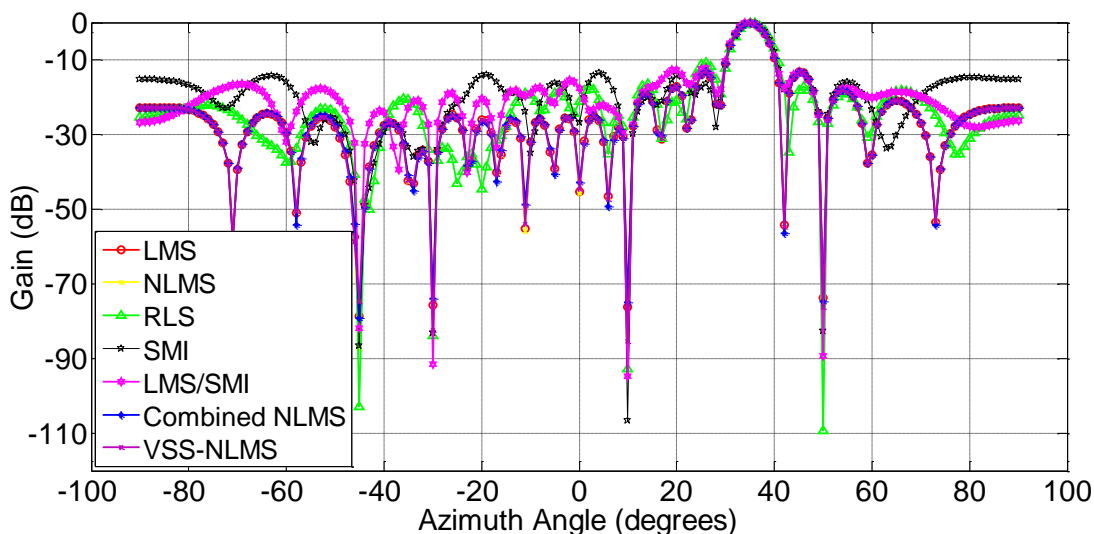


Figure 4.3: Linear array Normalized gain for 21 elements antenna array (SINR=30).

The resulting MSE's versus iterations for the LMS, NLMS, RLS, LMS with SMI initialization, combined NLMS and VSS-NLMS algorithms with different array sizes are shown Figures in 4.4 to 4.9, and Figure 4.10 shows the MSE of the SMI algorithm for each block of data.

It is noted that the SMI, RLS, and LMS with SMI weights initialization algorithms have faster convergence rates compared to the LMS, NLMS, and combined NLMS algorithms. However, the combined NLMS algorithm shows the lowest steady state error, RLS algorithm also shows very good performance in terms of the steady state error, whereas the LMS algorithm has the worst values. The proposed VSS-NLMS algorithm show similar behavior to the combined NLMS algorithm. Again, the performance of LMS shows improvement when using higher number of elements, whereas SMI gives higher MSE with larger array size. Compared to the LMS algorithm, the LMS/SMI algorithm MSE steps to the optimum value due to SMI

weights initialization, whereas in the combined NLMS algorithm and VSS-NLMS, an improved convergence rates are achieved compared to the NLMS algorithm.

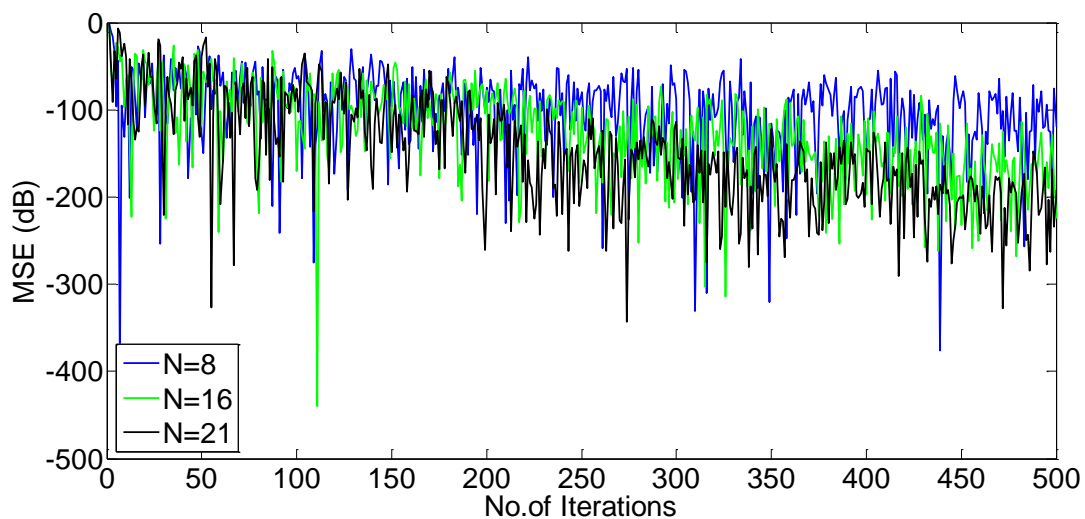


Figure 4.4: MSE versus iterations for LMS with 8, 16 and 21 elements linear arrays (SINR=30).

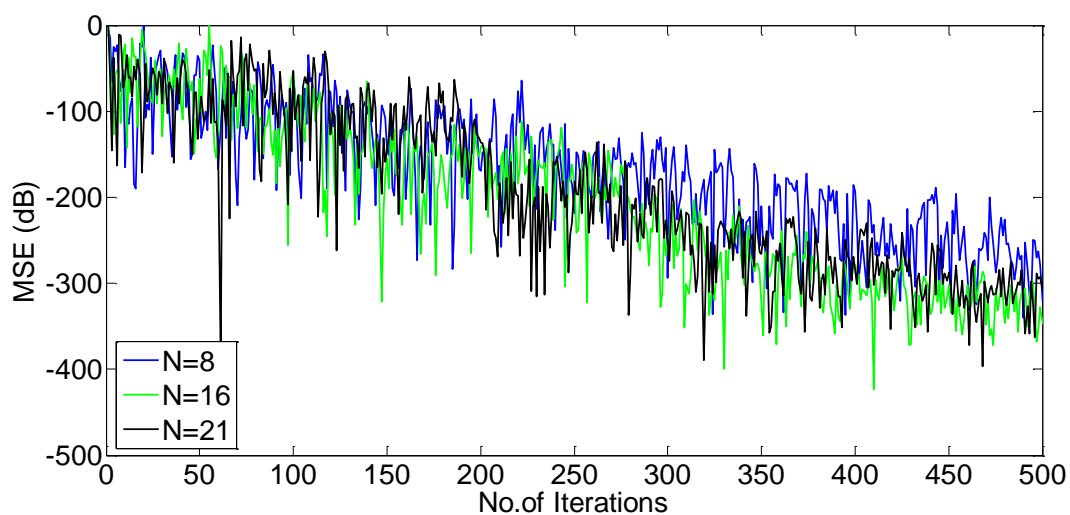


Figure 4.5: MSE versus iterations for NLMS with 8, 16 and 21 elements linear arrays (SINR=30).

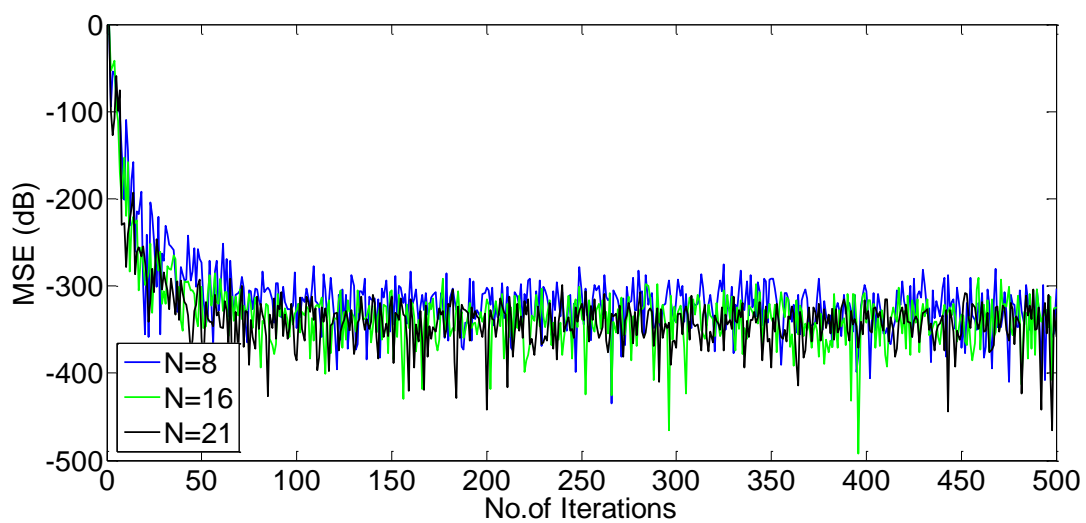


Figure 4.6: MSE versus iterations for RLS with 8, 16 and 21 elements linear arrays (SINR=30).

