NORMALIZED TIME VARYING MIXED NORM LMS-LMF ADAPTIVE ALGORITHM

BY

MUHAMMAD MOINUDDIN

A Thesis Presented to the

DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

MAY 2001

UMI Number: 1406485

UMI®

Bell & Howell Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by

Muhammad Moinuddin

under the direction of his thesis advisor and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

Thesis Committee

azzedine zerguine

Dr. Zerguine, A. (Chairman)

Dr. Sheikh, A.U.H. (Member)

Dr. Semari, S. A. Al. (Member)

Dr. Al. Baiyat, S. A. (Member)

Dr. Al. Suwailem, U. (Member)

Department Chairman

Dean of Graduate Studies

DEANSHIP OF GRADUATE STUDIES

22-07-2001

Date

Dedicated

to

My beloved Parents

Acknowledgements

In the name of ALLAH, The Most Gracious, Merciful and Beneficent. First and foremost all the praise goes to ALLAH *subhana-wa-ta' ala*, The Almighty, and The Creator of the universe, who granted me the grace and strength to complete this work. Peace and blessing of ALLAH be upon His last Prophet Hazrat Muhammad *sallallaho-alaihe-wassalam.*

I would like to express my profound gratitude and appreciation to my thesis committee chairman Dr. Azzedine Zerguine for his guidance, sincere advice, and continuous support throughout my thesis. Working with him was indeed a wonderful learning experience, which I thoroughly enjoyed. Thanks are also to my committee members Dr. Asrar Ul Haq Sheikh, Dr. S. A. Al. Baiyat, Dr. S. A. Al Semari and Dr. U. Al Suwailem for their cooperation and critical review of the thesis.

I shall ever remain indebted to my parents for their socio-religious guidance they gave to me. Their love and prayers led to this accomplishment. I pray ALLAH (SWT) to forgive their souls and admit them to Al-Jannah. I am also thankful to my brothers and the only sister for their moral support.

Acknowledgement is due to King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia for providing the support and facilities in carrying out this research.

I owe my deep appreciation to my seniors Faisal Ali Shah, Ajmal Khan and my colleague Wasif Naeem for helping me on issues relating to LaTeX and MAT-

LAB. I would also appreciate the help provided by my senior Shafayat Abrar in programming on MATLAB.

Finally, I would like to thank my school, college and university friends in Pakistan. I would also like to thank my friends in KFUPM, Wasif, Atif, Asif, Saad, Ahmar, Fareed, Ajmal, Junaid, Wasiq, Sajid, Aamir, Mahmood, Hamid, and Faisal Ali Shah for providing me a wonderful company. Thanks are also due to Raslan for Arabic abstract of the thesis.

Contents

List of Tables

List of Figures

Nomenclature

Symbols

J_n	Cost Function
$\mathbf{w_n}$	Vector representing weights of the adaptive filter
$\mathbf{w_{opt}}$	Vector representing the weights of the unknown system
$\mathbf{v_n}$	Weight error vector
$\mathbf{x_n}$	Vector representing the input sequence
$\mathbf{d_n}$	Desired response
$\mathbf{R}$	Autocorrelation matrix of the input sequence
$\mathbf{K}_n$	Autocorrelation matrix of the weight error vector
ξ_n	Additive noise to the system
e_n	Error between the output from the unknown system and that of from the adaptive filter
μ	Step-size used in the adaptation of the filter weights
α	Fixed mixing parameter
α_n	Time varying mixing parameter
p_n	Time varying parameter that gives the time averaged estimate

	of the autocorrelation between e_n and e_{n-1}
δ, β, γ	Constants used in updating α_n
$\mathbf{Q}$	Orthonormal matrix whose ith column is the eigenvector of $\mathbf{K}_n$
$\mathbf{G}_n$	Diagonalized autocorrelation matrix of the weight error vector
e_{an}	A priori estimation error
e_{pn}	A posteriori estimation error
$\hat{\mu}_n$	constant representing reciprocal of $\| \mathbf{x}_n \|^2$
$f(e_n)$	General function of the output estimation error
$\mathbf{w}_n^o$	Vector representing time varying weights of the unknown system
$\mathbf{w}^o$	Vector representing fixed weights of the unknown system
$\mathbf{q}_n$	Zero mean stationary random process which is responsible for random nostationarity in the weights of unknown system
$\mathbf{Q}_n$	Covariance matrix of $\mathbf{q}_n$
Ω	Frequency offset between transmitter and receiver carriers
$\mathbf{z}_n$	Component of the steady state weight error vector that is independent of the offset frequency Ω
$\mathbf{z}$	$E[\mathbf{z}_n]$
ζ	Steady-state excess mean square error
N	Filter length
σ_x^2	Variance of input sequence

σ_w^2	Variance of the noise
ϕ_w	High order moments of the noise

Operators

E[ ]	Expectation operator
tr( )	Trace operator
Re[ ]	Real part of [ ]
lim	Limit operator

Abbreviations

SNR	Signal to Noise Ratio
LMS	Least Mean Squares
LMF	Least Mean Fourth
NLMS	Normalized Least Mean Square
NLMF	Normalized Least Mean Fourth
prop	Proposed Agorithm
MSE	Mean Square Error
EMSE	Excess Mean Square Error
RLS	Recursive Least Squares
FRLS	Fast Recursive Least Squares

FAEST	Fast a posteriori error sequential technique
FTF	Fast transversal filter
ISI	Inter Symbol Interference

THESIS ABSTRACT

Name: MUHAMMAD MOINUDDIN

Title: NORMALIZED TIME VARYING MIXED NORM LMS-LMF ADAPTIVE ALGORITHM

Degree: MASTER OF SCIENCE

Major Field: ELECTRICAL ENGINEERING

Date of Degree: MAY 2001

The least-mean square (LMS) and least-mean fourth (LMF) algorithms are two important adaptive schemes. They have several advantages and disadvantages. These are combined in one which is named as mixed-norm algorithm in order to utilize the benefits of both the algorithms in which mixing parameter is fixed. Recently a new weighted sum of LMS and LMF algorithm has been proposed in which the mixing parameter is time varying and has the ability to adapt the variations in the environment. Due to its dependence on LMS and LMF algorithms, it is effected by the high eigen value spread of the input sequence. In order to overcome this problem, a normalized time varying mixed norm LMS-LMF algorithm is proposed in this thesis work. Convergence analysis for the mean and the mean square of the proposed algorithm are presented in this work and the bound for the convergence in the mean is also derived. Steady state analysis is carried out and the expressions for the steady-state excess mean-square error are derived for both NLMS and the proposed algorithms. Tracking analysis is also presented in the presence of two type of nonstationarities, random and cyclic. Also, the expressions for the corresponding steady-state excess mean-square error are derived for both algorithms, and then the analytical results are compared with the experimental one. Finally simulation results are presented in the support of theoretical analysis.

Master of Science Degree

King Fahd University of Petroleum and Minerals, Dhahran.
MAY 2001

ملخص الرسالة

ا لا سم: محمد معين الدين

عنوا ن الدراسة: خوارزم متكيف ذو معيار مختلط متغير مع الوقت مطبع.

التخصص: الهندسة الكهربائية

تاريخ التخر ج: مايو 2001

خوارزم متوسط المربعات ذو الدرجة الثانية (LMS) و متوسط المربعات ذو الدرجة الرابعة (LMF), هما خطتان تكيفيتان مهمتان. لديهما بضع العيوب والمميزات. هذان الخوارزمان يمكن أن يجمعا في خوارزم واحد يسمى خوارزم المعيار المخلوط بحيث نستفيد منهما ويكون خلط المتغيرات ثابت. حديثا تم إقتراح خوارزم يعتمد على المجموع المثقل ويكون خلط المتغيرات متغير مع الوقت، ولديه الإمكانية لتكييف التغيرات في البيئة. ونتيجة لاعتماده على خوارزم (LMS) و(LMF)، فهو يتأثر بال eigenvalue للسلسلة المدخلة. لكي نتغلب على هذه المشكلة. نقترح في هذة الرسالة خوارزم متكيف ذو معيار مختلط متغير مع الوقت ومطبع. تم أيضا عرض التحليل التقاربي للمتوسط ومتوسط الدرجة الثانية للخوارزم في هذا العمل و تم إشتقاق حدود التقارب أيضا. تم أيضا تحليل الحالة الثابتة وإشتقاق العبارة الجبرية للخطأ في متوسط مربع الفائض في الحالة الثابتة، لكل من NMLS والخوارزم المقترح. وعرضنا أيضا التحليل المساري في وجود نوعين من اللاثبوتية العشوائية والدورية. وكذلك تم إشتقاق متوسط مربع الفائض في الحالة الثابتة، لكلا الخوارزمين وتم مقارنة النتائج التحليلية بالنتائج المخبرية. وأخيرا تم عرض نتائج المحاكاة فأتت لتدعم النتيجة التحليلية.

درجة الماجستير في العلوم

جامعة الملك فهد للبترول و المعادن

الظهران ,المملكة العربية السعودية

مايو 2001

Chapter 1

Introduction

Adaptive systems are playing a vital role in the development of modern communications. The concept of adaptive filtering constitutes an important part of the statistical signal processing. Whenever there is a requirement to process signals that result from an unknown statistics of an environment, the use of an adaptive filter offers an attractive solution to the problem. Also, adaptive systems proved to be extremely effective in achieving high efficiency, high quality and high reliability of around-the-world ubiquitous telecommunication services. Thus, adaptive filters are successfully applied in such diverse fields as equalization [1], noise cancellation [2], linear prediction [3] and in system identification [4, 5].

The two most widely used algorithms for adaptive filters are the Least Mean Squares (LMS) [5, 6] and the Recursive Least Squares (RLS) [5] algorithms.

The RLS algorithm updates the inverse of the $L \times L$ covarience matrix (L is the length of the filter), therefore, its computational complexity amounts to $O(L^2)$ operations. To reduce the computational complexity of the RLS algorithms, some fast versions of the RLS (FRLS) algorithms have been developed, namely, the Fast Kalman [7], Fast a posteriori Error Sequential Technique(FAEST) [8], and Fast Transversal Filter(FTF) [9], which reduces the order of the complexity to $O(L)$. Practical use of the fast RLS algorithms in the real time applications has been prevented in the past because of the divergence due to numerical error accumulation in the prediction parameters. Efficient stabilization techniques with limited additional complexity that prevent numerical divergence without performance degradation have been proposed in [10, 11].

The LMS algorithm provides a solution to the optimal Weiner Filter criterion minimizing the mean square value of the error in a stochastic approximation sense. LMS belongs to the gradient type algorithmic schemes, thus inheriting their low computational complexity i.e. $O(L)$ operations, and their slow convergence, especially on highly correlated signals like speech. Hence, to whiten the input signal, one method is the projection algorithm or the affine projection algorithm [12], or the Normalized LMS algorithm [13].

There are many other algorithms derived from the LMS algorithm such as sign LMS [4], Leaky LMS [14] and Block LMS [15] algorithms just to name a few.

Another approach to improve the performance of the LMS is by employing a time varying step size in the standard LMS [16, 17]. This is based on using large step size when the algorithm is far from the optimal solution, thus speeding up the convergence rate, and when the algorithm is near the optimum, small step size is used to achieve a low level of misadjustment, thus achieving a better overall performance. This can be obtained by adjusting the step size in accordance to some criterion. Several criteria have been used, such as squared instantaneous error [16], sign changes of successive samples of the gradient [18], attempting to reduce the squared error at each instant [19], cross correlation of input and error [17]. In [20], a new variable step size LMS algorithm was proposed in which step size is adjusted according to the square of the time averaged estimate of the correlation of the error.

The Least Mean Fourth (LMF) algorithm, another member of the steepest descent algorithm with 4th error norm was suggested in [21], is a special case of the more general family of steepest decent algorithms [5] with $2k$ error norms, where k is a positive integer. While the LMS algorithm is well established in adaptive filtering, the LMF algorithm has recently gained attention [22, 23, 24].

LMS and LMF algorithms have some advantages and disadvantages. Recently, the utilization of the weighted sum of the two algorithms was proposed to combine the advantages of both in the mixed-norm adaptive filtering [25] in which the mixing parameter is fixed. Recently, in [26], a variable weight mixed norm LMS-LMF adaptive algorithm is proposed in which a self adapting time variable weighting factor is used that emphasizes the LMF algorithm when the coefficient vector is away from the optimal value and conversely emphasizes the LMS algorithm when it is close to the optimum. But due its reliance on the LMS and LMF, this algorithm is affected by the eigenvalue spread of the auto correlation matrix of the input signal. To overcome this problem, a normalized version of this algorithm is proposed in this work as was in the case of the LMS algorithm, that is the NLMS algorithm. But before this work is presented, let's review some of the background.

1.1 Adaptive Filters

When the filter is linear and all the pertinent statistics are known, the solution can be obtained from the *Wiener filter* [5], which is optimum in the mean-square sense. However, when the filter is required to operate in an environment of unknown statistics or a non stationary environment, an adaptive filter provides an elegant solution to such problems. Adaptive filters are generally defined as filters whose characteristics can be modified to achieve desired objectives and accomplish this modification or adaptation automatically without user intervention.

The adaptive filter relies on a recursive algorithm for its operation, which makes it possible for the filter to perform satisfactorily in an environment where complete knowledge of the relevant signal characteristics is not available. The algorithm starts from some predetermined set of initial conditions, representing complete ignorance about the environment. In a stationary environment, after successive iterations, the algorithm tries to converge to the optimum Wiener solution in some statistical sense.

There are different ways of classifying adaptive filters, depending on the feature of interest. When the feature of interest is input-output mapping, adaptive filters can be classified into two main groups: *linear and nonlinear.* Linear adaptive filters compute an estimate of the desired response by using a linear combination of the available set of the observables applied to the input. This form of input-output mapping satisfied by having a single layer of computational units or simply a single computational unit as the output layer. On the other hand, when the input-output mapping is required to be nonlinear, we need to use nonlinear adaptive filter. Typically, nonlinear adaptive filters involve the use of one or more layers of computational units in addition to the output layer [5].

1.2 Applications Of Adaptive Filters

Adaptive filtering has a number of applications in different fields. Although these applications are indeed quite different in nature, nevertheless, they have one basic common feature: an input signal and a desired response to compute the error, which is in turn used to control the values of a set of adjustable filter coefficients.

However, the main difference among the various applications is the manner in which the desired response is extracted. On this basis, adaptive filters are classified into the following four categories:

1.2.1 Inverse Modeling or Equalization

In this application, the adaptive filter is used to provide an inverse model that represents the best fit to the unknown system. Thus, at convergence, the inverse of the transfer function of the unknown system is approximated by the adaptive filter. A delay is introduced into the desired response path as shown in Figure 1.1, so as to ensure that the input to the adaptive filter is minimum phase and suitable for equalization. The primary use of the inverse modeling is for reducing the intersymbol

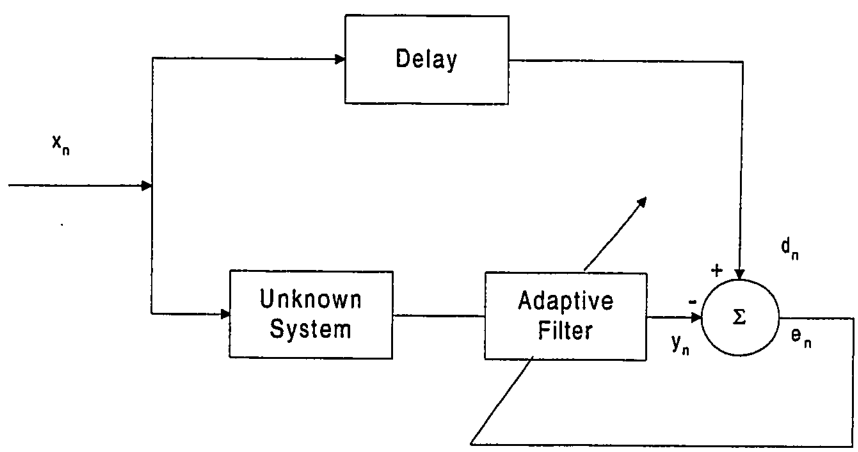

Figure 1.1: Inverse Modeling Scenario.

interference (ISI) in digital receivers. This is achieved through the use of channel equalization for digital communications [1].

1.2.2 Prediction

In this application, the adaptive filter is used to provide the best prediction of the present value of the input signal from its previous values. The configuration shown in Figure 1.2 is used for this purpose, where the desired signal, d_n, is the instantaneous value and the input to the adaptive filter is the delayed version of the same signal. This application is used in linear predictive coding (LPC) of speech [3]

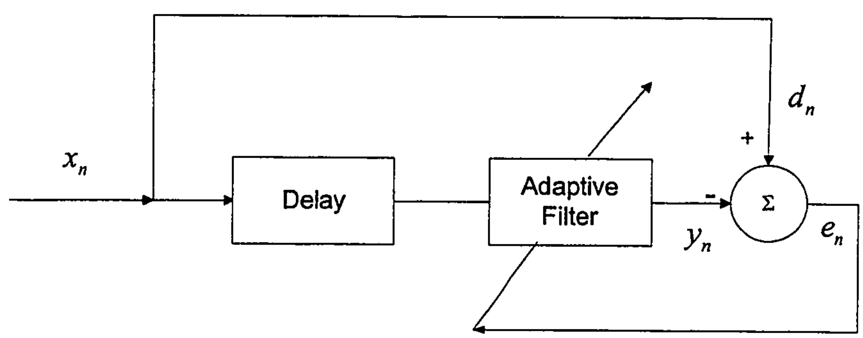

Figure 1.2: Prediction Scenario.

and in adaptive differential pulse-code modulation (DPCM) [27].

1.2.3 Noise Cancellation

In this class of applications, the adaptive filter is used to cancel unknown interference contained in a primary signal, as shown in Figure 1.3. The primary signal serves as the desired response of the adaptive filter. This type of application is used in adaptive noise cancellation [2], or adaptive beamforming or adaptive array processing [28].

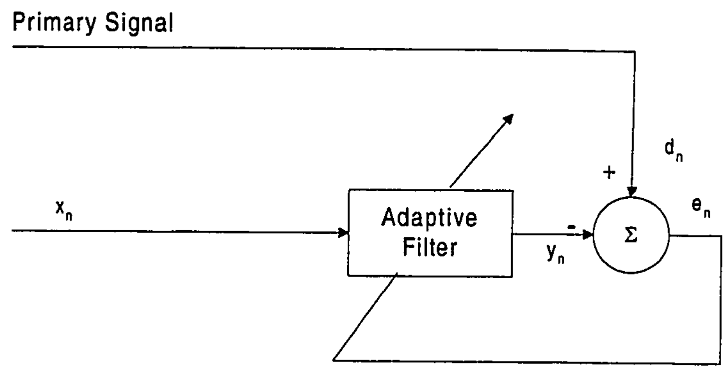

Figure 1.3: Noise Cancellation Scenario.

1.2.4 System Identification

System Identification is the experimental approach to the modeling of a process or a plant. It involves the following steps: experimental planning, the selection of a model structure, parameter estimation and model validation. The procedure of system identification, as pursued in practice, is iterative in nature in that we may have to go back and forth in these steps until a satisfactory model is built. The system to be identified is unknown which can be stationary or time varying. Figure 1.4 depicts the system identification scenario.

Now we will discuss briefly the idea of adaptive filtering algorithms for estimat-

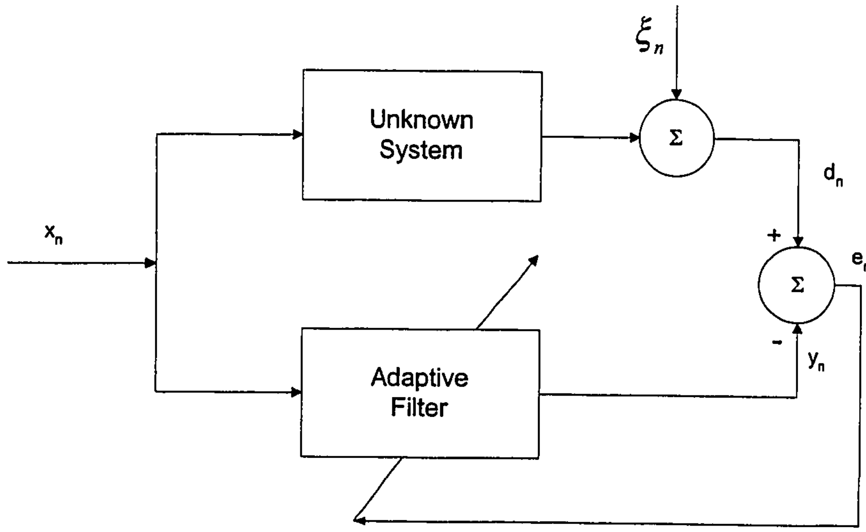

Figure 1.4: System Identification Scenario.

ing the parameters of an unknown system modeled as a transversal filter. Suppose that we have an unknown dynamic system that is linear and time varying. The system is characterized by a real valued set of discrete time measurements that describe the variations of the system output in response to a known stationary input. The requirement is to develop an on-line transversal filter model for this plant. The model consists of finite number of unit delay elements and a corresponding set of adjustable parameters (tap weights). Let the available input sequence at time n be denoted by the set of samples: $x_n, x_{n-1}...x_{n-M+1}$, where M is the number of adjustable parameters in the model. The input sequence is applied simultaneously to the system and the model. Let their respective outputs be denoted by d_n and y_n. The system output d_n serves as the purpose of desired response for the adaptive filtering algorithm employed to adjust the model parameters. The model output is given by

$$y_n = \sum_{K=0}^{M-1} \hat{w}_{kn} x_{n-k}. \tag{1.1}$$

where $\hat{w}_{0n}, \hat{w}_{1n}, ...\hat{w}_{(M-1)n}$ are the estimated model parameters at the nth iteration. The model output y_n is compared with the system output d_n. The difference between them defines the modeling(estimation) error. Let this error be denoted by e_n and defined as follows:

$$e_n = d_n - y_n. \tag{1.2}$$

Typically, at iteration n, the modeling error e_n is nonzero, implying that the model deviates from the system. In an attempt to account for this deviation, the error e_n is applied to an adaptive control algorithm. The samples of the input sequence $x_n, x_{n-1}...x_{n-M+1}$, are also applied to the algorithm. The combination of the transversal filter and the adaptive control algorithm constitutes the adaptive filtering algorithm. The algorithm is designed to control the adjustment made in the values of the model parameters. As a result, the model parameter assume a new set of values for use in the next iteration. Thus, at iteration $n+1$, a new model output is computed, and with it a new value for the modeling error. The operation described is then repeated. This process is continued for a sufficiently large number of iterations (starting from $n = 0$), until the deviation of the model from the system, measured by the magnitude of the modeling error e_n, becomes sufficiently small in a statistical sense.

When the system is time varying, the system output is non-stationary, and so is the desired response presented to the adaptive filtering algorithm. In such a situation, the adaptive filter has the task of not only keeping the modeling error small but also continually tracking the time variations in the dynamics of the system.

1.3 Adaptive Filtering Algorithms

An adaptive algorithm refers to the criteria by which a filter is adapted in response to the outside environmen. Let $\mathbf{w}_n$ be a vector of length L whose elements represent a time-varying finite impulse response of the adaptive filter. A general form for the algorithm that adapts the filter coefficient vector $\mathbf{w}_n$ is given by:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + Q(\mathbf{x}_n, e_n, \mu). \tag{1.3}$$

where $\mathbf{x}_n$ is the input sequence, e_n is the adaptive error, and μ denotes the algorithm step-size which may be time varying and Q is a function of all these quantities. In general the adaptive algorithms can be classified into two main categories, the least squares algorithms, and the least mean algorithms [29].

1.3.1 Least Squares Algorithms

Algorithms based on the method of least squares can be classified into three major classes [5] as follows:

1. Recursive least squares (RLS) algorithm.

2. QR-decomposition based recursive least squares algorithm (QRD-RLS).

3. Fast algorithms:

   - Fast transversal filters algorithm (FTF),
   - Recursive least squares lattice algorithm (LSL), and
   - QR-decomposition based least squares lattice algorithm (QRD-LSL).

Among all the above mentioned adaptive least squares algorithm, the most popular and widely used one is the RLS algorithm. Thus, only RLS is discussed here.

Recursive Least-Squares (RLS) Algorithm

In this class of algorithm, following is the minimization cost function:

$$J_n = \sum_{i=1}^{n} \beta_{(n,i)} e_i^2, \tag{1.4}$$

where the sequence $\beta_{(n,i)}$ represents a weighting function to gradually fade out the effect of previous data and to make the finite dimension arithmetic possible [30]. One possible choice for $\beta_{(n,i)}$ is the exponential weighting defined as [5]:

$$\beta_{(n,i)} = \lambda^{n-i} \tag{1.5}$$

where λ is a positive constant close to, but less than one [5]. The adaptation algorithm should read [5]:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mathbf{k}_{n+1}\alpha_{n+1}^* \tag{1.6}$$

where $\mathbf{k}_n$ is the gain vector, while α_n is the innovation. They are defined in [5] in terms of the autocorrelation matrix, $\mathbf{R}$, and the desired response d_n [5] as follows:

$$\mathbf{k}_n = \mathbf{R}_n^{-1}\mathbf{x}_n, \tag{1.7}$$

$$\alpha_n = d_n - \mathbf{w}_{n-1}^H \mathbf{x}_n, \tag{1.8}$$

The main advantage of the RLS algorithm is its faster convergence irrespective of the input statistics. But the main problem with RLS algorithm is its computational complexity.

1.3.2 Least Mean Algorithms

Instead of minimizing the time average of the error signal e_n as in the RLS algorithms, the least mean class of algorithms minimizes a statistical average of the error, i.e., $E[f(e_n)]$ is a convex function of the filter coefficients $\mathbf{w}_n$. Thus, $\mathbf{w}_n$ can be adapted using the steepest-descent algorithm as follows:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \mu \nabla E[f(e_n)], \tag{1.9}$$

where $\nabla E[f(e_n)]$ represents the gradient of $E[f(e_n)]$ with respect to $\mathbf{w}_n$. Since, a closed form of this gradient is not available, it is replaced by its stochastic approximation, i.e. $\nabla E[f(e_n)]$, or equivalently by $-\frac{df(e_n)}{de_n}\Big|_{e_n}\mathbf{x}_n$. Thus, different least mean algorithms are obtained for each choice of function $f(e_n)$, some well known of them are discussed here.

Least Mean Square (LMS) Algorithm

If $f(e_n) = [e_n^2]$, the least mean squares (LMS) algorithm is obtained which is one of the most popular algorithms in adaptive filtering. According to LMS algorithm, the filter coefficients are adapted according to the following recursion:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu e_n \mathbf{x}_n, \tag{1.10}$$

where the error signal e_n is defined in (1.2), $\mathbf{x}_n$ is the tap input vector, and $\mathbf{w}_n$ represents the tap weights of the adaptive filter. The parameter μ is a positive constant called step size which is used to control the size of the incremental correction applied to the tap weights as we proceed from one iteration to the next.

The LMS algorithm is simple to implement and yet capable of achieving satisfactory performance under the right conditions. Its major limitations are a relatively slow rate of convergence and a sensitivity to variations in the condition number of correlation matrix of the input signal; the condition number of a hermitian matrix is defined as the ratio of its largest eigenvalue to its smallest eigenvalue.

In a non-stationary environment, the orientation of the error-performance surface varies continuously with time. In this case, the LMS algorithm has the added task of continually tracking the bottom of the error performance surface. Indeed, tracking will occur provided that the input data varies slowly compared to the learning rate of the LMS algorithm [5].

