The Leaky Least Mean Fourth Algorithm

BY

OBAID UR REHMAN KHATTAK

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

June 2009

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **OBAID UR REHMAN KHATTAK** under the direction of his thesis adviser 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

Azeddine Zerguine

Dr. Azeddine Zerguine (Adviser)

Dr. Asrar U.H Sheikh (Member)

Dr. Samir Abdul-Jauwad (Member)

Dr. Samir Abdul-Jauwad
Department Chairman

Dr. Salam Zummo
Dean of Graduate Studies

9/12/09
Date

عمادة الدراسات العليا
DEANSHIP OF GRADUATE STUDIES

Dedicated to my loving Mother & Father

ACKNOWLEDGMENTS

In the name of Allah, the Most Beneficent Most Merciful

All praise is due to Allah, the source of all knowledge and strength. I acknowledge His infinite mercy and grace in making this work a success. And may His peace and blessings be upon his final messenger Muhammad, a guidance and inspiration to our lives. The successful completion of this work was made possible by a number of major contributions from different persons and organizations alike, to whom I wish to express my due gratitude.

I am grateful to King Fahd University of Petroleum & Minerals for giving me the opportunity to carry out this task under the guidance of a scholarly faculty and a proper research environment. My sincere gratitude and thanks goes to my thesis advisor Dr. Azeddine Zerguine for his meticulous attention, his guidance, and his patience with me. I am thankful to him for the multitudes of favors I have taken from him, and for all the things he has taught me. I would also like to extend my appreciation to my thesis committee members Dr. Samir Abdul-Jauwad and Dr. Asrar U.H. Sheikh for their helpful insights and advice.

Acknowledgment is due to Dr. Moinuddin for long hours of discussions on my thesis, and for his countless favors ever since I entered KFUPM.

I am grateful to all my friends I have made during my time here. I will never forget the time we spent together. We certainly enjoyed doing our theses during the same time, sharing long hours of frustration and hardwork together, spending long nights keeping awake for each other just so that thesis simulations would seem less difficult.

My heartfelt gratitude goes to my parents. I would never have been able to pursue this task without their cooperation and understanding. I am eternally indebted to my mother for all her prayers, concern, love and understanding. She has been my first university, and whatever knowledge or abilities I have today have their roots in the love and teachings with which she has nurtured me.

TABLE OF CONTENTS

LIST OF FIGURES

THESIS ABSTRACT

NAME:	Obaid ur Rehman Khattak
TITLE OF STUDY:	The Leaky Least Mean Fourth Algorithm
MAJOR FIELD:	Electrical Engineering
DATE OF DEGREE:	JUNE 2009

This thesis presents the work done on a leakage-based variant of the Least Mean Fourth algorithm. The performance assessment of the algorithm is carried out using the concepts of energy conservation. This includes its steady state, tracking and transient analysis.Finally, a number of simulations have been carried out to compare the proposed algorithm to the Leaky LMS and the conventional LMF as well as prove the accuracy of the theoretical findings.

Keywords: *Adaptive filters, LMF, Weight Drift, Stability, LLMF algorithm.*

خلاصة الرسالة

الأسم : عبيد الرحمن خطاك

عنوان الرسالة: الخوارزمية الماصة للمتوسط الرابع الاقل

الدرجة الممنوحة: ماجستير في العلوم

حقل التخصص: الهندسة الكهربائية

تاريخ منح الدرجة: يونيو 2009م

هذه الرسالة تقدم العمل الذي تم على نوع ماص من خوارزمية المتوسط الرابع الاقل. تقييم اداء هذه الخوارزمية تم باستخدام مبدأ حفظ الطاقة. هذا التقييم يشمل الحالة المستقرة و تحليل التتبع والحالة الانتقالية. في الاخير تمت محاكاة الخوارزمية و مقارنتتها بعدد من الخوارزميات الاخرى كخوارزمية المتوسط الرابع الاقل التقليدية و الخوارزمية الماصة للمتوسط الثاني الاقل.

مفردات الخوارزمية الماصة للمتوسط الرابع الاقل الخوارزمية الماصة للمتوسط الثاني الاقل, الحالة المستقرة.

درجة الماجستير في العلوم

جامعة الملك فهد للبترول و المعادن
يونيو 2009م

Nomenclature

Notations

$\mathbf{u}_k$	:	Input regression vector at time k
$\mathbf{R}$	:	Regressor covariance matrix
$\mathbf{w}_o$	:	Unknown system weight vector
$\mathbf{w}_k$	:	Adaptive filter weight vector
n_k	:	Noise at time k
σ_u^2	:	Variance of input regressors
σ_n^2	:	Variance of noise
ξ_n^k	:	k^{th} order moment of noise
M	:	Filter length
d_k	:	Output of unknown system at time k
e_k	:	Output error for adaptive system at time k
J	:	Cost function
μ	:	Step size of adaptive algorithm
α	:	Leakage Factor
$\mathbf{v}_k$	:	Weight error vector at time k
e_{ak}	:	a priori output estimation error at time k
$e_{ak}^{\mathbf{A}}$	:	weighted a priori output estimation error at time k for weighting matrix $\mathbf{A}$
$\mathbf{T}$	:	Companion Form Matrix of $\mathbf{R}$
$\mathbf{I}$	:	Identity matrix

Abbreviations

LMS	:	Least Mean Square
LMF	:	Least Mean Fourth
LLMF	:	Leaky Least Mean Fourth
MSE	:	Mean Squared Error
EMSE	:	Excess Mean Squared Error
MSD	:	Mean Square Deviation

CHAPTER 1

INTRODUCTION

This thesis work is concerned with a member of the least mean fourth family of on-line adaptive filter algorithms. Adaptive filters are widely used in our everyday lives in a variety of areas such as plant modeling or system identification, noise cancelation and adaptive equalization, to name a few. The theory, benefits and applications of adaptive filters have been widely described in literature (see [1], [2] and references therein). We will go into more detail into the aforementioned applications of adaptive filtering in the next section.

The most important motivation for the development of adaptive filter theory has been the tracking of changes in parameters of the environment in which the filter is being used. Of course, with changes in the environment, the parameters of the filters being used will also change to keep the behavior of the overall system of the filter and the environment to continue to be agreeable to our purposes.

As an example, consider the use of adaptive filters in wireless communication systems. An inherent property of wireless communication channels is their time-varying behavior which is shown by their changing amplitude and phase response characteristics. In order to combat the Inter Symbol Interference (ISI) occurring due to the multipath property of these channels, the inverse filter of the channel to remove the ISI requires the capability to change its parameters in accordance with changes in the wireless channels so that the behavior of the overall system of the channels and inverse filter, i.e., minimum ISI, is maintained. In communication literature, such an inverse filter is known as an ”equalizer” and equalizers which have the property of adapting themselves to the channel are known as ”adaptive equalizers” [3].

An adaptive filter is characterized by the adaptive algorithm that is implemented therein. These adaptive filter algorithms can be classified in a number of ways. For example, we can classify them according to batch-processing algorithms which process a collection of data inputs at the same time or online algorithms which process the input data as it arrives i.e in real time. They can also be categorized according to supervised and unsupervised adaptive filters where the former use a training sequence to adjust its parameters in the beginning and then switch to decision directed mode at the steady state to track variations in the environment whereas the latter do not use a training sequence at all and instead use the statistical properties of the signals.

The common property that all these algorithms share is the use of a cost function which describes the deviation of the actual behavior of the filter from the behavior that is needed. The algorithm then processes the signals with the aim of reducing this deviation, or equivalently, minimizing the cost function.

From this point onwards, we will only consider the supervised adaptive filtering category.

1.1 System Model for Adaptive Filters

Before proceeding to give a general overview of the prominent online adaptive algorithms, it is instructive that we formulate the problem that is solved using the theory of the adaptive filters. Consider the case of an adaptive identification problem as shown in Fig. 1.1. The output d_k is given by

$$d_k = \mathbf{u}_k \mathbf{w}_\mathrm{o} + n_k, \tag{1.1}$$

where

$$\mathbf{w}_\mathrm{o} = [w_{\mathrm{o}1}, w_{\mathrm{o}2}, ..., w_{\mathrm{oM}}]^T \tag{1.2}$$

is the vector of the unknown system parameters and

$$\mathbf{u}_k = [u_{1k}, u_{2k} ..., u_{Mk}] \tag{1.3}$$

is the input data vector at time k, n_k is the plant noise, M is the number of plant parameters and $[.]^T$ is the transpose operation. The inputs $u_{1k}, u_{2k} ..., u_{Mk}$ may be successive samples of some signal, such as in the case of adaptive echo cancelation and adaptive line enhancement. They may also be the instantaneous outputs of M parallel sensors, such as in the case of adaptive beamforming. The identification of the plant is performed by an adaptive FIR filter whose weight vector $\mathbf{w}_k$, assumed of dimension M, is adapted on the basis of error e_k given by

$$e_k = d_k - \mathbf{u}_k \mathbf{w}_k. \tag{1.4}$$

It is important to note at this point that regardless of whether the problem to be solved using adaptive filters is a system identification problem, a channel estimation problem or an inverse system estimation problem etc., the same adaptive filter algorithm can be used. The only difference between the different problems is the definition of e_k. For example e_k defined above for the plant identification problem is the difference between the known output of the unknown system and the output of the FIR adaptive filter whereas for the inverse system estimation problem, e_k is defined as the difference between the output of the inverse system and the known input d_k at time k to the system whose inverse system is to be estimated.

It is this error e_k which is used as the independent variable in the objective function for adaptive filtering. But since e_k is a function of the weight vector $\mathbf{w}_k$, the objective function can, therefore, be formulated as function of this weight vector and minimization of the cost function will give us the optimal weight vector in the sense of the objective function used. This important observation will be useful when we review some of the more important and prominent applications of adaptive filtering in the coming section.

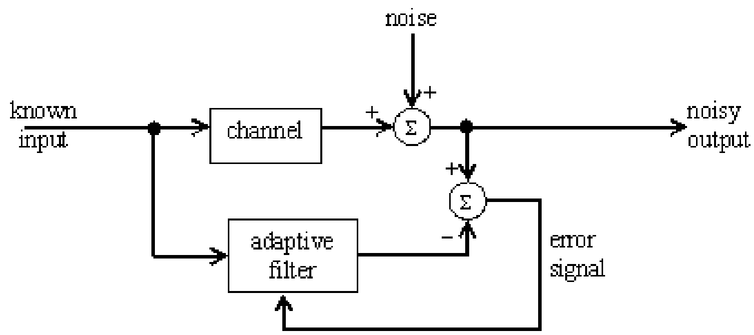

Figure 1.1: Adaptive filter.

1.2 Applications of Adaptive Filters

Adaptive filters has a number of applications, one of which was the system identification problem that was formulated in the previous section. Other applications that widely employ adaptive filters in their implementation are

- Inverse modeling or equalization
- Noise cancelation

Although these applications are quite different in nature, they have one important feature in common: An input signal and a desired output response signal. As was described at the end of the previous section, the main difference in formulating these problems into a structure suitable for applying an adaptive filtering solution is the manner in which the desired response is extracted.

Now we shall study in detail the applications of adaptive filtering that have been mentioned.

1.2.1 System Identification

The problem of system identification arises when we want to model a certain system or plant whose parameters are unknown to us and which may be time-varying. In this case, we feed the same known input into the system as well as an adaptive filter. The responses of the adaptive filter and the system are then compared and the difference between them i.e. the error, is then used to adjust the parameters of the adaptive filter iteratively. As the number of iterations increase, the parameters of the adaptive filter approach those of the system in a specific sense as dictated by the criterion used.

1.2.2 Inverse Modeling or Equalization

A brief overview and need for equalization was given in the introduction. However, we shall now explain how it can be formulated into a problem solvable by use of adaptive filters. Therefore, we can view equalization as the problem of estimating the inverse of an unknown noisy system as shown in Fig. 1.2. A delay is introduced into the desired response path to account for the delay of the signal through the channel. This ensures that the adaptive filter is causal and stable. The primary use of an equalizer is to reduce Inter-Symbol Interference (ISI) in digital communication receivers, as was described before.

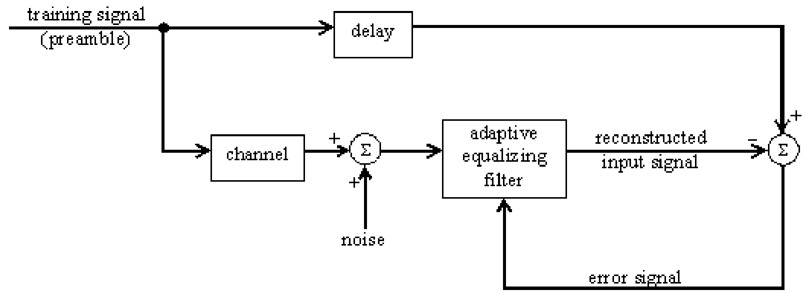

Figure 1.2: Inverse modeling problem.

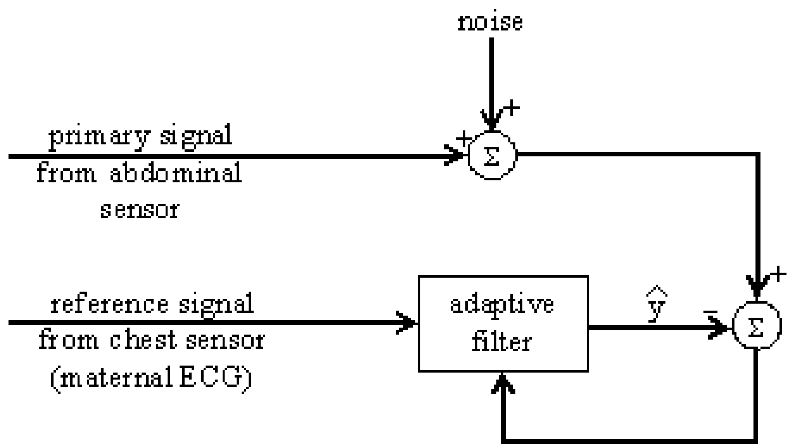

Figure 1.3: Noise cancelation problem.

1.2.3 Noise Cancelation

In this application, the adaptive filter is used to cancel unknown interference in a primary signal as shown in Fig. 1.3. The primary signal in this case serves as the desired response of the system. This type of application is used in adaptive beamforming and adaptive noise cancelation [1], [2].

1.3 Adaptive Filtering Algorithms Theory

In this section, a brief background on stochastic gradient algorithms, which include the Least Mean Square (LMS) and the Least Mean Fourth (LMF) family of algorithms, is given.

Concepts of cost functions, the steepest descent methods to achieve the minimum of these cost functions and how stochastic gradient algorithms stem from the steepest descent methods will also be discussed.

After that, we shall give a brief overview of the LMS and the LMF family of adaptive algorithms.