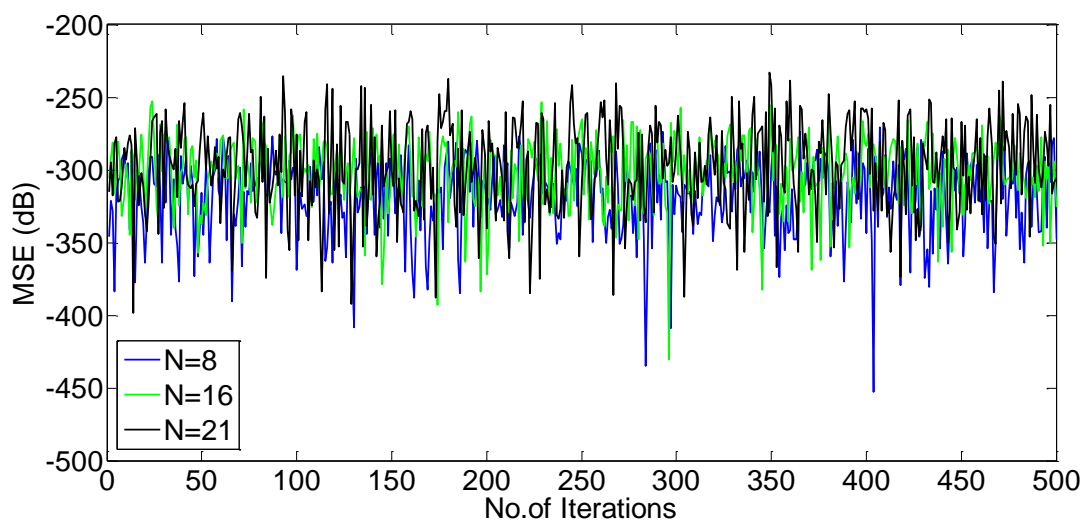


Figure 4.7: MSE versus iterations for LMS/SMI with 8, 16 and 21 elements linear arrays (SINR=30).

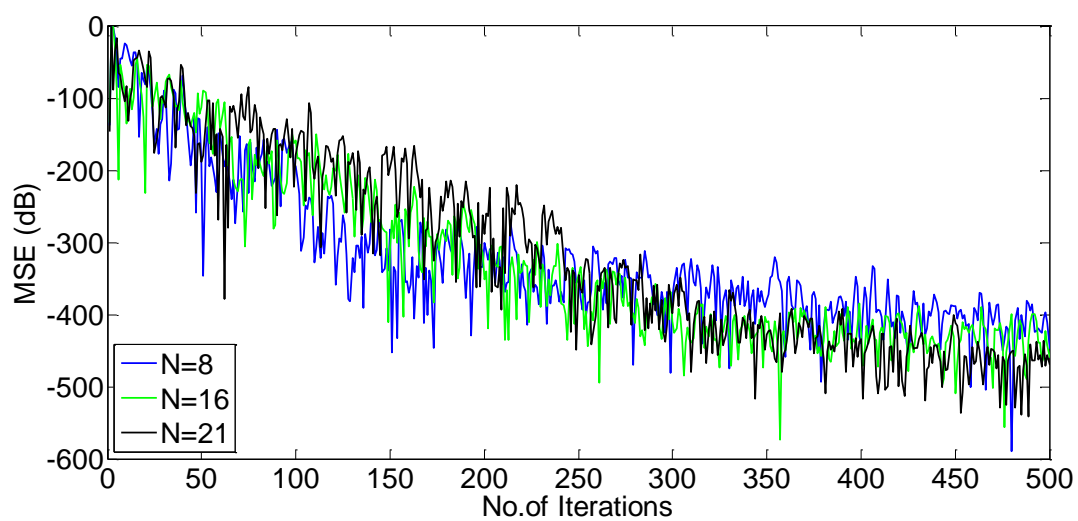


Figure 4.8: MSE versus iterations for Combined NLMS with 8, 16 and 21 elements linear arrays (SINR=30).

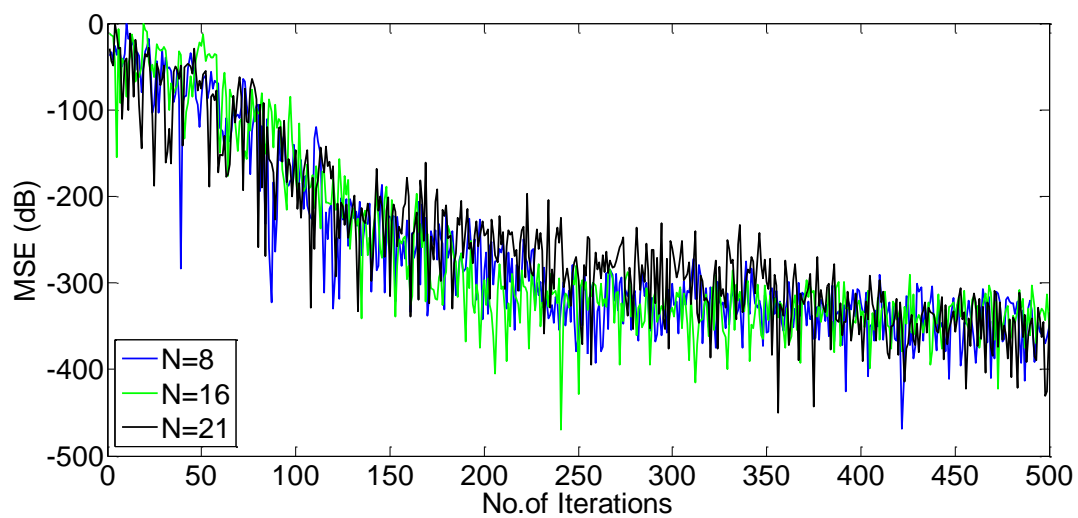


Figure 4.9: MSE versus iterations for VSS-NLMS with 8, 16 and 21 elements linear arrays (SINR=30).

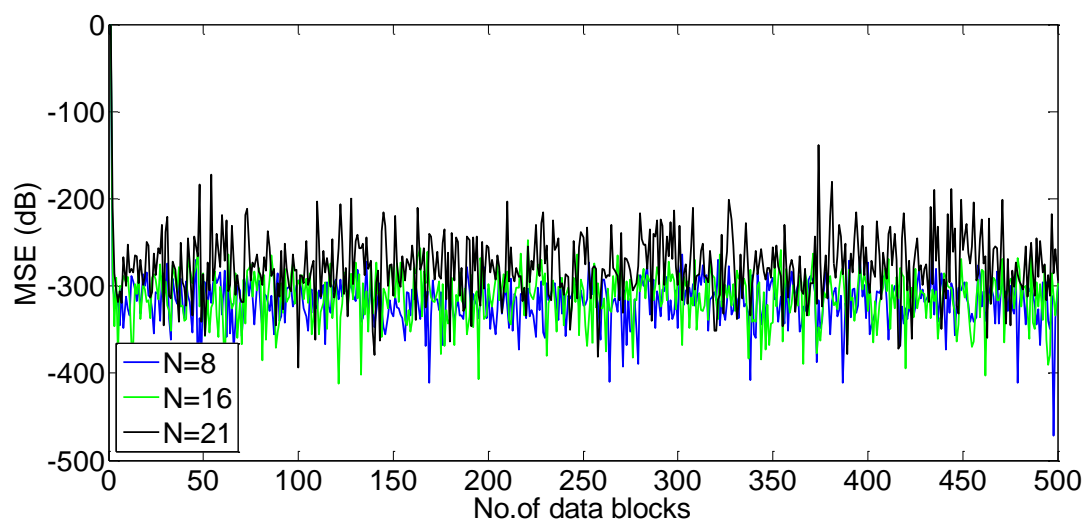


Figure 4.10: MSE versus iterations for SMI with 8, 16 and 21 elements linear arrays (SINR=30).

Using the same parameters values with a signal to interference noise ratio (SINR) of 10 dB, Figures 4.11 to 4.13 show the normalized array gain for the LMS, NLMS, RLS, SMI, LMS with SMI weights initialization, combined NLMS algorithms and the proposed VSS-NLMS, using 8, 16 and 21 elements respectively. Simulation results show that for high noise environment, that LMS, NLMS, RLS and VSS-NLMS show better robustness against noise than SMI, LMS with SMI weights initialization and combined NLMS algorithms which show poor behavior in terms of SLL and nulls depth. It can be also noticed that SMI and LMS with SMI weights initialization with higher number of elements do not achieve accurate beam steering.

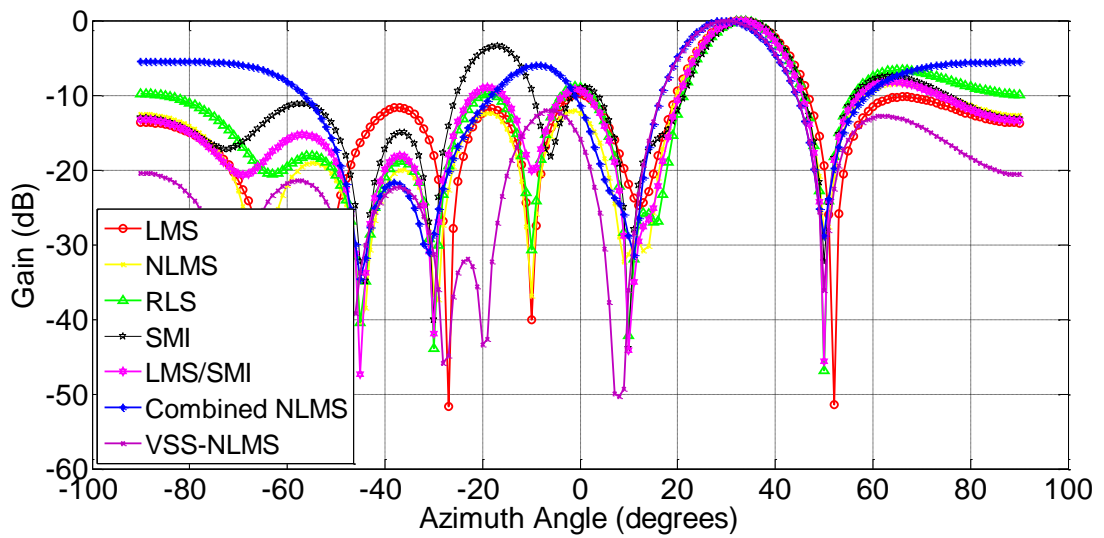


Figure 4.11: Linear array Normalized gain for 8 elements antenna array (SINR=10).