Least Mean Fourth (LMF) Algorithm

This algorithm consists of minimizing the fourth power of the error. Infact, it is a special case of the more general family of the steepest decent algorithms with $2k$ error norms [5], where k is a positive integer, and its weight update equation is defined as:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + 2\mu e_n^3 \mathbf{x}_n. \tag{1.11}$$

The LMF algorithm has a faster convergence as compared to LMS but has higher steady state error. The complexity of the LMF algorithm is more as compared to the LMS algorithm because of the higher power of e_n involved in the adaptation of the weights.

Least Mean p-power Algorithm

Recently, other minimization criteria have emerged, in which adaptive structures are derived from the minimization of a class of functions of the form [21]:

$$J_n = E[e_n^p]; \qquad p \geq 1 \tag{1.12}$$

where p is an integer constant. It is seen from (1.12) that when $p = 2$, it is reduced to the cost function of LMS algorithm while LMF when $p = 4$.

Least Mean Mixed Norm (LMMN) Algorithm

It has been seen that the LMS algorithm can achieve the lower steady-state error floor than the LMF algorithm but its convergence speed is slower. So, knowing the fact that the convex addition of two convex function is also a convex function, a class of mixed norm algorithm [25] was developed with the following cost function:

$$J_n = \alpha E[e_n^2] + (1 - \alpha) E[e_n^4]. \tag{1.13}$$

where α is the mixing parameter. The adaptation algorithm of (1.3.2) reduces to the LMS algorithm and LMF algorithm, respectively, for $\alpha = 1$ and $\alpha = 0$. Thus, it represents the generalization of both the LMS and LMF algorithm.

1.4 Normalized LMS (NLMS) Adaptive Algorithm

The LMS algorithm performs badly with correlated input signal like speech signals. The reason is that LMS algorithm is directly dependent on the input vector $\mathbf{x}_n$. Therefore, when the eigen value spread of $\mathbf{x}_n$ is large, LMS algorithm experiences a *gradient noise amplification.* In order to overcome this problem, in traditional normalized LMS algorithm, the input signal is normalized by the input signal power. Thus, the filter coefficients are adapted according to the following recursion [13]:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu \frac{e_n \mathbf{x}_n}{\| \mathbf{x}_n \|^2}. \tag{1.14}$$

Due to normalization, NLMS algorithm is made to have more stable behaviour for a known range ($0 < \mu < 2$), less sensitive to the colored input signal (as the effect of the eigen value spread of the input vector is reduced) [31], and converges faster than the LMS algorithm [31]. These facts gave us motivation to investigate the performance when the time varying mixed norm LMS-LMF algorithm [26] is normalized. The resulting algorithm, i.e., normalized time varying mixed norm LMS-LMF algorithm, promises an impressive improvement over the conventional NLMS algorithm.

1.5 Thesis Objectives and Organization

In this thesis work, the normalized time varying mixed norm LMS-LMF algorithm is proposed which minimizes a cost function defined as normalized weighted sum of LMS and LMF cost functions where the weighting factor is time varying and adapts itself so as to allow the algorithm to keep track of the variations in the environment.

The four objectives of this thesis are: (1) To examine the convergence properties of the proposed algorithm and to derive sufficient and necessary condition for the convergence in the mean and to find weight error vector recursion in the mean square sense. (2) To analyze the steady-state performance of the proposed algorithm and to derive the expression for excess mean square error at the steady-state. (3) To present the simulation scenario in support of the analytical analysis. (4) To do the tracking analysis of the proposed algorithm in the presence of both randomly and cyclicly varying nonstationarities.

Finally, the thesis is organized so as to achieve all the above mentioned objectives respectively. In Chapter 2, the proposed algorithm is studied. In particular, the convergence analysis is carried out and necessary condition for the convergence in the mean and the weight error vector recursion in the mean square sense are derived. Also, the steady-state analysis is performed and the expression for excess mean square error of the proposed algorithm at the steady-state is derived. A comparison of steady-state excess mean square error is carried out between proposed and the conventional NLMS and NLMF algorithms.

The tracking analysis of the proposed algorithm in the presence of both randomly and cyclicly varying nonstationarities is presented in the chapter 3 and mathematical expressions are derived for steady state excess mean square error for NLMS, NLMF and the proposed algorithms. These analytical results are then compared with the experimental results.

In support of the mathematical analysis, in Chapter 4, simulation scenario is presented in different environment (Uniform and Gaussian) and the analytical results of weight error recursion are compared with the experimental results. Finally, Thesis conclusions, contributions and the recommendations for Future Work are presented in Chapter 5.

Chapter 2

Proposed Normalized Time Varying Mixed Norm LMS-LMF Algorithm

2.1 Introduction

The least-mean squares (LMS) algorithm [6] is one of the most widely used adaptive schemes. It has several desirable features and some limitations. As such, several LMS-variants have been proposed that trade some of the LMS features for an enhanced performance in some of its limitations. Of particular importance is the class of least-mean square algorithms that employ an error nonlinearity $f(e_n)$ instead of the (linear) error term in LMS adaptation [32, 33, 34, 35]. Examples include the sign-error algorithm [36], the least-mean fourth (LMF) algorithm and its family [21], and the least-mean mixed norm algorithm [37], all of which are intuitively motivated. Table 2.1 defines $f(e_n)$ for many famous algorithms. Also, mentioned in Table 2.1 is $f(e_n) = \alpha e_n + (1-\alpha)e_n^3$ which is the error nonlinearity used in the mixed norm LMS-LMF algorithm [37, 26] with α as the mixing parameter. This algorithm is found to result in better performance than either the LMS or the LMF algorithms in Gaussian and non-Gaussian environments [37, 26].

Algorithm	$f(e_n)$
LMS	e_n
NLMS	$\frac{e_n}{\Vert \mathbf{x}_n \Vert^2}$
LMF	e_n^3
NLMF	$\frac{e_n^3}{\Vert \mathbf{x}_n \Vert^2}$
LMMN	$\alpha e_n + (1-\alpha)e_n^3$
Sign Error	$\text{sign}[e_n]$

Table 2.1: Examples for $f(e_n)$.

LMS and LMF algorithms have different convergence behavior and robustness to noise statistics (gaussian versus non-gaussian noise) [21]. For example, the LMF algorithm will clearly have a larger gradient driving it to converge faster when away from the optimum ($e_n^4 > e_n^2$ for $e_n^2 > 1$). However, the LMS will have more desirable characteristics in the neighborhood of the optimum.

Recently, the utilization of a weighted sum of the two performance measures was proposed to combine the advantages of both in the mixed-norm adaptive algorithm [25]. The mixed-norm LMS-LMF adaptive algorithm is defined by the following cost function:

$$J_n = \alpha E[e_n^2] + (1-\alpha)E[e_n^4]. \tag{2.1}$$

where the error is defined as

$$e_n = d_n - \mathbf{x}_n^T \mathbf{w}_n. \tag{2.2}$$

d_n is the desired value, $\mathbf{w}_n$ is the filter coefficient of the adaptive filter, $\mathbf{x}_n$ is the input vector and ξ_n is the additive noise, Figure 2.1 depicts this clearly.

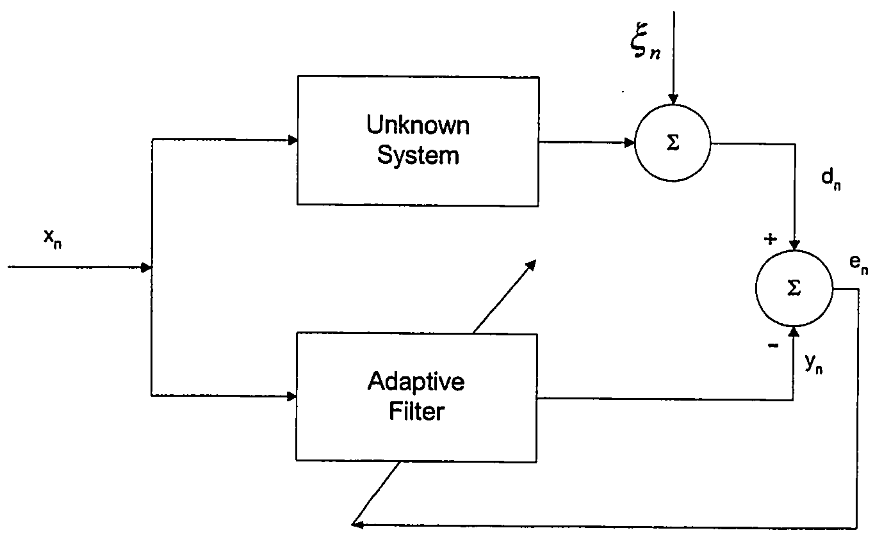

Figure 2.1: System Identification Scenario.

The algorithm defined in Equation (2.1) has a fixed mixing or weighting factor that is predetermined by the designer and hence will be unable to track variations in the environment. To overcome this difficulty, a time-variation in the weight parameters is proposed [26] and its cost function is defined as:

$$J_n = \alpha_n E[e_n^2] + (1 - \alpha_n) E[e_n^4]. \tag{2.3}$$

where α_n is the time varying mixing parameter which changes in accordance to the square of the time averaged estimate of the auto correlation of the e_n and e_{n-1}, i.e. it is updated as follows [20]:

$$\alpha_{n+1} = \delta \alpha_n + \gamma p_n^2, \tag{2.4}$$

$$p_n = \beta p_{n-1} + (1 - \beta) e_n e_{n-1}. \tag{2.5}$$

where δ, β, and γ are constants. The parameters δ and β, confined to the interval [0,1], are exponential weighting parameters that govern the averaging time constant, i.e., the quality of estimation, and $\gamma > 0$. But due to reliance of this algorithm on the LMS and LMF, it is affected by the eigenvalue spread of the auto correlation matrix of the input signal. To overcome this problem, a normalized version of this algorithm is proposed here which is called Normalized Time Varying Mixed Norm LMS-LMF algorithm.

2.2 The Normalized Time Varying Mixed Norm LMS-LMF Algorithm

Since LMS and LMF algorithms do not perform well when the input signal is colored, i.e., the autocorrelation matrix of the input signal has a larger eigen value spread, therefore, the variable weight mixed norm LMS-LMF algorithm defined by cost function (2.3), is also affected by the eigenvalue spread of the autocorrelation matrix of the input signal. Thus, in the proposed algorithm normalization of the input sequence is introduced into the time varying weighted sum of LMS and LMF algorithm and are updated according to the following recursion:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu[\alpha_n e_n + 2(1-\alpha_n)e_n^3]\frac{\mathbf{x}_n}{\| \mathbf{x}_n \|^2}. \tag{2.6}$$

2.3 Convergence Analysis

The following assumptions which are quite similar to what is usually assumed in literature and which can also be justified in several practical instances are used during the convergence and steady-state analysis of the proposed algorithm:

A1 The input $\mathbf{x}_n$ is a zero-mean, stationary, Gaussian process. Moreover, $\{\mathbf{x}_k, \mathbf{d}_k\}$ and $\{\mathbf{x}_n, \mathbf{d}_n\}$ are uncorrelated for $n \neq k$, where n and k show different time instants. This is commonly employed independence assumption [33, 38, 39] and is seldom true in practice. However, analysis employing this assumption have produced reliable results in the past.

A2 The noise ξ_n is a zero-mean iid process, and is independent of the input process. This assumption is justified in several practical examples.

A3 The mixing parameter α_n is independent of both the input signal and the weight error vector. This assumption can be justified by observing the equations

(2.4) and (2.5). Since parameter γ is very small, the dependency of the mixing parameter "α" on the input signal and the weight error vector is almost negligible.

A4 The auto-correlation matrix $\mathbf{R}$ of the input vector $\mathbf{x}_n$ is a diagonal matrix which is given by $\mathbf{R} = \sigma_x^2\mathbf{I}$, where σ_x^2 is the variance of the input sequence $\mathbf{x}_n$. This assumption is fairly restrictive but has been used [38] because of analytical difficulty. Furthermore, the white data model is a valid representation in many practical systems such as digital data transmission systems and analog systems that are sampled at the Nyquist rate and adapted using discrete-time algorithms.

Since $\mathbf{x}_k$ and $\mathbf{x}_n$ are uncorrelated for $n \neq k$, they are independent as well. Thus, it can be shown that input sequence $\mathbf{x}_n$ and the weight error vector $\mathbf{v}_n$ that will be defined later in Equation (2.7), are also independent. This assumption has been used in [40].

In the next section, the convergence analysis of the proposed algorithm is carried out using the above simplifying assumptions.

2.3.1 Convergence in the Mean

The weight update equation for the proposed algorithm is given by Equation 2.6. If $\mathbf{w}_{opt}$ is the optimum value of the weight according to the Wiener solution i.e. exact solution for the parameters of the actual system, then we can define weight error vector, $\mathbf{v}_n$, as follows:

$$\mathbf{v}_n = \mathbf{w}_n - \mathbf{w}_{opt}. \tag{2.7}$$

Now subtracting $\mathbf{w}_{opt}$ from both sides of the Equation (2.6), we will get:

$$\mathbf{v}_{n+1} = \mathbf{v}_n + \mu[\alpha_n e_n + 2(1-\alpha_n)e_n^3]\frac{\mathbf{x}_n}{\| \mathbf{x}_n \|^2}. \tag{2.8}$$

If the additive noise is represented by ξ_n, then the error in the system identification problem scenario which has been derived in Appendix **A**, can be written as:

$$e_n = \xi_n - \mathbf{x}_n^T \mathbf{v}_n. \tag{2.9}$$

Since, near the optimal solution, the term $\mathbf{x}_n^T \mathbf{v}_n$ is very small, we can use the following approximation:

$$(\mathbf{x}_n^T \mathbf{v}_n)^3 \approx \mathbf{x}_n^T \mathbf{v}_n \tag{2.10}$$

Using the above approximation for e_n^3 and substituting the values of e_n and e_n^3 into Equation (2.8), we get:

$$\begin{aligned}\mathbf{v}_{n+1} &= \mathbf{v}_n + [\mu\alpha_n\xi_n - \mu\alpha_n\mathbf{x}_n^T\mathbf{v}_n + 2\mu(1-\alpha_n)\xi_n^3 - 6\mu(1-\alpha_n)\xi_n^2\mathbf{x}_n^T\mathbf{v}_n \\ &\quad +6\mu(1-\alpha_n)\xi_n(\mathbf{x}_n^T\mathbf{v}_n)^2 - 2\mu(1-\alpha_n)\mathbf{x}_n^T\mathbf{v}_n)]\frac{\mathbf{x}_n}{\| \mathbf{x}_n \|^2}.\end{aligned} \tag{2.11}$$

Now taking the expectation on both sides of the above equation and using the assumptions mentioned earlier, we get:

$$\begin{aligned}E[\mathbf{v}_{n+1}] &= \Big(\mathbf{I} - \mu E[\alpha_n]E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\| \mathbf{x}_n \|^2}] - 6\mu(1-E[\alpha_n])\sigma_w^2 E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\| \mathbf{x}_n \|^2}] \\ &\quad -2\mu(1-E[\alpha_n])E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\| \mathbf{x}_n \|^2}]\Big)E[\mathbf{v}_n] \\ &= \Big\{\mathbf{I} - \mu\Big(E[\alpha_n] + 2(1-E[\alpha_n])\{3\sigma_w^2+1\}\Big) \\ &\quad E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\| \mathbf{x}_n \|^2}]\Big\}E[\mathbf{v}_n].\end{aligned} \tag{2.12}$$

where σ_w^2 is the variance of the noise. The necessary condition for convergence in the mean is represented by:

$$\left|1-\mu\Big(E[\alpha_n]+2(1-E[\alpha_n])\{3\sigma_w^2+1\}\Big)tr\{E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\|\mathbf{x}_n\|^2}]\}\right|<1 \qquad (2.13)$$

Thus, the value of μ is bounded in the range:

$$0<\mu<\frac{2}{\Big(E[\alpha_n]+2(1-E[\alpha_n])\{3\sigma_w^2+1\}\Big)tr\{E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\|\mathbf{x}_n\|^2}]\}}$$

Since, from [5] we have:

$$tr\{E[\frac{\mathbf{x}_n^T\mathbf{x}_n}{\|\mathbf{x}_n\|^2}]\}\approx 1$$

Thus, the range of μ is given by:

$$0<\mu<\frac{2}{\Big(E[\alpha_n]+2(1-E[\alpha_n])\{3\sigma_w^2+1\}\Big)} \qquad (2.14)$$

As it can be seen from 2.14 that μ depends only on the mean value of the mixing parameter and the variance of noise. It is also very clear from 2.14 that if $\alpha_n = 1$ and $\alpha_n = 0$ are substituted, the bound on μ for the NLMS and NLMF algorithm will be obtained, respectively, which are given below:

$$0 \;<\; \mu_{NLMS}<2 \qquad (2.15)$$

and

$$0 \;<\; \mu_{NLMF}<\frac{1}{3\sigma_w^2+1} \qquad (2.16)$$

These bounds are identical to those found in [41] and [42], respectively.

2.3.2 Convergence in the Mean Square

The update equation for the weight error vector is given by:

$$\begin{aligned}\mathbf{v}_{n+1} &= \{\mathbf{I} - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\}\mathbf{v}_n \\ &+\mu[\alpha_n\xi_n + 2(1-\alpha_n)\xi_n^3]\frac{\mathbf{x}_n}{\|\mathbf{x}_n\|^2}. \end{aligned} \tag{2.17}$$

Now taking transpose of the weight error vector:

$$\begin{aligned}\mathbf{v}_{n+1}^T &= \mathbf{v}_n^T\{\mathbf{I} - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\} \\ &+\mu[\alpha_n\xi_n + 2(1-\alpha_n)\xi_n^3]\frac{\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}. \end{aligned} \tag{2.18}$$

Post multiplying $\mathbf{v}_{n+1}$ by $\mathbf{v}_{n+1}^T$ gives the following:

$$\begin{aligned}\mathbf{v}_{n+1}\mathbf{v}_{n+1}^T &= \{\mathbf{v}_n\mathbf{v}_n^T - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T\mathbf{v}_n\mathbf{v}_n^T}{\|\mathbf{x}_n\|^2}\}\{\mathbf{I} - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\} \\ &+\mu[\alpha_n\xi_n + 2(1-\alpha_n)\xi_n^3]\{\mathbf{I} - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\}\frac{\mathbf{v}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2} \\ &+\mu[\alpha_n\xi_n + 2(1-\alpha_n)\xi_n^3]\frac{\mathbf{x}_n\mathbf{v}_n^T}{\|\mathbf{x}_n\|^2}\{\mathbf{I} - \mu[\alpha_n + 6(1-\alpha_n)\xi_n^2]\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\} \\ &+\mu^2[\alpha_n^2\xi_n^2 + 4(\alpha_n - \alpha_n^2)\xi_n^4 + 4(1-2\alpha_n+\alpha_n^2)\xi_n^6]\frac{\mathbf{x}_n\mathbf{x}_n^T}{(\|\mathbf{x}_n\|^2)^2}. \end{aligned} \tag{2.19}$$

Let $\mathbf{K}_n = E[\mathbf{v}_n\mathbf{v}_n^T]$ be the weight error correlation matrix which can be shown to be governed by the following recursion:

$$\begin{aligned}\mathbf{K}_{n+1} &= \mathbf{K}_n - \mu\Big\{E[\alpha_n] + 6(1-E[\alpha_n])\sigma_w^2\Big\}\{\mathbf{B}\mathbf{K}_n + \mathbf{K}_n\mathbf{B}\} \\ &+\mu^2\Big\{E[\alpha_n^2](1 - 12\sigma_w^2 + 36\phi_w^4) \\ &+12E[\alpha_n](\sigma_w^2 - 6\phi_w^4) + 36\phi_w^4\Big\}\mathbf{C} \\ &+\mu^2\Big\{E[\alpha_n^2](\sigma_w^2 - 4\phi_w^4 + 4\phi_w^6) \\ &+4E[\alpha_n](\phi_w^4 - 2\phi_w^6) + 4\phi_w^6\Big\}\mathbf{D}. \end{aligned} \tag{2.20}$$

where ϕ_w^4, and ϕ_w^6 are fourth and sixth moments of the noise respectively. Terms $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are defined as follows:

$$\begin{aligned}
\mathbf{B} &= E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}]. & (2.21)\\
\mathbf{C} &= E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\mathbf{v}_n\mathbf{v}_n^T\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}]. & (2.22)\\
\mathbf{D} &= E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{(\|\mathbf{x}_n\|^2)^2}]. & (2.23)
\end{aligned}$$

To simplify the analysis further, the following assumption is used which is suggested in [5]:

$$E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}] \approx \frac{E[\mathbf{x}_n\mathbf{x}_n^T]}{E[\|\mathbf{x}_n\|^2]}. \tag{2.24}$$

Therefore, we can find the values of $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ with the help of above approximation and the assumption **A4** described earlier as follows:

$$\begin{aligned}
\mathbf{B} &= E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}] \approx \frac{E[\mathbf{x}_n\mathbf{x}_n^T]}{E[\|\mathbf{x}_n\|^2]}\\
&= \frac{\mathbf{I}}{\mathbf{N}}. & (2.25)
\end{aligned}$$

where N is the size of filter.

Similarly,

$$\mathbf{C} = E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}\mathbf{v}_n\mathbf{v}_n^T\frac{\mathbf{x}_n\mathbf{x}_n^T}{\|\mathbf{x}_n\|^2}] \approx \frac{E[\mathbf{x}_n\mathbf{x}_n^T\mathbf{K}_n\mathbf{x}_n\mathbf{x}_n^T]}{E[(\|\mathbf{x}_n\|^2)^2]} \tag{2.26}$$

Since, it is assumed that the entries $\mathbf{x}_n$ are zero mean and white Gaussian random variables, therefore by using the "Gaussian Moment Factoring Theorem" [5, 43], fourth order expectations can be expressed as a sum of products of second order expectation. Hence, we can express the numerator of the above expectation which is derived in Appendix **B** in the following form:

$$E[\mathbf{x}_n\mathbf{x}_n^T\mathbf{K}_n\mathbf{x}_n\mathbf{x}_n^T] = 2\sigma_x^4\mathbf{K}_n + \sigma_x^4 tr\{\mathbf{K}_n\}\mathbf{I}. \tag{2.27}$$

Similarly, the expectation $E[(\| \mathbf{x}_n \|^2)^2]$ is derived in Appendix **C** and its result is given by:

$$E[(\| \mathbf{x}_n \|^2)^2] \;=\; N\sigma_x^4. \tag{2.28}$$

Thus, the value of **C** will be:

$$\mathbf{C} = \frac{1}{N}\{2\mathbf{K}_n + tr\{\mathbf{K}_n\}\mathbf{I}\}. \tag{2.29}$$

and in the same way **D** is evaluated as:

$$\begin{aligned} \mathbf{D} &= E[\frac{\mathbf{x}_n\mathbf{x}_n^T}{(\| \mathbf{x}_n \|^2)^2}] \approx \frac{E[\mathbf{x}_n\mathbf{x}_n^T]}{E[(\| \mathbf{x}_n \|^2)^2]} \\ &= \frac{\mathbf{I}}{\mathbf{N}\sigma_{\mathbf{x}}^2}. \end{aligned} \tag{2.30}$$

Thus, Equation (2.20) can be expressed in the following form and its detail is given in Appendix **D** :

$$\mathbf{K}_{n+1} = \mathbf{K}_n - \mu a_1[\mathbf{K}_n] + \mu^2 a_2[2\mathbf{K}_n + tr\{\mathbf{K}_n\}\mathbf{I}] + \mu^2 a_3\mathbf{I}, \tag{2.31}$$

with

$$a_1 \;=\; \frac{2}{N}\{E[\alpha_n] + 6(1 - E[\alpha_n])\sigma_w^2\}, \tag{2.32}$$

$$a_2 \;=\; \frac{1}{N}\{E[\alpha_n^2](1 - 12\sigma_w^2 + 36\phi_w^4) + 12E[\alpha_n](\sigma_w^2 - 6\phi_w^4) + 36\phi_w^4\}, \tag{2.33}$$

and

$$a_3 = \frac{1}{N\sigma_x^2}\{E[\alpha_n^2](\sigma_w^2 - 4\phi_w^4 + 4\phi_w^6) + 4E[\alpha_n](\phi_w^4 - 2\phi_w^6) + 4\phi_w^6\}. \quad (2.34)$$

where $E[\alpha_n^2]$, σ_x^2, ϕ_w^4 and ϕ_w^6 are the second moment of the mixing parameter, the variance of the input sequence, the fourth moment of the noise and the sixth moment of the noise, respectively.

Now diagonalizing the weight error correlation matrix $\mathbf{K}_n$ by using the unitary transformation as follows:

$$\mathbf{G}_n = \mathbf{Q}^T\mathbf{K}_n\mathbf{Q}, \quad (2.35)$$

where $\mathbf{Q}$ is the orthonormal matrix whose i^{th} column is the eigenvector of $\mathbf{K}_n$ associated with the i^{th} eigenvalue, that is

$$\mathbf{Q}^T\mathbf{Q} = \mathbf{I}, \quad (2.36)$$

hence Equation (2.31) will look like the following:

$$\mathbf{G}_{n+1} = \mathbf{G}_n - \mu a_1\mathbf{G}_n + \mu^2 a_2[2\mathbf{G}_n + tr\{\mathbf{G}_n\}\mathbf{I}] + \mu^2 a_3\mathbf{I}. \quad (2.37)$$

The above equation shows that the mean square behaviour of the weight error vector is dependent on the value of step-size μ, size of the filter N, mean of the mixing parameter $E[\alpha_n]$, second moment of $E[\alpha_n^2]$, variance of the input sequence σ_x^2, variance of the noise σ_w^2, fourth and sixth moment of the noise i.e. ϕ_n^4 and ϕ_n^6 respectively.

2.4 Steady-State Analysis

In general the adaptation scheme defined in [13] can be set into the following form:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu \mathbf{x}_n f(e_n). \tag{2.38}$$

where $f(e_n)$ denotes a general scalar function of the output estimation error e_n. In the case of the proposed algorithm, the error nonlinearity function $f(e_n)$ used is $\frac{\{\alpha_n e_n + 2(1-\alpha_n)e_n^3\}}{\|\mathbf{x}_n\|^2}$, with α_n being the time varying mixing parameter.

In the steady-state analysis of an adaptive algorithm, an important measure is its steady-state mean square error (MSE), which is defined as:

$$MSE = \lim_{n\to\infty} E[e_n^2] \tag{2.39}$$

$$= \lim_{n\to\infty} E\{[\xi_n - \mathbf{x}_n^T \mathbf{v}_n]^2\}, \tag{2.40}$$

Under the assumption **A2** defined earlier, the MSE is given by:

$$MSE = \sigma_w^2 + \lim_{n\to\infty} E\{[\mathbf{x}_n^T \mathbf{v}_n]^2\}, \tag{2.41}$$

which reduces to:

$$MSE = \sigma_w^2 + \lim_{n\to\infty} tr\{\mathbf{R}\mathbf{K}_n\}. \tag{2.42}$$

where $\mathbf{K}_n = E[\mathbf{v}_n \mathbf{v}_n^T]$ denotes the covariance matrix of the weight error vector, $\mathbf{v}_n$. During the steady-state analysis, the feedback approach [44, 45] is used for evaluating the MSE in a rather simpler way. The approach bypasses the need of directly using the covariance matrix of the weight error vector, i.e., Equation 2.37. Moreover, the feedback approach distinguish itself from earlier approaches in this respect as it does it in a general context where all the linear-nonlinear functions, $f(e_n)$, are obtained.