1.3.1 Adaptive Filter Theory

Now, a brief description of the fundamental ideas that are most widely used in the design of adaptive algorithms will be given. First we will describe what is meant by steepest descent and Newton's methods and then stochastic gradient methods in the context of adaptive filtering will be studied. Finally, we will list some of the prominent stochastic gradient algorithms that have been developed. These algorithms include the LMF [4] and NLMF [5] algorithm.

Steepest Descent Method

The steepest descent method [1],[2],[6] is a popular method used in unconstrained optimization. The basic idea of the steepest descent method is to use a scalar cost function of a variable, be it scalar-valued, vector-valued or matrix-valued, and iteratively find the optimum value of this independent variable such that the cost function is minimum at that optimal value.

The steepest descent method is a well-documented method of finding optimum values when the optimal values can not be found in closed form.

To put this in mathematical terms, consider a cost function $J(\mathbf{w})$ which is a continuously differentiable function of some unknown weight vector $\mathbf{w}$. This function maps the elements of $J(\mathbf{w})$ into real numbers. We want to find an optimal solution solution $\mathbf{w}_\text{o}$ that satisfies the following condition:

$$J(\mathbf{w}_\text{o}) \leq J(\mathbf{w}). \tag{1.5}$$

In the steepest descent method, we start with an initial guess for $\mathbf{w}_\text{o}$ and denote it by $\mathbf{w}_0$, generate a sequence of weight vectors $\mathbf{w}_1, \mathbf{w}_2, \ldots$, such that the cost function $J(\mathbf{w}_k)$ comes closer to a local minimum at each iteration k; that is,

$$J(\mathbf{w}_{k+1}) < J(\mathbf{w}_k). \tag{1.6}$$

Before proceeding further, it is necessary that the reason for stating that the cost function reaches it local minimum value be understood. The reason is that the function may not be a convex function in which case the only local minimum is the global minimum. Examples of such non-convex cost functions frequently arise in the study of unsupervised adaptive filtering algorithms [6] which have 2 or more local minima. This brings forth a drawback of the steepest descent method, that is, this method does not distinguish between local and global minima; depending on the choice of the initial guess, the cost function could converge to a value that is not the absolute minimum. Now, proceeding forward, we write in more explicit mathematical terms the steepest descent method by the recursive equation

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu \mathbf{p}, \tag{1.7}$$

where $\mathbf{w}_{k+1}$ is the updated weight vector at time $k+1$, $\mathbf{w}_k$ is the current weight vector, μ is the step size, k is the time index and $\mathbf{p}$ is the update direction vector. It is shown in [2] that a proper choice for $\mathbf{p}$ such that $\mathbf{w}$ converges to the proper value is given by

$$\mathbf{p} = -\mathbf{S}[\nabla_{\mathbf{w}} J(\mathbf{w}_k)], \tag{1.8}$$

where $\mathbf{S}$ is any positive-definite matrix and

$$\nabla_{\mathbf{w}} J(\mathbf{w}) = \frac{\partial J}{\partial \mathbf{w}}|_{\mathbf{w}=\mathbf{w}_k}. \tag{1.9}$$

This value for $\mathbf{p}$ has an interesting interpretation. The direction of $\mathbf{p}$ at a point is opposite to the direction in which the cost function is increasing, which incidently, is the direction of the gradient vector of the function at that point.Therefore, we move along the surface of the cost function towards its minimum because of the negative sign in (1.8). When the cost function reaches its local minimum, relative to the initial guess, the gradient will become zero and the weight vector converges to a finite value.

Another important aspect of the steepest descent method is the selection of a proper step size μ. A value too small will lead to slow convergence whereas a value too large might make the method unstable. For example, it can be shown that in the case of minimizing the mean square cost function, to be discussed later, the range of values μ can take while keeping the algorithm stable are between 0 and $\frac{2}{\lambda_{max}}$ where λ_{max} is the largest eigenvalue of the correlation matrix $\mathbf{R}$ of the input vector $\mathbf{u}_k$ given as

$$\mathbf{R} = E[\mathbf{u}_k^T \mathbf{u}_k]. \tag{1.10}$$

The main advantage of the steepest descent method is its simplicity. However, the convergence rate may be too slow in the case of steepest descent method. This is due to the fact that this method is based on the first order approximation of the error-performance surface around the current point in that it only uses the first-order derivatives i.e. the gradient, in its update equation.

A faster rate of convergence can be achieved by using a second-order approximation of the error-performance surface around the current point, which translates to assigning to $\mathbf{S}$ the value of the inverse of the Hessian matrix [1],[2] of the cost function. This method is known as Newton's method.

Stochastic Gradient Methods

There are two types of objective functions used in adaptive filtering-stochastic and deterministic. Objective functions which are given in terms of statistics of the input signals are stochastic objective functions whereas functions which act on the actual values of the signals are known as deterministic objective functions. Examples of the former is the least mean square and least mean fourth criteria whereas an example of the latter is the least squares criteria [1],[2],[6].

When using the steepest descent method to optimize stochastic cost functions, the gradient and the Hessian matrices of the stochastic cost functions with respect to the weight vector are also stochastic in nature. However, in practice, we do not have information about the stochastic properties of the signal and only have the instantaneous values. For this reason, when using the steepest descent method in this case, we try to approximate the gradient and/or the Hessian Matrix using functions. The resulting algorithms are known as Stochastic gradient algorithms.

Because we are using approximations to the true gradient and/or Hessian matrix, there will be a difference in the successive values the adaptive filter weight vector obtains using the steepest descent method and the corresponding stochastic gradient method at each iteration. This difference is termed as gradient noise. The more accurate the approximation functions, the closer the performance of the stochastic gradient algorithm will be to the corresponding steepest descent algorithm and smaller will the gradient noise [1],[2].

A stochastic gradient algorithm based on the steepest descent method to minimize the mean square error criterion is the Least Mean Square (LMS) algorithm and the stochastic gradient algorithm for the least mean fourth criterion is the Least Mean Fourth (LMF) [4] algorithm. The LMF algorithm is the subject of interest in this thesis.

1.3.2 Least Mean Square Family of Algorithms

The Least Mean Square (LMS) algorithm was first proposed in 1960 by Widrow and Hoff [7] and has since become the benchmark of adaptive filter theory. Very few algorithms in estimation and filtering theories have found so much success and used in so many widespread areas line echo cancelation, antenna beamforming and system identification, to name a few.

The LMS algorithm is a stochastic gradient algorithm that minimizes the Mean Square Error (MSE) criterion [1],[2] given by

$$J = E[e^2]. \tag{1.11}$$

The resulting LMS recursion equation is found to be [1],[2]

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu e_k \mathbf{u}_k^T. \tag{1.12}$$

Another variation of the LMS algorithm is the Normalized LMS (NLMS) algorithm proposed independently by Nagumo and Noda [8] and Albert and Gardner [9] which is given by

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \frac{\mu}{\epsilon + \|\mathbf{u}_k\|^2} e_k \mathbf{u}_k^T. \tag{1.13}$$

The NLMS has the advantage of removing the bias of the norm of the input $\mathbf{u}_k$ on the update of the weight vectors, which has an adverse effect on the performance of the LMS algorithm [1],[2].

An important variant of the LMS algorithm is the Leaky LMS [13]. The Leaky LMS was proposed to stabilize the weight drift problem (i.e. the possibility of unbounded weight estimates) that may occur in LMS in the presence of noise or in finite word-length implementations. The weight drift phenomenon causes overflow and degrades performance in many applications. More details about the Leaky LMS and the weight drift problem will be discussed in a later section as the Leaky LMS is the inspiration for the Leaky LMF proposed in this thesis.

1.3.3 Least Mean Fourth Family of Algorithms

The Least Mean Fourth family of adaptive algorithms were first proposed in [4] to the Least Mean Square (LMS) algorithm [1],[2]. The goal of the algorithms was to give a lower steady-state of misadjustment for a given rate of convergence using a different cost function where misadjustment Υ is defined as

$$\Upsilon = \frac{MSE - J_{min}}{J_{min}}, \tag{1.14}$$

where MSE is the steady state mean square error given as

$$MSE = lim_{k \to \infty} E[e_k^2]. \tag{1.15}$$

At this point, we can provide another useful expression for e_k which is given as follows

$$\begin{aligned} e_k &= d_k - \mathbf{u}_k \mathbf{w}_k \\ &= \mathbf{u}_k(\mathbf{w}_\mathrm{o} - \mathbf{w}_k) + n_k \\ &= e_{ak} + n_k, \end{aligned} \tag{1.16}$$

with e_{ak} being the *a priori output estimation error* given as

$$e_{ak} = \mathbf{u}_k(\mathbf{w}_\mathrm{o} - \mathbf{w}_k). \tag{1.17}$$

The cost function from which the LMF is derived is given as

$$J(\mathbf{w}) = E[(d_k - \mathbf{u}_k \mathbf{w}_k)^4], \tag{1.18}$$

with the corresponding update equation characterizing LMF being

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu e_k^3 \mathbf{u}_k^T. \tag{1.19}$$

It is shown in [4] that for the LMF algorithm to converge, the step size μ must be between 0 and $\frac{1}{3\sigma_n^2 \lambda_{max}}$, exclusive, where σ_n^2 is the variance of the noise and λ_{max} is the largest eigenvalue of the correlation matrix $\mathbf{R}$ as defined before.

To overcome the dependance of the step size value on the input statistics, the normalized version of LMF, known as Normalized LMF (NLMF), was proposed in [5] which has the following recursive equation:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \mu e_k^3 \frac{\mathbf{u}_k^T}{\|\mathbf{u}_k\|^2}. \tag{1.20}$$

1.4 The Weight Drift and Leaky LMS

There are a number of papers on this phenomenon which is a cause of instability in LMS adaptive filters [10]-[13].To begin with, recall that the conventional LMS recursion is given by

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{w}_k + \mu e_k \mathbf{u}_k^T, & (1.21) \\ e_k &= d_k - \mathbf{u}_k \mathbf{w}_k. & (1.22) \end{aligned}$$

The weight drift problem can be understood by the following example: Assume, that at iteration k, the input vector $\mathbf{u}_k$ is orthogonal to the weight error vector $\mathbf{v}_k = \mathbf{w}_\mathrm{o} - \mathbf{w}_k$ i.e $\mathbf{u}_k \mathbf{v_k}$ It then follows that $d_k - \mathbf{u}_k \mathbf{w}_k = n_k$. Consequently, the weight error vector satisfies the update equation

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \mu \mathbf{u}_k^T n_k. \tag{1.23}$$

Taking norms, we get

$$\|\mathbf{v}_{k+1}\|^2 = \|\mathbf{v}_k\|^2 + \mu^2 \|\mathbf{u}_k\|^2 n_k^2. \tag{1.24}$$

Solving this recursion for $||\mathbf{v}_N||^2$, we get

$$||\mathbf{v}_N||^2 = ||\mathbf{v}_0||^2 + \sum_{i=1}^{N} \mu^2 \|\mathbf{u}_i\|^2 n_i^2. \tag{1.25}$$

This relation shows that $||\mathbf{v}_N||^2 \to \infty$ as $N \to \infty$, if $\mu\|\mathbf{u}_i\|^2 n_i$ is not a finite energy sequence. This situation usually occurs in practical scenarios when the following two conditions are satisfied [2]:

1. The input covariance matrix is singular. This phenomenon occurs in digitally implemented fractionally space equalizers.

2. The quantization noise or output noise is non-zero mean.

This situation does not occur with Leaky LMS algorithm [13] described by the following update equation:

$$\mathbf{w}_{k+1} = (1 - \mu\alpha)\mathbf{w}_k + \mu \mathbf{u}_k^T e_k. \tag{1.26}$$

where α is the leakage parameter. The term *leakage* stems from the fact that, unlike the conventional LMS, where the weights remain static in case of stalling i.e. the input sequence becomes zeros, in Leaky LMS, the weights *leak out* i.e. become zeros.

To see how Leaky LMS mitigates the drift problem in LMS algorithm, using the same example and by the same steps of computation, we get

$$||\mathbf{v}_{k+1}||^2 = (1 - \mu\alpha)^2 \|\mathbf{v}_k\|^2 + \mu^2 \|\mathbf{u}_k\|^2 n_k^2, \tag{1.27}$$

so that $\|\mathbf{v}_{k+1}||^2$ remains bounded for $0 < \mu\alpha < 1$.

However, the Leaky LMS does add bias to the solution and $\|\mathbf{v}_{k+1}||^2$ does not reach 0 except for when $\alpha = 0$ which is the case for LMS [14]. To remove this bias, there are a number of variants of the Leaky LMS algorithm which mitigate the weight drift problem yet give the same misadjustment as the conventional LMS algorithm. Examples of these include Circular Leaky LMS [14] and the Subspace Leaky LMS [15].

1.5 Motivation for Leaky LMF

The description and use of the Leaky LMS was described in the previous section. The LMF algorithm, just like the LMS algorithm, suffers from the weight drift problem. Considering this fact, we shall employ the leakage technique to the LMF algorithm and refer to the resulting algorithm as the Leaky LMF.

1.6 Thesis Objectives

The aim of this thesis is to derive the Leaky-LMF algorithm which would be the LMF counterpart of the Leaky-LMS algorithm, establish the condition for convergence and then compare the performance to the LMF algorithm. We will also perform the steady-state, transient and tracking analysis on the proposed algorithm.

We shall now tabulate the objectives of the thesis:

1. To derive the recursive update equation of the Leaky LMF adaptive algorithm.

2. To find the range of values for which the step size in the recursive update equation of the Leaky-LMF guarantees convergence of the algorithm.

3. To derive the steady-state analysis of Leaky LMF.

4. To derive the tracking analysis of Leaky LMF where we will see how capable the newly proposed algorithm is of tracking changes in the environment.

5. To derive the transient analysis of the Leaky LMF algorithm.

CHAPTER 2

THE LEAKY LEAST MEAN FOURTH ADAPTIVE ALGORITHM

2.1 Introduction

In this chapter, the cost function used in the development of the proposed algorithm is presented followed by the derivations of the corresponding steepest descent algorithm and stochastic gradient algorithms. The resulting stochastic gradient update equation formulated will then fully describe the proposed leaky Least Mean Fourth algorithm. Following this, we will discuss the fundamental weighted energy relation that will be used in the transient, steady state and tracking analysis of the Leaky Least Mean Fourth algorithm.

2.2 Proposed Algorithm

The algorithm that is proposed in this thesis is the leaky Least Mean Fourth Algorithm. The assumptions used in the analysis are stated

A1 There exists a vector $\mathbf{w}_o$ such that $d_k = \mathbf{u}_k\mathbf{w}_o + n_k$.

A2 The noise sequence $\{n_k\}$ is i.i.d. with zero odd order moments and variance $\sigma_n^2 = E[n_k^2]$.

A3 The sequence n_k is independent of $\mathbf{u}_j$,$\mathbf{w}_k$ for all j,k.