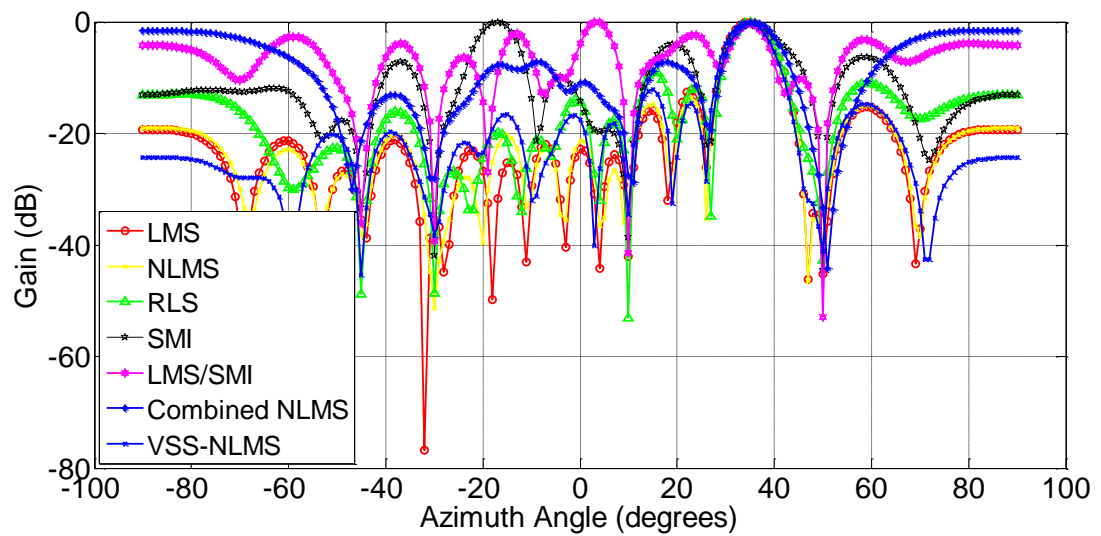


Figure 4.12: Linear array Normalized gain for 16 elements antenna array (SINR=10).

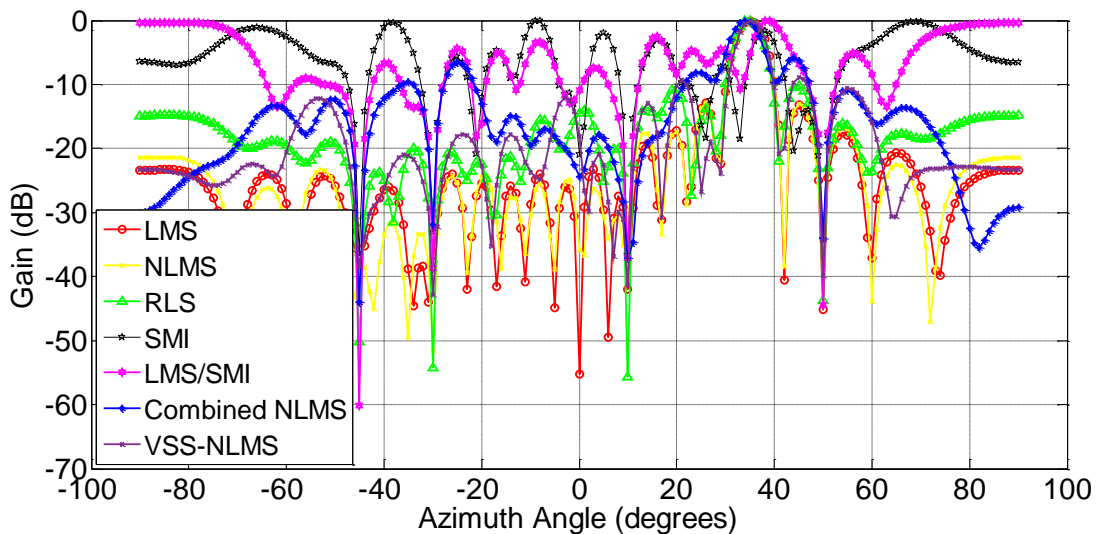


Figure 4.13: Linear array Normalized gain for 21 elements antenna array (SINR=10).

It can be observed from Figures 4.14 to 4.20 which show the MSE for all simulated algorithms with different array sizes for high noise environment, that all algorithms show a similar behavior in terms of convergence when compared to the lower noise scenario. However, the steady state error values are getting relatively higher.

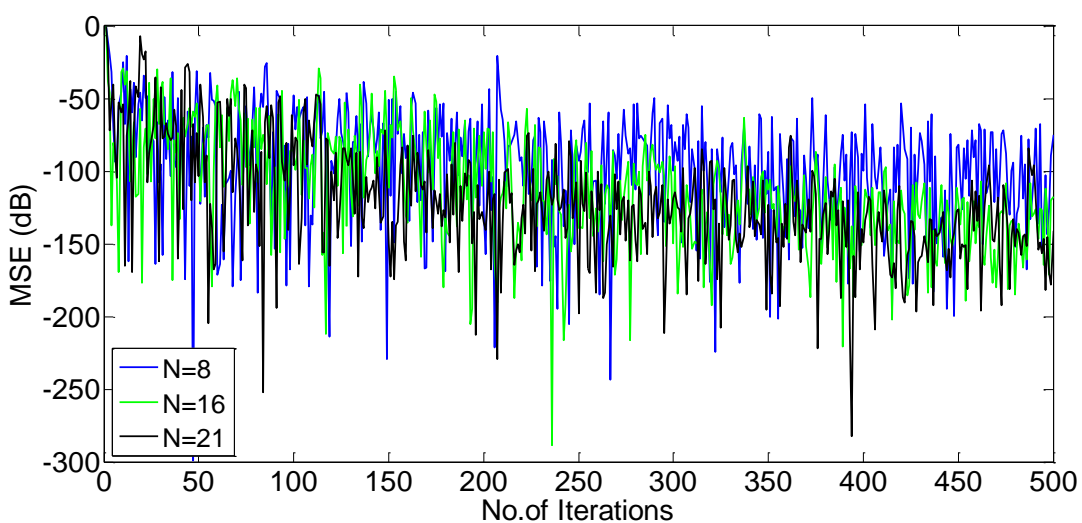


Figure 4.14: MSE versus iterations for LMS with 8, 16 and 21 elements linear arrays (SINR=10).

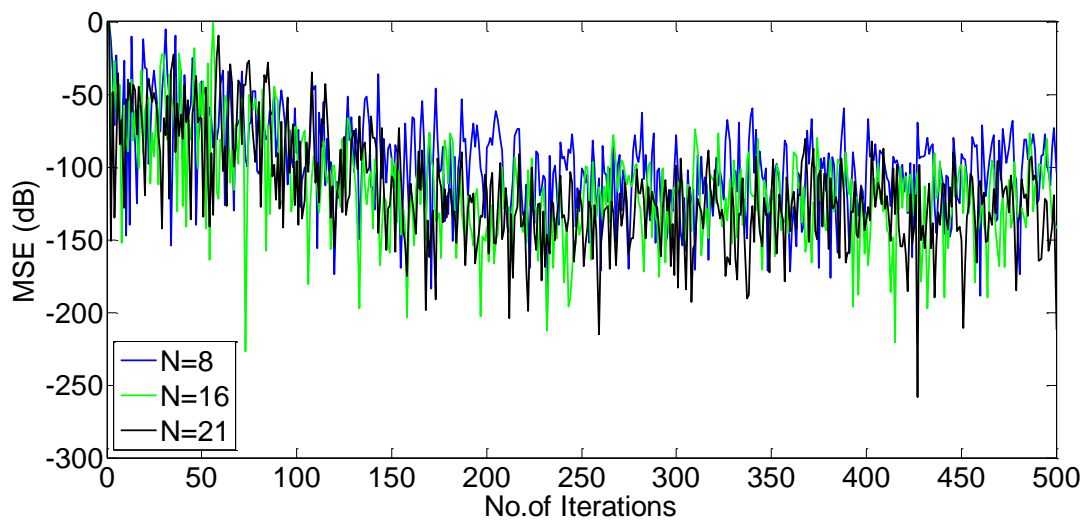


Figure 4.15: MSE versus iterations for NLMS with 8, 16 and 21 elements linear arrays (SINR=10).

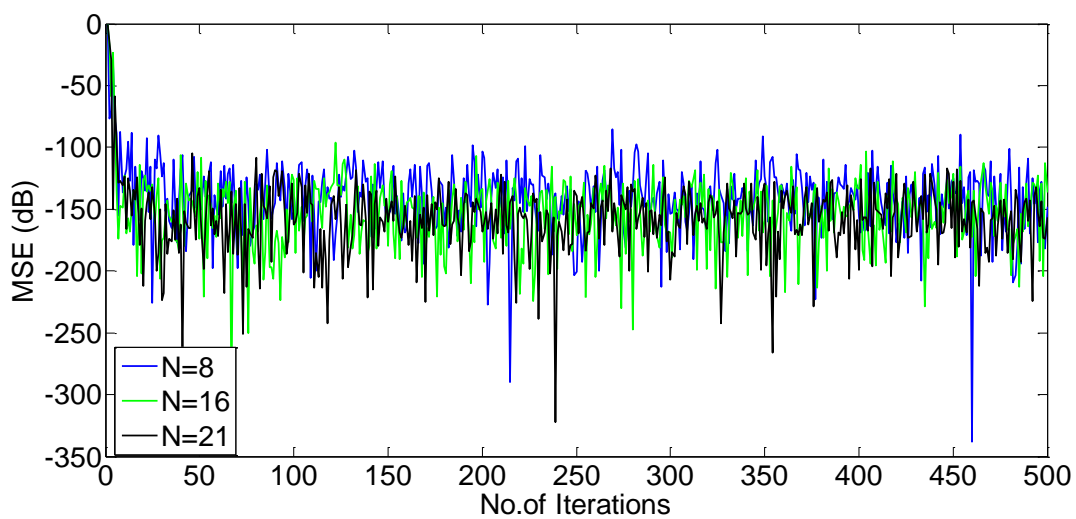


Figure 4.16: MSE versus iterations for RLS with 8, 16 and 21 elements linear arrays (SINR=10).