2.4.1 Fundamental Energy Relation

In this section the steady-state error analysis of the proposed algorithm is carried out using the concept of the feedback approach. First let's define the following so-called a priori estimation error, $e_{an} = \mathbf{x}_n^T \mathbf{v}_n$ and a posteriori estimation error, $e_{pn} = \mathbf{x}_n^T \mathbf{v}_{n+1}$. If we define the weight error vector as $\mathbf{v}_n = \mathbf{w}_{opt} - \mathbf{w}_n$, then it is very easy to show that the estimation error, e_n, and the a-priori error, e_{an}, are related via $e_n = e_{an} + \xi_n$. Also it is very easy show that the a-posteriori error, e_{pn}, can be set up into the following:

$$e_{pn} = e_{an} - \frac{\mu}{\hat{\mu}_n} f(e_n) \tag{2.43}$$

where $\hat{\mu}_n = 1/ \parallel \mathbf{x}_n \parallel^2$. Substituting Equation (2.43) into Equation(2.38) results into the following update relation:

$$\mathbf{v}_{n+1} = \mathbf{v}_n - \hat{\mu}_n \mathbf{x}_n [e_{an} - e_{pn}]. \tag{2.44}$$

By evaluating the energies of both the sides of the above equation, the new relation is obtained:

$$\parallel \mathbf{v}_{n+1} \parallel^2 + \hat{\mu}_n \parallel e_{an} \parallel^2 = \parallel \mathbf{v}_n \parallel^2 + \hat{\mu}_n \parallel e_{pn} \parallel^2 . \tag{2.45}$$

This energy relation hold for all adaptive filtering algorithms whose recursions are of the form given by Equation (2.38). No approximations or assumptions are needed to establish Equation (2.45); it is an exact relation that shows how the energies of the weight error vectors at two successive time instants are related to the energies of the a-priori and a-posteriori estimation errors. The relation has also an interesting system-theoretic interpretation. It establishes that the mapping from the variables

$\{\mathbf{v}_n, \sqrt{\hat{\mu}_n}e_{pn}\}$ to $\{\mathbf{v}_{n+1}, \sqrt{\hat{\mu}_n}e_{an}\}$ is energy preserving (or lossless) [44]. Moreover, combining Equations (2.43) and (2.45), it is very easy to notice that both of these expressions establish the existence of the feedback configuration shown in Figure 2.2, where the operator Γ denotes the lossless map from $\{\mathbf{v}_n, \sqrt{\hat{\mu}_n}e_{pn}\}$ to $\{\mathbf{v}_{n+1}, \sqrt{\hat{\mu}_n}e_{an}\}$, and where q^{-1} denotes the unit delay operator.

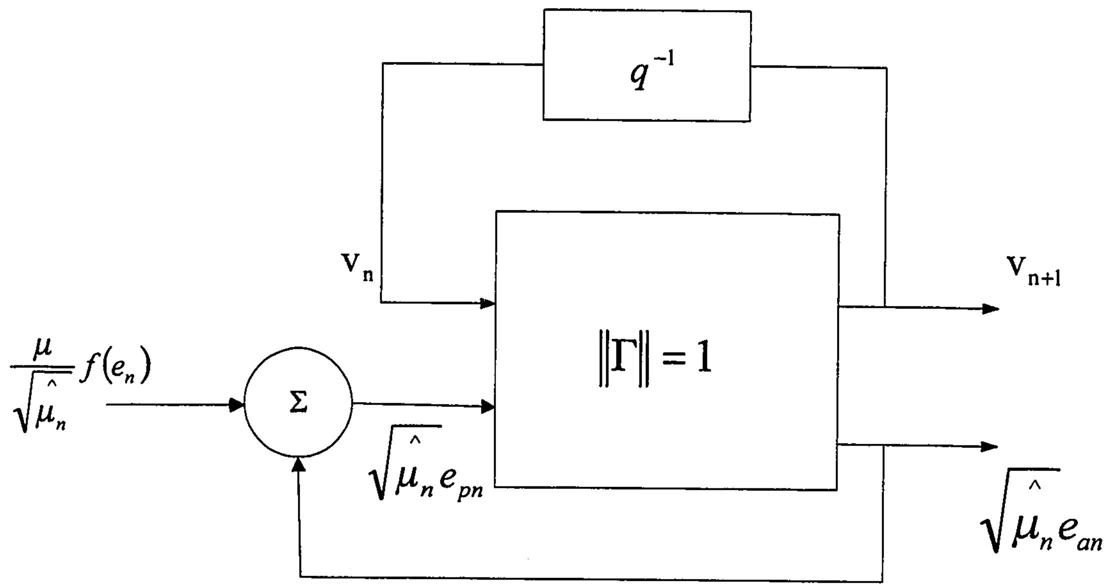

Figure 2.2: Lossless Mapping and a feedback loop.

2.4.2 Steady-State Performance Analysis

As it has been discussed earlier that the performance measure for the steady-state analysis is MSE of the adaptive filter when it reaches the steady-state, therefore in this section expression for the steady-state MSE of the proposed algorithm is derived. To do so, Equation (2.45) and the fact that in steady-state $E[\| \mathbf{v}_{n+1} \|^2] = E[\| \mathbf{v}_n \|^2]$ are used. By taking expectations of both sides of Equation (2.45), the following equality is obtained:

$$E[\hat{\mu}_n \| e_{an} \|^2] = E[\hat{\mu}_n \| e_{pn} \|^2]. \tag{2.46}$$

Since e_{pn} itself a function of e_{an}, Equation (2.46) reduces to the fundamental error variance relation in terms of the desired unknown e_{an} and ξ_n only in the following form:

$$E[\hat{\mu}_n \| e_{an} \|^2] = E[\hat{\mu}_n \| e_{an} - \frac{\mu}{\hat{\mu}_n} f(e_n) \|^2]. \tag{2.47}$$

The above equation can be solved for the steady-state excess mean-square-error (EMSE) defined as:

$$\begin{aligned} \zeta &= \lim_{n\to\infty} E\{[\mathbf{x}_n^T \mathbf{v}_n]^2\} \\ &= \lim_{n\to\infty} E[e_{an}^2]. \end{aligned} \tag{2.48}$$

It can be observed from Equation (2.41) that the desired MSE is given by $MSE = \sigma_w^2 + \zeta$, so that finding ζ is equivalent to finding the MSE. Equation (2.47) can be set into the following:

$$2\mu E[e_{an} f(e_n)] = \mu^2 E[\| \mathbf{x}_n \|^2 \, f(e_n)^2]. \tag{2.49}$$

For the case of the proposed Normalized Time Varying Mixed Norm adaptive algorithm $f(e_n) = \frac{\{\alpha_n e_n + 2(1-\alpha_n)e_n^3\}}{\|\mathbf{x}_n\|^2}$. Since e_{an} is small as the adaptive algorithm tries to achieve the optimum solution, its higher powers can be ignored. Thus e_n^3 can be approximated as follows:

$$\begin{aligned} e_n^3 &= (e_{an} + \xi_n)^3 \\ &\approx 3e_{an}\xi_n^2 + \xi_n^3 \end{aligned}$$

So, the approximated value of $f(e_n)$ will be:

$$f(e_n) \approx \frac{\{\alpha_n(e_{an} + \xi_n) + 2(1-\alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}}{\| \mathbf{x}_n \|^2}. \tag{2.50}$$

Now the value of $f(e_n)$ is substituted from Equation (3.29) into Equation (2.49) to get:

$$\begin{aligned} 2\mu E\Big[\frac{\alpha_n e_{an}^2 + \alpha_n e_{an}\xi_n + 6(1-\alpha_n)e_{an}^2\xi_n^2 + 2(1-\alpha_n)e_{an}\xi_n^3}{\| \mathbf{x}_n \|^2}\Big] &= \\ \mu^2 E\Big[\frac{\{\alpha_n(e_{an} + \xi_n) + 2(1-\alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}^2}{\| \mathbf{x}_n \|^2}\Big]. \end{aligned} \tag{2.51}$$

When the assumptions mentioned in the section 2.3 are taken into account, Equation (2.51) resulted in:

$$\begin{aligned} 2\Big\{E[\alpha_n] + 6(1-E[\alpha_n])\sigma_w^2\Big\}E\left[\frac{e_{an}^2}{\| \mathbf{x}_n \|^2}\right] &= \mu\Big\{\{E[\alpha_n^2] + 12(E[\alpha_n] - E[\alpha_n^2])\sigma_w^2 \\ &\quad +36(1 - 12E[\alpha_n] + E[\alpha_n^2])\phi_w^4\}E\left[\frac{e_{an}^2}{\| \mathbf{x}_n \|^2}\right] \\ &\quad +\{E[\alpha_n^2]\sigma_w^2 + 4(E[\alpha_n] - E[\alpha_n^2])\phi_w^4 \\ &\quad +4E[(1-\alpha_n)^2]\phi_w^6\}E\left[\frac{1}{\| \mathbf{x}_n \|^2}\right]\Big\}. \end{aligned} \tag{2.52}$$

Two cases can be considered for the evaluation of the expression $E\left[\frac{e_{an}^2}{\|\mathbf{x}_n\|^2}\right]$. These are detailed in the ensuing analysis. But before doing the analysis one more assumption is stated which is needed for the derivations.

A5 At steady-state, $\| \mathbf{x}_n \|^2$ is statistically independent of e_{an}^2

Case 1: Under the assumption **A5**, expression $E\left[\frac{e_{an}^2}{\|\mathbf{x}_n\|^2}\right]$ will look like:

$$E\left[\frac{e_{an}^2}{\| \mathbf{x}_n \|^2}\right] = E\left[e_{an}^2\right]E\left[\frac{1}{\| \mathbf{x}_n \|^2}\right], \tag{2.53}$$

so that Equation (2.52) leads to the excess MSE (ζ) for the proposed algorithm:

$$\zeta = \frac{\mu Y1}{Y2}. \tag{2.54}$$

where, $Y1$ and $Y2$ are defined as follows:

$$Y1 = \left\{E[\alpha_n^2]\sigma_w^2 + 4(E[\alpha_n] - E[\alpha_n^2])\phi_w^4 + 4(1 - 2E[\alpha_n] + E[\alpha_n^2])\phi_w^6\right\}, \tag{2.55}$$

and,

$$Y2 = 2E[\alpha_n] - \mu E[\alpha_n^2] + \{12(1 - E[\alpha_n]) - 12\mu(E[\alpha_n] - E[\alpha_n^2])\}\sigma_w^2 - 36\mu(1 - 2E[\alpha_n] + E[\alpha_n^2])\phi_w^4. \tag{2.56}$$

The excess MSE for the Normalized LMS (NLMS) algorithm can be obtained by substituting the value $\alpha_n = 1$ in Equation (2.54):

$$\zeta_{NLMS} = \frac{\mu\sigma_w^2}{2 - \mu}. \tag{2.57}$$

Similarly, for the Normalized LMF (NLMF) algorithm, $\alpha_n = 0$ is substituted in Equation (2.54),

$$\zeta_{NLMF} = \frac{\mu\phi_w^6}{3\sigma_w^2 - 9\mu\phi_w^4}. \tag{2.58}$$

Case 2: In the second case, the assumption of ([5],pp. 443) is used to evaluate $E\left[\frac{e_{an}^2}{\|\mathbf{x}_n\|^2}\right]$, i.e.

$$E\left[\frac{e_{an}^2}{\| \mathbf{x}_n \|^2}\right] \approx \frac{E[e_{an}^2]}{E[\ \| \mathbf{x}_n \|^2\ ]}. \tag{2.59}$$

Thus, Equation (2.52) leads to the excess MSE for the proposed algorithm:

$$\zeta = \frac{\mu Y1}{Y2} E[\frac{1}{\| \mathbf{x}_n \|^2}] tr\{\mathbf{R}\}. \tag{2.60}$$

We can derive the expressions for NLMS and NLMF algorithms from the above generalized equation. Thus, for the case of NLMS algorithm, the above expression is reduced to the following:

$$\zeta_{NLMS} = \frac{\mu \sigma_w^2}{2 - \mu} E\Big[\frac{1}{\| \mathbf{x}_n \|^2}\Big] tr\{\mathbf{R}\}. \tag{2.61}$$

Similarly for the NLMF algorithm, Equation (2.60) can be set to the following:

$$\zeta_{NLMF} = \frac{\mu \phi_w^6}{3\sigma_w^2 - 9\mu \phi_w^4} E\Big[\frac{1}{\| \mathbf{x}_n \|^2}\Big] tr\{\mathbf{R}\}. \tag{2.62}$$

To find the EMSE, we have to find the steady-state values of both mean and mean square value of α_n.The steady state value for the mean of the mixing parameter in terms of the steady-state mean-square error. From Equation (2.5), p_n can be set recursively in the following form:

$$p_n = (1 - \beta) \sum_{i=0}^{n-1} \beta^i e_{n-i} e_{n-i-1}, \tag{2.63}$$

and therefore

$$p_n^2 = (1 - \beta)^2 \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \beta^i \beta^j e_{n-i} e_{n-i-1} e_{n-j} e_{n-j-1}. \tag{2.64}$$

Now, we are in a position to find an expression for the mean of the mixing parameter, namely $E[\alpha_n]$ and it is given by

$$E[\alpha_{n+1}] = \delta E[\alpha_n] + \gamma (1 - \beta)^2 \sum_{i=0}^{n-1} \beta^{2i} E[e_{n-i}^2] E[e_{n-i-1}^2]. \tag{2.65}$$

In the above Equation, we have assumed that the algorithm has converged, and in this case the samples of the error e_n can be assumed uncorrelated, i.e.,

$$E[e_{n-i}e_{n-j}] = 0, \forall\, i \neq j. \tag{2.66}$$

Ultimately, the the steady state value for the mean of the mixing parameter is given by

$$\begin{aligned} E[\alpha_\infty] &= \frac{\gamma(1-\beta)\left\{E[e_\infty^2]\right\}^2}{(1-\delta)(1+\beta)} \\ &= \frac{\gamma(1-\beta)[\epsilon_{min}+\zeta_{excess}]^2}{(1-\delta)(1+\beta)}. \end{aligned} \tag{2.67}$$

To end this study, the value of $E[\alpha_\infty^2]$ can evaluated and found to be:

$$E[\alpha_\infty^2] = \frac{2\gamma^2\delta(1-\beta)^2\left[\epsilon_{min}+\zeta_{excess}\right]^4}{(1-\delta^2)(1-\delta)(1+\beta)^2}. \tag{2.68}$$

The derivations for the $E[\alpha_\infty]$ and $E[\alpha_\infty^2]$ has already done in [26].

Remarks

In the above two cases considered, it is found that the steady-state excess MSE of the proposed algorithm is dependent on the mean and mean-square value of the mixing parameter, α_n, choice of step-size parameter, μ, fourth and sixth moments of noise (ϕ_w^4 and ϕ_w^6). The steady-state performance of the proposed algorithm can be determined from the knowledge of these parameters. The selection of these parameters are very important and better performance of the adaptive algorithm can be achieved by their appropriate choice.

Chapter 3

Tracking Analysis In Cyclicly And Randomly Varying Environments

Cyclic and random system nonstationarities are common impairment in communication systems and especially in applications that involve channel estimation, channel equalization, and inter-symbol-interference cancellation. Cyclic system nonstationarities arise in communication systems due to mismatches between the transmitter and receiver carrier generator. The ability of adaptive filtering algorithms to track such systems variations are not fully understood. A recent contribution in this regard is the work [46], which performed a first order analysis of the performance of the LMS algorithm in the presence of the carrier frequency offset. In [47], first time a general framework for the tracking analysis of adaptive algorithms was developed that can handle both cyclic as well as random system nonstationarities simultaneously. The framework was based on energy conversation relation which was first noted in [48]-[44], holds for all adaptive algorithms whose recursion are of the form (2.38), i.e., given by:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu \mathbf{x}_n f(e_n).$$

where $f(e_n)$ denotes a general scalar function of the output estimation error e_n. In the case of the proposed algorithm, the function $f(e_n) = \dfrac{\{\alpha_n e_n + 2(1-\alpha_n)e_n^3\}}{\| \mathbf{x}_n \|^2}$, with α_n being the time varying mixing parameter.

3.1 The System Model

Consider the noisy measurements d_n that arise from a model of the form:

$$d_n = \mathbf{x}_n^T \mathbf{w}_n^o e^{j\Omega n} + \xi_n, \tag{3.1}$$

where ξ_n is the measurement noise and $\mathbf{w}_n^o$ is the unknown system that is to be tracked. The multiplicative term $e^{j\Omega n}$ accounts for a possible frequency offset between the transmitter and receiver carriers in a digital communication scenario. Furthermore it is assumed that the unknown system vector $\mathbf{w}_n^o$ is randomly changing according to:

$$\mathbf{w}_n^o = \mathbf{w}^o + \mathbf{q}_n. \tag{3.2}$$

where $\mathbf{w}^o$ is a fixed vector, and $\mathbf{q}_n$ is assumed to be a zero-mean stationary random vector process with a positive definite covariance matrix $\mathbf{Q}_n$. It is also statistically independent of the sequences $\{\xi_n\}$ and $\{x_n\}$.

Thus, from the generalized system model, given by Equation (3.1) and Equation (3.2), it can be seen that the effects of both cyclic and random system nonstationarities are included in it.

3.2 Steady-State Tracking Performance Measure

In the steady-state analysis of an adaptive algorithm, an important measure is its steady-state mean square error (MSE), which is defined by Equation (2.39) and reproduced here for clearity :

$$\begin{aligned} \text{MSE} &= \lim_{n\to\infty} E[e_n^2] && (3.3)\\ &= \lim_{n\to\infty} E\{[\xi_n + \mathbf{x}_n^T\mathbf{v}_n]^2\}. && (3.4)\end{aligned}$$

where $\mathbf{v}_n$ is the weight error vector defined as:

$$\mathbf{v}_n = \mathbf{w}_n^o e^{j\Omega n} - \mathbf{w}_n, \tag{3.5}$$

Under assumption the **A2** that noise sequence is iid and statistically independent of the input sequence, the MSE is given by:

$$\text{MSE} = \sigma_w^2 + \lim_{n\to\infty} E\{[\mathbf{x}_n^T\mathbf{v}_n]^2\}. \tag{3.6}$$

The second term on the right hand side of the above equation is defined as steady-state excess mean square error (EMSE), denoted by ζ, and thus it is given by:

$$\zeta = \lim_{n\to\infty} E\{[\mathbf{x}_n^T\mathbf{v}_n]^2\}. \tag{3.7}$$

3.3 Fundamental Energy Relation

In this section, the fundamental energy conservation relation will be developed. Using the Equation (2.38) and Equation (3.2), following recursion has been derived in Appendix **E**:

$$\mathbf{v}_{n+1} = \mathbf{v}_n - \mu \mathbf{x}_n f(e_n) + \mathbf{c}_n e^{j\Omega n}, \tag{3.8}$$

where $\mathbf{c}_n$ is defined as:

$$\mathbf{c}_n = \mathbf{w}^o(e^{j\Omega n} - 1) + \mathbf{q}_{n+1} e^{j\Omega n} - \mathbf{q}_n. \tag{3.9}$$

Now, let's define the following so-called a priori estimation error, $e_{an} = \mathbf{x}_n^T \mathbf{v}_n$ and a posteriori estimation error, $e_{pn} = \mathbf{x}_n^T(\mathbf{v}_{n+1} - \mathbf{c}_n e^{j\Omega n})$. It is very easy to show that the estimation error, e_n, and the a-priori error, e_{an}, are related via $e_n = e_{an} + \xi_n$. Also the a-posteriori error, e_{pn}, is already defined by Equation (2.43). Substituting Equation (2.43) into Equation(2.38) results into the following update relation:

$$\mathbf{v}_{n+1} = \mathbf{v}_n - \hat{\mu}_n \mathbf{x}_n [e_{an} - e_{pn}] + \mathbf{c}_n e^{j\Omega n}. \tag{3.10}$$

By evaluating the energies of both the sides of the above equation, the new relation is obtained:

$$\| \mathbf{v}_{n+1} - \mathbf{c}_n e^{j\Omega n} \|^2 + \hat{\mu}_n \| e_{an} \|^2 = \| \mathbf{v}_n \|^2 + \hat{\mu}_n \| e_{pn} \|^2. \tag{3.11}$$

It can be seen that if $\Omega = 0$(i.e. no frequency offset between transmitter and receiver), both the Equations (2.45) and (3.11) match exactly. Thus, Equation (3.11) represents the generalized tracking model.

3.4 Tracking Analysis

The energy relation (3.11) will be used to evaluate the excess mean square error at the steady state. But before starting the analysis, first the following assumption is stated:

A6 In steady-state, the weight error vector $\mathbf{v}_n$ takes the generic form $\mathbf{z}_n e^{j\Omega n}$, with the stationary random process $\mathbf{z}_n$ independent of the frequency offset Ω. Let $\mathbf{z}$ denote E[$\mathbf{z}_n$].

Using the Equation (2.43), assumption **A6**, and taking expectation of both sides of Equation (3.11) and the fact that at the steady state $E[\mathbf{v}_{n+1}] = E[\mathbf{v}_n]$, following relation can be obtained:

$$\begin{aligned} E[\hat{\mu}_n \parallel e_{an} \parallel^2] &= 2tr\{\mathbf{Q}_n\} + \parallel \mathbf{w}^o \parallel^2 |1 - e^{j\Omega}|^2 - 2ReE[\mathbf{q}_n(\mathbf{z}_n - \mu \mathbf{x}_n f(e_n) e^{-j\Omega n})] \\ & -2Re[(1 - e^{j\Omega})^* \mathbf{w}^{o*} E(\mathbf{z}_n - \mu \mathbf{x}_n f(e_n) e^{-j\Omega n})] \\ & + E\Big[\hat{\mu}_n \parallel e_{an} - \frac{\mu}{\hat{\mu}_n} f(e_n) \parallel^2 \Big], \end{aligned} \tag{3.12}$$

where $tr\{\mathbf{Q}_n\} = E[\mathbf{q}_n \mathbf{q}_n^T]$. The above equation can be solved for the steady-state excess mean square error (EMSE).

To find the value of $\mathbf{z}$, Equation (3.8) is used. First of all it is multiplied by the term $e^{-j\Omega n}$ and then expectation is taken both sides to get:

$$(1 - e^{-j\Omega})\mathbf{z} = \mu E\Big(\mathbf{x}_n f(e_n) e^{-j\Omega n}\Big) + \mathbf{w}^o(1 - e^{-j\Omega}). \tag{3.13}$$

The above equation has been derived in Appendix **F** and can be used for evaluating the value of $\mathbf{z}$ at steady-state. First of all, the analysis is carried out for the NLMS algorithm and then for the proposed algorithm.

3.4.1 NLMS Algorithm

For the case of NLMS algorithm, the function $f(e_n)$ is $\frac{e_n}{\| \mathbf{x}_n \|^2}$. The value of $f(e_n)$ is substituted in Equation (3.13) and using **A2**, **A6**, solved for $\mathbf{z}$ as follows:

$$\begin{aligned} \mathbf{z} &= \frac{\mu}{(1-e^{-j\Omega})} E\Big[\mathbf{x}_n (\frac{e_{an}+\xi_n}{\| \mathbf{x}_n \|^2}) e^{-j\Omega n}\Big] + \mathbf{w}^o \\ &= \frac{\mu}{(1-e^{-j\Omega})} E\Big[\frac{\mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n e^{-j\Omega n}}{\| \mathbf{x}_n \|^2}\Big] + \mathbf{w}^o. \end{aligned} \tag{3.14}$$

Considering the approximation used in [5], it can be written as:

$$E\Big[\frac{\mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n e^{-j\Omega n}}{\| \mathbf{x}_n \|^2}\Big] \approx \frac{E[\mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n e^{-j\Omega n}]}{E[\| \mathbf{x}_n \|^2]}. \tag{3.15}$$

Thus, the Equation (3.14) can be set to the following form:

$$\begin{aligned} \mathbf{z} &= \frac{\mu}{(1-e^{-j\Omega})} \frac{E[\mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n e^{-j\Omega n}]}{E[\| \mathbf{x}_n \|^2]} + \mathbf{w}^o \\ &= \frac{\mu}{(1-e^{-j\Omega})} \frac{\mathbf{R}\mathbf{z}}{tr\{\mathbf{R}\}} + \mathbf{w}^o \\ &= \left[\mathbf{I} - \frac{\mu \mathbf{R}}{(1-e^{-j\Omega}) tr\{\mathbf{R}\}}\right]^{-1} \mathbf{w}^o. \end{aligned} \tag{3.16}$$

where $\mathbf{R} = E[\mathbf{x}_n \mathbf{x}_n^T]$. Now, lets go back to Equation (3.12)and is solved after substituting the value of $f(e_n)$ and using the results of Appendices **G** and **H** to find the following result:

$$\begin{aligned} \frac{E[|\mathbf{x}_n^T \mathbf{v}_n|^2]}{E[\| \mathbf{x}_n \|^2]} &= 2tr\{\mathbf{Q}_n\} + \| \mathbf{w}^o \|^2 \, |1-e^{j\Omega}|^2 - 2ReE[\mathbf{q}_n(\mathbf{z}_n \\ &\quad -\mu \mathbf{x}_n (\frac{\mathbf{x}_n^T \mathbf{v}_n + \xi_n}{\| \mathbf{x}_n \|^2}) e^{-j\Omega n})] - 2Re[(1-e^{j\Omega})^* \mathbf{w}^{o*} E(\mathbf{z}_n \end{aligned}$$

$$-\mu \mathbf{x}_n\left(\frac{\mathbf{x}_n^T \mathbf{v}_n + \xi_n}{\| \mathbf{x}_n \|^2}\right)e^{-j\Omega n})]$$
$$+E\Big[\frac{1}{\| \mathbf{x}_n \|^2}\Big\| e_{an} - \mu \| \mathbf{x}_n \|^2 \left(\frac{\mathbf{x}_n^T \mathbf{v}_n + \xi_n}{\| \mathbf{x}_n \|^2}\right)\Big\|^2\Big]. \tag{3.17}$$

After solving the expectation terms, we will obtain:

$$\begin{aligned}\frac{\zeta^{NLMS}}{tr\{\mathbf{R}\}} &= 2tr\{\mathbf{Q}_n\} + \| \mathbf{w}^o \|^2 \, |1 - e^{j\Omega}|^2 - 2tr\{\mathbf{Q}_n\} + 2\mu \frac{tr\{\mathbf{Q}_n \mathbf{R}\}}{tr\{\mathbf{R}\}} \\ & -2Re\Big[(1 - e^{j\Omega})^* \mathbf{w}^{o*}\Big(\mathbf{z} - \mu \frac{\mathbf{R}\mathbf{z}}{tr\{\mathbf{R}\}}\Big)\Big] \\ & +(1-\mu)^2 \frac{\zeta^{NLMS}}{tr\{\mathbf{R}\}} + \frac{\mu^2 \sigma_w^2}{tr\{\mathbf{R}\}}.\end{aligned} \tag{3.18}$$

Substituting the value of $\mathbf{z}$ in the above equation, it is solved for ζ in Appendix **I** to get the following:

$$\boxed{\zeta^{NLMS} = \frac{tr\{\mathbf{R}\}}{(2\mu - \mu^2)}\Big\{\frac{2\mu tr\{\mathbf{Q}_n \mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2 \sigma_w^2}{tr\{\mathbf{R}\}} + \beta_o\Big\}.} \tag{3.19}$$

where $\beta_o = |1 - e^{-j\Omega}|^2 Re\Big\{tr(\mathbf{W}^o(\mathbf{I} - 2\mathbf{X}))\Big\}$, $\mathbf{W}^o$and $\mathbf{X}$ are defined, respectively, as:

$$\mathbf{W}^o = \mathbf{w}^o \mathbf{w}^{o*}, \tag{3.20}$$

and

$$\mathbf{X} = (\mathbf{I} - \frac{\mu \mathbf{R}}{tr\{\mathbf{R}\}})[\mathbf{I} - \frac{\mu \mathbf{R}}{tr\{\mathbf{R}\}} - e^{-j\Omega}\mathbf{I}]^{-1}. \tag{3.21}$$

Approximate ζ^{NLMS} for the Gaussian white input

If it is assumed that Gaussian white input signal is available then the approximate expression for the ζ^{NLMS} can be obtained. For gaussian white input signal following properties are valid:

$$\mathbf{R} = \sigma_x^2 \mathbf{I}, \tag{3.22}$$

$$tr\{\mathbf{R}\} = N\sigma_x^2. \tag{3.23}$$

where N is the filter length. Now using the above facts for the gaussian white input signal, Equation (3.19) can be set to the following:

$$\zeta^{NLMS} = \frac{1}{(2-\mu)}\Big[2\sigma_x^2 tr\{\mathbf{Q}_n\} + \mu\sigma_w^2 + c\Big]. \tag{3.24}$$

where expression for c is derived in Appendix **J** and its result is given by:

$$c \;=\; \left[\frac{\sigma_x^2(2N-\mu)|1-e^{j\Omega}|^2}{|1-\frac{\mu}{N}-e^{-j\Omega}|^2} \parallel \mathbf{w}^o \parallel^2\right], \tag{3.25}$$

or can be expressed in the following form:

$$c \;=\; \Big[\frac{\sigma_x^2(2N-\mu)\{(1-cos\Omega)^2+sin^2\Omega\}}{\{(1-\frac{\mu}{N}-cos\Omega)^2+sin^2\Omega\}} \parallel \mathbf{w}^o \parallel^2\Big]. \tag{3.26}$$

To simplify the analysis, it is assumed that Ω is very small, so that the term $(1-cos\Omega)$ can be approximated to Ω. Also $\frac{\mu}{N} >> (1-cos\Omega)$, which is usually valid in practical cases. Thus, c can be approximated as:

$$c \approx \Big[\frac{N^2\sigma_x^2(2N-\mu)\Omega^2}{\mu^2} \parallel \mathbf{w}^o \parallel^2\Big] \tag{3.27}$$

Hence, the approximate result for the ζ^{NLMS} is:

$$\boxed{\zeta^{NLMS} \approx \frac{1}{(2-\mu)}\Big[2\sigma_x^2 tr\{\mathbf{Q}_n\} + \mu\sigma_w^2 + \frac{N^2\sigma_x^2(2N-\mu)\Omega^2}{\mu^2} \parallel \mathbf{w}^o \parallel^2\Big].} \tag{3.28}$$

Remark

Thus From the above result it can be seen that, unlike the stationary case, the steady-state EMSE is not a monotonically increasing function of the step-size μ. The EMSE is composed of three terms. The first term increases with the random nonstationarity term $tr\{\mathbf{Q}_n\}$. The second term increases with the step-size μ and the noise variance σ_w^2. The third term decreases with μ and increases with the frequency offset Ω. This term becomes dominant for small values of μ and causes the EMSE to increase with the order of μ^2.