A4 The regressor covariance matrix is $\mathbf{R} = E[\mathbf{u}_k^T\mathbf{u}_k] > 0$.

A5 The random variables $\{d_k, \mathbf{u}_k, n_k\}$ have zero means.

To develop the proposed algorithm, we will be using the system identification model (1.1). The stochastic cost function which is used as a basis for the proposed algorithm is given as

$$J(\mathbf{w}) = E[(d_k - \mathbf{u}_k\mathbf{w}_k)^4] + \alpha||\mathbf{w}_k||^2, \tag{2.1}$$

where α is the leakage factor. The direction vector of this function $\mathbf{p}$ (1.8) is then given as

$$\mathbf{p} = -\nabla_{\mathbf{w}} J(\mathbf{w}_k) = 4E[(d_k - \mathbf{u}_k\mathbf{w}_k)^3] + 2\alpha\mathbf{w}_k. \tag{2.2}$$

Putting this in (1.7), we get the resulting steepest descent update equation to minimize (2.1) given as

$$\mathbf{w}_{k+1} = (1 - 2\mu\alpha)\mathbf{w}_k + \mu\mathbf{u}_k^T E[e_k^3], \tag{2.3}$$

where we have used (1.22).The stochastic gradient update equation originating from this to minimize (2.1) is found by removing the expectation operator and is

$$\mathbf{w}_{k+1} = (1 - \mu\alpha)\mathbf{w}_k + \mu\mathbf{u}_k^T e_k^3, \tag{2.4}$$

where the factor 2 is absorbed into μ.

As we can see,(2.4) is similar in form to the Leaky LMS update equation (1.26) and we will show through simulations how this algorithm helps to prevent the weight drift problem due to finite precision effects from occurring.

2.3 Weighted Energy Conservation Relation

The fundamental energy conservation relation [2][16]-[18] is a very useful framework for the analysis of adaptive filters which will be used in this thesis to study the performance behavior of the Leaky Least Mean Fourth adaptive algorithm. Its main advantage is that it can be applied across a wide spectrum of adaptive filter algorithms without resorting to restrictive assumptions that are generally used in the literature in the study of adaptive filtering algorithms. These restrictive assumptions include the Gaussianity assumption on the noise. The general nature of this approach also allows for easy comparison between different algorithms.

Before we proceed, the weighted squared Euclidean norm of a column vector $\mathbf{x}$ is first defined as

$$||\mathbf{x}||^2_{\mathbf{A}} = \mathbf{x}^T\mathbf{A}\mathbf{x}, \tag{2.5}$$

where $\mathbf{A}$ is some positive-definite symmetric weighting matrix. The choice $\mathbf{A} = \mathbf{I}$ results in the standard Euclidean norm of x

$$||\mathbf{x}||^2 = \mathbf{x}^T\mathbf{x}. \tag{2.6}$$

To commence the energy conservation relation for the Leaky LMF, we shall start with the Leaky LMF update equation given by (2.4). Subtracting both sides of (2.4) from $\mathbf{w}_{\mathrm{o}}$, we get

$$\mathbf{v}_{k+1} = (1 - \mu\alpha)\mathbf{v}_k + \mu\alpha\mathbf{w}_{\mathrm{o}} - \mu\mathbf{u}_k^T e_k^3. \tag{2.7}$$

Taking the weighted norms of both sides (2.7), using some positive definite weighting matrix $\mathbf{A}$, we get the following weighted energy relation for the Leaky LMF:

$$\begin{aligned}
||\mathbf{v}_{k+1}||^2_{\mathbf{A}} &= (1-\mu\alpha)^2||\mathbf{v}_k||^2_{\mathbf{A}} + ||\mu\alpha\mathbf{w}_o||^2_{\mathbf{A}} + \mu^2||\mathbf{u}_k||^2_{\mathbf{A}}e_k^6 \\
&\quad +2\mu\alpha(1-\mu\alpha)\mathbf{w}_o^T\mathbf{A}\mathbf{v}_k - 2\mu(1-\mu\alpha)e_k^3\mathbf{u}_k\mathbf{A}\mathbf{v}_k \\
&\quad -2\mu^2\alpha\mathbf{w}_o^T\mathbf{A}e_k^3\mathbf{u}_k \\
&= (1-\mu\alpha)^2||\mathbf{v}_k||^2_{\mathbf{A}} + ||\mu\alpha\mathbf{w}_o||^2_{\mathbf{A}} + \mu^2||\mathbf{u}_k||^2_{\mathbf{A}}e_k^6\mathbf{v}_k \\
&\quad -2\mu(1-\mu\alpha)e_k^3e_{ak}^{\mathbf{A}} + 2\mu\alpha\mathbf{w}_o^T\mathbf{A}\left[(1-\mu\alpha)\mathbf{v}_k - \mu e_k^3\mathbf{u}_k\right]. \qquad (2.8)
\end{aligned}$$

where $e_{ak}^{\mathbf{A}} = \mathbf{u}_k\mathbf{A}\mathbf{v}_k$ is the weighted a-priori estimation error. For $\mathbf{A} = \mathbf{I}$, we have the standard a-priori estimation error e_{ak} as defined in (1.17).

(2.8) is the weighted energy conservation relation that will be used in the coming chapters to study the performance of the Leaky LMF adaptive algorithm in terms of

Steady State Analysis, which relates to determining the steady state values of $E[||\mathbf{v}_k||^2]$, $E[e_{ak}^2]$ and $E[e_k^2]$.

Stability, which relates to determining the range of values of the step-size over which $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$ remain bounded.

Transient Analysis, which is concerned with studying the time evolution of $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$.

CHAPTER 3

TRANSIENT ANALYSIS OF THE PROPOSED LEAKY LMF ADAPTIVE ALGORITHMS

3.1 Introduction

The transient analysis of adaptive algorithms lends its importance to the requirement of the algorithms to adapt to changes in the signal statistics in a quick and stable manner. Therefore, the study of the transient behavior of the adaptive algorithms in terms of convergence rates and stability conditions is an essential part of adaptive filter performance analysis. As was stated in the previous chapter, (2.8) will be used to pursue the transient analysis.

3.2 Transient Analysis

The transient analysis carried out in this chapter will deal with the following three questions that frequently arise when dealing with adaptive filter properties:

1. What are the ranges of the step size for which the $||\mathbf{v}_k||$ and e_k remain bounded in both the mean and mean-square sense?

2. How does $E[||\mathbf{v}_k||^2]$ and $E[e_k^2]$evolve with time?

The first question is answered by finding the bounds on the step size values for which $||\mathbf{v}_k||$ remains bounded. The second question is answered by formulating a suitable model that predicts the values of $E[||\mathbf{v}_k||^2]$ and $E[e_k^2]$ for each time instant k.Therefore, we shall proceed by first finding the conditions for which $||\mathbf{v}_k||$ remains stable in the mean; then we will move on to construct a suitable framework to accurately model the time-evolution of $E[||\mathbf{v}_k||^2]$ and $E[e_k^2]$. After this, we shall find the conditions for which $||\mathbf{v}_k||$ remains stable in the mean-square sense.

3.2.1 Mean Convergence Behavior

To begin with, taking expectations of both sides of (2.7), we get

$$E[\mathbf{v}_{k+1}] = (1 - \mu\alpha)E[\mathbf{v}_k] + \mu\alpha\mathbf{w}_{\mathrm{o}} - \mu E[\mathbf{u}_k^T e_k^3]. \tag{3.1}$$

To solve for $E[\mathbf{u}_k^T e_k^3]$, we will use the following assumption:

A6 The regressors $\mathbf{u}_k$ are Gaussian distributed.

Although **A6** is not practical in communication scenarios where the information data is not Gaussian, it is useful in making the analysis tractable for performance comparisons with other adaptive algorithms [19]. Using this, we can find the following expression for $E[\mathbf{u}_k^T e_k^3]$ [19]:

$$E[\mathbf{u}_k^T e_k^3] = 3(\sigma_n^2 + \zeta)\mathbf{R}E[\mathbf{v}_k]. \tag{3.2}$$

where $\zeta = E[e_{ak}^2]$.

Putting the above expression for $E[\mathbf{u}_k^T e_k^3]$ in (3.1) and simplifying, we get

$$E[\mathbf{v}_{k+1}] = [I - \mu\{\alpha\mathbf{I} + 3(\sigma_n^2 + \zeta)\mathbf{R}\}]E[\mathbf{v}_k] + \mu\alpha\mathbf{w}_\text{o}. \tag{3.3}$$

To find the mean convergence condition on the step-size, we will use the approach used in [20]. Therefore, let $\vartheta \leq \zeta$ be the Cramer-Rao bound associated with estimating $\mathbf{u}_k\mathbf{w}_\text{o}$ by $\mathbf{u}_k\mathbf{w}_k$; then from (3.3), we see that $\mathbf{v}_k$ is convergent in the mean if the eigenvalues of $[I - \mu\{\alpha I + 3(\sigma_n^2 + \vartheta)\mathbf{R}\}]$ lie between -1 and 1. From this, we can find the range of step-size values for which the $\mathbf{v}_k$ remains bounded in the mean sense which is given as

$$0 < \mu < \frac{2}{\alpha + 3(\sigma_n^2 + \vartheta)\lambda_{\max}(\mathbf{R})}. \tag{3.4}$$

3.2.2 Constructing the Learning Curves

In this subsection, we will construct a state-space model that describes the time evolution of $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$. Taking the expectation of both sides of (2.8), we get

$$
\begin{aligned}
E\left[||\mathbf{v}_{k+1}||_{\mathbf{A}}^2\right] \;=\; & (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||_{\mathbf{A}}^2\right] + ||\mu\alpha\mathbf{w}_{\mathrm{o}}||_{\mathbf{A}}^2 + \mu^2 E\left[||\mathbf{u}_k||_{\mathbf{A}}^2 e_k^6\right] \\
& -2\mu(1-\mu\alpha)E\left[e_k^3 e_{ak}^{\mathbf{A}}\right] \\
& +2\mu\alpha\mathbf{w}_{\mathrm{o}}^T\mathbf{A}\left[(1-\mu\alpha)E\left[\mathbf{v}_k\right] - \mu E\left[e_k^3\mathbf{u}_k^T\right]\right]. \qquad (3.5)
\end{aligned}
$$

To proceed further with the analysis, we have to evaluate $E\left[||\mathbf{u}_k||_{\mathbf{A}}^2 e_k^6\right]$ and $E\left[e_k^3 e_{ak}^{\mathbf{A}}\right]$.

Evaluation of Term $E\left[e_k^3 e_{ak}^{\mathbf{A}}\right]$

Since e_{ak} and $e_{ak}^{\mathbf{A}}$ are jointly Gaussian by **A6** and independent of n_k by **A2**, then using Price's theorem [2] and (1.17), we can express $E\left[e_k^3 e_{ak}^{\mathbf{A}}\right]$ as

$$E\left[e_k^3 e_{ak}^{\mathbf{A}}\right] = E\left[e_{ak} e_{ak}^{\mathbf{A}}\right]\mathcal{G}_k, \qquad (3.6)$$

where $\mathcal{G}_k$ in our case is found to be

$$\mathcal{G}_k = 3(\sigma_n^2 + \zeta). \qquad (3.7)$$

Evaluation of Term $[||\mathbf{u}_k||^2_{\mathbf{A}} e_k^6]$

To evaluate this term, we will the following approximation [2]:

A7 The adaptive filter is long enough so that $||\mathbf{u}_k||^2_{\mathbf{A}}$ is independent of e_k.

Simulations done in this report have shown that even for filter lengths of 5, **A7** is reasonable. This assumption allows us to write $E\,[||\mathbf{u}_k||^2_{\mathbf{A}} e_k^6]$ as

$$E\left[||\mathbf{u}_k||^2_{\mathbf{A}} e_k^6\right] = E\left[||\mathbf{u}_k||^2_{\mathbf{A}}\right] E\left[e_k^6\right]. \tag{3.8}$$

From this, we can express $E\,[||\mathbf{u}_k||^2_{\mathbf{A}} e_k^6]$ further as

$$E\left[||\mathbf{u}_k||^2_{\mathbf{A}} e_k^6\right] = tr(\mathbf{R}\mathbf{A})\mathcal{Z}_k, \tag{3.9}$$

where

$$\begin{aligned} \mathcal{Z}_k &= E\left[e_k^6\right] \\ &= 15\zeta^3 + 45\zeta^2\sigma_n^2 + 15\zeta\xi_n^4 + \xi_n^6, \end{aligned} \tag{3.10}$$

with ξ_n^4 and ξ_v^n being the fourth and sixth order moments of n_k, respectively.

Using (3.2),(3.6) and (3.9) in (3.5) and some algebraic manipulation, we get the following result:

$$\begin{aligned} E\left[||\mathbf{v}_{k+1}||^2_{\mathbf{A}}\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||^2_{\mathbf{A}}\right] + ||\mu\alpha\mathbf{w}_{\text{o}}||^2_{\mathbf{A}} + \mu^2 tr(\mathbf{R})\mathcal{Z}_k \\ &\quad -2\mu(1-\mu\alpha)E\left[e_{ak}e_{ak}^{\mathbf{A}}\right]\mathcal{G}_k + 2\mu\alpha\mathbf{w}_{\text{o}}^T\mathbf{A}\mathbf{H}E\left[\mathbf{v}_k\right], \end{aligned} \tag{3.11}$$

where

$$\mathbf{H} = \mathbf{I} - \mu\{\alpha\mathbf{I} + 3(\sigma_n^2 + \zeta)\mathbf{R}\}. \tag{3.12}$$

More is needed in order to evaluate (3.11) since it is hard to evaluate $E\left[e_{ak}e_{ak}^{\mathbf{A}}\right]$ due to the dependencies among the regressors $\mathbf{u}_k$. Therefore, will make the following assumption [2],[16]:

A8 The sequence of vectors $\mathbf{u}_k$ are independent and identically distributed.