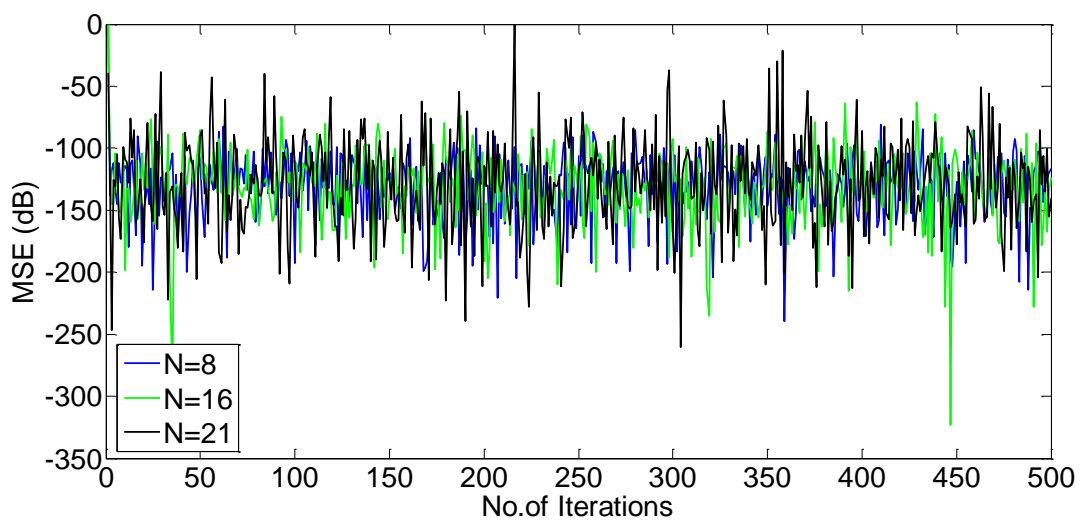


Figure 4.17: MSE versus iterations for SMI with 8, 16 and 21 elements linear arrays (SINR=10).

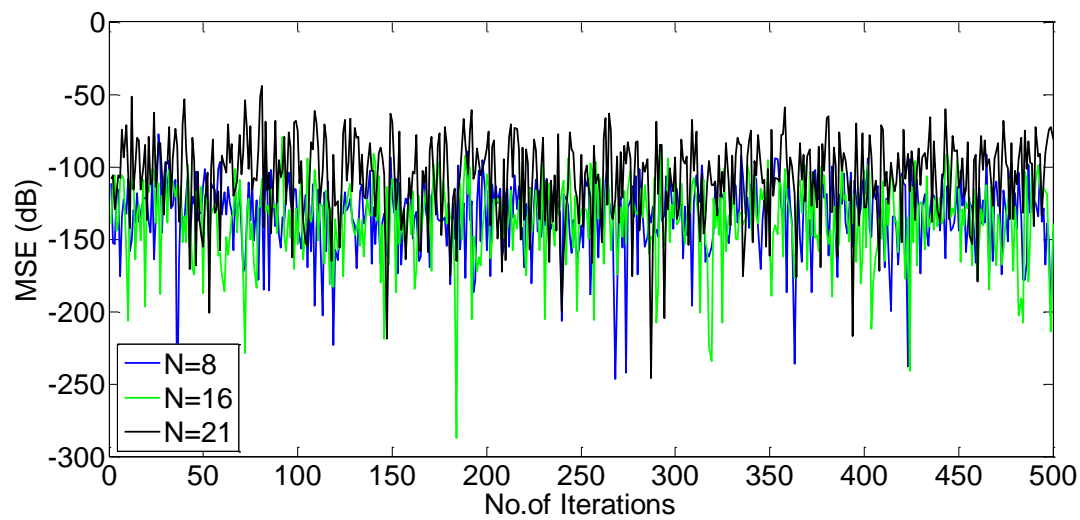


Figure 4.18: MSE versus iterations for LMS/SMI with 8, 16 and 21 elements linear arrays (SINR=10).

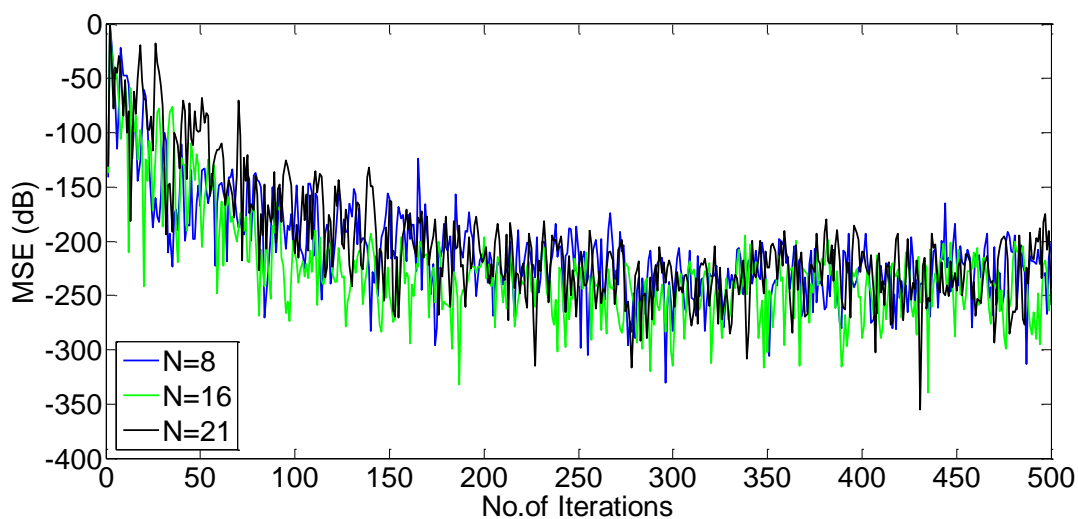


Figure 4.19: MSE versus iterations for combined NLMS with 8, 16 and 21 elements linear arrays (SINR=10).

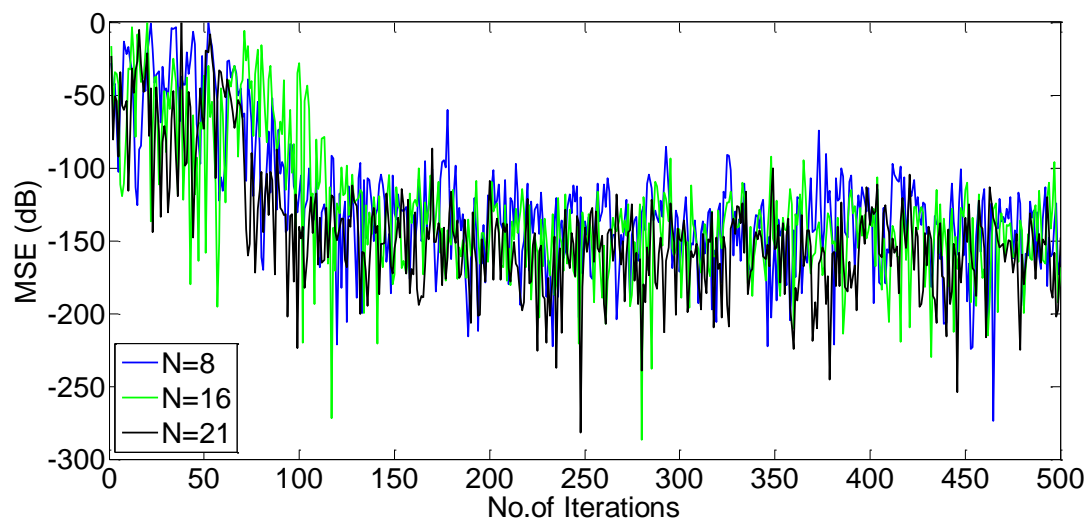


Figure 4.20: MSE versus iterations for VSS-NLMS with 8, 16 and 21 elements linear arrays (SINR=10).

4.2 Rectangular Array for Non-Sparse Algorithms

In this section, rectangular arrays with different sizes are used to evaluate the LMS, NLMS, RLS, SMI, LMS with SMI weights initialization, combined NLMS and the proposed VSS-NLMS algorithms using the similar algorithms parameters' values

used in the previous section. The array receives five narrowband signals, a desired signal and four interference signals from the azimuth of 35, 50, 10, -30 and -45 respectively, with SINR of 30 dB and the spacing between array elements is set to be $\lambda/2$.

The normalized array gain for the LMS, NLMS, RLS, SMI, LMS with SMI weights initialization, combined NLMS and the proposed VSS-NLMS, using $N \times N$ rectangular array with $N = 8$ and 16 are shown in Figures 4.21 and 4.22 respectively. Algorithms resulting array patterns show similar characteristics compared to the linear array patterns discussed in the previous section, with slightly deeper nulls in the rectangular array pattern. However, the convergence speed and the mean square error are significantly improved as illustrated in Figures 4.23 to 4.29.

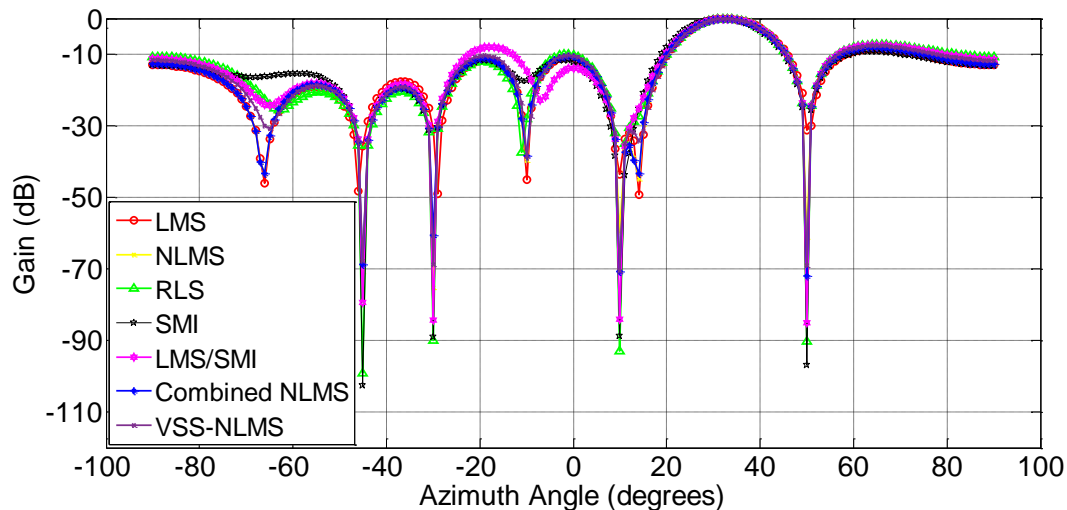


Figure 4.21: Normalized gain for 8×8 rectangular antenna array.