3.5 Proposed Algorithm

For the case of the proposed algorithm $\dfrac{f(e_n) = \{\alpha_n e_n + 2(1-\alpha_n)e_n^3\}}{\| \mathbf{x}_n \|^2}$. To simplify the analysis $f(e_n)$ can be approximated as follows:

$$f(e_n) \approx \frac{\{\alpha_n(e_{an} + \xi_n) + 2(1-\alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}}{\| \mathbf{x}_n \|^2}. \tag{3.29}$$

The value of $f(e_n)$ is substituted in Equation (3.13) and using **A2**, **A6**, solved for $\mathbf{z}$ as follows:

$$\mathbf{z} = \frac{\mu}{(1-e^{-j\Omega})} E\Big[\mathbf{x}_n \Big\{ \frac{\{\alpha_n(e_{an} + \xi_n) + 2(1-\alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}}{\| \mathbf{x}_n \|^2} \Big\} e^{-j\Omega n}\Big] + \mathbf{w}^o. \tag{3.30}$$

Using the same approximation as was used for the NLMS algorithm, the expectation in the above equation can be solved as:

$$\begin{aligned} \mathbf{z} &= \frac{\mu}{(1-e^{-j\Omega})}\Big[E[\alpha_n]\frac{\mathbf{Rz}}{tr\{\mathbf{R}\}} + 6(1-E[\alpha_n])\sigma_w^2 \frac{\mathbf{Rz}}{tr\{\mathbf{R}\}}\Big] + \mathbf{w}^o \\ &= \frac{\mu}{(1-e^{-j\Omega})}\gamma_o \mathbf{Rz} + \mathbf{w}^o \\ &= \Big[\mathbf{I} - \frac{\mu\gamma_o\mathbf{R}}{(1-e^{-j\Omega})}\Big]^{-1}\mathbf{w}^o. \end{aligned} \tag{3.31}$$

where γ_o is defined as:

$$\gamma_o = \frac{1}{tr\{\mathbf{R}\}}\Big[E[\alpha_n] + 6E[\bar{\alpha_n}]\sigma_w^2\Big]. \tag{3.32}$$

and $\bar{\alpha_n} = (1 - \alpha_n)$. Now Equation (3.12) is solved after substituting the value of $f(e_n)$ and each term of its right hand side is considered separately. Starting with the third term as follows:

$$\begin{aligned}
3^{rd} \text{ term} &= -2\text{Re E}\Big[\mathbf{q}_n\mathbf{z}_n - \mu\mathbf{q}_n\mathbf{x}_n\Big\{\frac{\alpha_n e_{an} + \alpha_n\xi_n + 2\bar{\alpha_n}(3e_{an}\xi_n^2 + \xi_n^3)}{\|\mathbf{x}_n\|^2}\Big\}e^{-j\Omega n}\Big] \\
&= -2\text{Re}\Big[E[\mathbf{q}_n\mathbf{z}_n] - \mu E[\alpha_n]\frac{E[\mathbf{q}_n\mathbf{x}_n e_{an}e^{-j\Omega n}]}{tr\{\mathbf{R}\}} \\
&\quad -6\mu E[\bar{\alpha_n}]\frac{E[\mathbf{q}_n\mathbf{x}_n e_{an}e^{-j\Omega n}]}{tr\{\mathbf{R}\}}\sigma_w^2\Big] \\
&= -2tr\{\mathbf{Q}_n\} + 2\mu\Big(E[\alpha_n] + 6E[\bar{\alpha_n}]\sigma_w^2\Big)\frac{tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}}. \qquad (3.33)
\end{aligned}$$

The fourth term is given by:

$$\begin{aligned}
4^{th} \text{ term} &= -2Re\Big[(1 - e^{j\Omega})^*\mathbf{w}^{o*}E(\mathbf{z}_n - \mu\mathbf{x}_n\Big\{\frac{\alpha_n e_{an} + \alpha_n\xi_n + 2\bar{\alpha_n}(3e_{an}\xi_n^2 + \xi_n^3)}{\|\mathbf{x}_n\|^2}\Big\}e^{-j\Omega n})\Big] \\
&= -2Re\Big[(1 - e^{j\Omega})^*\mathbf{w}^{o*}\Big(\mathbf{z} - \mu E[\alpha_n]\frac{\mathbf{Rz}}{tr\{\mathbf{R}\}} - 6\mu E[\bar{\alpha_n}]\sigma_w^2\frac{\mathbf{Rz}}{tr\{\mathbf{R}\}}\Big)\Big] \\
&= -2Re\Big[(1 - e^{j\Omega})^*\mathbf{w}^{o*}(\mathbf{I} - \mu\gamma_o\mathbf{R})\mathbf{z}\Big] \\
&= -2|1 - e^{j\Omega}|^2\text{Re}\Big[\mathbf{W}^o\mathbf{X}_p\Big]. \qquad (3.34)
\end{aligned}$$

where, $\mathbf{X}_p = (\mathbf{I} - \mu\gamma_o\mathbf{R})[\mathbf{I} - \mu\gamma_o\mathbf{R} - e^{-j\Omega}\mathbf{I}]^{-1}$. Finally, the 5^{th} term is given by:

$$\begin{aligned}
5^{th} \text{ term} &= E[\frac{e_{an}^2}{\|\mathbf{x}_n\|^2}] - 2\mu E[e_{an}f(e_n)] + \mu^2 E[\|\mathbf{x}_n\|^2 f(e_n)^2] \\
&= E[\frac{e_{an}^2}{\|\mathbf{x}_n\|^2}] - 2\mu E\Big[e_{an}(\frac{\{\alpha_n(e_{an} + \xi_n) + 2(1 - \alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}}{\|\mathbf{x}_n\|^2})\Big] \\
&\quad +\mu^2 E\Big[\|\mathbf{x}_n\|^2(\frac{\{\alpha_n(e_{an} + \xi_n) + 2(1 - \alpha_n)(3e_{an}\xi_n^2 + \xi_n^3)\}}{\|\mathbf{x}_n\|^2})^2\Big] \\
&= \frac{\zeta^{prop}}{tr\{\mathbf{R}\}} - 2\mu\gamma_o\zeta^{prop} + \frac{\mu^2}{tr\{\mathbf{R}\}}(a\zeta^{prop} + b). \qquad (3.35)
\end{aligned}$$

where, ζ^{prop} is the steady-state excess mean square error for the proposed algorithm, while a and b are defined as follows:

$$\begin{aligned} a &= E[\alpha_n^2] + 36E[\bar{\alpha_n}^2]\phi_w^4 + 12E[\alpha_n]E[\bar{\alpha_n}]\sigma_w^2, \\ b &= E[\alpha_n^2]\sigma_w^2 + 4E[\bar{\alpha_n}^2]\phi_w^6 + 4E[\alpha_n]E[\bar{\alpha_n}]\phi_w^4. \end{aligned} \tag{3.36}$$

Now all these terms are substituted in Equation (3.12) and solved for ζ^{prop}:

$$\begin{aligned} \frac{\zeta^{prop}}{tr\{\mathbf{R}\}} &= 2tr\{\mathbf{Q}_n\} + \mathbf{W}^o|1-e^{j\Omega}|^2 - 2tr\{\mathbf{Q}_n\} + 2\mu\gamma_o tr\{\mathbf{Q}_n\mathbf{R}\} - 2|1-e^{j\Omega}|^2 Re[\mathbf{W}^o\mathbf{X}] \\ &+ \frac{\zeta^{prop}}{tr\{\mathbf{R}\}} - 2\mu\gamma_o\zeta^{prop} + \frac{\mu^2}{tr\{\mathbf{R}\}}(a\zeta^{prop} + b), \end{aligned} \tag{3.37}$$

or it can be set up into the following form:

$$\boxed{\zeta^{prop} = \frac{2\gamma_o tr\{\mathbf{R}\}}{(2\gamma_o tr\{\mathbf{R}\} - \mu a)}\Big[tr\{\mathbf{Q}_n\mathbf{R}\} + \frac{\beta_{op}}{2\mu\gamma_o} + \frac{b\mu}{2\gamma_o tr\{\mathbf{R}\}}\Big].} \tag{3.38}$$

where, $\beta_{op} = |1-e^{-j\Omega}|^2 Re\Big\{tr(\mathbf{W}^o(\mathbf{I}-2\mathbf{X}_p))\Big\}$. The above result can be verified by substituting the value of $\alpha_n = 1$ for the NLMS algorithm. If $\alpha_n = 1$, it is observed that $\gamma_o = \frac{1}{tr\{\mathbf{R}\}}$, $\beta_{op} = \beta_o$, $a = 1$ and $b = \sigma_w^2$. Thus after substituting these values in Equation (3.38), following result is obtained:

$$\zeta^{prop} = \frac{1}{(2\mu - \mu^2)}\Big\{2\mu tr\{\mathbf{Q}_n\mathbf{R}\} + \mu^2\sigma_w^2 + \beta_o tr\{\mathbf{R}\}\Big\} \tag{3.39}$$

which is the same as that of NLMS given by Equation (3.19).

Similarly substituting $\alpha_n = 0$ for the NLMF algorithm, we get$\gamma_o = \frac{6\sigma_w^2}{tr\{\mathbf{R}\}}$, $a = 36\phi_w^4$ and $b = 4\phi_w^6$. Thus excess mean square error for the NLMF algorithm is given by:

$$\boxed{\zeta^{NLMF} = \frac{\sigma_w^2}{\mu(\sigma_w^2 - 3\mu\phi_w^4)}\Big\{\mu tr\{\mathbf{Q}_n\mathbf{R}\} + \frac{\mu^2\phi_w^6}{3\sigma_w^2} + \frac{\beta_{op} tr\{\mathbf{R}\}}{12\sigma_w^2}\Big\}.} \tag{3.40}$$

Approximate ζ^{prop} for the Gaussian white input

If the input is Gaussian white signal then for small values of Ω, the approximate expression of steady-state excess mean square error, ζ^{prop}, can be obtained by using Equation (3.22), Equation (3.23) and Equation (3.38) as follows:

$$\zeta^{prop} = \frac{2p}{2p-\mu a}\Big[\sigma_x^2 tr\{\mathbf{Q}_n\} + \frac{b\mu}{2p} + c_p\Big]$$

where $c_p = \frac{\beta_{op}}{2\mu\gamma_o}$. It is simplified in the same as carried out for c in Equation (3.25) and its result is given by:

$$c_p = \left[\frac{\sigma_x^2(2N-\mu)|1-e^{j\Omega}|^2}{2|1-\frac{\mu p}{N}-e^{-j\Omega}|^2} \parallel \mathbf{w}^o \parallel^2\right], \tag{3.41}$$

or can be expressed in the following form:

$$c_p \;=\; \Big[\frac{\sigma_x^2(2N-\mu)\{(1-cos\Omega)^2+sin^2\Omega\}}{\{(1-\frac{\mu p}{N}-cos\Omega)^2+sin^2\Omega\}} \parallel \mathbf{w}^o \parallel^2\Big]. \tag{3.42}$$

To simplify the analysis, it is assumed that Ω is very small, so that the term $(1-cos\Omega)$ can be approximated to Ω. Also $\frac{\mu p}{N} >> (1-cos\Omega)$, which is usually valid in practical cases. Thus, c_p can be approximated as:

$$c_p \approx \Big[\frac{N^2\sigma_x^2(2N-\mu)\Omega^2}{2\mu^2p^2} \parallel \mathbf{w}^o \parallel^2\Big] \tag{3.43}$$

Hence, the approximate result for the ζ^{prop} is:

$$\boxed{\zeta^{prop} = \frac{2p}{2p-\mu a}\Big[\sigma_x^2 tr\{\mathbf{Q}_n\} + \frac{b\mu}{2p} + \frac{N^2\sigma_x^2(2N-\mu)\Omega^2}{2\mu^2p^2} \parallel \mathbf{w}^o \parallel^2\Big].} \tag{3.44}$$

where, p is given by:

$$p = E[\alpha_n] + 6(1-E[\alpha_n])\sigma_w^2. \tag{3.45}$$

Remarks

In this section, exact and approximate expressions of excess mean square error (EMSE) are derived for both NLMS and the proposed algorithm. Their expression shows that they depend on the values of step-size μ, variance of noise, variance of input sequence, frequency offset between transmitter and receiver, filter size. The EMSE is composed of three terms. The first term increases with the random nonstationarity term $tr\{\mathbf{Q}_n\}$. The second term increases with the frequency offset Ω, and decreases with ste-size. The third term increases with μ. The second term becomes dominant for small values of μ and causes the EMSE to increase with the order of μ^2.

3.6 Results of the comparison between Analytical and Experimental EMSE (ζ)

In this sections, the analytical results that are derived for the NLMS and the proposed algorithm in the previous section are compared with the experimental one. For the analytical results, approximated expressions for the steady-state error is used, i.e., Equation 3.28 and Equation 3.44, respectively, for the NLMS and the proposed algorithms.

Figure 3.1 and Figure 3.2 represent the comparison for the NLMS algorithm for the two different values of frequency offset Ω. The results show that analytical and experimental results are closer to each other for very small step-size, and then the difference is increases. But near the optimum value of the excess mean square error, they again come closer to each other. This shows that there exists an optimum value of the step-size at which the excess mean square error is minimum, and the analytical approximate expressions derived are more realistic in its vicinity. Figure 3.1 shows that at $\Omega = 0.001$, the optimum value of step-size is nearly 0.2, while at $\Omega = 0.002$ it is approximately equal to 0.25, as shown in Figure 3.2. This indicates that by increasing the frequency offset between the transmitter and the receiver, the NLMS algorithm is able to track the variations in the environment at higher values of step-size.

For the case of proposed algorithm, the approximated analytical results are far away from the experimental one atlower values of step-size, but they come close to each other at higher values of step-size. This shows that approximate expressions are more valid at higher values of step-size. By comparing the two Figures, i.e., Figure 3.3 and Figure 3.4, it is clear that by increasing the frequency offset, the ability of the proposed algorithm to track the variations in environment is degraded at smaller values of step-size.

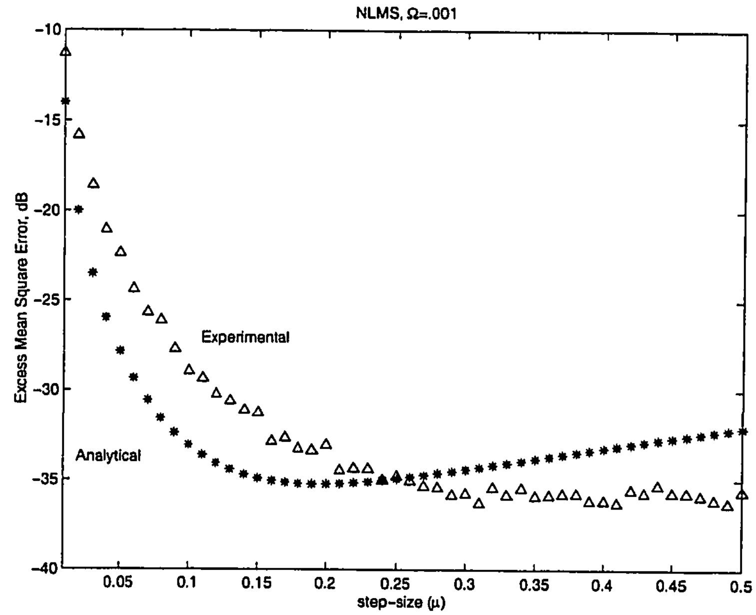

Figure 3.1: Comparison between Analytical and Experimental ζ_{NLMS} at $\Omega = .001$

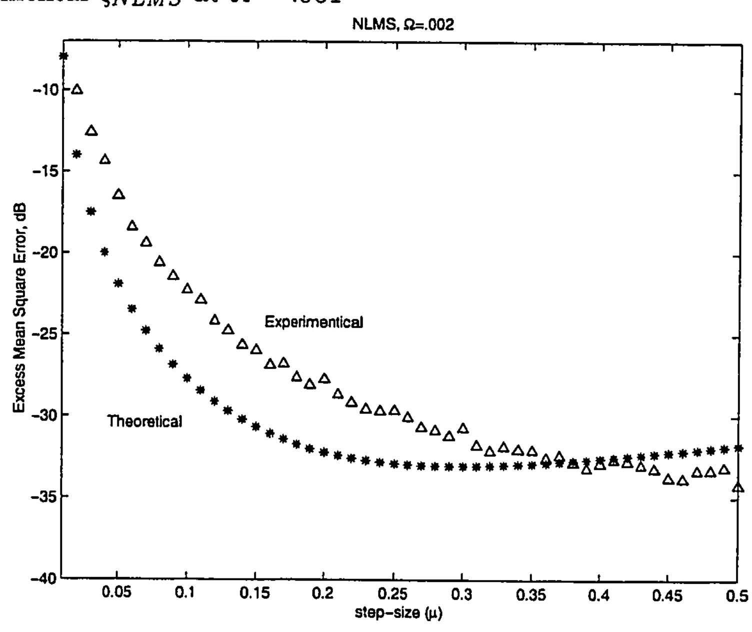

Figure 3.2: Comparison between Analytical and Experimental ζ_{NLMS} at $\Omega = .002$

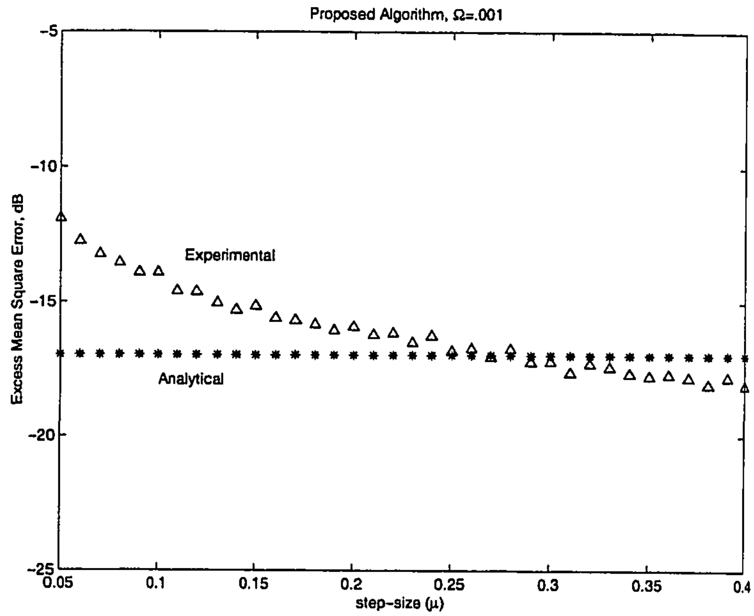

Figure 3.3: Comparison between Analytical and Experimental ζ_{prop} at $\Omega = .001$

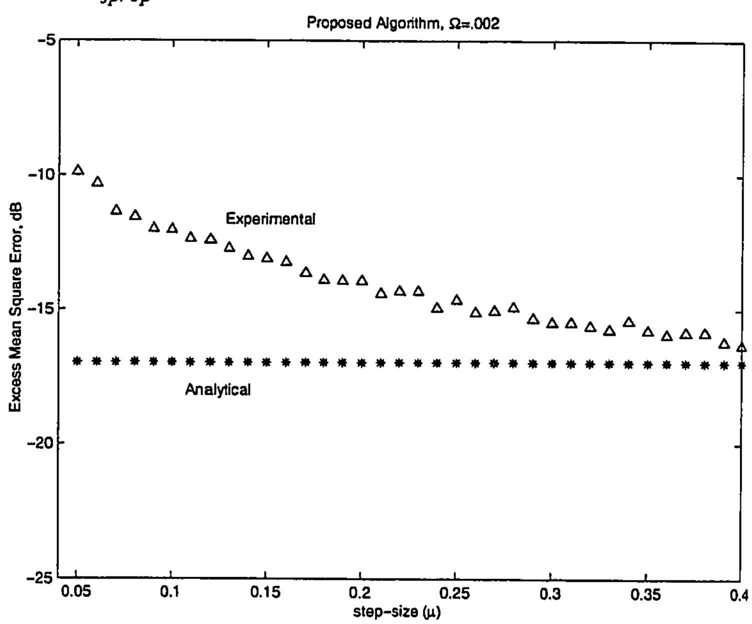

Figure 3.4: Comparison between Analytical and Experimental ζ_{prop} at $\Omega = .002$

Chapter 4

Simulation Results

In this chapter the results of the computer simulations are presented which are made to investigate the performance behaviors of the proposed time-varying normalized mixed norm LMS-LMF algorithm. These results are compared with the results of traditional Normalized Least Mean Square (NLMS) algorithm in unknown system identification problem which shows better performance of the proposed algorithm in terms of convergence speed and the steady-state error. The performance measure considered is the normalized weight error norm expressed in deci bell (dB)i.e. $(10log_{10}||\mathbf{w}_n - \mathbf{w}_{opt}||^2/||\mathbf{w}_{opt}||^2)$.

Traditional NLMS algorithm can achieve fastest convergence at step-size equal to 1 ($\mu = 1$), but its steady-state error is higher. The results have shown that the proposed algorithm can achieve a lower steady-state error at the same convergence speed. Another superiority of the proposed algorithm is that it can achieve the lowest possible steady-state error achievable by the traditional NLMS in lesser convergence time.

In our simulation work, the following performance measures are investigated:

- Steady-state Normalized Weight Error Norm,
- Convergence Time,
- Recovery time after an abrupt change in the environment, and
- Behavior of the time-varying mixing parameter(α_n).

All these performance measures are analyzed with both white (uncorrelated) and correlated input sequences. The performance is investigated in uniform and gaussian environment. The results are obtained by averaging over 50 independent runs.

4.1 The Adaptive System Identification Problem Scenario

We have chosen unknown systems having FIR model given by $[0.5,\ 1]^T$ to be identified. The input sequence $\mathbf{x}_n$ to both the unknown system and the adaptive filter is obtained by passing a zero mean uniformly distributed sequence through a channel $[0.3482,\ 0.8704,\ 0.3482]^T$ that is used to make the eigenvalue spread of the autocorrelation matrix of the input signal equal to 68.9. In gaussian environment the additive noise is zero mean gausssian distributed while in uniform environment it is zero mean uniformly distributed. The observation noise is uncorrelated with the input sequence. The variance of the noise is taken as 0.01 in one case and 0.001 in another case. The length of the adaptive filter is chosen equal to the length of the unknown system. Following are the aims of this simulation work:

1. Comparison of the convergence speed of the NLMS algorithm and Proposed algorithm in achieving the same steady-state error with
   - white input sequence, and
   - highly correlated input sequence.
2. Comparison of steady-state error of the NLMS algorithm and Proposed algorithm in achieving the fastest convergence with
   - white input sequence, and
   - highly correlated input sequence.
3. Comparison for the Recovery Time after an abrupt change in the environment with
   - white input sequence, and
   - highly correlated input sequence.
4. Comparison between the analytical and experimental mean square behaviour of the weight error vector.

Each case is considered separately in the following sections.

4.1.1 Comparison of the convergence speed of the NLMS algorithm and Proposed algorithm in achieving the same steady-state error with white input sequence

In this section the NLMS algorithm and proposed time-varying normalized mixed norm LMS-LMF algorithm are compared in terms of the convergence time when the input is white. In Figure 4.1, 4.5, 4.3, and 4.7, it is shown that the proposed algorithm has achieved the same noise floor in a lesser number of iterations as compared to the traditional NLMS algorithm. In Figure 4.1, it is shown that in 20 dB SNR and uniform environment the proposed algorithm achieved the same steady state in 18000 iterations earlier than the NLMS algorithm, while in 30 dB SNR, in uniform environment Figure 4.3, the proposed algorithm has converged nearly 20000 iterations earlier. Hence, it can be inferred from Figure 4.1 and Figure 4.3 that in uniform environment the proposed algorithm has better convergence speed in 30 dB SNR as compared to 20 dB SNR.

In Figure 4.5 and Figure 4.7 proposed lagorithm is compared with the NLMS algorithm in gaussian environment. In Figure 4.5, it is shown that both algorithm have same convergence time in the case of 20 dB SNR, while in the case of 30 dB SNR shown in Figure 4.7, there is a difference of 6000 iterations. So it can be concluded that in the gaussian environment, the proposed algorithm is performing better with 30 dB SNR.

The behavior of the time varying mixing parameter is also plotted for each case. The curves for the mixing parameter in the case of gaussian environment have slower convergence as compared to that of uniform environment as shown in Figure 4.6 and Figure 4.8. Thus, it shows the better convergence ability of the mixing parameter in uniform environment.