Using this assumption, $\mathbf{u}_k$ and $\mathbf{v}_k$ become independent since now $\mathbf{v}_k$ depends only on $\mathbf{u}_{k-1}$ which is assumed to be independent of $\mathbf{u}_k$. Therefore, we can express $E\left[e_{ak}e_{ak}^{\mathbf{A}}\right]$ as

$$\begin{aligned}
E\left[e_{ak}e_{ak}^{\mathbf{A}}\right] &= E\left[\mathbf{v}_k^T\mathbf{u}_k^T\mathbf{u}_k\mathbf{A}\mathbf{v}_k\right] \\
&= E\left[\mathbf{v}_k^T E\left[\mathbf{u}_k^T\mathbf{u}_k|\mathbf{v}_k\right]\mathbf{A}\mathbf{v}_k\right] \\
&= E\left[\mathbf{v}_k^T E\left[\mathbf{u}_k^T\mathbf{u}_k\right]\mathbf{A}\mathbf{v}_k\right] \\
&= E\left[\mathbf{v}_k^T\mathbf{R}\mathbf{A}\mathbf{v}_k\right] \\
&= E\left[||\mathbf{v}_k||^2_{\mathbf{RA}}\right].
\end{aligned} \tag{3.13}$$

From (3.13) we can see that for $\mathbf{A} = \mathbf{I}$, (3.13) results in

$$\begin{aligned}
E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] &= E\left[e_{ak}e_{ak}\right] \\
&= \zeta.
\end{aligned} \tag{3.14}$$

Now, $E\left[e_{ak}e_{ak}^{\mathbf{A}}\right]$, $\mathcal{Z}_k$, and $\mathcal{G}_k$ are functions of $\mathbf{v}_k$, so that (3.11) becomes

$$\begin{aligned} E\left[||\mathbf{v}_{k+1}||^2_{\mathbf{A}}\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||^2_{\mathbf{A}}\right] + ||\mu\alpha\mathbf{w}_{\mathrm{o}}||^2_{\mathbf{A}} + \mu^2 tr(\mathbf{RA})\mathcal{Z}_k \\ & -2\mu(1-\mu\alpha)\mathcal{G}_k E\left[||\mathbf{v}_k||^2_{\mathbf{RA}}\right] + 2\mu\alpha\mathbf{w}_{\mathrm{o}}^T\mathbf{AH}E\left[\mathbf{v}_k\right]. \quad (3.15) \end{aligned}$$

We can now use the above relation to study the transient behavior of the proposed Leaky LMF adaptive algorithm for both white as well as correlated input data. We will now develop a state-space model for both cases.

3.2.3 Transient Analysis for White Input Data

For white input data i.e $\mathbf{R} = \sigma_u^2$, using (3.14), we get

$$\begin{aligned} \zeta &= E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] \\ &= \sigma_u^2 E\left[||\mathbf{v}_k||^2\right]. \quad (3.16) \end{aligned}$$

From this, we can see that for white input data,

$$\begin{aligned} \mathcal{G}_k &= 3(\sigma_n^2 + \sigma_u^2 E\left[||\mathbf{v}_k||^2\right]), \quad (3.17) \\ \mathcal{Z}_k &= 15\left(\sigma_u^2 E\left[||\mathbf{v}_k||^2\right]\right)^3 + 45\left(\sigma_u^2 E\left[||\mathbf{v}_k||^2\right]\right)^2\sigma_n^2 \\ & +15\left(\sigma_u^2 E\left[||\mathbf{v}_k||^2\right]\right)\xi_n^4 + \xi_n^6, \quad (3.18) \\ tr(\mathbf{R}) &= M\sigma_u^2, \quad (3.19) \\ \mathbf{H} &= 1-\mu\{\alpha + 3(\sigma_n^2 + \sigma_u^2 E\left[||\mathbf{v}_k||^2\right])\sigma_u^2. \quad (3.20) \end{aligned}$$

Using (3.3) and (3.17)-(3.20), we can compactly represent the evolution of the $E\left[\mathbf{v}_k\right]$ and $E\left[||\mathbf{v}_k||^2\right]$ by the following state space equation:

$$\begin{bmatrix} E\left[||\mathbf{v}_{k+1}||^2\right] \\ E\left[\mathbf{v}_{k+1}\right] \end{bmatrix} = \begin{bmatrix} f_1 & f_2 \\ 0 & \mathbf{H} \end{bmatrix} \begin{bmatrix} E\left[||\mathbf{v}_k||^2\right] \\ E\left[\mathbf{v}_k\right] \end{bmatrix} + \mu \begin{bmatrix} \mu\alpha^2||\mathbf{w}_\mathrm{o}||^2 + M\sigma_u^2\xi_n^6 \\ \alpha\mathbf{w}_\mathrm{o} \end{bmatrix} \tag{3.21}$$

where

$$\begin{aligned} f_1 \;=\; & (1-\mu\alpha)^2 + \mu^2 15M\sigma_u^4\xi_n^4 - 6\mu(1-\mu\alpha)\sigma_n^2\sigma_u^2 + \mu^2 45M\sigma_u^6\sigma_n^2 E\left[||\mathbf{v}_k||^2\right] \\ & -6\mu(1-\mu\alpha)\sigma_u^4 E\left[||\mathbf{v}_k||^2\right] + \mu^2 15M\sigma_u^8 E\left[||\mathbf{v}_k||^2\right]^2, \end{aligned} \tag{3.22}$$

and

$$f_2 = 2\mu\alpha\mathbf{H}\mathbf{w}_\mathrm{o}^T. \tag{3.23}$$

The time evolution of $E[e_{ak}^2]$ can be found using (3.16) and (3.21). The time evolution of $E[e_k^2]$ is then found by using

$$E[e_k^2] = E[e_{ak}^2] + \sigma_n^2. \tag{3.24}$$

3.2.4 Transient Analysis for Correlated Data

For uncorrelated data, we see from (3.15) that only unweighted norms of $\mathbf{v}_k$ and $\mathbf{v}_{k+1}$ appear on both sides of the equation. However, when the input data is correlated i.e. $\mathbf{R}$ is a non-diagonal matrix, different weighting matrices will appear on both sides of the equation. To solve this problem, we shall start with (3.15) and for $\mathbf{A} = \mathbf{I}$, we get

$$\begin{aligned} E\left[||\mathbf{v}_{k+1}||^2\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||^2\right] + ||\mu\alpha\mathbf{w}_{\mathrm{o}}||^2_{\mathbf{R}} + \mu^2 tr(\mathbf{R})\mathcal{Z}_k \\ &\quad -2\mu(1-\mu\alpha)\mathcal{G}_k E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] + 2\mu\alpha\mathbf{w}_{\mathrm{o}}^T\mathbf{H}E\left[\mathbf{v}_k\right]. \end{aligned} \quad (3.25)$$

It can be seen that a weighted norm of $\mathbf{v}_k$ appears with a weighting matrix . This can be inferred from (3.15) for $\mathbf{A} = \mathbf{R}$, which leads to

$$\begin{aligned} E\left[||\mathbf{v}_{k+1}||^2_{\mathbf{R}}\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] + ||\mu\alpha\mathbf{w}_{\mathrm{o}}||^2_{\mathbf{A}} + \mu^2 tr(\mathbf{R}^2)\mathcal{Z}_k \\ &\quad -2\mu(1-\mu\alpha)\mathcal{G}_k E\left[||\mathbf{v}_k||^2_{\mathbf{R}^2}\right] + 2\mu\alpha\mathbf{w}_{\mathrm{o}}^T\mathbf{R}\mathbf{H}E\left[\mathbf{v}_k\right]. \end{aligned} \quad (3.26)$$

We see that a weighted norm of $\mathbf{v}_k$ appears again, this time with a weighting matrix $\mathbf{A} = \mathbf{R}^2$, which can then in turn be inferred from (3.15) for $\mathbf{A} = \mathbf{R}^3$. Continuing in this fashion, (3.15) for $\mathbf{A} = \mathbf{R}^{M-1}$ becomes

$$\begin{aligned} E\left[||\mathbf{v}_{k+1}||^2_{\mathbf{R}^{M-1}}\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_n||^2_{\mathbf{R}^{M-1}}\right] + ||\mu\alpha\mathbf{w}_{\mathrm{o}}||^2_{\mathbf{A}} + \mu^2 tr(\mathbf{R}^M)\mathcal{Z}_k \\ &\quad -2\mu(1-\mu\alpha)\mathcal{G}_k E\left[||\mathbf{v}_k||^2_{\mathbf{R}^M}\right] \\ &\quad +2\mu\alpha\mathbf{w}_{\mathrm{o}}^T\mathbf{R}^{M-1}\mathbf{H}E\left[\mathbf{v}_k\right]. \end{aligned} \quad (3.27)$$

where we see now that a weighted norm of $\mathbf{v}_k$ appears again, this time with a weighting matrix $\mathbf{A} = \mathbf{R}^M$.

Using the Cayley-Hamilton theorem [2], we can write $\mathbf{R}^M$ as

$$\mathbf{R}^M = -p_{M-1}\mathbf{R}^{M-1} - p_{M-2}\mathbf{R}^{M-2} - \ldots - p_1\mathbf{R} - p_0\mathbf{I}, \tag{3.28}$$

where p_0,p_1,...,$p_{M-1}$are the coefficients of the characteristic polynomial of $\mathbf{R}$, given as

$$p(x) = \det(x\mathbf{I} - \mathbf{R}). \tag{3.29}$$

Using (3.28), we have

$$E\left[||\mathbf{v}_k||^2_{\mathbf{R}^M}\right] = -p_{M-1}E\left[||\mathbf{v}_k||^2_{\mathbf{R}^{M-1}}\right] - p_{M-2}E\left[||\mathbf{v}_k||^2_{\mathbf{R}^{M-2}}\right] - \ldots - p_1E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] - p_0E\left[||\mathbf{v}_k||^2\right]. \tag{3.30}$$

Ultimately, we can combine (3.3) and (3.25)-(3.27) as

$$\underbrace{\begin{bmatrix} \mathbf{A}_{k+1} \\ E\left[\mathbf{v}_{k+1}\right] \end{bmatrix}}_{\mathbf{W}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 \\ \mathbf{0} & \mathbf{H} \end{bmatrix}}_{\mathbf{F_k}} \underbrace{\begin{bmatrix} \mathbf{A}_k \\ E\left[\mathbf{v}_k\right] \end{bmatrix}}_{\mathbf{W}_k} + \mu \underbrace{\begin{bmatrix} \mathbf{L}_k \\ \alpha\mathbf{w}_\mathrm{o} \end{bmatrix}}_{\mathbf{Y}_k} \tag{3.31}$$

with $\mathbf{A}_k, \mathbf{L}_k, \mathbf{F}_2, \mathbf{F}_1$ are given as

$$\mathbf{A}_k = \begin{bmatrix} E\left[||\mathbf{v}_k||^2\right] \\ E\left[||\mathbf{v}_k||^2_{\mathbf{R}}\right] \\ E\left[||\mathbf{v}_k||^2_{\mathbf{R}^2}\right] \\ \vdots \\ E\left[||\mathbf{v}_k||^2_{\mathbf{R}^{M-2}}\right] \\ E\left[||\mathbf{v}_k||^2_{\mathbf{R}^{M-1}}\right] \end{bmatrix} \tag{3.32}$$

$$\mathbf{L}_k = \mu\mathcal{Z}_k \begin{bmatrix} tr(\mathbf{R}) \\ tr(\mathbf{R}^2) \\ tr(\mathbf{R}^3) \\ \vdots \\ tr(\mathbf{R}^M) \end{bmatrix} + \mu\alpha^2 \begin{bmatrix} ||\mathbf{w}_\text{o}||^2 \\ ||\mathbf{w}_\text{o}||^2_{\mathbf{R}} \\ ||\mathbf{w}_\text{o}||^2_{\mathbf{R}^2} \\ \vdots \\ ||\mathbf{w}_\text{o}||^2_{\mathbf{R}^{M-1}} \end{bmatrix} \tag{3.33}$$

$$\mathbf{F}_2 = 2\mu\alpha\mathbf{w}_\text{o}^T \begin{bmatrix} \mathbf{I} \\ \mathbf{R} \\ \mathbf{R}^2 \\ \vdots \\ \mathbf{R}^{M-1} \end{bmatrix} \mathbf{H} \tag{3.34}$$

where $\mathbf{H}$ comes from (3.12) and

$$\mathbf{F}_1 = \begin{bmatrix} k_1 & -k_2 & 0 & 0 & \cdots & 0 \\ 0 & k_1 & -k_2 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & k_1 & -k_2 \\ k_2 p_0 & k_2 p_1 & \cdots & \cdots & k_2 p_{M-2} & k_1 + k_2 p_{M-1} \end{bmatrix} \tag{3.35}$$

where

$$k_1 = (1 - \mu\alpha)^2, \tag{3.36}$$

and

$$k_2 = 2\mu(1 - \mu\alpha)\mathcal{G}_k. \tag{3.37}$$

From this, we can see that that the evolution of $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$ can be described by the first and second entries of the state vector W_{k+1}, respectively. The resulting learning curve of the filter is then

$$E[e_k^2] = E[e_{ak}^2] + \sigma_n^2. \tag{3.38}$$

We can also see that for $\mathbf{R} = \sigma_u^2\mathbf{I}$, (3.31) degenerates to (3.21).

3.2.5 Mean Square Stability

As can be seen from the block triangular structure of $\mathbf{F}_k$ in (3.31), we find that one of the conditions for the mean-square stability of the Leaky LMF algorithm is that it be mean convergent. The mean convergence condition was found before and shown in (3.4). To find the second condition for the mean-square stability of the Leaky LMF to hold, we will use the same approach as was done for finding the mean convergence on the step size.

Therefore, let $\vartheta \leq \zeta$ be the Cramer-Rao bound associated with estimating $\mathbf{u}_k \mathbf{w}_\text{o}$ by $\mathbf{u}_k \mathbf{w}_k$; then $\mathcal{G}^*$ and $\mathcal{Z}^*$ are defined as

$$\mathcal{G}^* = 3(\sigma_n^2 + \vartheta), \tag{3.39}$$

$$\mathcal{Z}^* = 15\vartheta^3 + 45\vartheta^2\sigma_n^2 + 15\vartheta\xi_n^4 + \xi_n^6. \tag{3.40}$$

Using this, let us define $\mathbf{F}_1^*$ and $\mathbf{L}^*$ as follows

$$\mathbf{F}_1^* = \mathbf{F}_1|_{\mathcal{G}_k = \mathcal{G}^*}, \tag{3.41}$$

$$\mathbf{L}^* = \mathbf{L}_k|_{\mathcal{Z}_k = \mathcal{Z}^*}. \tag{3.42}$$

$\mathbf{F}_1^*$ can then be written as

$$\mathbf{F}_1^* = \mathbf{I} - \mu\mathbf{G}_1 + \mu^2\mathbf{G}_2, \tag{3.43}$$

where

$$\mathbf{G}_1 = 2(\alpha\mathbf{I} + \mathcal{G}^*\mathbf{T}), \tag{3.44}$$

and

$$\mathbf{G}_2 = \alpha(\alpha I + 2\mathcal{G}^*\mathbf{T}), \tag{3.45}$$

where in (3.44) and (3.45),

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 \\ -p_0 & -p_1 & \cdots & \cdots & -p_{M-2} & -p_{M-1} \end{bmatrix} \tag{3.46}$$

From [2], a sufficient condition for $\mathbf{F}_1$ to be stable, and thus constitute the second condition for the mean square stability of the proposed algorithm is that the step size lies in the following range:

$$0 < \mu < \frac{1}{\lambda_{\max}(\mathbf{G}_1^{-1}\mathbf{G}_2)}. \tag{3.47}$$

Combining (3.4) and (3.47), we find that the condition for $\mathbf{v}_k$ to converge in both the mean and mean square sense is

$$0 < \mu < \min\left(\frac{2}{\alpha + \mathcal{G}^*\lambda_{\max}(\mathbf{R})}, \frac{1}{\lambda_{\max}(\mathbf{G}_1^{-1}\mathbf{G}_2)}\right). \tag{3.48}$$

To be more explicit, we first note from (3.46) that $\mathbf{T}$ is a companion form matrix of $\mathbf{R}$. Therefore it has the same eigenvalues as $\mathbf{R}$. Let λ_i,λ_i', and λ_i'' be the i^{th} eigenvalues of $\mathbf{T}$, $\mathbf{G}_1$, and $\mathbf{G}_2$, repectively. Then, from using the matrix eigenvalue properties [21] and (3.44)-(3.46), the relations between them are given as,

$$\begin{aligned}
\lambda_i' &= 2(\alpha + \mathcal{G}^*\lambda_i), \\
\lambda_i'' &= \alpha(\alpha + 2\mathcal{G}^*\lambda_i).
\end{aligned}$$

Furthermore, $\mathbf{G}_1$, $\mathbf{G}_2$ and $\mathbf{T}$ will have the same eigenvectors.