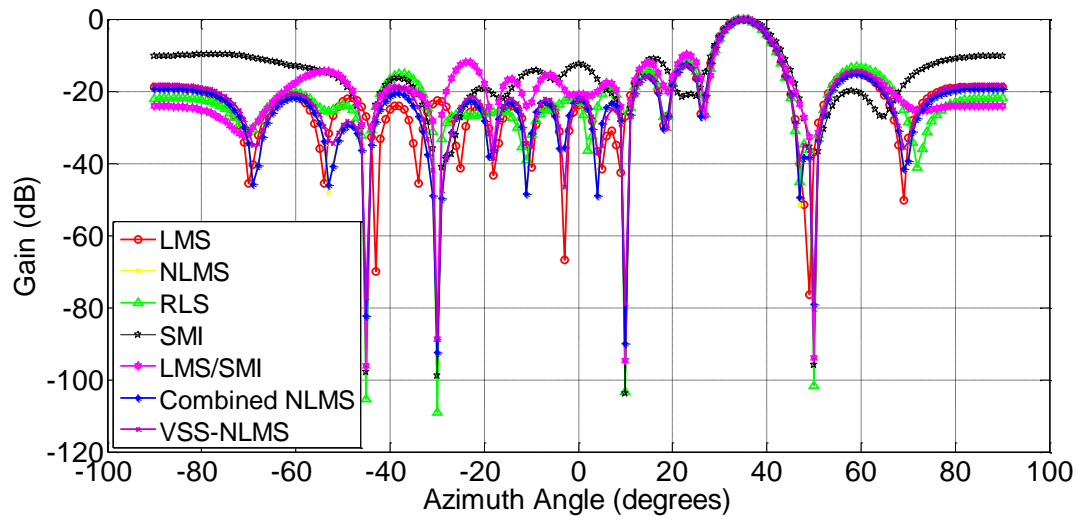


Figure 4.22: Normalized gain for 16×16 rectangular antenna array.

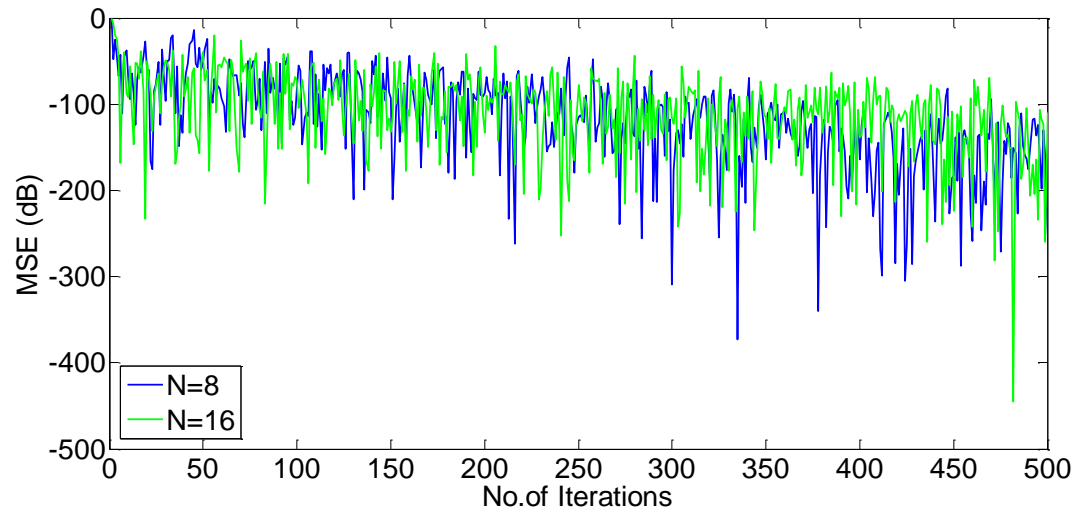


Figure 4.23: MSE versus iterations for LMS for different sizes rectangular arrays.

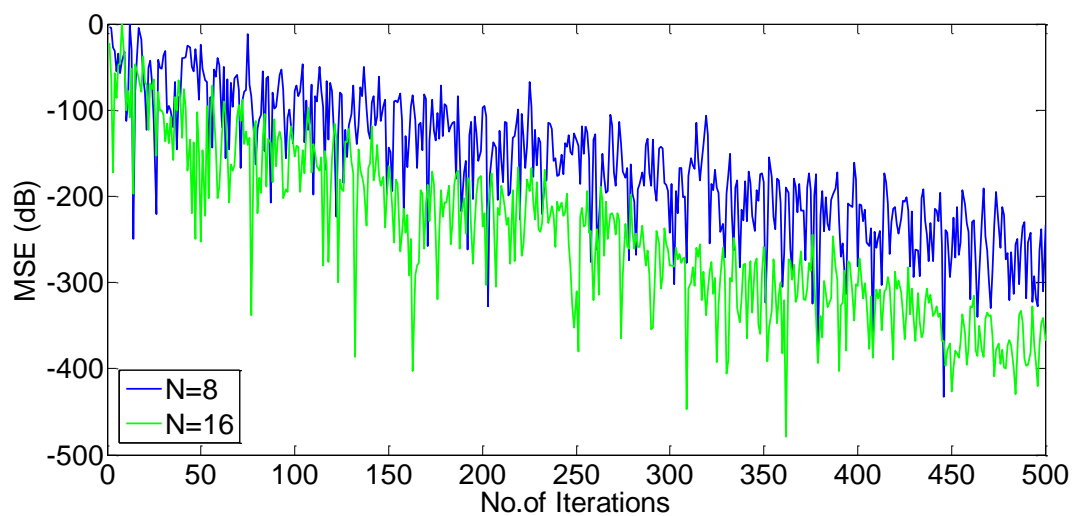


Figure 4.24: MSE versus iterations for NLMS for different sizes rectangular arrays.

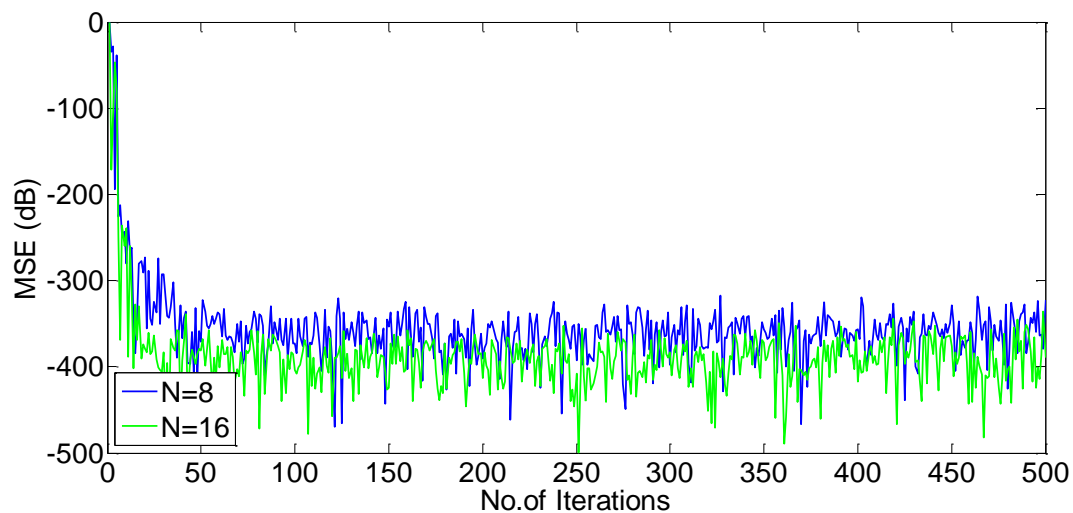


Figure 4.25: MSE versus iterations for RLS for different sizes rectangular arrays.

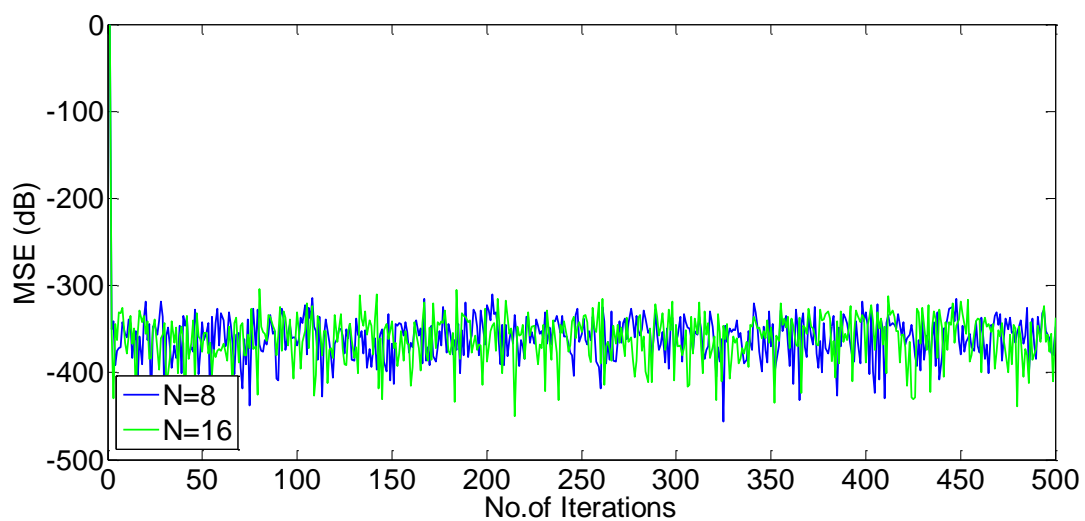


Figure 4.26: MSE versus iterations for SMI for different sizes rectangular arrays.