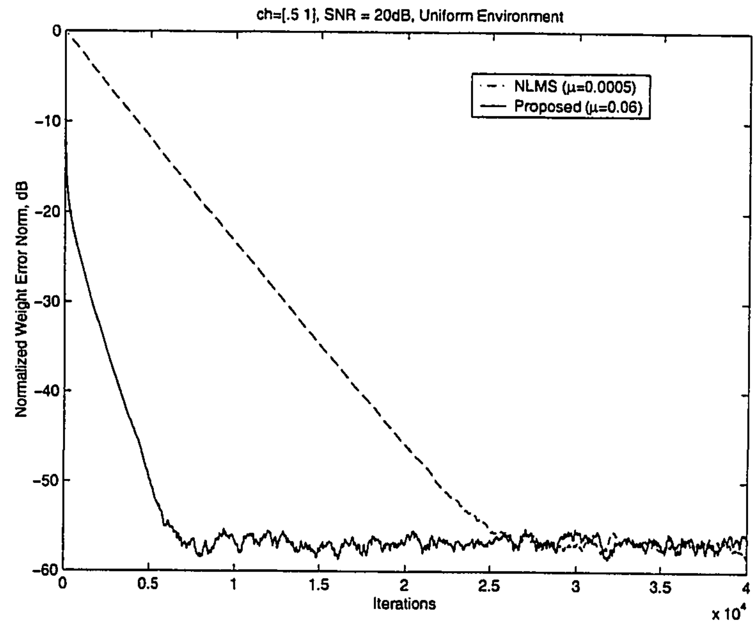

Figure 4.1: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

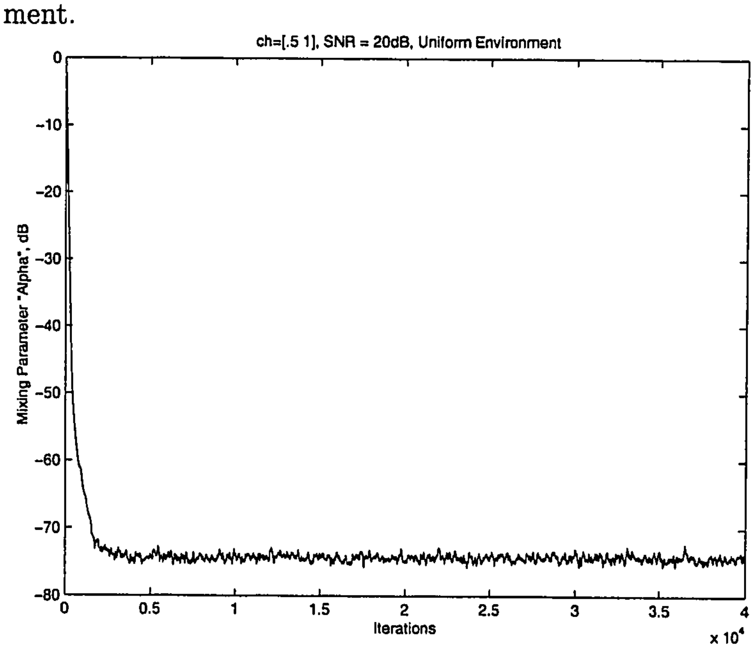

Figure 4.2: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

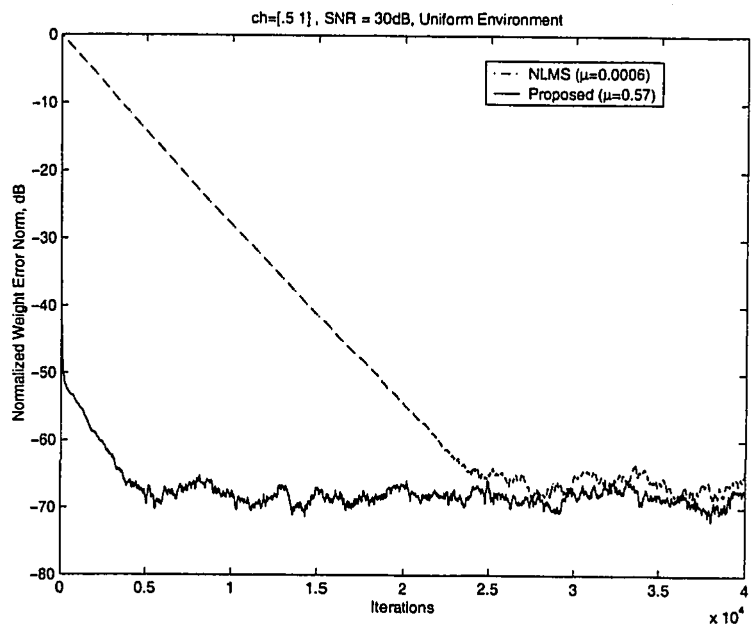

Figure 4.3: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in uniform environment.

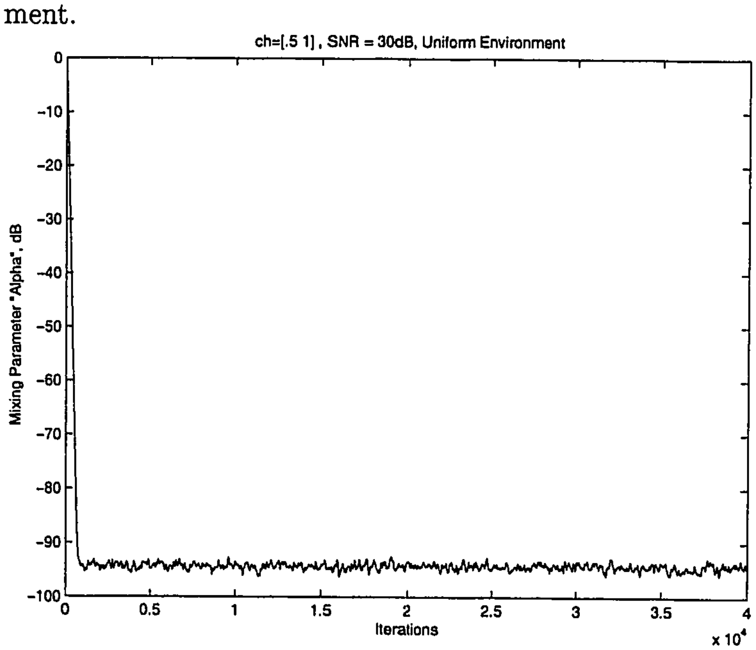

Figure 4.4: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

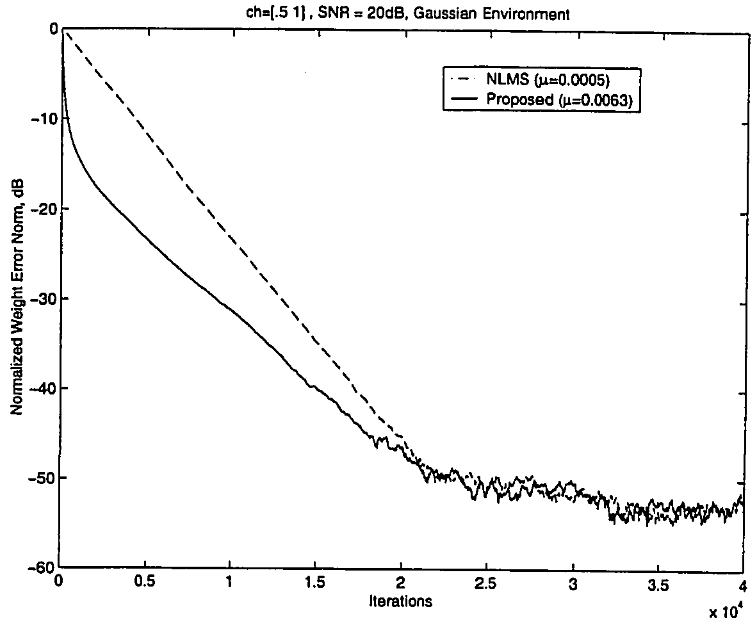

Figure 4.5: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

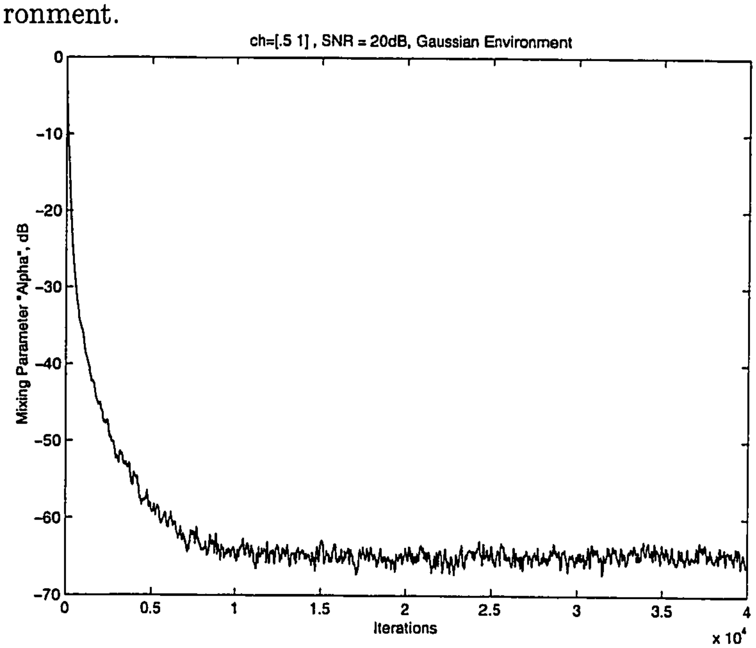

Figure 4.6: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

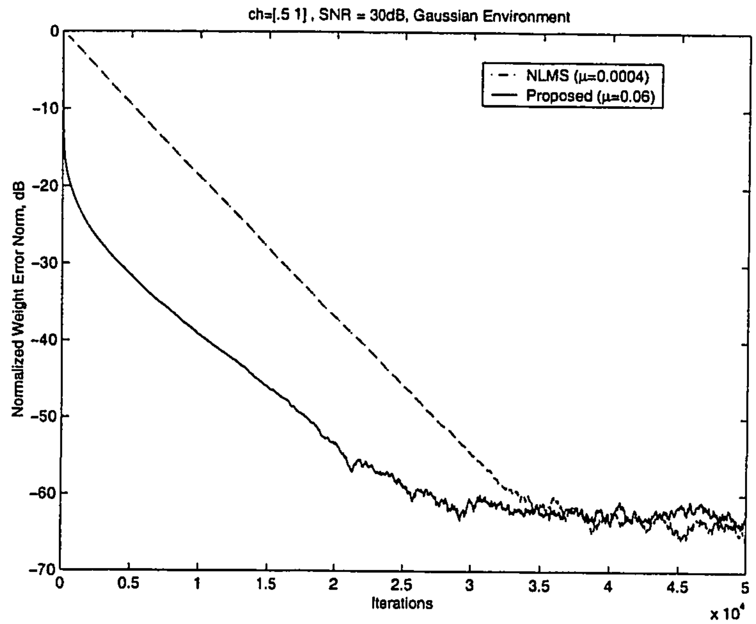

Figure 4.7: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment.

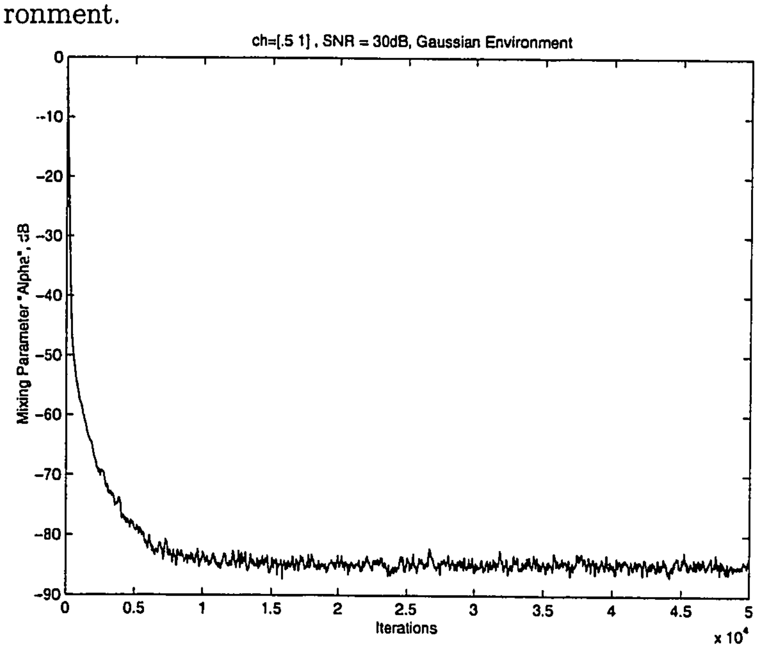

Figure 4.8: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.1.2 Comparison of steady-state error of the NLMS algorithm and Proposed algorithm for fastest convergence with white input sequence

In this section, the steady state error for the fastest convergence of NLMS algorithm and proposed algorithm are compared with white input sequence. In Figure 4.9, 4.13, 4.11 and 4.15, it is shown that the proposed algorithm can achieve the lower noise floor in its fastest convergence as compared to the traditional NLMS algorithm. The behavior of the time varying mixing parameter is also plotted for each case.

In Figure 4.9, in 20 dB SNR and uniform environment, it is shown that the proposed algorithm has achieved the steady state 35 dB below than the NLMS algorithm, while 32 dB below in the case of 30 dB SNR, as shown in Figure 4.11. In gaussian environment proposed algorithm has achieved the lower steady state as compared to the uniform environment. In case of 20 dB SNR, proposed algorithm has almost reached the -34 dB floor in gaussian environment and its final steady state is 15 dB below than that of NLMS algorithm. In 30 dB SNR, shown in Figure 4.13 and Figure 4.15, the proposed algorithm has reached nearly -50 dB floor but the margin between the two algorithm is greater than that of uniform environment, i.e., of 20 dB. So, it can be stated that the proposed algorithm has better performance in terms of steady state error in its fastest convergence and is further improved by increasing the SNR.

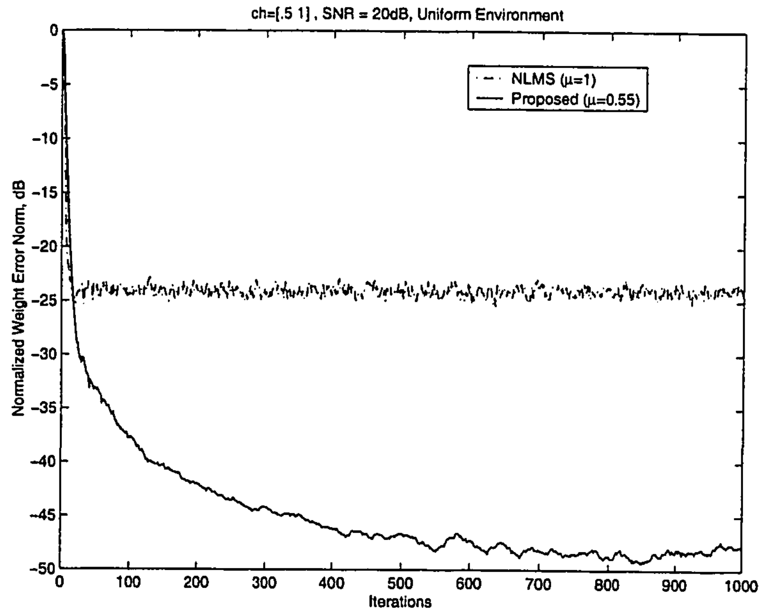

Figure 4.9: Comparison of the steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

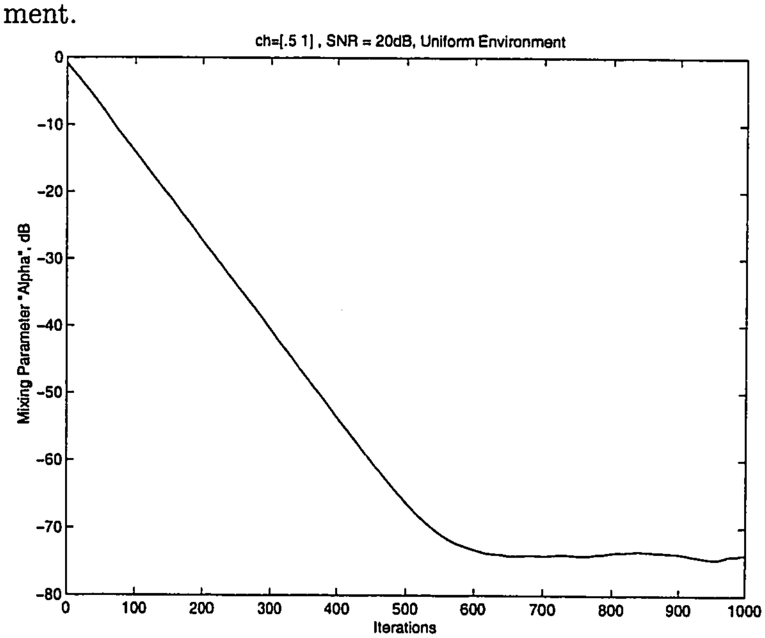

Figure 4.10: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

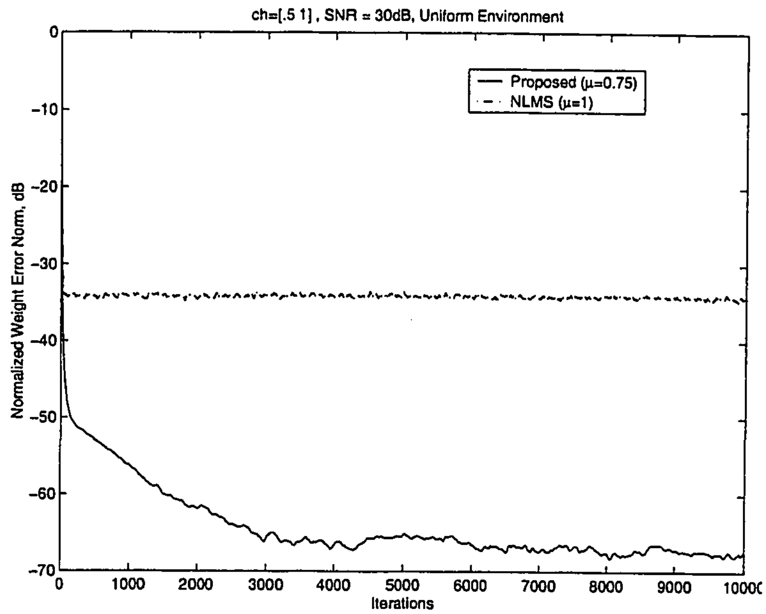

Figure 4.11: Comparison of steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in uniform environment.

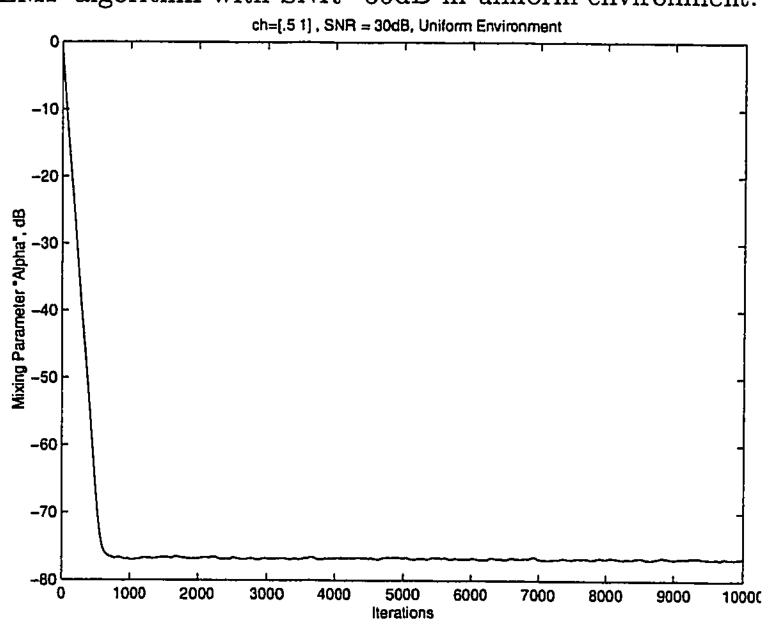

Figure 4.12: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

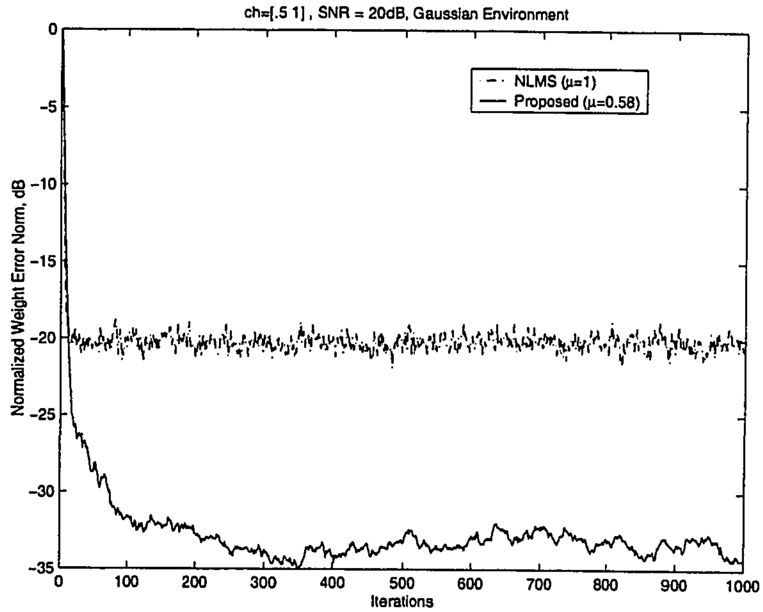

Figure 4.13: Comparison of steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

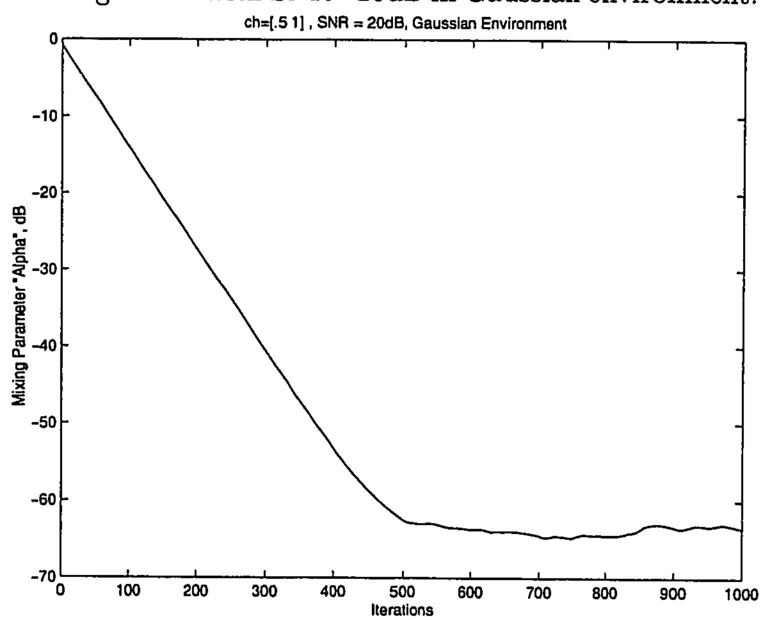

Figure 4.14: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

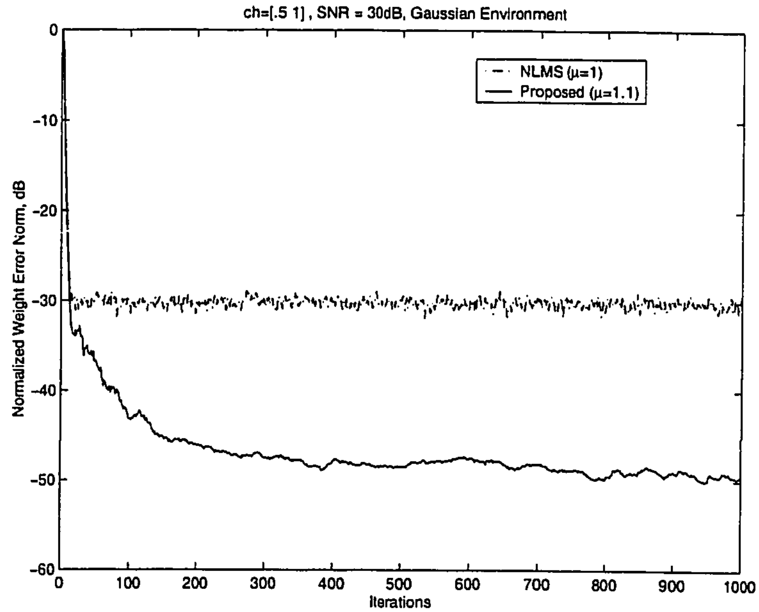

Figure 4.15: Comparison of steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment.

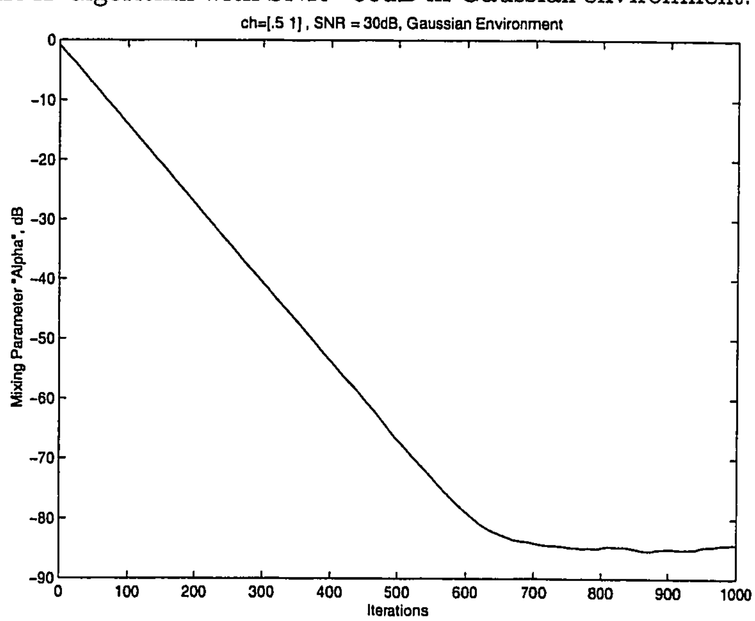

Figure 4.16: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.1.3 Comparison of the convergence speed of the NLMS algorithm and Proposed algorithm in achieving the same steady-state error with highly correlated input sequence

Here same analysis is carried out as it has been done in section 4.1.1 but for highly correlated input sequence $\mathbf{x}_n$ which is produced by using the relation given in [18]

$$\mathbf{x}_n = 0.9 * \mathbf{x}_{n-1} + \mathbf{a}_n, \tag{4.1}$$

where $\mathbf{a}_n$ is zero mean, uncorrelated gaussian noise of unity variance. This input is resulted in flattened elliptical contours, which usually cause difficulties in the convergence of gradient algorithms. When this input sequence is used, it is observed that with 20 dB SNR, the proposed algorithm has better performance both in uniform and gaussian environment. In uniform environment Figure 4.17, it has reached the same steady state nearly 300 iterations earlier while in gaussian Figure 4.21, approximately 3000 iterations earlier. This indicates that it is more advantageous in gaussian environment. But in 30 dB SNR it has better performance only in gaussian environment in which it has achieved the same steady state 600 iterations earlier, shown in Figure 4.23. In uniform environment Figure 4.19, its final steady state is just above than that of the NLMS algorithm. The behaviour of the mixing parameter is same for all the cases except that of 30 dB SNR and uniform environment in which the mixing parameter has slower convergence.