Using this, we find that the i^{th} eigenvalue of $\mathbf{G}_1^{-1}\mathbf{G}_2$ is given by

$$\begin{aligned}
\lambda_i^{\mathbf{G}_1^{-1}\mathbf{G}_2} &= \frac{\lambda_i''}{\lambda_i'} \\
&= \frac{\alpha(\alpha + 2\mathcal{G}^*\lambda_i)}{2(\alpha + \mathcal{G}^*\lambda_i)} \\
&= \alpha - \frac{\alpha^2}{2(\alpha + \mathcal{G}^*\lambda_i)}.
\end{aligned} \tag{3.49}$$

Furthermore

$$\begin{aligned}
\lambda_{\max}^{\mathbf{G}_1^{-1}\mathbf{G}_2} &= \alpha - \frac{\alpha^2}{2(\alpha + \mathcal{G}^*\lambda_{\max}(\mathbf{R}))} \\
&= \frac{\alpha(\alpha + 2\mathcal{G}^*\lambda_{\max}(\mathbf{R}))}{2(\alpha + \mathcal{G}^*\lambda_{\max}(\mathbf{R}))},
\end{aligned} \tag{3.50}$$

and

$$\frac{1}{\lambda_{\max}^{\mathbf{G}_1^{-1}\mathbf{G}_2}} = \frac{2(\alpha + \mathcal{G}^*\lambda_{\max}(\mathbf{R}))}{\alpha(\alpha + 2\mathcal{G}^*\lambda_{\max}(\mathbf{R}))}. \tag{3.51}$$

Now, by comparing (3.4) and (3.51) and after some algebraic manipulation, we get the following result for the upper bound $\mu_{\max}$ on the step size to ensure mean and mean square stability:

$$\mu_{\max}=\begin{cases} \frac{2}{\alpha+\mathcal{G}_k^*\lambda_{\max}(\mathbf{R})}, & \alpha > \frac{\mathcal{G}_k^*\lambda_{\max}(\mathbf{R})}{4}, \\ \frac{1}{\lambda_{\max}(\mathbf{G}_1^{-1}\mathbf{G}_2)}, & \text{otherwise.} \end{cases} \tag{3.52}$$

CHAPTER 4

STEADY STATE ANALYSIS OF LEAKY LMF

In this chapter, the steady state analysis of the proposed Leaky LMF algorithm is carried out. We will be using the assumptions used in the previous chapter in addition to the following assumption:

A9 The regressors $\mathbf{u}_k$ have covariance matrix $\mathbf{R} = \sigma_u^2\mathbf{I}$.

The reason for using this restrictive assumption is to make the analysis more tractable. For the case of correlated regressors, we end up with a single equation with two variables $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$ which have do not have a linear relation between them. Thus we end up with an under-determined system. However, for white Gaussian regressors, we have an additional equation that relates $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$ given by (3.16).

Therefore, we will use (3.16) and (3.21) in our study of the steady-state behavior of the Leaky LMF. To begin with, we use (3.21) to get

$$
\begin{aligned}
E\left[||\mathbf{v}_{k+1}||^2\right] &= f_1 E\left[||\mathbf{v}_k||^2\right] + f_2 E\left[\mathbf{v}_k\right] + \mu^2\alpha^2||\mathbf{w}_\mathrm{o}||^2 + \mu M\sigma_u^2\xi_n^6, &(4.1)\\
E\left[\mathbf{v}_{k+1}\right] &= \mathbf{H}E\left[\mathbf{v}_k\right] + \mu\alpha\mathbf{w}_\mathrm{o}, &(4.2)
\end{aligned}
$$

where the terms inside the equations are given by (3.17)-(3.20).

Assuming the step size satisfies the mean and mean square convergence conditions, then at steady state (as $k \to \infty$), we have

$$
\begin{aligned}
\lim_{k\to\infty} E\left[||\mathbf{v}_{k+1}||^2\right] &= \lim_{k\to\infty} E\left[||\mathbf{v}_k||^2\right] = E\left[||\mathbf{v}_\infty||^2\right] &(4.3)\\
\lim_{k\to\infty} E\left[\mathbf{v}_{k+1}\right] &= \lim_{k\to\infty} E\left[\mathbf{v}_k\right] = E\left[\mathbf{v}_\infty\right] &(4.4)
\end{aligned}
$$

Then, taking the limit as $k \to \infty$ on both sides of (4.1)-(4.2), we have

$$
\begin{aligned}
E\left[||\mathbf{v}_\infty||^2\right] &= f1_\infty E\left[||\mathbf{v}_\infty||^2\right] + f2_\infty E\left[\mathbf{v}_\infty\right] + \mu^2\alpha^2||\mathbf{w}_\mathrm{o}||^2 + \mu M\sigma_u^2\xi_n^6, &(4.5)\\
E\left[\mathbf{v}_\infty\right] &= \mathbf{H}_\infty E\left[\mathbf{v}_\infty\right] + \mu\alpha\mathbf{w}_\mathrm{o}, &(4.6)
\end{aligned}
$$

where

$$
\begin{aligned}
f_{1\infty} &= (1-\mu\alpha)^2 + \mu^2 15M\sigma_u^4\xi_n^4 - 6\mu(1-\mu\alpha)\sigma_n^2\sigma_u^2 + \mu^2 45M\sigma_u^6\sigma_n^2 E\left[||\mathbf{v}_\infty||^2\right]\\
&\quad -6\mu(1-\mu\alpha)\sigma_u^4 E\left[||\mathbf{v}_\infty||^2\right] + \mu^2 15M\sigma_u^8 E\left[||\mathbf{v}_\infty||^2\right]^2, &(4.7)
\end{aligned}
$$

$$f2_\infty = 2\mu\alpha\mathbf{H}_\infty \mathbf{w}_\mathrm{o}^T, \tag{4.8}$$

$$\mathbf{H}_\infty = 1 - \mu\left\{\alpha + 3(\sigma_n^2 + \sigma_u^2 E\left[||\mathbf{v}_\infty||^2\right])\sigma_u^2\right\}. \tag{4.9}$$

From (4.6) and using (4.9), we get

$$E\left[\mathbf{v}_\infty\right] = \frac{\alpha\mathbf{w}_\mathrm{o}}{\alpha + 3(\sigma_n^2 + \sigma_u^2 E\left[||\mathbf{v}_\infty||^2\right])\sigma_u^2}. \tag{4.10}$$

Let

$$C = \alpha + 3(\sigma_n^2 + \sigma_u^2 E\left[||\mathbf{v}_\infty||^2\right])\sigma_u^2. \tag{4.11}$$

Then using (4.10) in (4.5), we get

$$E\left[||\mathbf{v}_\infty||^2\right] = f_{1\infty} E\left[||\mathbf{v}_\infty||^2\right] + \frac{2\mu\alpha^2(1-C)||\mathbf{w}_\mathrm{o}||^2}{C} + \mu^2\alpha^2||\mathbf{w}_\mathrm{o}||^2 + \mu M\sigma_u^2\xi_n^6. \tag{4.12}$$

Multiplying both sides of (4.12) by C, we get

$$\begin{aligned} CE\left[||\mathbf{v}_\infty||^2\right] &= Cf_{1\infty}E\left[||\mathbf{v}_\infty||^2\right] + 2\mu\alpha^2(1-C)||\mathbf{w}_\mathrm{o}||^2 + C\mu^2\alpha^2||\mathbf{w}_\mathrm{o}||^2 \\ &\quad + \mu MC\sigma_u^2\xi_n^6. \end{aligned} \tag{4.13}$$

Opening this expression and grouping together coefficients of different powers of $E\left[||\mathbf{v}_\infty||^2\right]$ together, then after some algebra, we get the following quartic polynomial in $E\left[||\mathbf{v}_\infty||^2\right]$:

$$\sum_{j=0}^{4} \beta_j (E\left[||\mathbf{v}_\infty||^2\right])^j = 0, \tag{4.14}$$

where

$$\begin{aligned}
\beta_0 &= \left(2\mu\alpha^2 - \mu^2\alpha^3 - 3\sigma_n^2\sigma_u^2\mu^2\alpha^2\right)||\mathbf{w}_\mathrm{o}||^2, && (4.15)\\
\beta_1 &= \mu^2 3M\sigma_u^4(\alpha 5\xi_n^4 + 15M\sigma_n^2\sigma u^2\xi_n^4 + \sigma_u^2\xi_n^6) \\
&\quad -6\mu\sigma_n^2\sigma u^2\left(\alpha(2-\mu\alpha) - 3(1-\mu\alpha)\sigma_n^2\sigma_u^2\right) \\
&\quad -2\mu\alpha^2 - \mu^2\alpha^3 + 3\mu^2\alpha^2\sigma_u^2(\sigma u^2||\mathbf{w}_\mathrm{o}||^2 + \sigma_n^2), && (4.16)\\
\beta_2 &= \mu^2 45M\sigma u^6(\alpha\sigma_n^2 + 3\sigma_n^4\sigma u^2 + \xi_n^4\sigma u^2) \\
&\quad -6\mu(1-\mu\alpha)\sigma u^4(\alpha + 6\sigma_n^2\sigma u^2) - 3\sigma u^4, && (4.17)\\
\beta_3 &= 3\mu\sigma u^8(\mu\alpha 5M + \mu 60M\sigma_n^2\sigma u^2 - 6(1-\mu\alpha)), && (4.18)\\
\beta_4 &= \mu^2 45M\sigma u^{12}. && (4.19)
\end{aligned}$$

Since $E\left[||\mathbf{v}_\infty||^2\right]$ will be very small, we can assume $\left(E\left[||\mathbf{v}_\infty||^2\right]\right)^4$ to be negligible and the problem of finding $E\left[||\mathbf{v}_\infty||^2\right]$ is now solved by finding the roots of the following polynomial equation:

$$\sum_{j=0}^{3} \chi_j (E\left[||\mathbf{v}_\infty||^2\right])^j = 0, \tag{4.20}$$

where

$$\chi_j = \frac{\beta_j}{\beta_4}. \tag{4.21}$$

Equation (4.20) has three roots [21]. From simulations, we found that the smallest positive square root of the polynomial gives $E\left[||\mathbf{v}_{\infty}||^{2}\right]$.

CHAPTER 5

TRACKING ANALYSIS OF LEAKY LMF

The aim of tracking analysis of an adaptive filter is to provide a quantitative measure of how well the adaptive algorithm is able to track variations in the signal statistics. In this chapter, the tracking analysis of the proposed algorithm is carried out. Both the random walk model and the Rayleigh fading model (single path and multipath) to model the time varying channels and the analysis is carried out in the same way as was done for the steady state analysis.

5.1 Random Walk Model

The first order random-walk model for a channel is given as

$$\mathbf{c}_{k+1} = \mathbf{c}_k + \mathbf{q}_k, \tag{5.1}$$

where $\mathbf{c}_k$ is the time-varying wide-sense stationary unknown system that is to be tracked and $\mathbf{q}_k$ is assumed to be a zero-mean stationary random vector process with a positive-definite covariance matrix $\mathbf{Q}$. It is also statistically independent of all other parameters of the adaptive filter. The noisy measurement that arises from the random walk model is given by

$$d_k = \mathbf{u}_k \mathbf{c}_k + n_k. \tag{5.2}$$

It can be seen from the assumptions used for $\mathbf{q}_k$ and (5.1) that

$$E\left[\mathbf{c}_{k+1}\right] = E\left[\mathbf{c}_k\right] = \mathbf{c}. \tag{5.3}$$

Now it was observed in [2] that the covariance matrix of $\mathbf{c}_{k+1}$ $\mathbf{C}_{k+1}$ is given by

$$\begin{aligned}
\mathbf{C}_{k+1} &= E\left[\left(\mathbf{c}_{k+1} - \mathbf{c}\right)\left(\mathbf{c}_{k+1} - \mathbf{c}\right)^T\right] && (5.4)\\
&= E\left[\left(\mathbf{c}_k + \mathbf{q}_k - \mathbf{c}\right)\left(\mathbf{c}_k + \mathbf{q}_k - \mathbf{c}\right)^T\right] && (5.5)\\
&= E\left[\left(\mathbf{c}_k - \mathbf{c}\right)\left(\mathbf{c}_k - \mathbf{c}\right)^T\right] + \mathbf{Q} && (5.6)\\
&= \mathbf{C}_{k+1} + \mathbf{Q}. && (5.7)
\end{aligned}$$

We see that a positive-definite matrix is added to the covariance matrix of the the unknown system vector at each iteration and thus grows unbounded. A more practical model that can be used is by replacing (5.1) by

$$\mathbf{c}_{k+1} - \mathbf{c} = \varrho(\mathbf{c}_k - \mathbf{c}) + \mathbf{q}_k, \tag{5.8}$$

for some scalar $|\varrho| < 1$. In this case, the covariance matrix of $\mathbf{c}_{k+1}$ would tend to a finite steady-state value given by

$$\lim_{k\to\infty} \mathbf{C}_{k+1} = \frac{\mathbf{Q}}{1 - |\varrho|^2}. \tag{5.9}$$

However, the tracking analysis of this model is more demanding. As mentioned in [2], it was found that in the literature it is a convention to assume the value of ϱ to be sufficiently close to 1 to warrant the use of model (5.1) which simplifies our analysis greatly.For this reason , we have used the model model (5.1) for tracking analysis of the Leaky LMF.