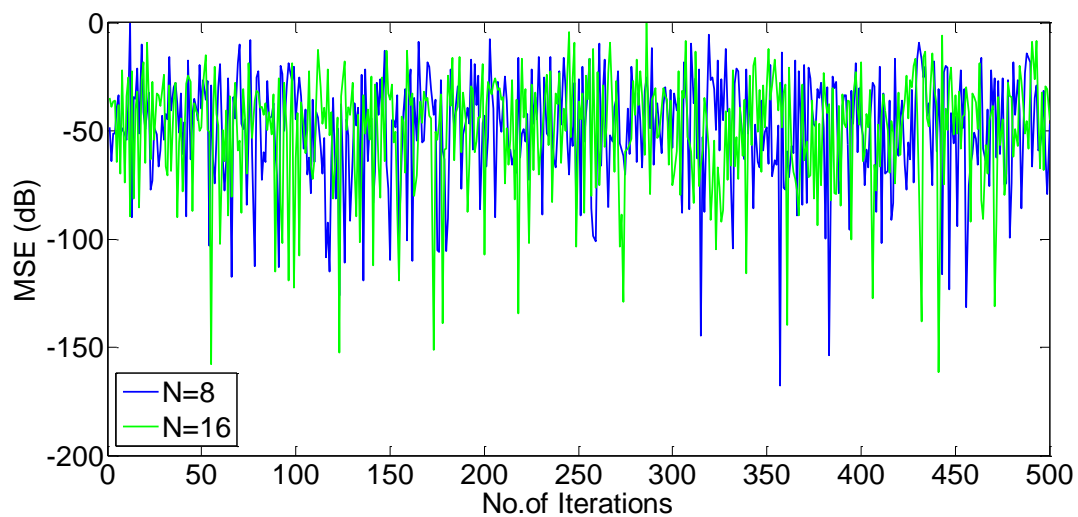


Figure 4.27: MSE versus iterations for LMS/SMI for different sizes rectangular arrays.

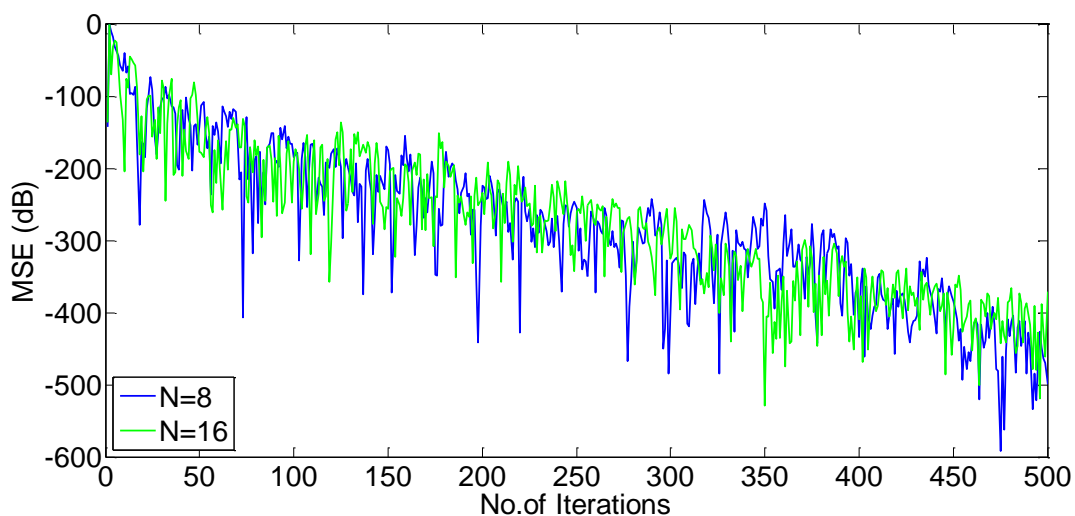


Figure 4.28: MSE versus iterations for combined NLMS for different sizes rectangular arrays.

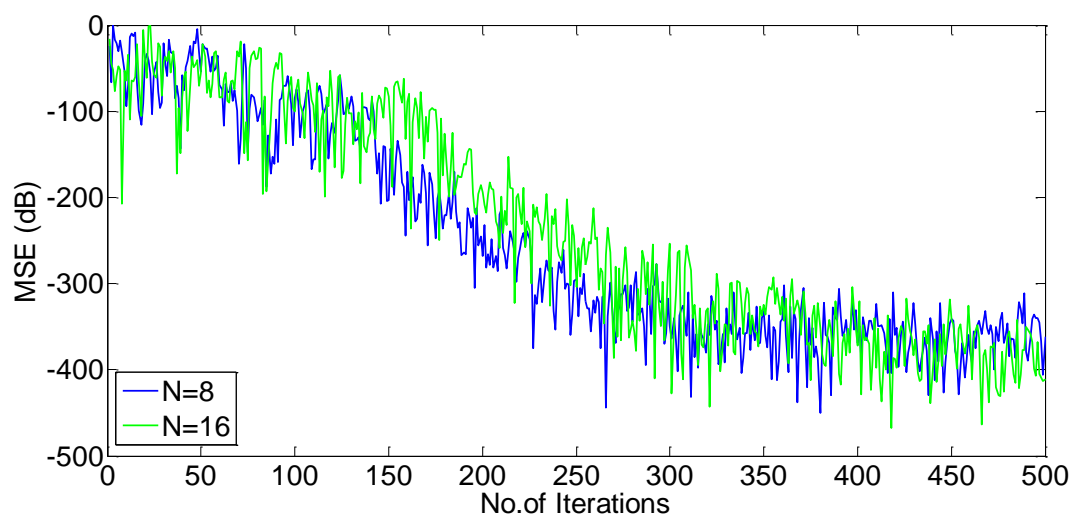


Figure 4.29: MSE versus iterations for VSS-NLMS for different sizes rectangular arrays.

4.3 Sparsity Aware Adaptive Algorithms

In this section a linear array is used to evaluate sparsity aware algorithms, different array sizes are simulated using PNLMS and LP-PNLMS. In addition to that a variable step size applied to both algorithms. Similar to the previous sections the

array receives five narrowband signals, a desired signal and four interference signals from the azimuth of 35, 50, 10, -30 and -45 respectively, and the spacing between array elements is set to be $\lambda/2$.

Parameters for the PNLMS are selected as follows $\mu_{PNLMS} = 1.2$, $\delta_{PNLMS} = \frac{\sigma_s}{N}$, $\delta_P = 0.01$, and $\rho_g = \frac{5}{N}$, where the LP-PNLMS parameters are set as $p = 0.85$, $\varepsilon_{lp} = \frac{\delta_x^2}{N}$, $\mu_{lp} = 1.2$ and $\gamma_{lp} = 3 \times 10^{-7}$.

The normalized array gain for 8, 16 and 21 elements are shown in Figures 4.30 to 4.32 respectively, and Figures 3.33 to 3.36 show the resulting MSE for each algorithm with different array sizes. Each of PNLMS, LP-PNLMS, VSS-PNLMS and VSSLP-PNLMS show very close results in terms of array pattern characteristics. On the other hand, the MSE behavior show faster convergence for both LP-PNLMS and VSSLP-PNLMS. Moreover, no significant improvement over the of PNLMS, LP-PNLMS is noticed when using variable step sizes. However, in general sparse algorithms are showing improved performance in terms of convergence rates over the non-sparsity aware algorithms.

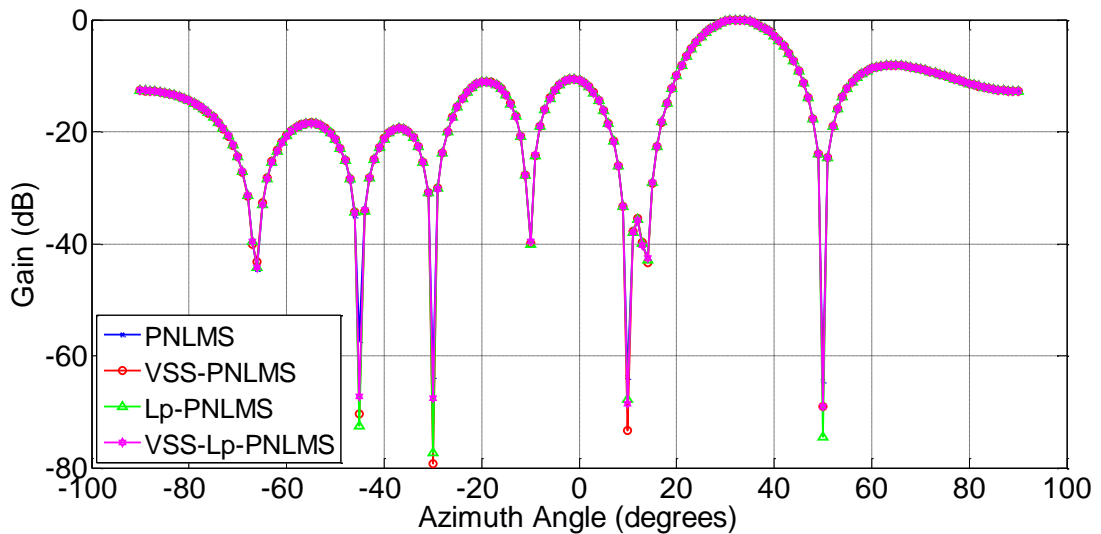


Figure 4.30: Sparse algorithms normalized gain for 8 elements antenna array.