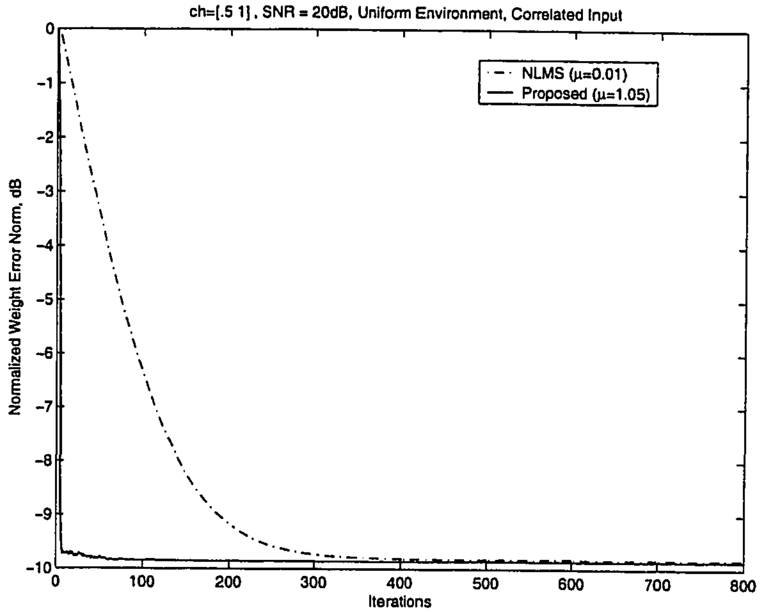

Figure 4.17: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

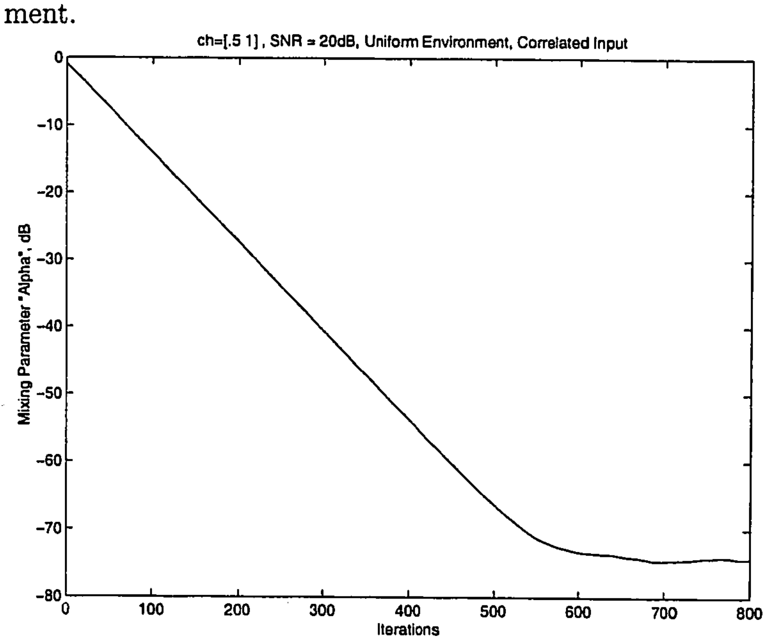

Figure 4.18: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

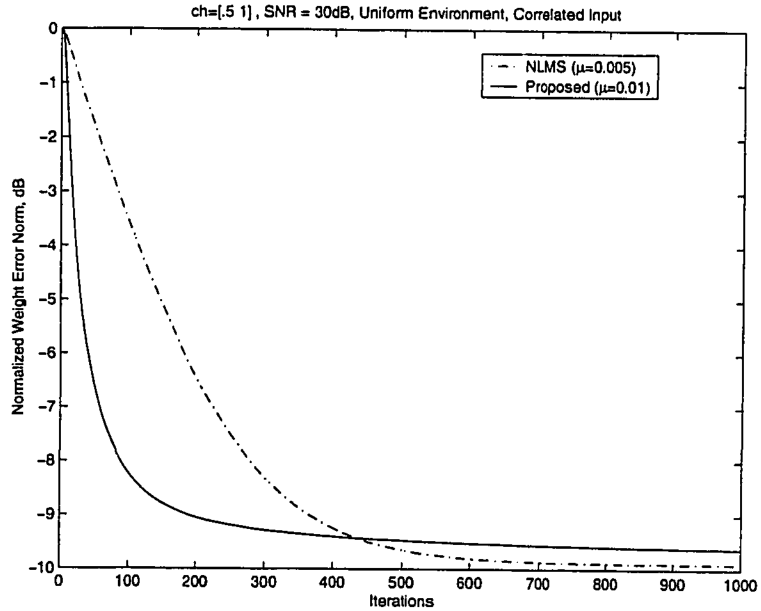

Figure 4.19: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in uniform environment.

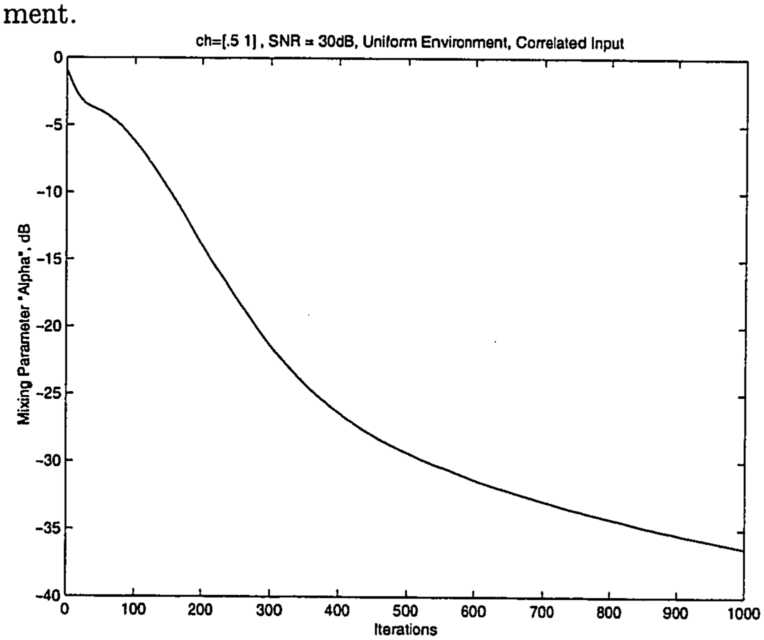

Figure 4.20: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

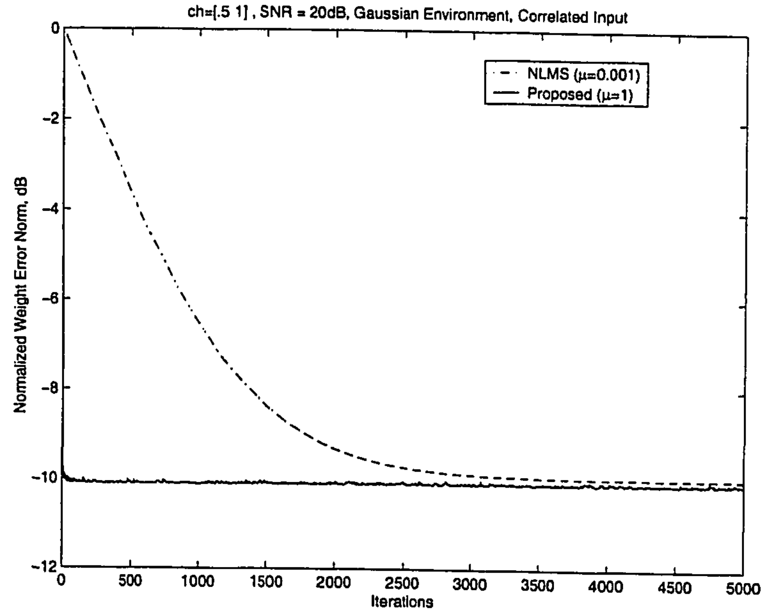

Figure 4.21: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

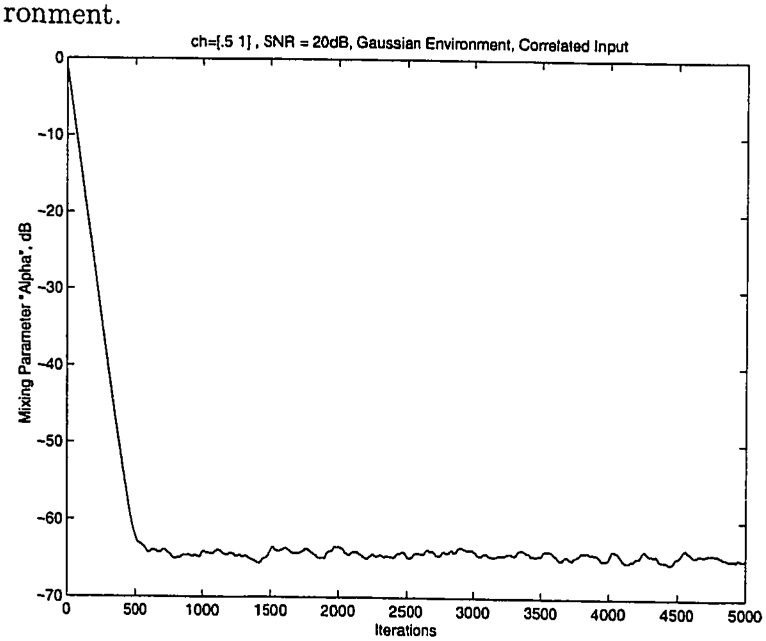

Figure 4.22: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

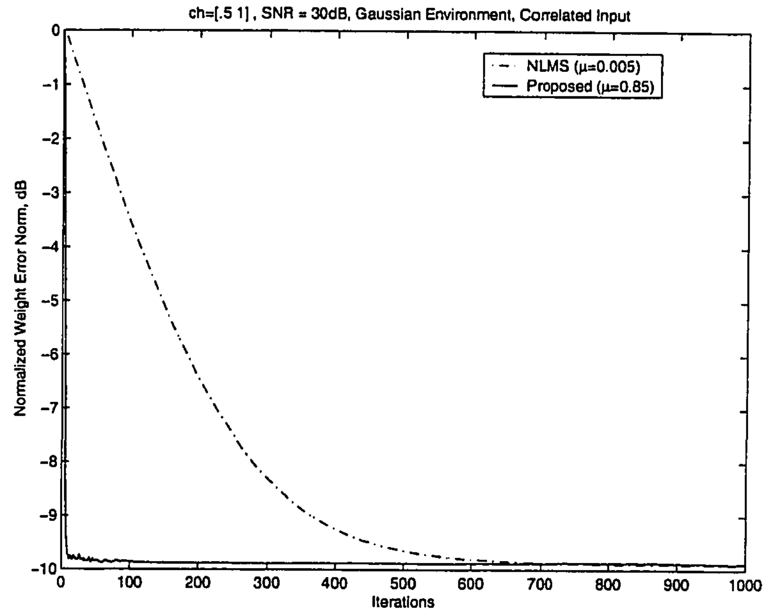

Figure 4.23: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment.

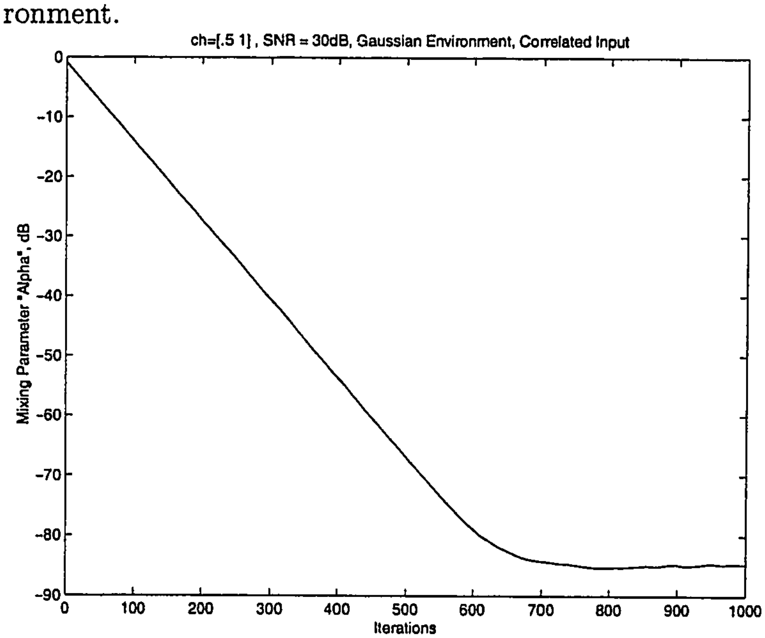

Figure 4.24: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.1.4 Comparison of steady-state error of the NLMS algorithm and Proposed algorithm for fastest convergence with highly correlated input sequence

Here the steady state error for the fastest convergence of NLMS algorithm and proposed algorithm are compared for the highly correlated input similar to that has been done in section 4.1.2. High correlated sequence is obtained with the same relation given by Equation (4.1). In uniform environment with 20 dB SNR Figure 4.25, it is shown that the proposed algorithm has achieved almost the same steady state in its fastest convergence as that of traditional NLMS algorithm, while with 30 dB SNR its steady state 1.5 dB below than that of NLMS algorithm, Figure 4.27. In the case of gaussian environment with 20 dB SNR, Figure 4.29, the proposed algorithm has superiority of 2 dB over the traditional NLMS algorithm while with 30 dB SNR, Figure 4.31, it is nearly 3 dB below than that of NLMS algorithm. Thus it can be inferred that the proposed algorithm with highly correlated input sequence has better performance in almost all the cases but it has best performance in gaussian environment with 30 dB SNR

The behavior of the time varying mixing parameter is also plotted for each case. Its convergence speed is almost the same as in the case of achieving the same steady state, and these are reported in Figures 4.26, 4.28, 4.30, and 4.32.

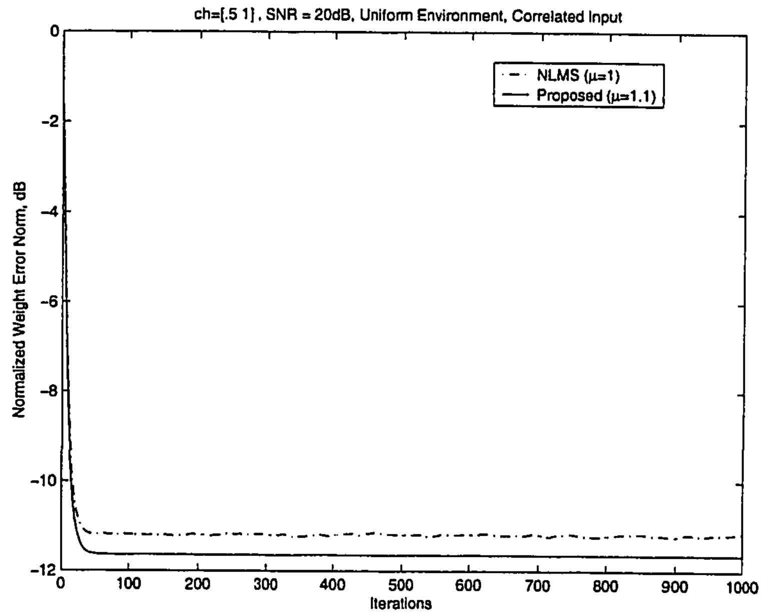

Figure 4.25: Comparison of the steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

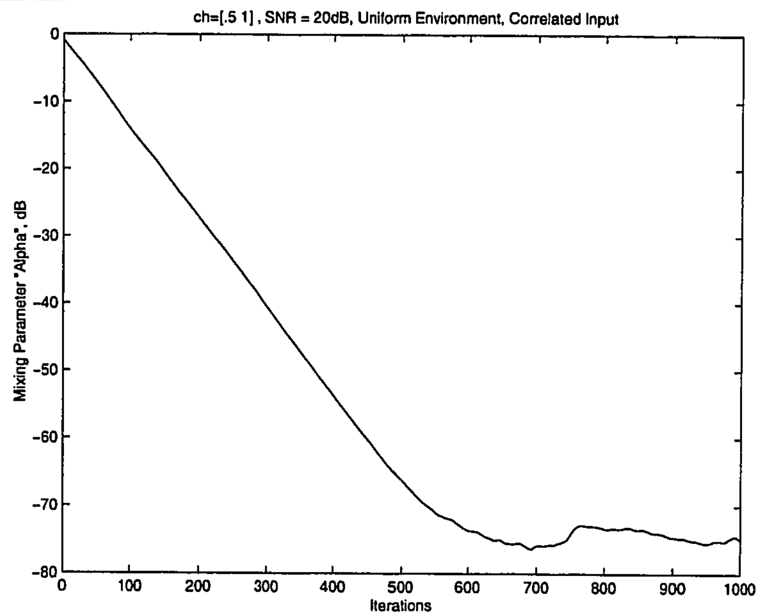

Figure 4.26: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

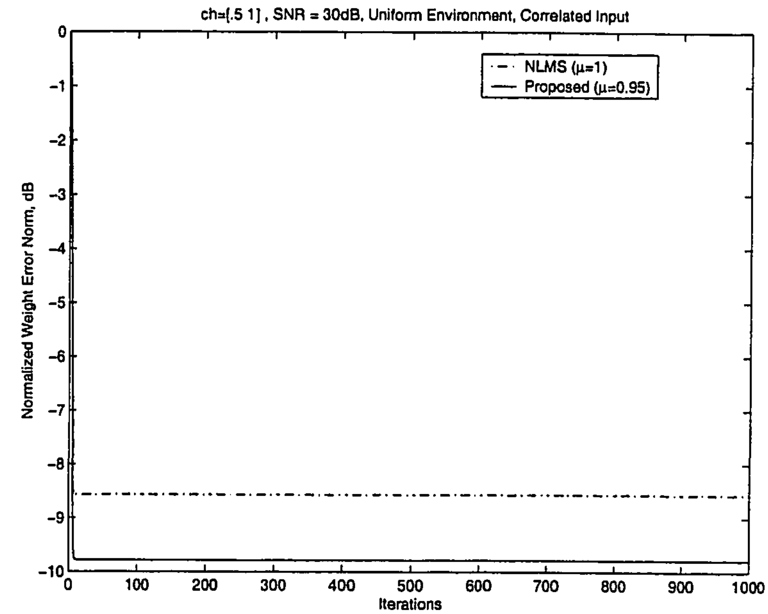

Figure 4.27: Comparison of the steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in uniform environment.

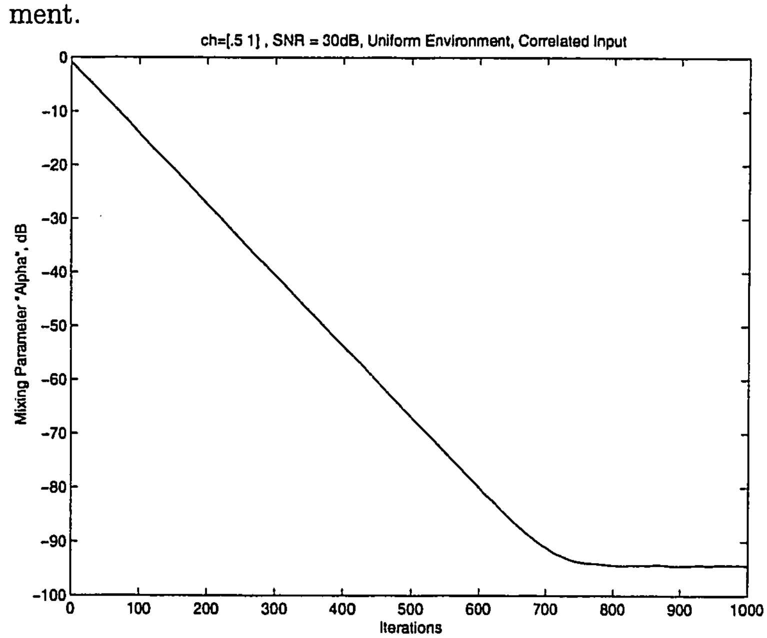

Figure 4.28: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

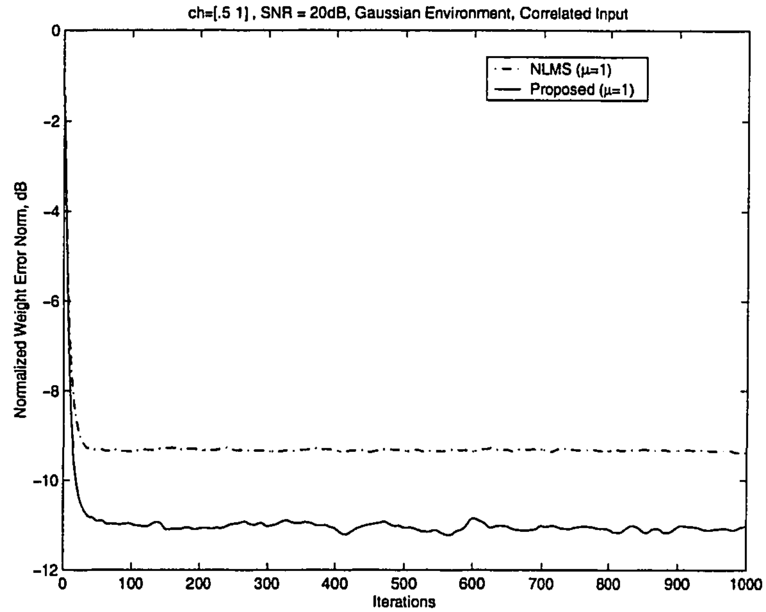

Figure 4.29: Comparison of steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

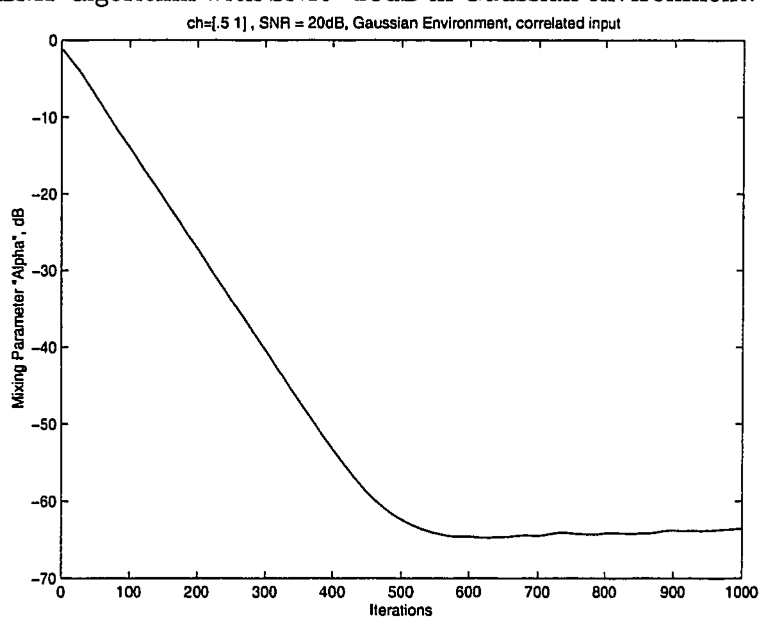

Figure 4.30: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

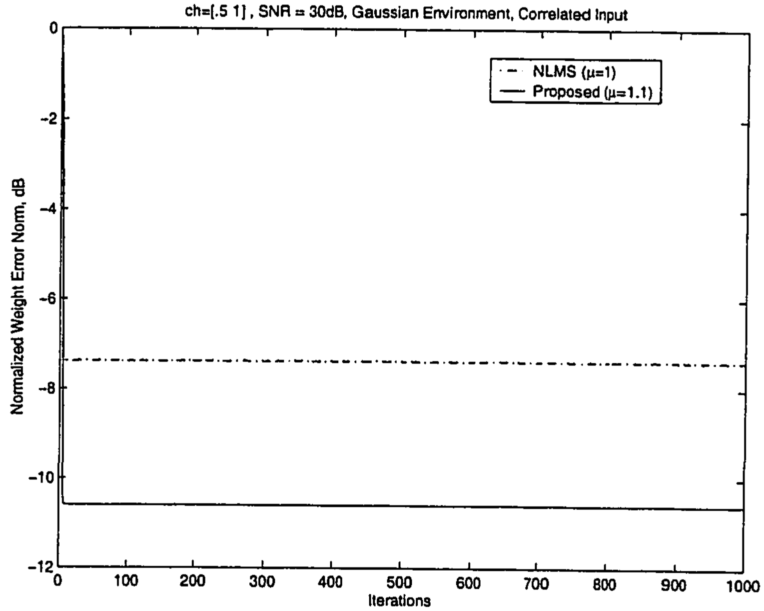

Figure 4.31: Comparison of steady-state error of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment.

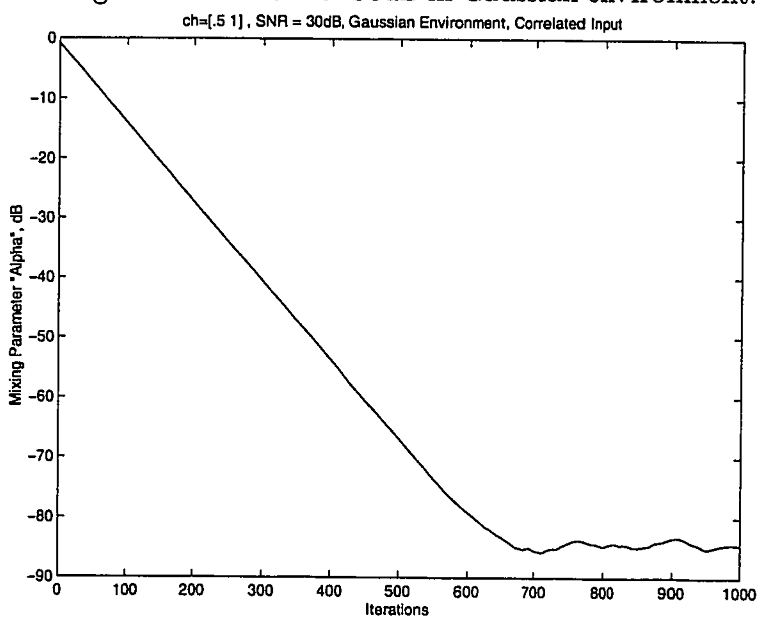

Figure 4.32: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.1.5 Results for the Recovery Time after an abrupt change in the Environment with the white input sequence

In this section, the ability of adaptive algorithm to attain the steady-state after a sudden change in the environment is investigated. After an abrupt change in the environment, the coefficient of the weight vector of the adaptive filter are made zeros, so adaptive algorithm has to identify the system in that operating environment from scratch. It is observed that in uniform environment with SNR 20 dB and 30 dB, the proposed algorithm has better convergence speed and tracking ability to the abrupt change in the environment than the traditional NLMS algorithm as shown in Figure 4.33 and 4.35, respectively. With 20 dB SNR, it has tracked the steady state 19000 iterations earlier than the NLMS algorithm, while with 30 dB SNR, it has tracked 20000 iterations earlier. In the case of gaussian environment with 20 dB SNR Figure 4.37, both algorithm have same performance but with 30 dB SNR Figure 4.39, it has tracked 14000 iterations earlier than the NLMS algorithm.

The corresponding time varying mixing parameter is also plotted for each case. There is very fast increase in the mixing parameter after the abrupt change in the environment but soon its steady state is attained, these are depicted in Figureures 4.34, 4.36, 4.38, and 4.40.