5.2 Tracking Analysis of Leaky LMF for Random Walk Model

To begin with, we shall rewrite the Leaky LMF update equation , taking the non-stationarity of the channel into account, we get the following recursion:

$$\mathbf{w}_{k+1} = (1 - \mu\alpha)\mathbf{w}_k + \mu\mathbf{u}_k^T e_k^3. \tag{5.10}$$

Let $\mathbf{v}_j = \mathbf{c}_j - \mathbf{w}_j$, then

$$\begin{aligned}
\mathbf{c}_{k+1} - \mathbf{w}_{k+1} &= \mathbf{c}_{k+1} - (1-\mu\alpha)\mathbf{w}_k - \mu\mathbf{u}_k^T e_k^3 \\
&= \mathbf{c}_{k+1} - \mathbf{w}_k + \mu\alpha\mathbf{w}_k - \mu\mathbf{u}_k^T e_k^3 \\
&= \mathbf{c}_{k+1} - \mathbf{w}_k + \mu\alpha\left(\mathbf{c}_k - \mathbf{v}_k\right) - \mu\mathbf{u}_k^T e_k^3 \\
&= \mathbf{c}_k + \mathbf{q}_k - \mathbf{w}_k + \mu\alpha\left(\mathbf{c}_k - \mathbf{v}_k\right) - \mu\mathbf{u}_k^T e_k^3 \\
&= \mathbf{v}_k + \mathbf{q}_n + \mu\alpha\mathbf{c}_k - \mu\alpha\mathbf{v}_k - \mu\mathbf{u}_k^T e_k^3 \\
\mathbf{v}_{k+1} &= (1-\mu\alpha)\mathbf{v}_k + \mu\alpha\mathbf{c}_k - \mu\mathbf{u}_k^T e_k^3 + \mathbf{q}_k. \qquad (5.11)
\end{aligned}$$

Taking the weighted norms of both sides of (5.11), with $\mathbf{A}$ being the symmetric weighting matrix, and using **A1-A9** along with the assumptions on the statistics of $\mathbf{q}_k$, we get

$$\begin{aligned}
E\left[||\mathbf{v}_{k+1}||^2_{\mathbf{A}}\right] &= (1-\mu\alpha)^2 E\left[||\mathbf{v}_k||^2_{\mathbf{A}}\right] + ||\mu\alpha\mathbf{c}||^{\mathbf{2}}_{\mathbf{A}} + \mu^2 tr(\mathbf{RA})\mathcal{Z}_k \\
&\quad -2\mu(1-\mu\alpha)\mathcal{G}_k E\left[||\mathbf{v}_k||^2_{\mathbf{RA}}\right] + 2\mu\alpha\mathbf{c}^T\mathbf{A}JE\left[\mathbf{v}_k\right] \\
&\quad + tr\left(\mathbf{QA}\right). \qquad (5.12)
\end{aligned}$$

We see that the only difference between (3.15) and (5.12) is the addition term $tr\left(\mathbf{QA}\right)$. Using this fact, we can approach the problem of tracking analysis of the Leaky LMF in the same way as was done for the steady state analysis for white gaussian data.

Furthermore, after applying the same steps and assumptions done for transient analysis of stationary environment to non-stationary environment expressed by the random walk model, we get the following state space equation representing the evolution of $E[||\mathbf{v}_k||^2]$ and $E[e_{ak}^2]$ in a random walk model:

$$\underbrace{\begin{bmatrix} \mathbf{A}_{k+1} \\ E\left[\mathbf{v}_{k+1}\right] \end{bmatrix}}_{\mathbf{W}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 \\ \mathbf{0} & \mathbf{H} \end{bmatrix}}_{\mathbf{F_k}} \underbrace{\begin{bmatrix} \mathbf{A}_k \\ E\left[\mathbf{v}_k\right] \end{bmatrix}}_{\mathbf{W}_k} + \underbrace{\begin{bmatrix} \mathbf{M}_k \\ \mu\alpha\mathbf{c} \end{bmatrix}}_{\mathbf{Y}_k} \tag{5.13}$$

where the only difference between (3.31) and (5.13) is the term $\mathbf{M}_k$ given by

$$\mathbf{M}_k = \mu^2 \mathcal{Z}_k \begin{bmatrix} tr(\mathbf{R}) \\ tr(\mathbf{R}^2) \\ tr(\mathbf{R}^3) \\ \vdots \\ tr(\mathbf{R}^M) \end{bmatrix} + \mu^2\alpha^2 \begin{bmatrix} ||\mathbf{c}||^2 \\ ||\mathbf{c}||^2_{\mathbf{R}} \\ ||\mathbf{c}||^2_{\mathbf{R}^2} \\ \vdots \\ ||\mathbf{c}||^2_{\mathbf{R}^{M-1}} \end{bmatrix} + \begin{bmatrix} tr(\mathbf{Q}) \\ tr(\mathbf{QR}) \\ tr(\mathbf{QR}^2) \\ \vdots \\ tr(\mathbf{QR}^{M-1}) \end{bmatrix} \tag{5.14}$$

5.3 Rayleigh Fading Channel Model

In a wireless communications environment, the transmitted signal suffers from multipath reflections while traveling from the transmitter to the receiver so that the receiver gets several replicas of the transmitted signal with different amplitude and phase distortions at different delays so that the overall receiver signal is the sum of all the reflections. Based on the relative phases of the reflections, the signal may add constructively or destructively at the receiver. Furthermore, if the receiver is moving with respect to the transmitter, then these interferences will vary with time, This phenomenon is described in [2] [3] as channel fading.

The impulse response of a single tap (i.e single path) fading channel can be described as

$$h(n) = \psi x(n)\delta(n - n_o), \tag{5.15}$$

where $\{x(n)\}$ is a time-variant unit variance complex sequence that models the channel variations in the channel, and n_o is the channel delay. ψ^2 is the power attenuation that a signal will undergo when it passes through the channel. Although there are several models to describe the fading characteristics of $\{x(n)\}$ the most widely used is the Rayleigh fading model. In this model, for each time instant n, the amplitude $|\{x(n)\}|$ has a Rayleigh distribution given by

$$f_{|x(n)|}(|x(n)|) = |x(n)|\, e^{\frac{-|x(n)|^2}{2}}, \tag{5.16}$$

while the phase $\angle x(n)$ is assumed to be uniformly distributed within $[-\pi, \pi]$

$$f_{\angle x(n)}(\angle x(n)) = \frac{1}{2\pi}, -\pi \leq \angle x(n) \leq \pi. \tag{5.17}$$

Zeroth order Bessel function has been used extensively in the literature to model the autocorrelation function of $\{x(n)\}$. This model is based on the assumption that all the scatterers are uniformly distributed around the receiver, so that its power spectral density has a U-shaped function. This function is expressed as

$$r(k) \cong E[x(n)x(n-k)] = J_o(2\pi f_D T_s k), n = \cdots, -1, 0, 1, \cdots \tag{5.18}$$

where T_s is the sampling period, f_D is the maximum Doppler frequency of the Rayleigh fading channel and J_o is the zeroth order Bessel function defined by

$$J_o(y) = \frac{1}{\pi}\int_0^{\pi} \cos(y \sin\theta)d\theta. \tag{5.19}$$

The Doppler frequency is related to the speed of the mobile user v and to the carrier frequency f_c as follows

$$f_D = \pm\frac{vf_c}{c} = \pm\frac{v}{\lambda_{sig}}. \tag{5.20}$$

where c is the speed of light and λ_{sig} is the wavelength of the signal.

Therefore, the weight vector we wish to estimate has the form

$$\begin{bmatrix} 0 & 0 & x(n) & 0 & 0 \end{bmatrix} \tag{5.21}$$

When we investigate further into the fading phenomenon, we find in the case where the reflections originate from far off objects like mountains and buildings, then the signal replicas corresponding to these reflections arrive at a much larger delay as compared to the first group of reflections in which case, a single path Rayleigh fading channel is not sufficient and we can use a finite impulse response model to simulate the channel which can be expressed as

$$h(n) = \sum_{k=1}^{L} \psi_k x_k(n) \delta(n - k + 1), \tag{5.22}$$

where ψ_k and x_k are, respectively, the path loss and fading sequence of the k^{th} cluster of reflectors. In this analysis, a two-path Rayleigh fading channel has been assumed where the signals along both paths are assumed to fade independently with same Doppler frequency. Although this assumption is unrealistic, it does allow for us to express Rayleigh fading using a random walk model.The channel impulse response is assumed to consist of an initial delay of 2 samples followed by a Rayleigh fading path and the signal arriving on the second path one sampling delay after the first one such that the channel vector that we wish to estimate has the form of a 5-tap FIR filter with coefficients expressed in vector form as

$$\mathbf{c}_n = \begin{bmatrix} 0 & 0 & x_2(n) & 0 & x_4(n) \end{bmatrix} \tag{5.23}$$

As mentioned in [2], a first order approximation for the variation of the Rayleigh fading coefficient $x(n)$ is to assume that $x(n)$ varies according to the AR model given by

$$x(n) = r(1)x(n-1) + \sqrt{1 - |r(1)|^2}\,\eta(n), \tag{5.24}$$

where $r(1) = J_o(2\pi f_D T_s)$ and $\eta(n)$ denotes the white noise process with unit variance.

The above approximation indicates that the fluctuations in the channel weight vector could be approximated as

$$\mathbf{c}_{k+1} = \tau \mathbf{c}_k + \mathbf{q}_k, \tag{5.25}$$

where the covariance matrix of $\{\mathbf{q}_n\}$ is $\mathbf{Q} = (1 - \tau^2)\mathbf{I}$ where $\tau = r(1)$. It is clear from (5.18) that the value of τ depends on f_D and if τ is chosen to be approximately equal to 1, then the results of the analysis that we have done for the random walk model can be applied for the Rayleigh fading channel estimation problem as well.

CHAPTER 6

PERFORMANCE ANALYSIS OF THE PROPOSED LEAKY LMF ALGORITHM

In this chapter, the results of the computer simulations to investigate the performance behavior of the Leaky LMF are presented. A number of simulation results are carried out to corroborate the theoretical findings.

First, we will show how the Leaky LMF mitigates the weight drift problem that is encountered in the conventional LMF algorithm. After that, we will show how the Leaky LMF provides better performance in terms of the mean-square deviation as compared to the Leaky LMS algorithm for different noise environments. After this, we will present a number of simulations which show that there is a good match between the theoretical findings of the Leaky LMF and the simulation results. These simulations can be divided into the following two categories:

1. Comparison of the transient performance of the Leaky LMF and the simulation results for Gaussian, Uniform and Laplacian noise environments at noise variance of 0.1, 0.01 and 0.001.

2. Comparison of the tracking performance of the Leaky LMF and the simulation results for Gaussian and Uniform noise environments at noise variance of 0.1, 0.01 and 0.001.

6.1 Comparison of LMF and Leaky LMF in Weight Drift Environment

In this section, we will present the simulation to show how weight drift problem occurs in the LMF algorithm and how it can be prevented from happening using the Leaky LMF. In this simulation, the parameters have been chosen to speed up the weight drift phenomenon as was done in [14]. The true weight error vector is given by $[0.7071\ -0.7071]^T$ while the input regressor vector is randomly assigned values of $\pm[0.5\ -0.5]$ with equal probability so that the input covariance matrix is singular. The output noise and the quantization noise are grouped together and modeled as a Gaussian random vector with mean $[0.49\ -0.49]^T$ whose elements are independent of each other and have a variance of 10^{-3} . The number of quantization bits for the adaptive filter coefficients and the regressor values are set to 10. The step size was taken to be 0.0156 and the product of the step size and the leakage factor was set at 0.002. We make a single run over 10^4 samples and have taken the infinite norms of the updated weight vectors in case of both the LMF and the Leaky LMF.

As can be seen from Fig. 6.1, we see that in the case of LMF, the parameter drift causes the adaptive filter weights to blow up while in the case of the Leaky LMF, the adaptive filter weights are bounded.

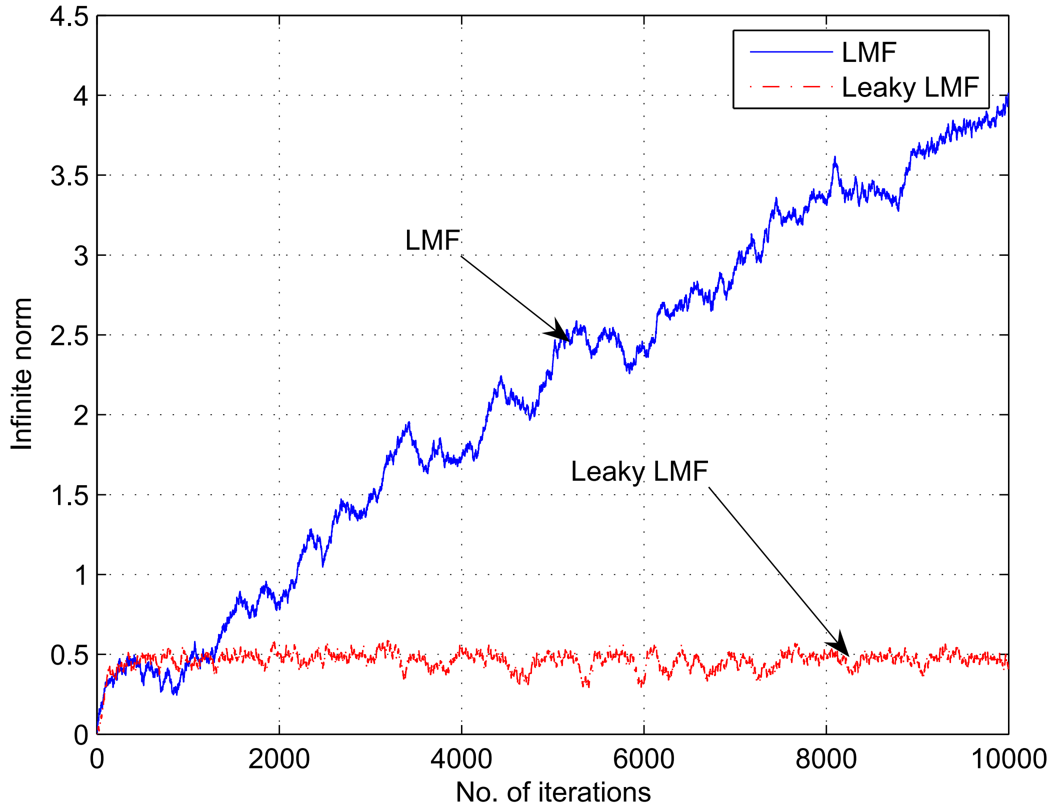

Figure 6.1: Weight drift situation.

6.2 Comparison of the Leaky LMF to the Leaky LMS

We shall now compare the Leaky LMF to the Leaky LMS algorithm and show that for the same step size, the Leaky LMF outperforms the Leaky LMS in the mean square deviation (MSD) sense. The true weight vector was chosen to be

$$\begin{bmatrix} 0.227 & 0.460 & 0.688 & 0.460 & 0.227 \end{bmatrix}^T$$

The step size values for the Leaky LMS and the Leaky LMF were set at 0.01 and 0.09, respectively, while the leakage factor for both algorithms was set at 10^{-5}. A white Gaussian input process with zero mean and unit variance was fed into both the Leaky LMF and the Leaky LMS algorithms while the output noise was set as a zero mean random process with variance 0.001. The experiment was conducted for Gaussian, Laplacian and Uniformly distributed noisy environments and the results were averaged over 20 trials while number of samples used was set at 3×10^4. Fig. 6.2, 6.3 and 6.4 shows the result of the simulations in Gaussian, Uniform and Laplacian noise environments, respectively.