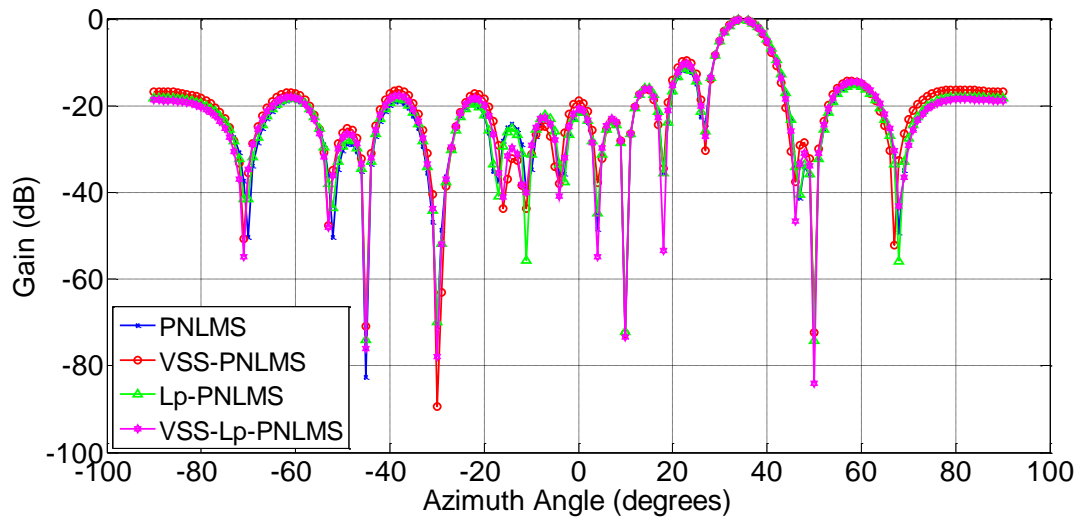


Figure 4.31: Sparse algorithms normalized gain for 16 elements antenna array.

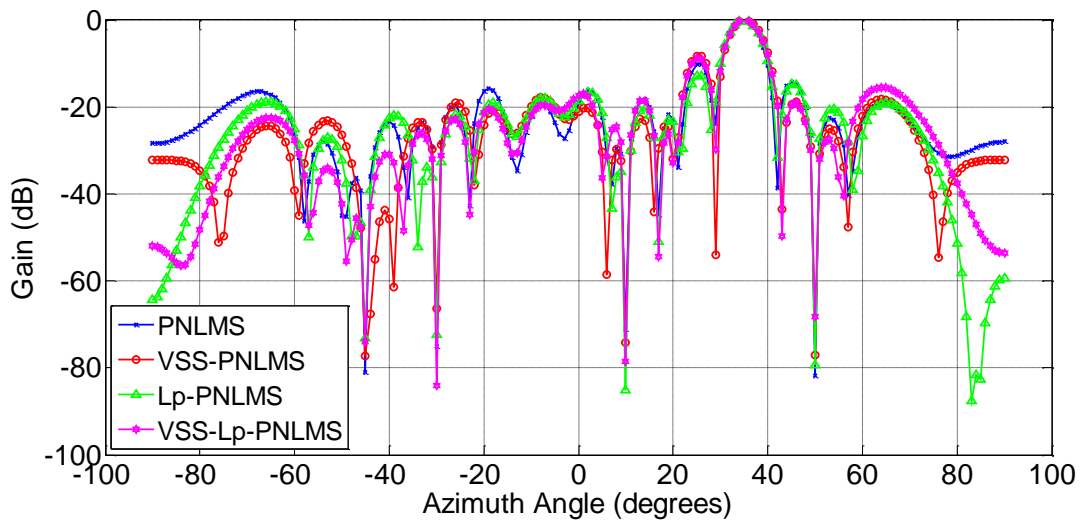


Figure 4.32: Sparse algorithms normalized gain for 21 elements antenna array.

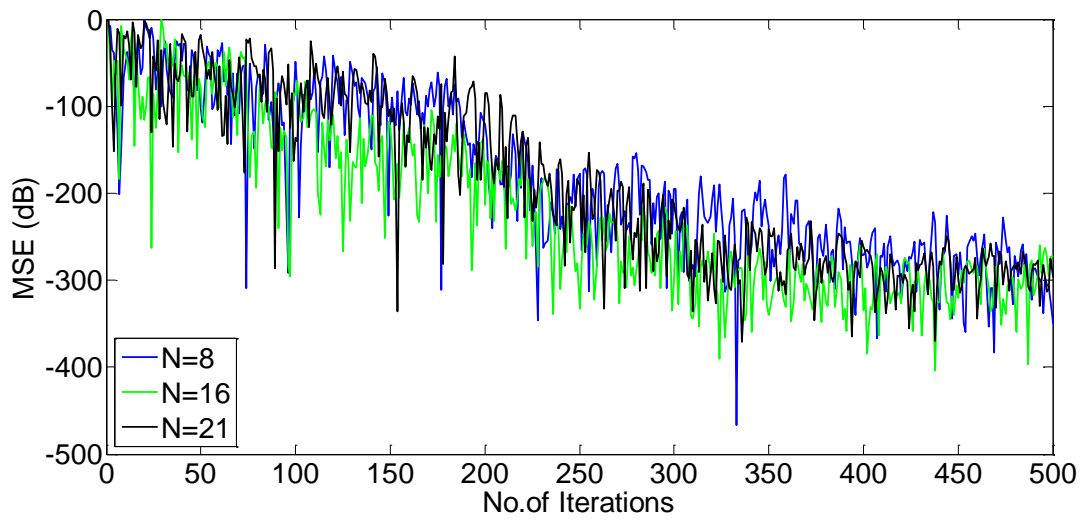


Figure 4.33: MSE versus iterations for PNLMS with 8, 16 and 21 elements.

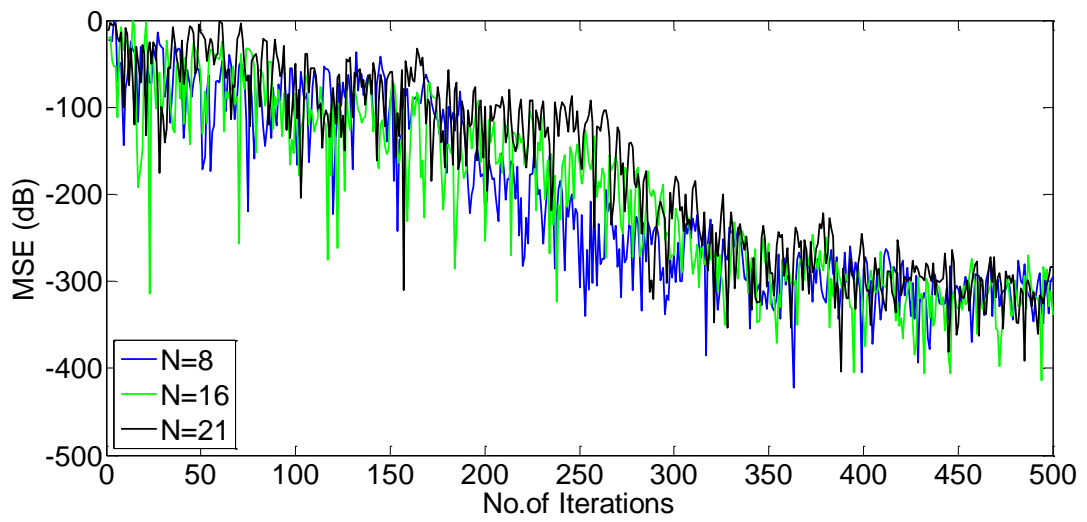


Figure 4.34: MSE versus iterations for VSS-PNLMS with 8, 16 and 21 elements.

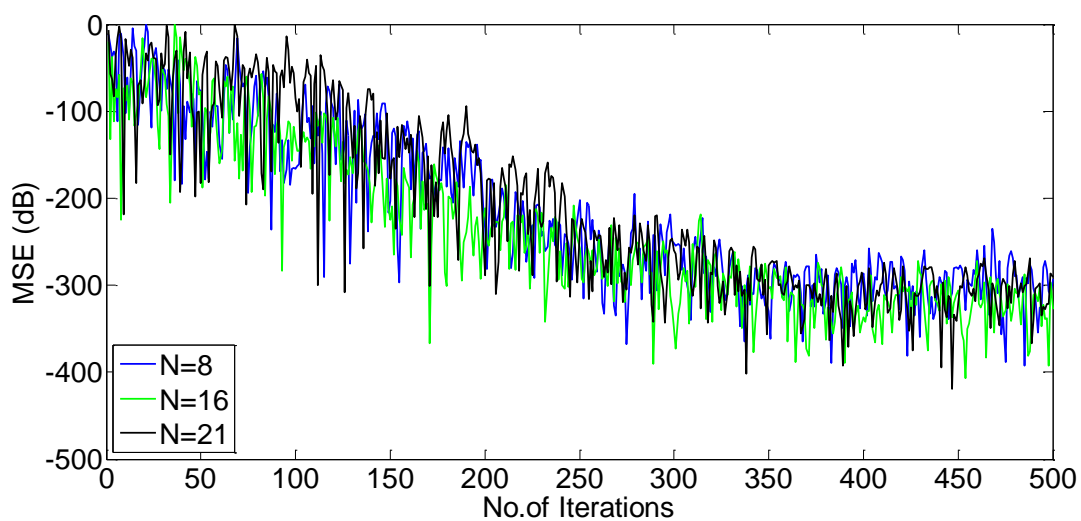


Figure 4.35: MSE versus iterations for Lp-PNLMS with 8, 16 and 21 elements.