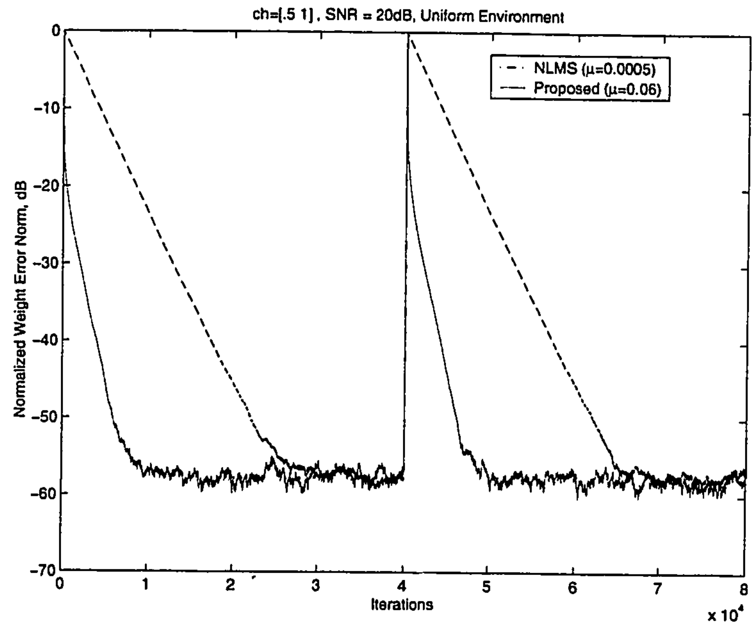

Figure 4.33: Comparison of the Recovery Time of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

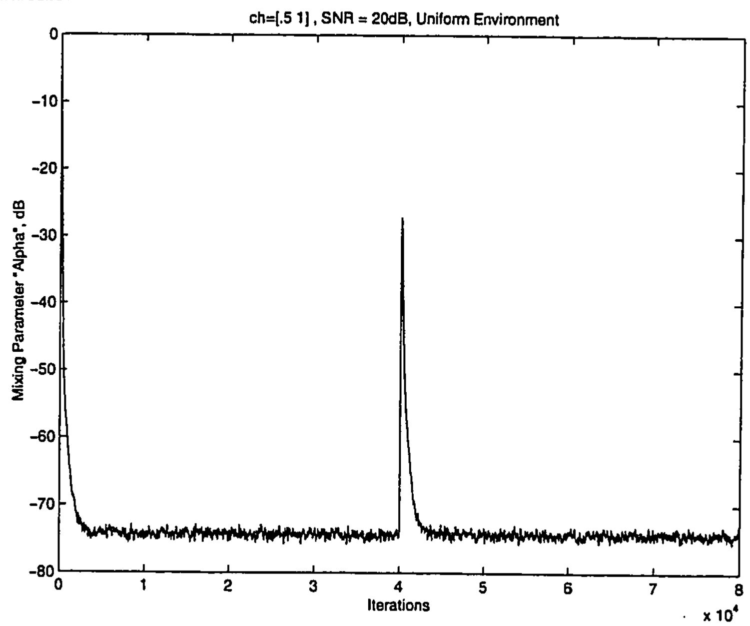

Figure 4.34: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

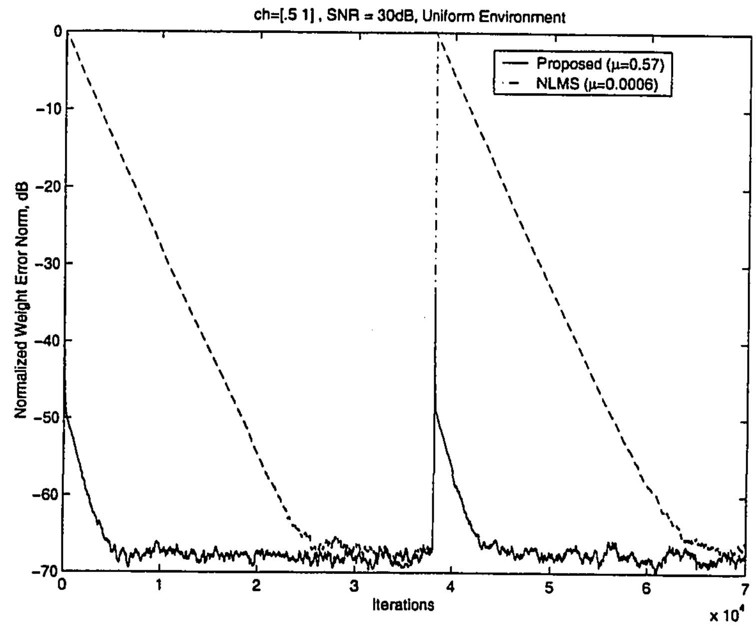

Figure 4.35: Comparison of the Recovery Time of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in uniform environment.

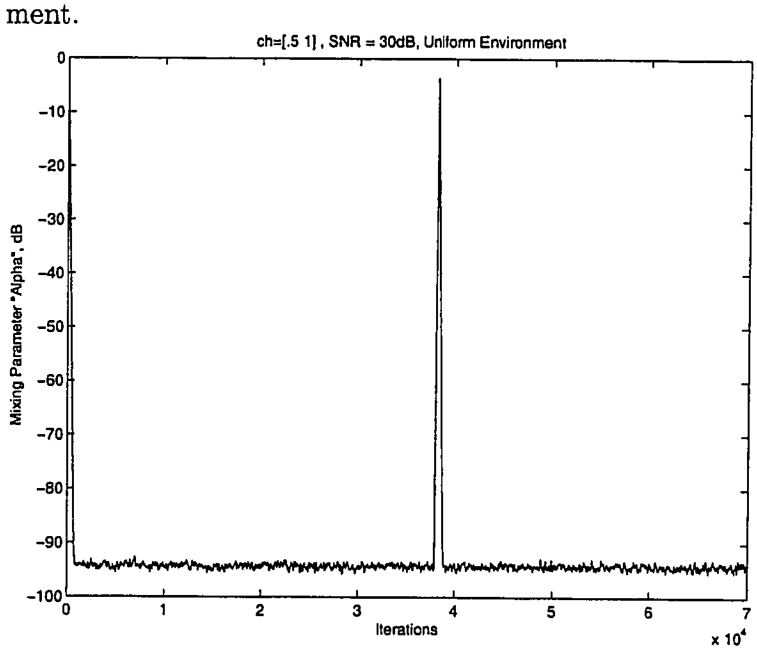

Figure 4.36: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

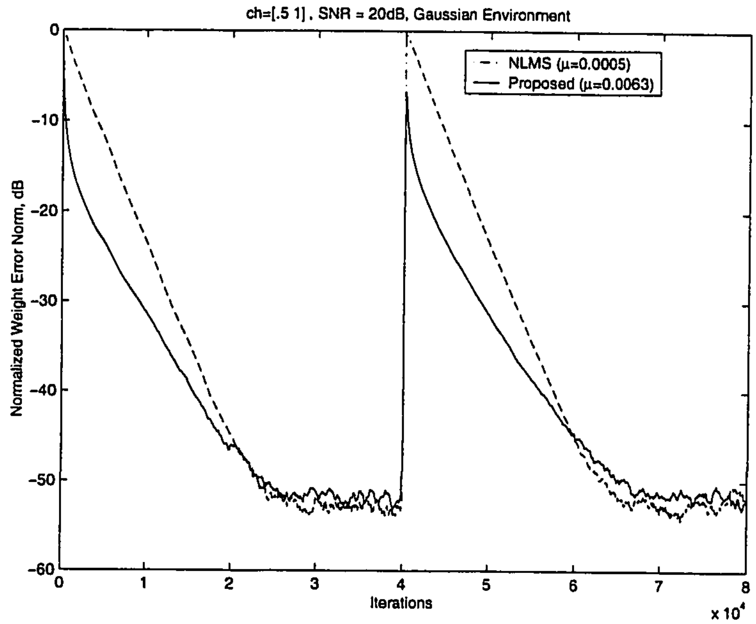

Figure 4.37: Comparison of the Recovery Time of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

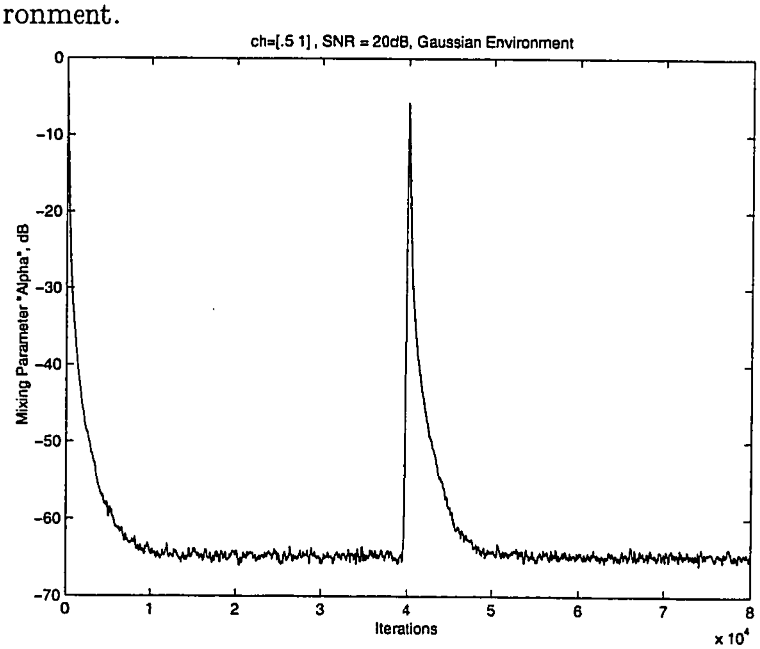

Figure 4.38: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

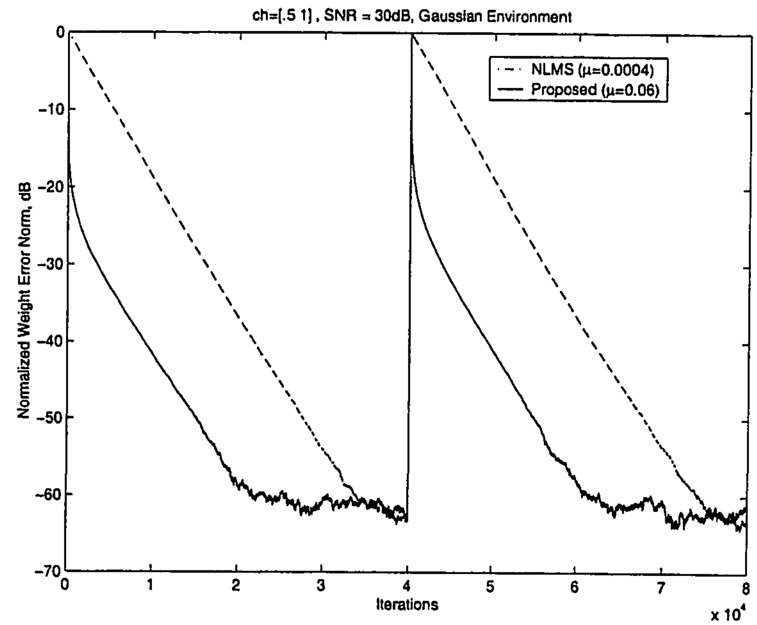

Figure 4.39: Comparison of the convergence speed of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment.

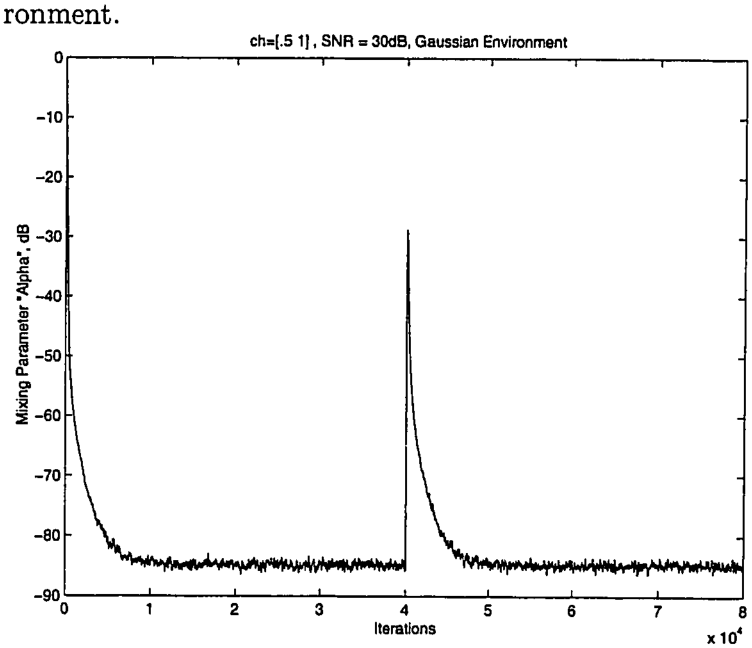

Figure 4.40: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.1.6 Results for the Recovery Time after an abrupt change in the Environment for the First System with highly correlated input sequence

In this section, the same investigation is carried out as done in the previous section (**4.1.5**) but when the input sequence is highly correlated given by Equation (4.1). In uniform environment Figure 4.41, the proposed algorithm has tracked the system after abrupt change in the environment nearly 200 iterations earlier than the NLMS algorithm, while in gaussian environment Figure 4.43, it has tracked 250 iterations earlier than NLMS algorithm, but in both cases, the tracking speed of the proposed algorithm is less with correlated input sequence as compared to the white input sequence which has been discussed in the previous section.

The corresponding time varying mixing parameter is also plotted for each case, shown in Figure 4.42 and 4.42. The change in the mixing parameter is different in the case of correlated input sequence. There is an sudden increase in it due to abrupt change in the environment but time taken by it to attain the steady state is higher as compared to that with the white input sequence.

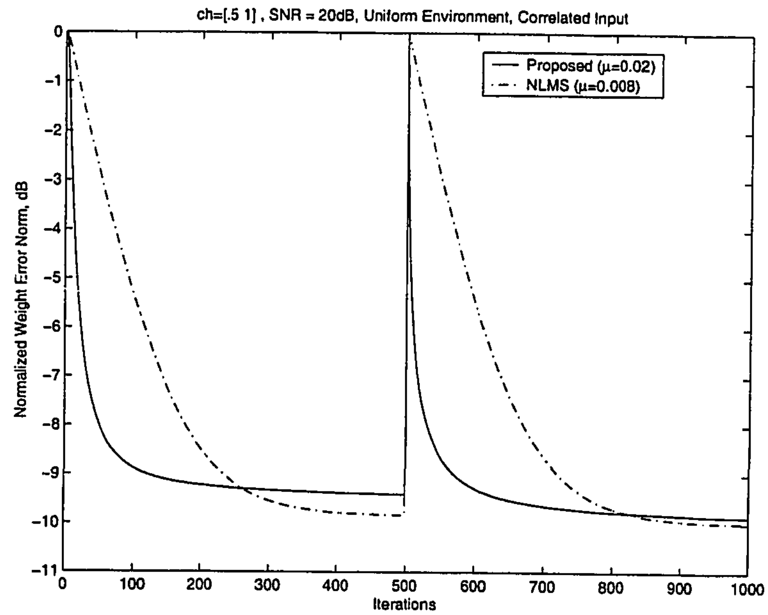

Figure 4.41: Comparison of the Recovery Time of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in uniform environment.

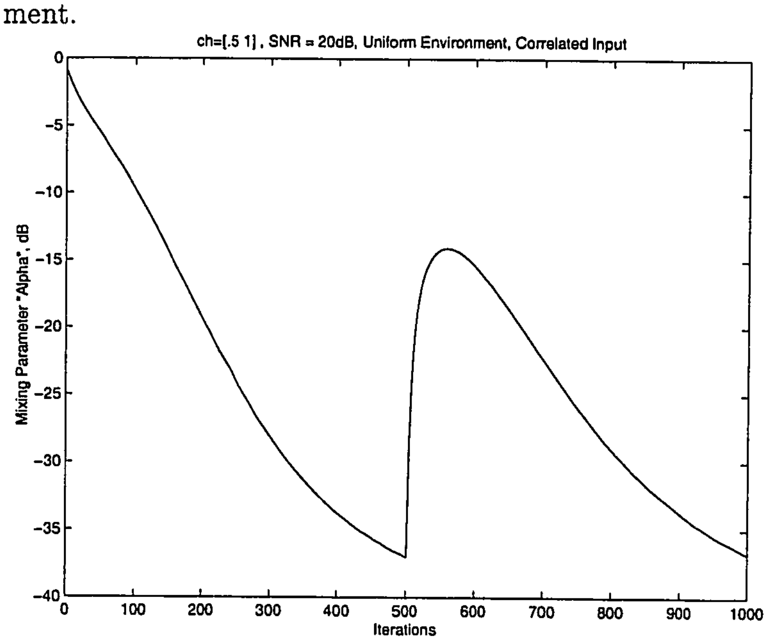

Figure 4.42: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

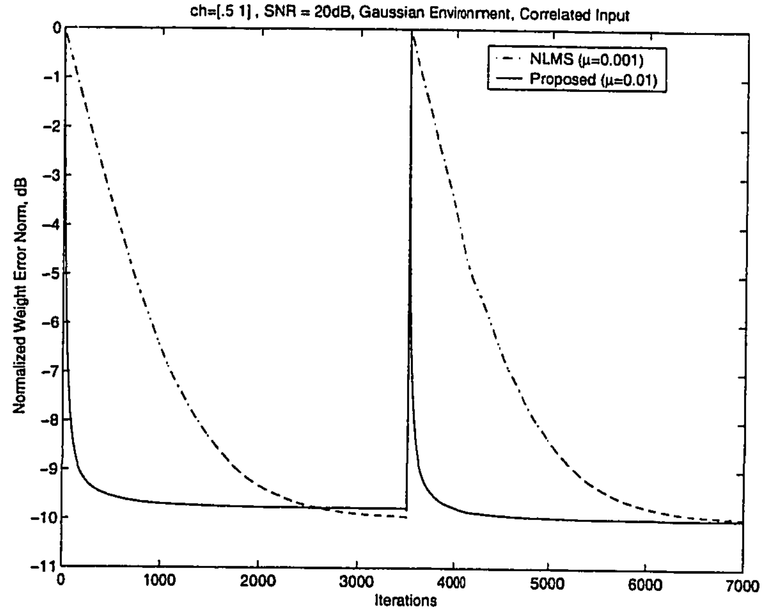

Figure 4.43: Comparison of the Recovery Time of the NLMS algorithm and the proposed normalized mixed norm LMS-LMF algorithm with SNR=20dB in Gaussian environment.

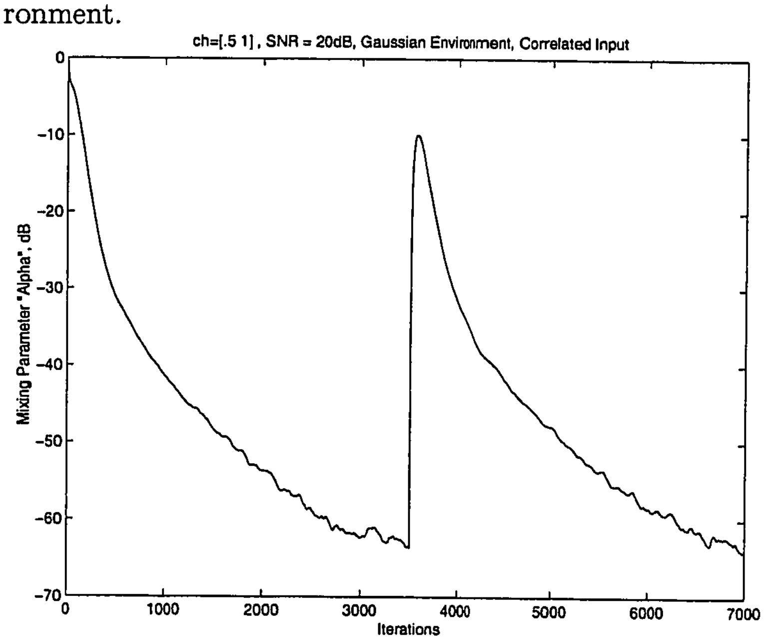

Figure 4.44: Behavior of the time-varying mixing parameter(α_n)for the corresponding case.

4.2 Results on the comparison between Analytical and Experimental Mean Square Behaviour of the Weight Error Vector

In this section, analytical mean square behaviour of the weight vector given by the Equation 2.37 is compared with the experimental results of its mean square behaviour. For gaussian distributed signal, we can use the relation given in [5]:

$$E[(\xi_w)^{2k}] = k!(E[\xi_w^2])^k \tag{4.2}$$

where **k** is an integer. Using this relation, we have evaluated the values of the high order moments of noise i.e. ϕ_n^4 and ϕ_n^6 etc. The unknown system used has response [.5 1], and variance of the input sequence, i.e., σ_x^2 is assumed to be unity. Now using all these, analytical results of mean square behaviour of the weight error vector are plotted for the case of both uniform and gaussian environment with SNR= 30 dB. These results are then compared with the experimental values on the same graph, which shows that experimental results are approximately approaching the analytical results.

In Figures 4.45 and 4.46, the results for analytical and experimental trace of weight error vector of the proposed algorithm has plotted in the Gaussian and Uniform environment, respectively . It can be observed that in the case of Gaussian environment, the two curves are more closer to each other as compared to that in the uniform environment. It should be the case, because the analysis has been done assuming the gaussian noise. These results have verified the analysis which has been carried out during the analysis of the proposed algorithm.

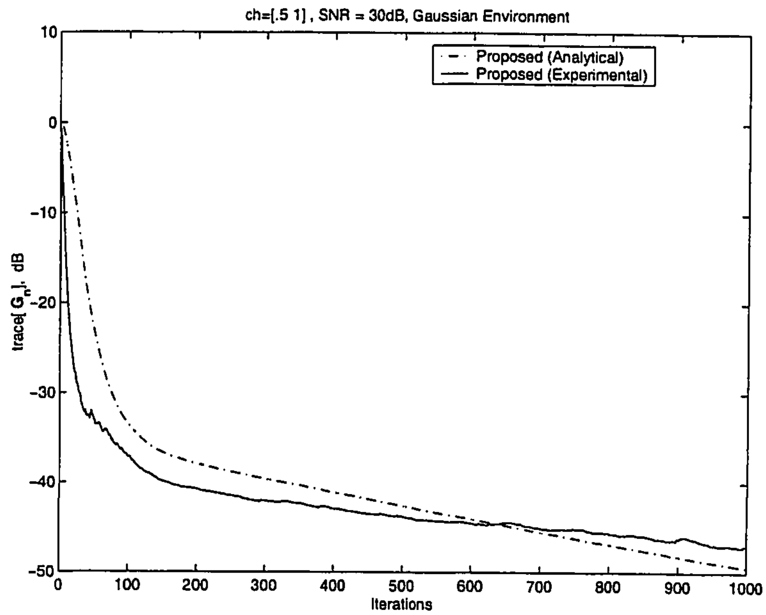

Figure 4.45: Comparison of Analytical and Experimental Trace of weight error vector of the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Gaussian environment step size=1.

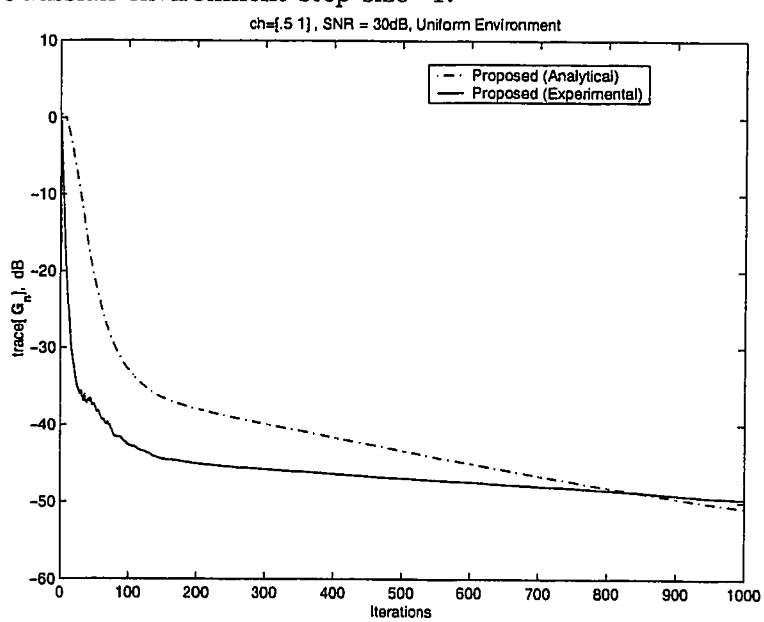

Figure 4.46: Comparison of Analytical and Experimental Trace of weight error vector of the proposed normalized mixed norm LMS-LMF algorithm with SNR=30dB in Uniform environment step size=0.7.

Chapter 5

Thesis Contributions, Conclusions, and Recommendations for Future Works

5.1 Thesis Contributions

This work has successfully presented a normalized mixed norm algorithm which is time varying, and it is given the name as normalized time varying mixed norm LMS-LMF algorithm. This algorithm is analyzed in terms of convergence properties, steady state performances, and the tracking ability. The performance of the proposed algorithm has been supported by presenting the simulation scenario. The major contributions of this thesis work are following:

1. A new normalized time varying mixed norm LMS-LMF algorithm has been

proposed in which the mixing parameter is time varying and it is adapted according to time averaged estimate of the correlation of the error.

2. The convergence analysis of the proposed algorithm is carried out in terms of mean and mean square sense. A bound for the step-size is derived that ensures the convergence in the mean.

3. Steady state analysis is presented. Exact and approximate expressions of the steady-state excess mean-square error for both NLMS and the proposed algorithms are provided.

4. Tracking ability of the NLMS and proposed algorithms are analyzed in the presence of two types of nonstationarities i.e., random and cyclic. The expressions of the steady-state excess mean-square error for this case is also derived. The analytical results are then compared with the experimental results which support the analysis.

5.2 Conclusions

Three main aspects are observed and following conclusions are drawn:

1. A major problem with the NLMS algorithm is that at higher convergence speed, the steady state error is high. The proposed algorithm has overcome this, and it has been shown through simulation results that at faster convergence, its steady state error is also low as compared to the NLMS algorithm.

For example, with white input sequence in uniform environment, and 20 dB SNR, it is found that the final steady state error floor of the proposed algorithm is 24 dB below than that of the NLMS algorithm. This difference is increased up to 32 dB when the SNR=30 dB. Similarly in gaussian environment, it has better performance with 30 dB SNR. So, it can be concluded that at high SNR, the performance of the proposed algorithm is improved. With the same SNR, it has better performance in uniform environment as compared to gaussian. In the case of highly correlated input sequence, even though, the proposed algorithm has better performance, the performance margin is reduced. For example, in gaussian environment with SNR=30 dB, the margin is of only 1.5 dB. Also, similar to the white input case, its performance is improved by increasing the SNR.

2. The second achievement of the proposed algorithm is in terms of the convergence speed. The NLMS algorithm achieves the lower steady state error when the step-size is very small, due to which its convergence speed becomes slow. However, the results have shown that the proposed algorithm can reach the same steady state error floor in a very lesser number of iterations, thus improving the convergence speed. For example, when using the white input sequence in uniform environment, the proposed algorithm reached the noise floor of NLMS algorithm 20000 iterations earlier with SNR= 20 dB, and 22000 iterations with SNR=30 dB. So, its convergence speed is increased by SNR. While

in gaussian environment, it has same convergence speed with SNR=20 dB, but increased with SNR=30 dB. With highly correlated input, its performance is better in gaussian environment as compared to uniform environment.