We can see from the resulting simulations that the Leaky LMF performance better in the MSD sense even with a larger step-size.

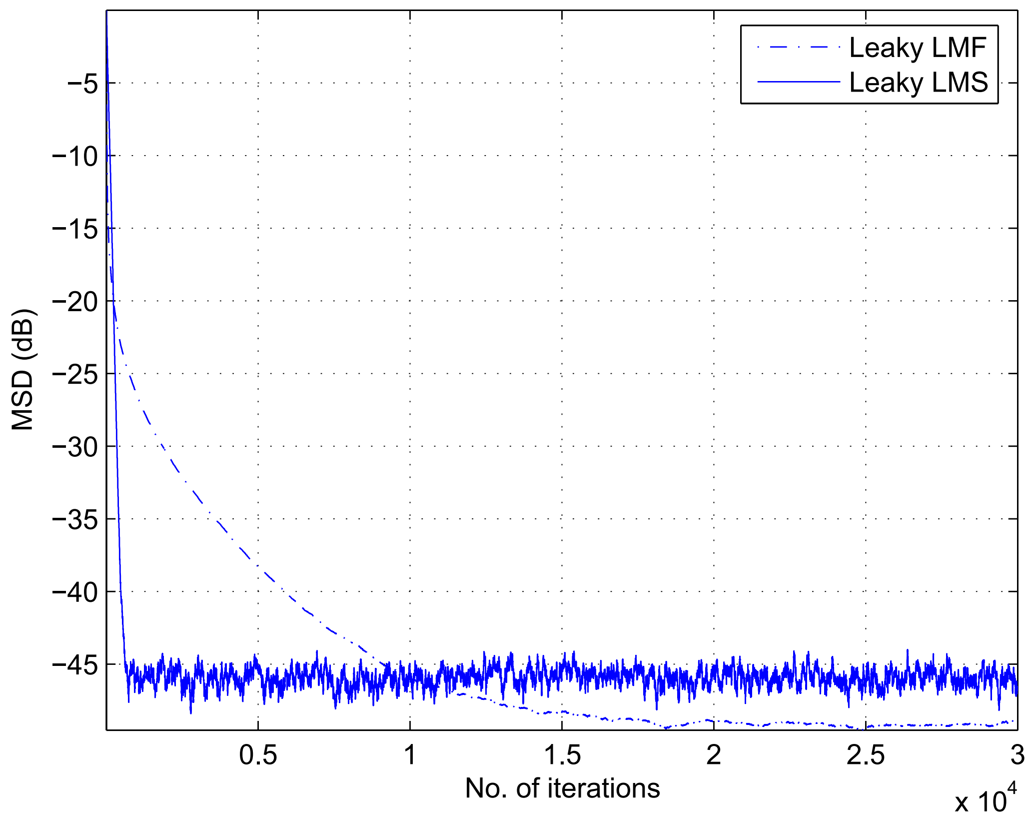

Figure 6.2: Performance of leaky LMF vs. leaky LMS in Gaussian noise.

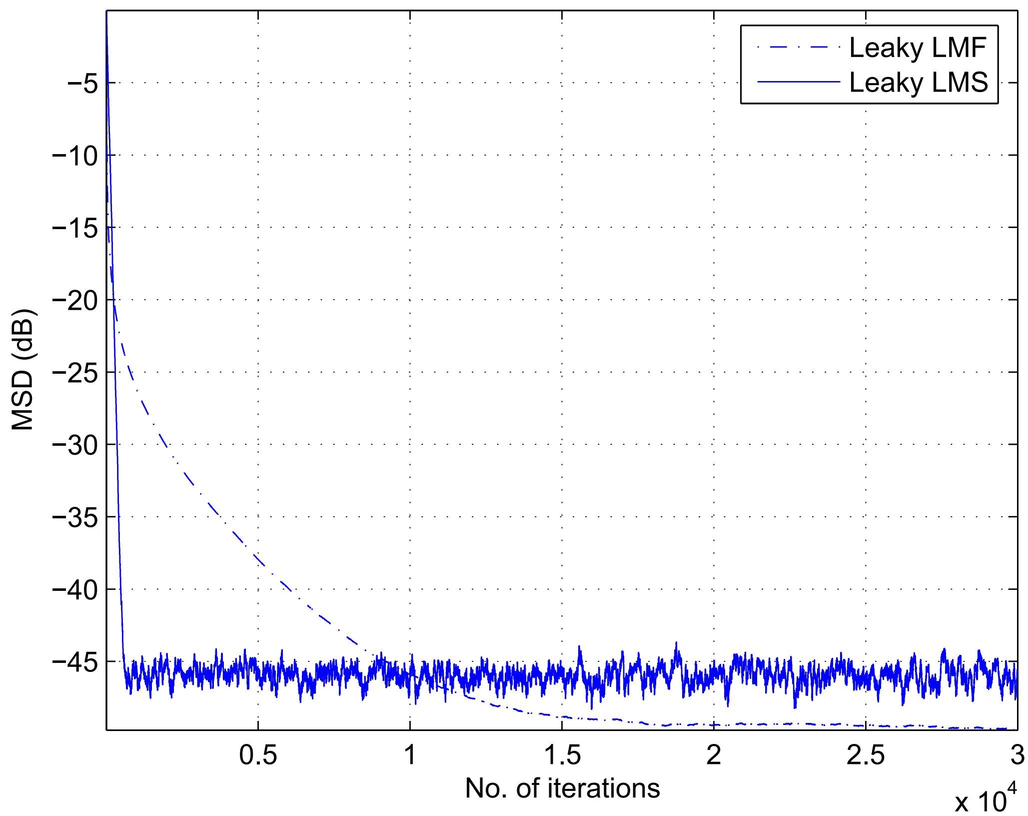

Figure 6.3: Performance of leaky LMF vs. leaky LMS in uniform noise.

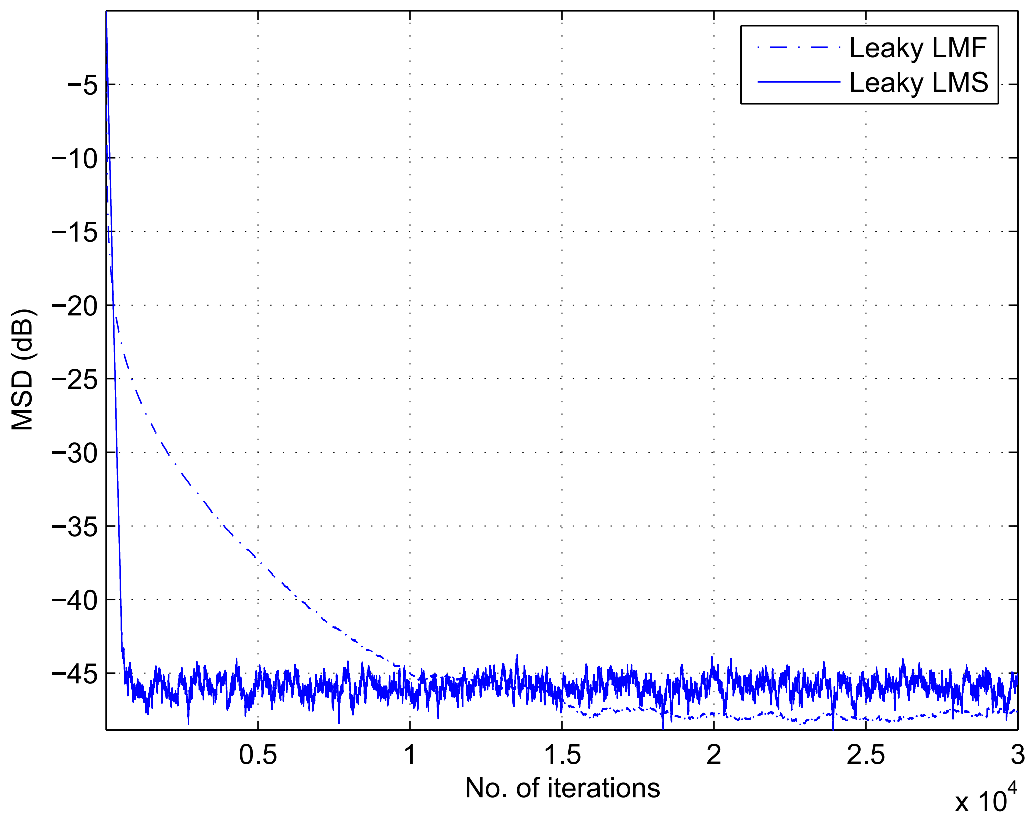

Figure 6.4: Performance of leaky LMF vs. leaky LMS for laplacian noise.

6.3 Comparison of the Theoretical and Simulation Results For Transient Analysis

In this section, we will try to see if the theoretical findings pertaining to the transient analysis of the Leaky LMF agree with the simulation results. A random normalized system weight vector was generated with the number of taps set at 5.

For white input data, with variance of the regressors was set to unity. The step size and the leakage factor were set at 0.01 and 0.001, respectively, while the number of trials and the number of samples used in the experiment were set to 500 and 10^4, respectively.For correlated input data, the eigenvalue spread of the regressor covariance matrix was set to 5. All other parameters are the same as for white data. The simulations were performed for uniform and Gaussian noise environments with the noise variance values set at 0.1, 0.01 and 0.001. The theoretical curves were generated by using (3.31).As we can see from the Fig. 6.5-6.28, there is a very good match between theory and simulation results.

We can see that the rate of convergence is must more in a given noise environment i.e. type of noise and variance value, for white data as compared to correlated data. The reason for this is that the increase in the eigenspread value of $\mathbf{R}$ decreases the speed of convergence [2].

We also note that for the same nature of input data i.e. correlated or white, and noise variance, the MSE performance of the Leaky LMF is much better in uniform noise than gaussian noise. This is to be expected as the conventional LMF also performs better in non-gaussian noise scenarios [4],[5].

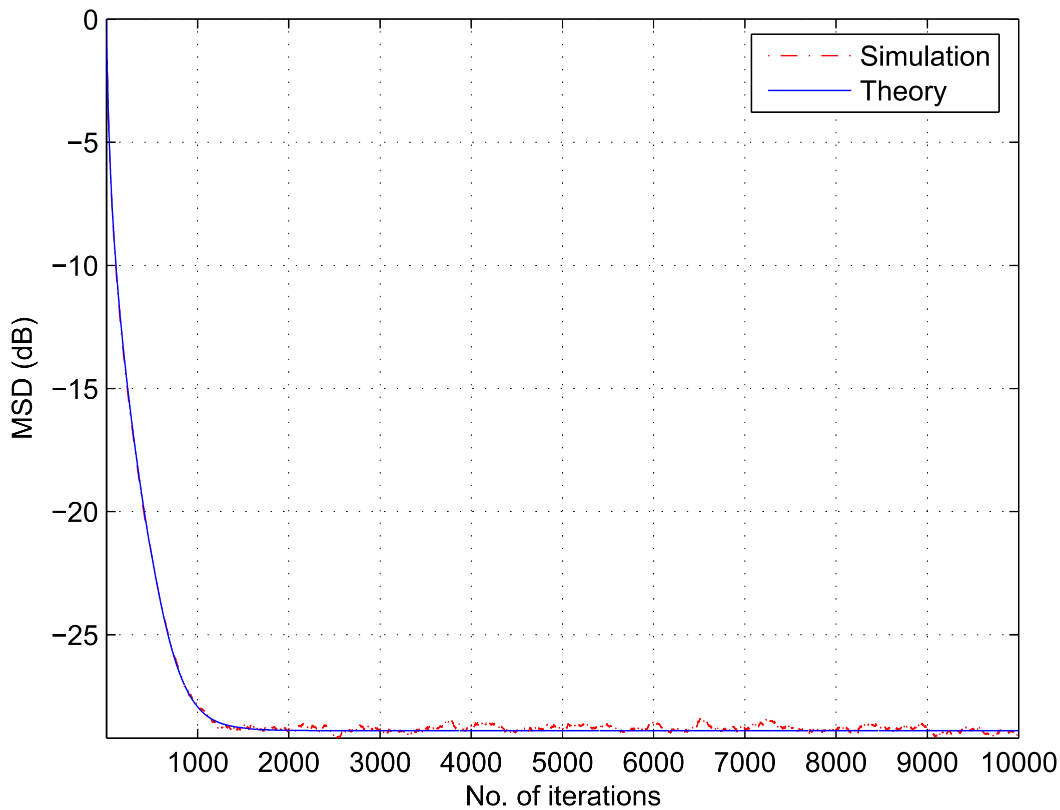

Figure 6.5: Leaky LMF MSD in Gaussian noise with white data and noise variance 0.1.

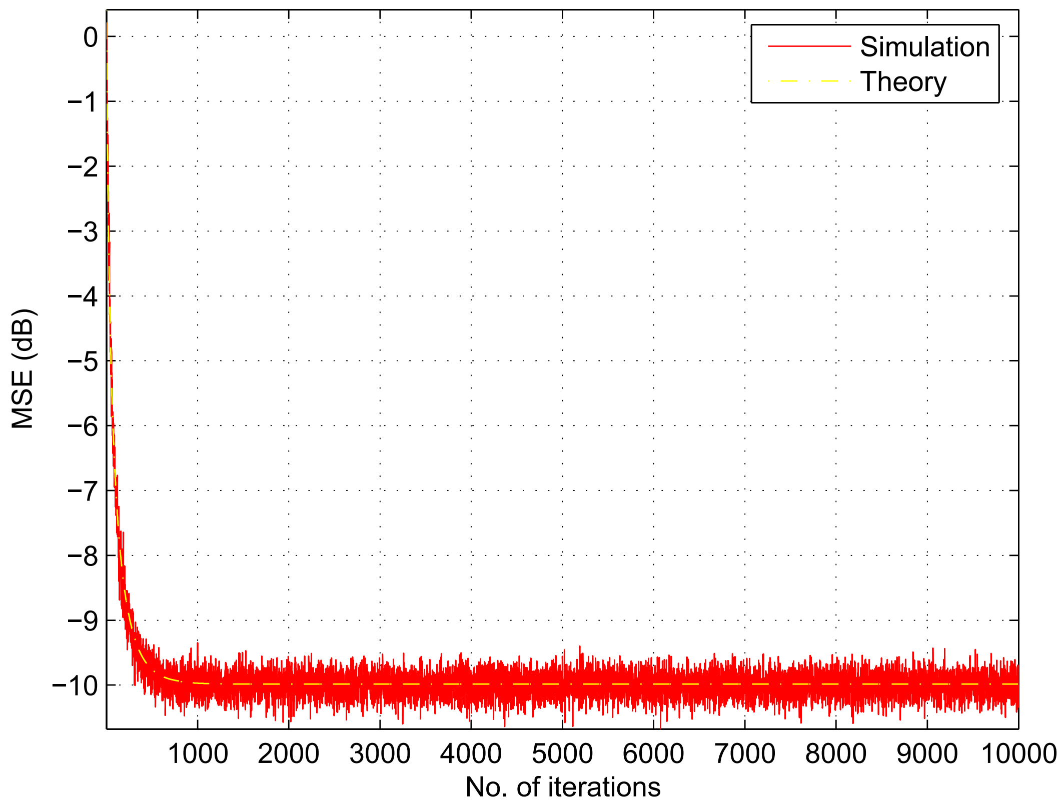

Figure 6.6: Leaky LMF MSE in Gaussian noise with white data and noise variance 0.1.