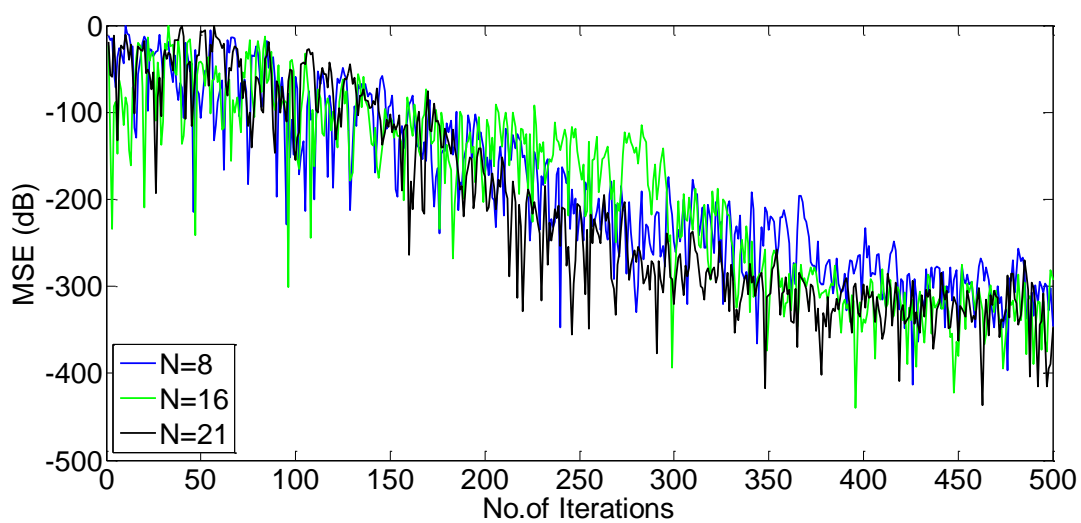


Figure 4.36: MSE versus iterations for VSS-Lp-PNLMS with 8, 16 and 21 elements.

Chapter 5: Conclusion and Future Work

In this thesis, various beam forming algorithms such as: LMS, NLMS, RLS, SMI, LMS with SMI weights initialization, and combined NLMS filter with a variable mixing parameter have been presented and analyzed. Moreover, VSS-NLMS algorithm is also proposed to achieve an improved convergence and maintain system robustness. Simulation results for linear and rectangular arrays, show that each algorithm can achieve the desired performance, to steer the beam towards and signal of interest and forming nulls towards the interference sources. However, each algorithm has advantages and weaknesses. In the terms of convergence speed and nulls depth RLS and SMI show better performance, whereas LMS, NLMS are simpler and give lower SSL. However, it can be observed that some of these weaknesses can be reduced by using combined algorithms, where LMS/SMI and combined NLMS filters have an improved convergence speed compared to the LMS and NLMS algorithms with an acceptable increase in the computation complexity. It can be noticed the proposed VSS_NLMS achieves a similar performance to the combined NLMS filters. In addition, to fully utilize the sparse characteristics of the system and overcome restriction related consumption and processing complexity of the large arrays, sparsity aware algorithms such as the PNLMS and LP-PNLMS are also studied and analyzed. Further, to increase system robustness and achieved an improved convergence, a variable step-size is also proposed for both of these algorithms. As expected, sparse algorithms achieve an improved performance over the non-sparsity aware algorithms in terms of convergence speed and the steady state mean square error.

Many of the methods and algorithms introduced in this thesis have potential applications in other systems outside the scope of this thesis, and as a future work,

different ideas can be extended from this work for the autonomous network, where the network uses AI-based algorithm for audit, self-healing and even for predicting any possible degradation in the performance. Smart antennas can automatically recognize scenarios and configure beams through AI-based operations and maintenance. The antennas change beams with various widths and directions based on the beam configurations, improving the user experience.

References

- [1] Al-Falahy, N., & Alani, O. (2017). Technologies for 5G Networks: Challenges and Opportunities. *IT Professional*, 19(1), 12-20.
- [2] Niu, Y., Li, Y., Jin, D., Su, L. & Vasilakos, A. (2015). A survey of millimeter wave communications (mmWave) for 5G: opportunities and challenges. *Wireless Networks*, 21(8), 2657-2676.
- [3] Ghosh, S. & Sen, D. (2019). An Inclusive Survey on Array Antenna Design for Millimeter-Wave Communications. *IEEE Access*, 7, 83137-83161.
- [4] Balanis, C., & Ioannides, P. (2007). Introduction to smart antennas. *Synthesis Lectures on Antennas*, 5, 1-179. Retrieved from:
<https://doi.org/10.2200/S00079ED1V01Y200612ANT005>
- [5] Balanis, C. (2005). Antenna theory: analysis and design. Third edition, New Jersey, USA: John Wiley & Sons.
- [6] Van Veen, B. & Buckley, K. (1988). Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine*, 5(2), 4-24.
- [7] Gross, F. (2005). Smart Antenna for Wireless Communication. Second edition, New York, USA: McGraw-Hill.
- [8] Patel, D., Makwana, B., & Parmar, P. (2016). Comparative analysis of adaptive beamforming algorithm LMS, SMI and RLS for ULA smart antenna. *2016 International Conference on Communication and Signal Processing (ICCSP)* (pp. 1029-1033). Melmaruvathur, India, 6-8 April 2016.

- [9] Saxena, P. & Kothari, A. (2014). Performance Analysis of Adaptive Beamforming Algorithms for Smart Antennas. *IERI Procedia*, 10, 131-137.
- [10] Al-Sadoon, M., Abd-Alhameed, R., Elfergani, I., Noras, J., Rodriguez, J., & Jones, S. (2016). Weight Optimization for Adaptive Antenna Arrays Using LMS and SMI Algorithms. *WSEAS Transactions on Communications*, 15, 206-214.
- [11] Shubair, R., Jimaa, S., & Omar, A. (2009). Enhanced adaptive beamforming using LMMN algorithm with SMI initialization. *2009 IEEE Antennas and Propagation Society International Symposium*, Charleston, SC, USA, 1-5 June 2009. (Accessed 24/7/2009).
- [12] Khalaf, A., El-Daly, A., & Hamed, H. (2018). A Hybrid NLMS/RLS Algorithm to Enhance the Beamforming Process of Smart Antenna Systems. *Journal of Telecommunication, Electronics and Computer Engineering*, 10, 15-22.
- [13] Lu, L., Zhao, H., He, Z., & Chen, B. (2015). A novel sign adaptation scheme for convex combination of two adaptive filters. *AEU-International Journal of Electronics and Communications*, 69(11), 1590-1598.
- [14] Huang, F., Zhang, J., & Zhang, S. (2016). Combined-Step-Size Affine Projection Sign Algorithm for Robust Adaptive Filtering in Impulsive Interference Environments. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 63(5), 493-497.
- [15] Candido, R., Silva, M., & Nascimento, V. (2010). Transient and Steady-State Analysis of the Affine Combination of Two Adaptive Filters. *IEEE Transactions on Signal Processing*, 58(8), 4064-4078.

- [16] Tan, J. & Ouyang, J. (1997). A novel variable step-size LMS adaptive filtering algorithm based on sigmoid function. *Journal of Data Acquisition & Processing*, 12(3), 171-174.
- [17] Aboulnasr T., & Mayyas K. (1997). A robust variable step-size LMS-type algorithm analysis and simulation. *IEEE Transactions on Signal Processing*, 45(3), 631-639.
- [18] Zhu, Y., Li, Y., Guan, S. & Chen, Q. (2012). A Novel Variable Step-Size NLMS Algorithm and Its Analysis. *Procedia Engineering*, 29, 1181-1185.
- [19] Chen Y., Gu Y., Hero A. (2009) Sparse LMS for system identification. *Proceedings of IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP'09)* (pp. 3125-3128), Taipei, Taiwan, 19-24 April 2009.
- [20] Taheri, O., & Vorobyov, S. (2011). Sparse channel estimation with Lp-norm and reweighted L1-norm penalized least mean squares. *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2864-2867), Prague, Czech Republic, 22-27 May 2011.
- [21] Li, Y., & Hamamura, M. (2015). Zero-attracting variable-step-size least mean square algorithms for adaptive sparse channel estimation. *International Journal of Adaptive Control and Signal Processing*. 29, 1189-1206.
- [22] Shi, W., Li, Y., & Yin, J. (2019). Improved Constraint NLMS Algorithm for Sparse Adaptive Array Beamforming Control Applications. *ACES Journal*, 34 (3), 419-424.

- [23] Shi, W., Li, Y., Sun, L., Yin, J., & Zhao, L. (2019). Norm Constrained Noise-free Algorithm for Sparse Adaptive Array Beamforming. *ACES Journal*, 34 (5), 709-715.
- [24] Paleologu, C., Benesty, J., & Ciochina, S. (2008). A Variable Step-Size Proportionate NLMS Algorithm for Echo Cancellation. *Rev. Roum. Sci. Techn. Électrotechn. et Énerg*, 53, 309-317.
- [25] Li, Y., & Hamamura, M. (2014). An Improved Proportionate Normalized Least-Mean-Square Algorithm for Broadband Multipath Channel Estimation. *The Scientific World Journal*, 2014(8). DOI: 10.1155/2014/572969. Retrieved from: <https://www.hindawi.com/journals/tswj/2014/572969/>
- [26] Das, R. & Chakraborty, M. (2012). A zero attracting proportionate normalized least mean square algorithm. *Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference* (pp. 1-4), Hollywood, CA, USA, 3-6 December 2012.
- [27] Das, R., & Chakraborty, M. (2016). Improving the Performance of the PNLMS Algorithm Using l_1 Norm Regularization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(7), 1280-1290.
- [28] Balanis, C. (1992). Antenna theory: a review. *Proceedings of the IEEE*, 80(1), 7-23.
- [29] Haykin, S. (2002). Adaptive Filter Theory. Fourth edition. Upper Saddle River, New Jersey, USA: Prentice-Hall.

- [30] Van Trees, H. (2002). Detection, Estimation, and Modulation Theory, Part IV: Optimum Array Processing. New York, USA: John Wiley & Sons.