3. Third is in terms of recovery time after a sudden change in the environment. It is found that in uniform environment, its performance is better with both SNR=20 dB and SNR=30 dB. While in gaussian environment, it has same performance with SNR=20 dB but improved by increasing the SNR.

5.3 Recommendations for Future Work

There are few suggestions regarding the future works. In the thesis, convergence analysis of the proposed algorithm is carried out and the bound on step-size is derived which ensures the convergence in the mean . In a similar way, the bound on step-size to ensure convergence in the mean square can be derived.

Performance of the proposed algorithm is dependent on two parameters that are δ, and γ. These are bounded between the range [0,1]. Another suggestion is that the optimum values of these parameters can be derived for the optimal solution.

Appendix A

Derivation for Equation (2.9)

In system identification problem the error e_n is defined according to Equation (2.2) which is reproduced here:

$$e_n = d_n - \mathbf{x}_n^T \mathbf{w}_n.$$

where d_n is the desired response which consists of output from the unknown system and additive noise i.e.

$$d_n = \mathbf{x}_n^T \mathbf{w}_{opt} + \xi_n.$$

Substituting the value of d_n in Equation (2.2) and using the definition of weight error vector $\mathbf{v}_n$, we will obtain the following

$$\begin{aligned} e_n &= \xi_n - \mathbf{x}_n^T(\mathbf{w}_n - \mathbf{w}_{opt}) \\ &= \xi_n - \mathbf{x}_n^T \mathbf{v}_n. \end{aligned}$$

Appendix B

Derivation for Expectation term $E[\mathbf{x}_n\mathbf{x}_n^T\mathbf{K}_n\mathbf{x}_n\mathbf{x}_n^T]$

According to the Gaussian Factoring Moment Theorem for zero mean Gaussian random variables x_i, $i = 1, \cdots, 4$, the following result holds:

$$E(x_1x_2x_3x_4) = E(x_1x_2)E(x_3x_4) + E(x_1x_3)E(x_2x_4) + E(x_1x_4)E(x_2x_3). \quad \text{(B.1)}$$

Applying the above result for a zero mean gaussian random vector $\mathbf{x}_n$ with $E(\mathbf{x}_n\mathbf{x}_n^T) = \mathbf{R}$, where $\mathbf{R} = \sigma_x^2\mathbf{I}$, we will obtain following:

$$\begin{aligned}
E[\mathbf{x}_n\mathbf{x}_n^T\mathbf{K}_n\mathbf{x}_n\mathbf{x}_n^T] &= E[\mathbf{x}_n\mathbf{x}_n^T]\mathbf{K}_nE[\mathbf{x}_n\mathbf{x}_n^T] + E[\mathbf{x}_n\mathbf{x}_n^T]\mathbf{K}_n^TE[\mathbf{x}_n\mathbf{x}_n^T] + E[\mathbf{x}_n\mathbf{x}_n^T]tr(E[\mathbf{x}_n\mathbf{x}_n^T]\mathbf{K}_n) \\
&= \mathbf{R}\mathbf{K}_n\mathbf{R} + \mathbf{R}\mathbf{K}_n^T\mathbf{R} + \mathbf{R}tr(\mathbf{R}\mathbf{K}_n) \\
&= (\sigma_x^2\mathbf{I})\mathbf{K}_n(\sigma_x^2\mathbf{I}) + (\sigma_x^2\mathbf{I})\mathbf{K}_n(\sigma_x^2\mathbf{I}) + (\sigma_x^2\mathbf{I})tr((\sigma_x^2\mathbf{I})\mathbf{K}_n) \\
&= 2\sigma_x^4\mathbf{K}_n + \sigma_x^4 tr\{\mathbf{K}_n\}\mathbf{I}
\end{aligned}$$

Appendix C

Derivation for Expectation term $E[(\| \mathbf{x}_n \|^2)^2]$

Since the term $(\| \mathbf{x}_n \|^2)^2$ is a number, we can apply the trace operator to simplify the expectation. Now using the property of trace operator i.e. $E[tr(A)] = tr(E[A])$, the expectation of Equation 2.28 can be solved as follows:

$$\begin{aligned} E[(\| \mathbf{x}_n \|^2)^2] &= E[\mathbf{x}_n^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{x}_n] \\ &= tr(E[\mathbf{x}_n^T \mathbf{x}_n] E[\mathbf{x}_n^T \mathbf{x}_n]) \\ &= tr(\mathbf{RR}) \\ &= tr(\sigma_x^4 \mathbf{I}) \\ &= N\sigma_x^4 \end{aligned}$$

Appendix D

Derivation for Equation (2.31)

First of all Equation (2.20) is reproduced here:

$$
\begin{aligned}
\mathbf{K}_{n+1} &= \mathbf{K}_n - \mu\Big\{E[\alpha_n] + 6(1 - E[\alpha_n])\sigma_w^2\Big\}\{\mathbf{B}\mathbf{K}_n + \mathbf{K}_n\mathbf{B}\} \\
&+\mu^2\Big\{E[\alpha_n^2](1 - 12\sigma_w^2 + 36\phi_w^4) \\
&+12E[\alpha_n](\sigma_w^2 - 6\phi_w^4) + 36\phi_w^4\Big\}\mathbf{C} \\
&+\mu^2\Big\{E[\alpha_n^2](\sigma_w^2 - 4\phi_w^4 + 4\phi_w^6) \\
&+4E[\alpha_n](\phi_w^4 - 2\phi_w^6) + 4\phi_w^6\Big\}\mathbf{D}.
\end{aligned}
$$

Now substituting the values of constants $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ from equations (2.25), (2.29), and (2.30) respectively to obtain the following result:

$$
\begin{aligned}
\mathbf{K}_{n+1} &= \mathbf{K}_n - \mu\Big\{E[\alpha_n] + 6(1 - E[\alpha_n])\sigma_w^2\Big\}\Big\{\frac{\mathbf{I}}{\mathbf{N}}\mathbf{K}_n + \mathbf{K}_n\frac{\mathbf{I}}{\mathbf{N}}\Big\} \\
&+\mu^2\Big\{E[\alpha_n^2](1 - 12\sigma_w^2 + 36\phi_w^4) \\
&+12E[\alpha_n](\sigma_w^2 - 6\phi_w^4) + 36\phi_w^4\Big\}\Big\{\frac{2\mathbf{K}_n + tr\{\mathbf{K}_n\}\mathbf{I}}{N}\Big\}
\end{aligned}
$$

$$
\begin{aligned}
&+\mu^2\Big\{E[\alpha_n^2](\sigma_w^2 - 4\phi_w^4 + 4\phi_w^6)\\
&+4E[\alpha_n](\phi_w^4 - 2\phi_w^6) + 4\phi_w^6\Big\}\frac{\mathbf{I}}{\mathbf{N}\sigma_{\mathbf{x}}^{\mathbf{2}}}\\
= \;& \mathbf{K}_n - \frac{2\mu}{N}\{E[\alpha_n] + 6(1 - E[\alpha_n])\sigma_w^2\}[\mathbf{K}_n] + \frac{\mu^2}{N}\{E[\alpha_n^2](1 - 12\sigma_w^2 + 36\phi_w^4)\\
&+12E[\alpha_n](\sigma_w^2 - 6\phi_w^4) + 36\phi_w^4\}[2\mathbf{K}_n + tr\{\mathbf{K}_n\}\mathbf{I}] + \frac{\mu^2}{N\sigma_x^2}\{E[\alpha_n^2](\sigma_w^2 - 4\phi_w^4 + 4\phi_w^6)\\
&+4E[\alpha_n](\phi_w^4 - 2\phi_w^6) + 4\phi_w^6\}\mathbf{I}\\
= \;& \mathbf{K}_n - \mu a_1[\mathbf{K}_n] + \mu^2 a_2[2\mathbf{K}_n + tr\{\mathbf{K}_n\}\mathbf{I}] + \mu^2 a_3\mathbf{I}
\end{aligned}
$$

Appendix E

Derivation for Equation (3.8)

Subtracting terms $\mathbf{w}^o_{n+1}e^{j\Omega(n+1)}$ and $\mathbf{w}^o_n e^{j\Omega n}$ from both sides of Equation (2.38) and using the definition of weight error vector given by Equation (3.5), we will obtain the following:

$$\begin{aligned}\mathbf{v}_{n+1} &= \mathbf{v}_n - \mu \mathbf{x}^*_n f(e_n) - \mathbf{w}^o_n e^{j\Omega n} + \mathbf{w}^o_{n+1} e^{j\Omega(n+1)} \\ &= \mathbf{v}_n - \mu \mathbf{x}^*_n f(e_n) - e^{j\Omega n}(\mathbf{w}^o_{n+1} e^{j\Omega} - \mathbf{w}^o_n)\end{aligned}$$

Now using the relation given by 3.2 substitute the values of $\mathbf{w}^o_{n+1}$ and $\mathbf{w}^o_n$ in the above equation, we will get the following:

$$\begin{aligned}\mathbf{v}_{n+1} &= \mathbf{v}_n - \mu \mathbf{x}^*_n f(e_n) - e^{j\Omega n}(\mathbf{w}^o(e^{j\Omega} - 1) + \mathbf{q}_{n+1} e^{j\Omega n} - \mathbf{q}_n) \\ &= \mathbf{v}_n - \mu \mathbf{x}^*_n f(e_n) - e^{j\Omega n}\mathbf{c}_n.\end{aligned}$$

Appendix F

Derivation for Equation (3.13)

To solve for the value of $\mathbf{z}$, Equation (3.8) is first multiplied both sides by term $e^{-j\Omega n}$ and using assumption **A6**, we will get:

$$\mathbf{z}_{n+1} e^{j\Omega(n+1)} e^{-j\Omega n} = \mathbf{z}_n e^{j\Omega n} e^{-j\Omega n} - \mu \mathbf{x}_n^* f(e_n) e^{-j\Omega n} + \mathbf{c}_n$$

Now taking expectation both sides and using the fact that at steady state $E[\mathbf{z}_{n+1}] = E[\mathbf{z}_n] = \mathbf{z}$, we will get:

$$\mathbf{z}(1 - e^{j\Omega}) = \mu E[\mathbf{x}_n^* f(e_n) e^{-j\Omega n}] - E[\mathbf{w}^o(e^{j\Omega} - 1)] + E[\mathbf{q}_{n+1} e^{j\Omega n}] - E[\mathbf{q}_n]$$

Since, the random nostationarity term $\mathbf{q}_n$ has zero mean, we will get:

$$\mathbf{z}(1 - e^{j\Omega}) = \mu E[\mathbf{x}_n^* f(e_n) e^{-j\Omega n}] - \mathbf{w}^o(e^{j\Omega} - 1)$$

Appendix G

Derivation for Expectation term $E[\mathbf{q}_n^* \mathbf{z}_n]$

Using assumption**A6**, substitute the value of $\mathbf{z}_n$ and using the Equations (3.2) and (3.5), we will obtain the following:

$$\begin{aligned} E[\mathbf{q}_n^* \mathbf{z}_n] &= E[\mathbf{q}_n^* \{(\mathbf{w}^o + \mathbf{q}_n)e^{j\Omega n} - \mathbf{w}_n\} e^{-j\Omega n}] \\ &= E[\mathbf{q}_n^* \mathbf{w}^o e^{-j\Omega n}] + E[\mathbf{q}_n^* \mathbf{q}_n] + E[\mathbf{q}_n^* \mathbf{w}_n e^{-j\Omega n}] \end{aligned}$$

One conclusion drawn from the assumption **A1** is that weight error vector is independent of the input sequence at the same time instance, so we will get:

$$\begin{aligned} E[\mathbf{q}_n^* \mathbf{z}_n] &= E[\mathbf{q}_n^* \mathbf{q}_n] \\ &= tr(\mathbf{Q}_n) \end{aligned}$$

Appendix H

Derivation for Expectation term $E[\mathbf{q}_n^* \mathbf{x}_n^* \mathbf{x}_n \mathbf{z}_n]$

Using the property of trace operator i.e. $E[tr(A)] = tr(E[A])$ and the Gaussian Factoring Moment Theorem, we will get:

$$
\begin{aligned}
E[\mathbf{q}_n^* \mathbf{x}_n^* \mathbf{x}_n \mathbf{z}_n] &= E[tr(\mathbf{q}_n^* \mathbf{x}_n^* \mathbf{x}_n \mathbf{z}_n)] \\
&= E[tr(\mathbf{q}_n^* \mathbf{z}_n \mathbf{x}_n^* \mathbf{x}_n)] \\
&= tr(E[\mathbf{q}_n^* \mathbf{z}_n] E[\mathbf{x}_n^* \mathbf{x}_n]) \\
&= tr(\mathbf{Q}_n \mathbf{R})
\end{aligned}
$$

Appendix I

Derivation for Equation (3.19)

Substituting the value of $\mathbf{z}$ from Equation (3.16) into Equation (3.18), we will get the following:

$$\begin{aligned}
\zeta^{NLMS}(\frac{2\mu-\mu^2}{tr(\mathbf{R})}) &= \parallel \mathbf{w}^o \parallel^2 |1-e^{j\Omega}|^2 + 2\mu\frac{tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} \\
&\quad -2Re\Big[(1-e^{j\Omega})^*\mathbf{w}^{o*}\Big(\mathbf{I}-\mu\frac{\mathbf{R}}{tr\{\mathbf{R}\}}\Big)\mathbf{z}\Big] + \frac{\mu^2\sigma_w^2}{tr(\mathbf{R})} \\
\zeta^{NLMS} &= \frac{tr(\mathbf{R})}{(2\mu-\mu^2)}\Big[2\mu\frac{tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2\sigma_w^2}{tr(\mathbf{R})} + \parallel \mathbf{w}^o \parallel^2 |1-e^{j\Omega}|^2 \\
&\quad -2Re\Big[(1-e^{j\Omega})^*\mathbf{w}^{o*}\Big(\mathbf{I}-\frac{\mu\mathbf{R}}{tr\{\mathbf{R}\}}\Big)\Big[\mathbf{I}-\frac{\mu\mathbf{R}}{(1-e^{-j\Omega})tr\{\mathbf{R}\}}\Big]^{-1}\mathbf{w}^o\Big] \\
&= \frac{tr(\mathbf{R})}{(2\mu-\mu^2)}\Big[2\mu\frac{tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2\sigma_w^2}{tr(\mathbf{R})} + \parallel \mathbf{w}^o \parallel^2 |1-e^{j\Omega}|^2 \\
&\quad -2Re\Big[|1-e^{j\Omega}|^2 \parallel \mathbf{w}^o \parallel^2 \Big(\mathbf{I}-\frac{\mu\mathbf{R}}{tr(\mathbf{R})}\Big)[\mathbf{I}-\frac{\mu\mathbf{R}}{tr\{\mathbf{R}\}}-e^{-j\Omega}\mathbf{I}]^{-1}\Big] \\
&= \frac{tr(\mathbf{R})}{(2\mu-\mu^2)}\Big[2\mu\frac{tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2\sigma_w^2}{tr(\mathbf{R})} + \parallel \mathbf{w}^o \parallel^2 |1-e^{j\Omega}|^2 \\
&\quad -2Re\Big[|1-e^{j\Omega}|^2\mathbf{W}^o\mathbf{X}\Big]
\end{aligned}$$

where, $\mathbf{W}^0$ and $\mathbf{X}$ are defined in Equations (3.20) and (3.21), respectively. The above equation can be further simplified as follows:

$$\begin{aligned}\zeta^{NLMS} &= \frac{tr\{\mathbf{R}\}}{(2\mu-\mu^2)}\left[\frac{2\mu tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2\sigma_w^2}{tr\{\mathbf{R}\}} + |1-e^{-j\Omega}|^2 Re\Big\{tr(\mathbf{W}^o(\mathbf{I}-2\mathbf{X}))\Big\}\right] \\ &= \frac{tr\{\mathbf{R}\}}{(2\mu-\mu^2)}\left[\frac{2\mu tr\{\mathbf{Q}_n\mathbf{R}\}}{tr\{\mathbf{R}\}} + \frac{\mu^2\sigma_w^2}{tr\{\mathbf{R}\}} + \beta_o\right]\end{aligned}$$

Appendix J

Derivation for Equation (3.25)

Equation (3.19) can be written in the following form:

$$\zeta^{NLMS} = \frac{1}{(2-\mu)}\Big[2tr(\mathbf{Q}_n\mathbf{R}) + \mu\sigma_w^2 + c\Big]$$

where $c = \frac{\beta_o tr(\mathbf{R})}{\mu}$. After substituting the value of β_o, it can be expressed as follows:

$$c = \frac{tr(\mathbf{R})}{\mu}|1-e^{-j\Omega}|^2 Re\Big[tr(\mathbf{W}^o(\mathbf{I}-2\mathbf{X}))\Big]$$

substitute the value of $\mathbf{X}$, we can get the following:

$$\begin{aligned}
c &= \frac{tr(\mathbf{R})}{\mu}|1-e^{-j\Omega}|^2 Re\Big[tr(\mathbf{W}^o(\mathbf{I}-2(\mathbf{I}-\frac{\mu\mathbf{R}}{tr\{\mathbf{R}\}})[\mathbf{I}-\frac{\mu\mathbf{R}}{tr\{\mathbf{R}\}}-e^{-j\Omega}\mathbf{I}]^{-1}))\Big] \\
&= \frac{tr(\mathbf{R})}{\mu}|1-e^{-j\Omega}|^2 Re\Big[tr(\mathbf{W}^o(\mathbf{I}-2(1-\frac{\mu}{N})(1-\frac{\mu}{N}-e^{-j\Omega})^{-1}\mathbf{I}))\Big] \\
&= \frac{tr(\mathbf{R})}{\mu}|1-e^{-j\Omega}|^2 \mathbf{W}^o Re\Big[tr\Big((1-\frac{2(1-\frac{\mu}{N})}{(1-\frac{\mu}{N}-e^{-j\Omega})})\mathbf{I}\Big)\Big] \\
&= \frac{tr(\mathbf{R})}{\mu}|1-e^{-j\Omega}|^2 \parallel \mathbf{w}^o \parallel^2 Re\Big[\frac{N}{|1-\frac{\mu}{N}-e^{-j\Omega}|^2}\Big\{(\frac{2\mu}{N}-\frac{\mu^2}{N^2})+j2Sin\Omega(1-\frac{\mu}{N})\Big\}\Big] \\
&= \sigma_x^2|1-e^{-j\Omega}|^2 \parallel \mathbf{w}^o \parallel^2 \Big[\frac{(2N-\mu)}{|1-\frac{\mu}{N}-e^{-j\Omega}|^2}\Big]
\end{aligned}$$

Bibliography

[1] S. U. H. Qureshi. Adaptive equalization. *IEEE Proceedings*, 53:1349–1387, 1985.

[2] B. Widrow, J. R. Glover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, J. R. Zeidler, E. Dong, Jr., and R. C. Goodlin. Adaptive noise cancelling: Principles and applications. *IEEE Proceedings*, 63(12):1692–1716, 1975.

[3] L. Makhoul. Linear predition a tutorail review. *IEEE Proceedings*, 63(4):561–580, 1975.

[4] C. F. N. Cowan and P. M. Grant. *Adaptive Filters.* Englewood Cliffs, NJ: Prentice Hall, 1985.

[5] S. Haykin. *Adaptive Filter Theory.* Englewood Cliffs, NJ: Prentice Hall, 1996.

[6] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson. Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter. *IEEE Proceedings*, 64(8):1151–1162, 1976.

[7] L. Ljung, M. Morf and D. Falconer. Fast calculation of gain matrices for recursive estimation schemes. *Int. J. Contr*, 27:1–19, 1978.

[8] D. Manolakis G. Carayannis and N. Kalouptsidis. A fast sequential algorithm for least squares filtering and prediction. *IEEE transactions on Acoustic, Speech and Signal Processing*, 31:1394–1402, 1983.

[9] J. M. Cioffi and T. Kailath. Fast recursive least squares transversal filters for adaptive filtering. *IEEE Transaction, Acoustic, Speech, Signal Processing,*, 32:304–337, 1984.

[10] A. Benallal and A. Gillorie. A new method to stabilize Fast RLS algorithms based on a First Order Model of the propagation of numerical errors. *Proc. ICASSP88*, pages 1373–1376, 1988.

[11] D. Mansour and A. H. Gray. Unconstrained frequency domain adaptive filters. *IEEE Transaction Acoustic Speech and Signal Processing*, 30(5):726–734, 1982.

[12] K. Ozeki and T. Umeda. An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties. *IEICE Transactions, Japan*, J67-A:126–132, Feb 1984.

[13] J. I., Nagumo and A. Noda. A learning method for system identification. *IEEE Transactions, Automatic control*, AC-12:282–287, 1967.

[14] R. D. Gitlin, H. C. Meadors, and S. B. Weinstein. The Tap-Leakage Algorithm: An Algorithm for the stable operation of a digitally implemented, Fractionally Spaced adaptive equalizer. *Bell Syst. Tech. J.*, 61:1817–1839, 1982.

[15] G. A. Clark, S. K. Mitra, and S. R. Parker. Block implementatiom of adaptive digital filters. *IEEE Transaction Acoustic Speech and Signal Processing*, 29(3):744–752, 1981.

[16] R. H. Kwong and E. W. Johnston. A Variable Stepsize LMS algorithm. *IEEE transactions on signal procssing*, 40:1633–1642, 1992.

[17] T. J. Shan and T. Kailath. Adaptive algorithms with an automatic gain control feature. *IEEE transactions on Acoustic, Speech and Signal Processing*, 35:122–127, 1988.

[18] R. W. Harris, D. M. Chabries, and F. A. Bishop. A Variable Stepsize (VS) algorithm. *IEEE transactions on Acoustic, Speech and Signal Processing*, 34:499–510, 1986.

[19] V. J. Methews and Z. Xie. A stochastic gradient adaptive filter with gradient adaptive step size. *IEEE transactions on Signal Processing*, 41:2075–2087, 1993.

[20] T. Aboulnasr and K. Mayyas. A Robust Variable Step-Size LMS Type Algorithm: Analysis and Simulation. *IEEE transactions on Signal Processing*, 45:631–639, 1997.

[21] E. Walach and B. Widrow. The Least Mean Fourth(LMF) adaptive algorithm and its family. *IEEE transactions, Information Theory*, 30(2):275–283, 1984.

[22] A. Zerguine, C. F. N. Cowan, and M. Bettayeb. Novel LMS-LMF adaptive scheme for echo cancellation. *Electronic Letters.*, 32(19):1776–1778, 1996.

[23] A. Zerguine, M. Bettayeb, and C. F. N. Cowan. Hybrid LMS-LMF scheme for echo cancellation. *IEE Proceedings-VIS Image Signal Processing.*, 146(4):173–180, 1999.

[24] A. Zerguine, C. F. N. Cowan, and M. Bettayeb. Adaptive echo cancellation using least mean mixed-norm. *IEEE Transactions Signal Processing.*, SP-45(5):1340–1343, 1997.

[25] J. A. Chambers, O. Tanrikulu and A. G. Constantinides. Least mean mixed-norm adaptive filtering. *Electronic Letters*, 30(19):1574–1575, 1994.

[26] T. Aboulnasr and A. Zerguine. Variable Weight Mixed-Norm LMS-LMF Adaptive Algorithm. *IEEE conference on signals, systems and computers*, 1:791–794, 1999.

[27] J. G. Proakis. *Digital Communications.* Singapore: McGraw-Hill, 1983.

[28] B. Widrow, P.E. Mantey, L. J. Griffiths, and B.B. Goode. Adaptive Antenna Systems Array. *IEEE Proceedings*, 55:2143–2159, 1967.

[29] C. W. K. Gritton and D. W. Lin. Echo Cancellation Algorithms. *IEEE ASSP Magazine*, pages 30–37, 1984.

[30] K. Murano, S. Unagami, and F. Amano. Echo Cancellation and Applications. *IEEE Communication Magazine*, 28(1):49–55, 1990.

[31] N. J. Bershad. Analysis of the normalized LMS algorithm with gaussian inputs. *IEEE transactions on Acoustic, Speech and Signal Processing*, ASSP-34:793–806, 1986.

[32] N. J. Bershad. On the optimum data nonlinearity in LMS adaptation. *IEEE transactions Signal Processing*, 34(1):69–76, 1986.

[33] S. C, Douglas and T. H. Y. Meng. Stochastic gradient adaptation under general error criteria. *IEEE transactions Signal Processing*, 42(6):1352–1365, 1994.

[34] T. Y. Al-Naffouri, A. Zerguine, and M. Bettayeb. A unifying view of error nonlinearities in LMS adaptation. *Proceedings of the 1998 IEEE ICASSP*, 3:1697–1700, 1998.

[35] T. Y. Al-Naffouri, A. Zerguine, and M. Bettayeb. Convergence properties of mixed-norm algorithms under general error criteria. *IEEE ISCAS '99*, 3:211–214, 1999.

[36] H. Sari. Performance evaluation of three adaptive equalization algorithms. *Proceedings ICASSP '82*, pages 1385–1388, 1982.

[37] O. Tanrikulu and J. A. Chambers. Convergence and steady-state properties of the least mean mixed-norm(lmmn) adaptive filtering. *IEE proceedings-VIS Image Signal Processing.*, 143(3):137–142, 1996.

[38] V. J. Mathews and Z. Xie. A stochastic gradient adaptive filter with gradient adaptive step-size. *IEEE transactions on Signal Processing.*, 41(6):2075–2087, 1993.

[39] M. Tarrab, and A. Feuer. Convergence and Performance Analysis of the Normalized LMS algorithm with Uncorrelated Gaussian data. *IEEE Tansactions on Information Theory*, 34:680–691, 1998.

[40] L. L. Horowitz and K. D. Senne. Performance advantage of complex LMS for controlling narrow-band adaptive arrays. *IEEE Transactions Circuits and Systems.*, 28(6):562–576, 1981.

[41] D. T. M. Solck. On the Convergence behavior of the LMS and NLMS Algorithms. *IEEE transactions on Signal Processing.*, 41:2811–2825, 1993.

[42] A. Zerguine. Convergence and Steady-State Analysis of the Normalized Least Mean Fourth Algorithm. *Submitted in IEEE transactions on Signal Processing.*

[43] H. Urkowitz. *Signal Theory and Random Processes.* Dedham, MA: Artech House, 1983.

[44] M. Rupp and A. H. Sayed. A Time-Domain Feedback Analysis Of Filtered-Error Adaptive Gradient Algorithms. *IEEE transactions on Signal Processing.*, 44:1428–1439, 1996.

[45] A. H. Sayed and M. Rupp. *Robustness Issues in Adaptive Filtering, DSP Handbook.* CRC Press, 1998.

[46] M. Rupp. LMS tracking behavior under periodically changing systems. *Proceedings European Signal Processing Conference, Island of Rhodes, Greece*, 1998.

[47] N. R. Yousef and A. H. Sayed. A Generalized Tracking Analysis of Adaptive Filtering Algorithms in Cyclicly and Randomly varying Environments. *Proceedings International Conference On on Acoustic, Speech and Signal Processing (ICASSP)*, 1:440–443, 2000.

[48] M. Rupp and A. H. Sayed. A Time-Domain Feedback Analysis Of Adaptive Algorithms via the Small Gain Theorem . *Proceedings. SPIE*, 2563:458–469, 1995.

Vitae

- Muhammad Moinuddin.
- Born in Karachi, Pakistan.
- Received Bachelor of Engineering (B.E.) degree in Electrical Engineering from NED University of Engineering and Technology, Karachi, Pakistan in April 1998.
- Joined the Department of Electrical Engineering at KFUPM as a Research Assistant in September 1999.
- Completed Master of Science (M.S.) in Electrical Engineering at KFUPM in May 2001.