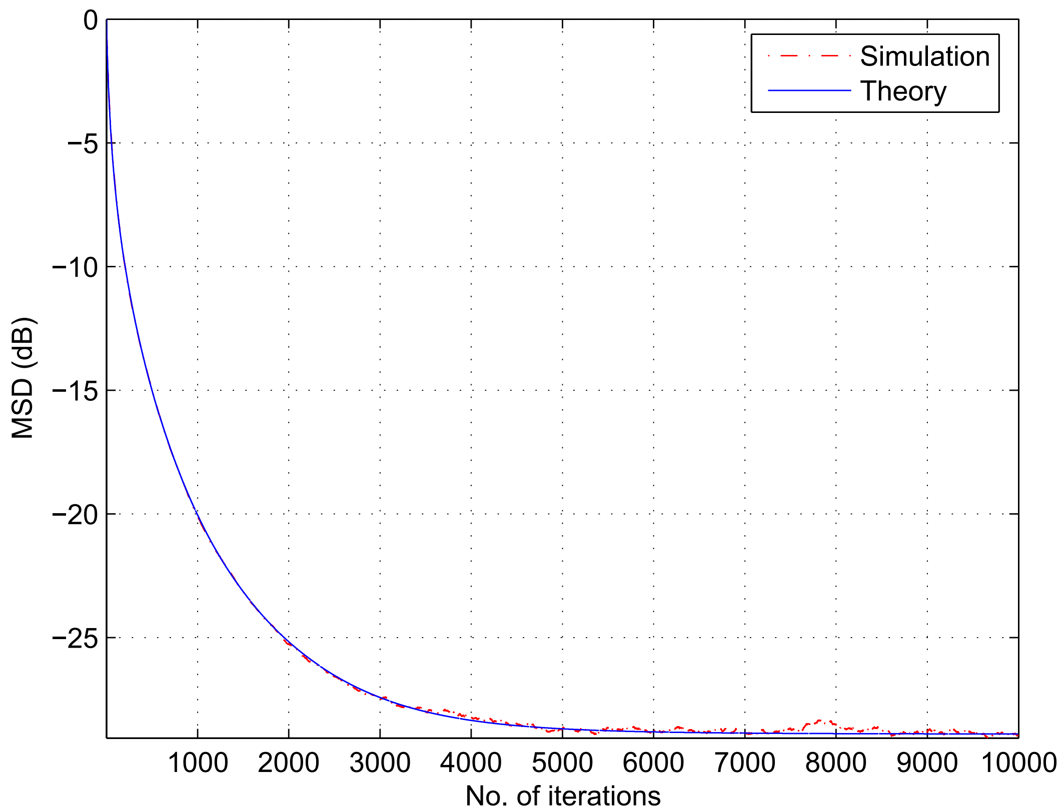

Figure 6.7: Leaky LMF MSD in Gaussian noise with correlated Data and noise Variance 0.1.

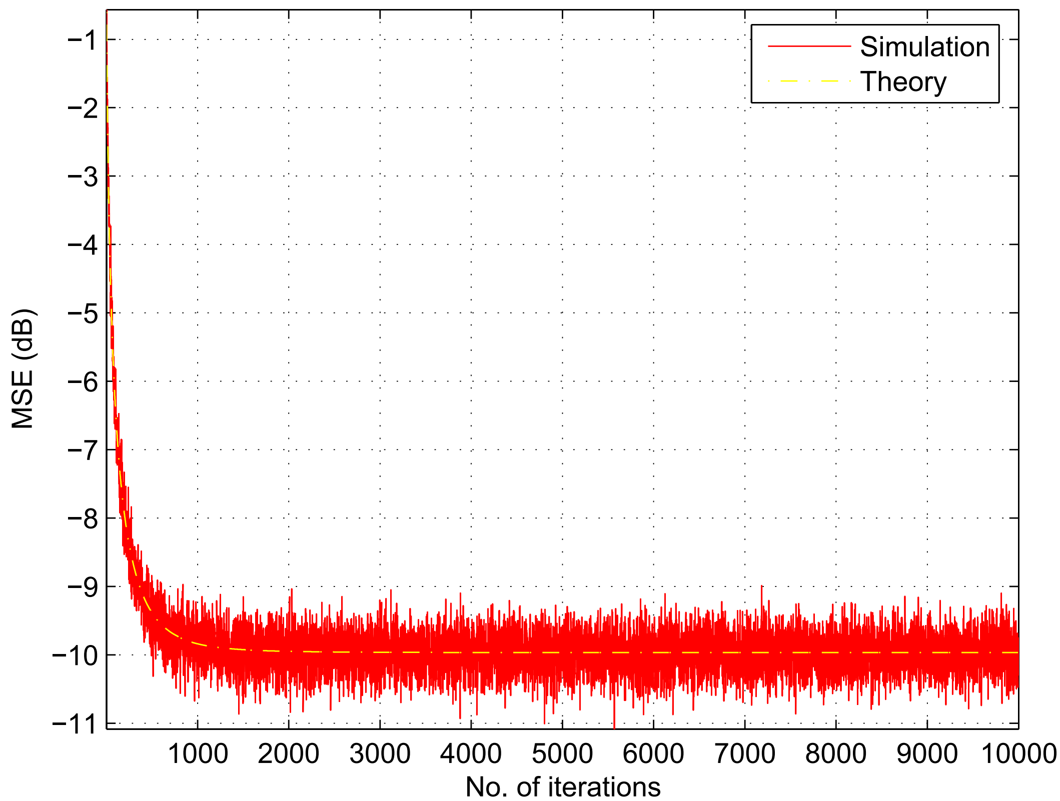

Figure 6.8: Leaky LMF MSE in Gaussian noise with correlated data and noise variance 0.1.

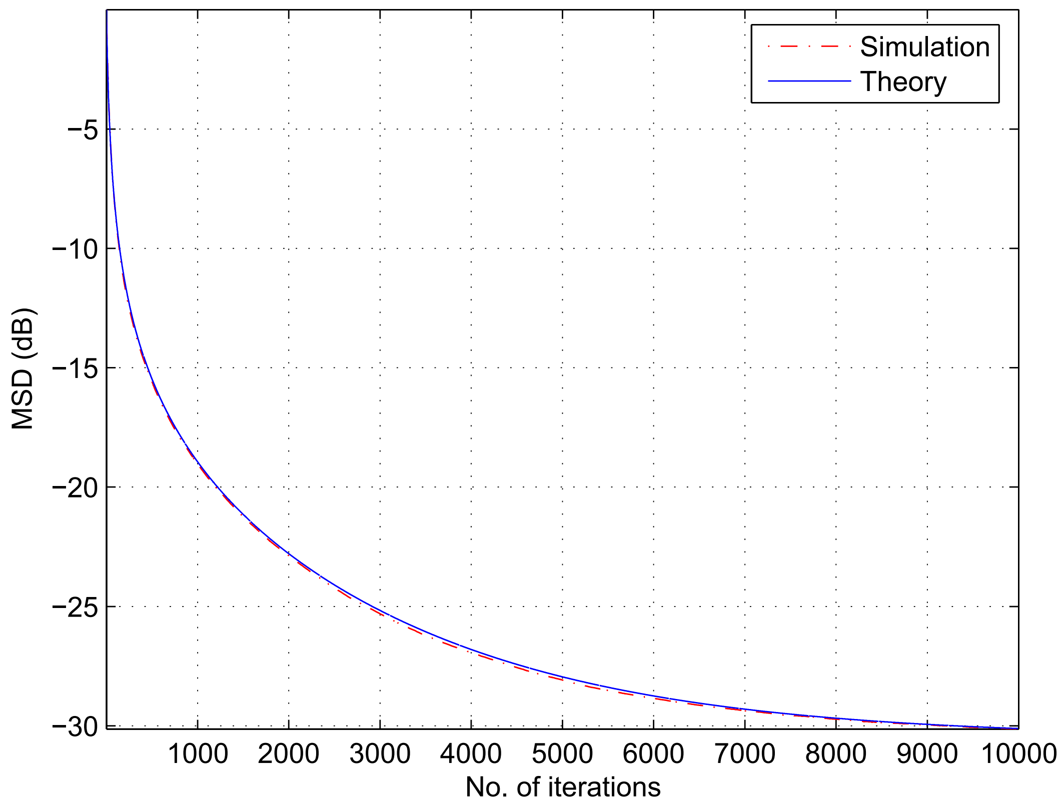

Figure 6.9: Leaky LMF MSD in Gaussian Noise with white data and noise variance 0.01.

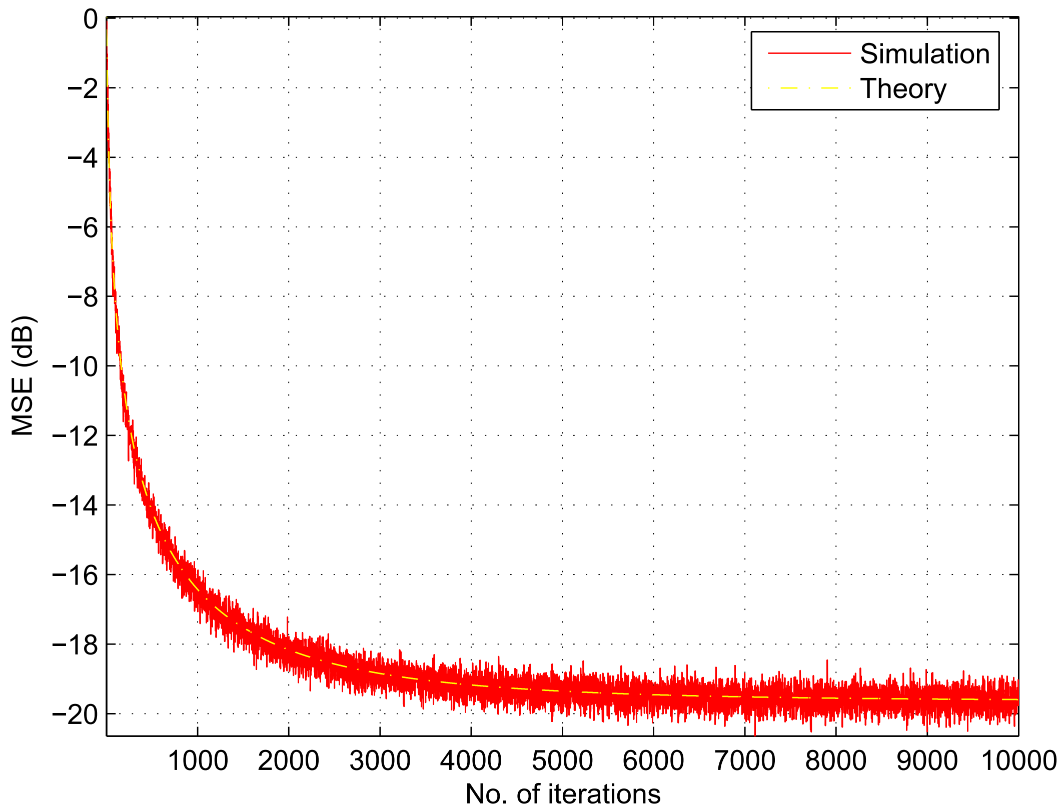

Figure 6.10: Leaky LMF MSE in Gaussian noise with white data and noise variance 0.01.

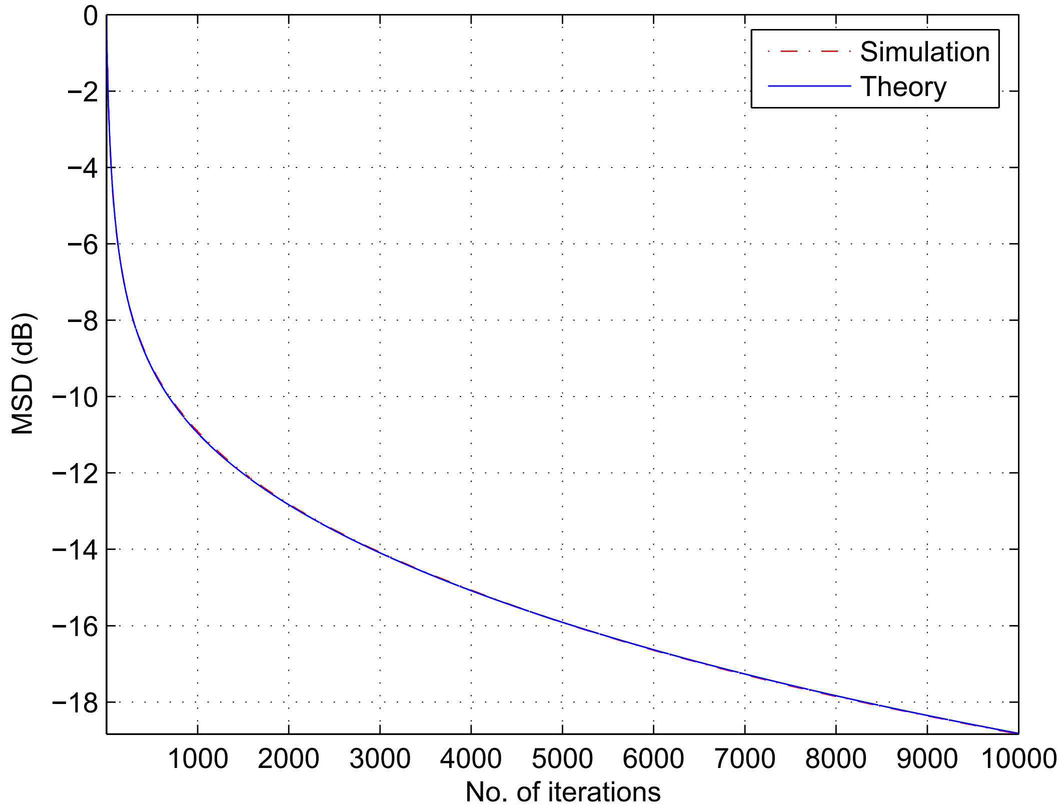

Figure 6.11: Leaky LMF MSD in Gaussian noise with correlated data and noise variance 0.01.

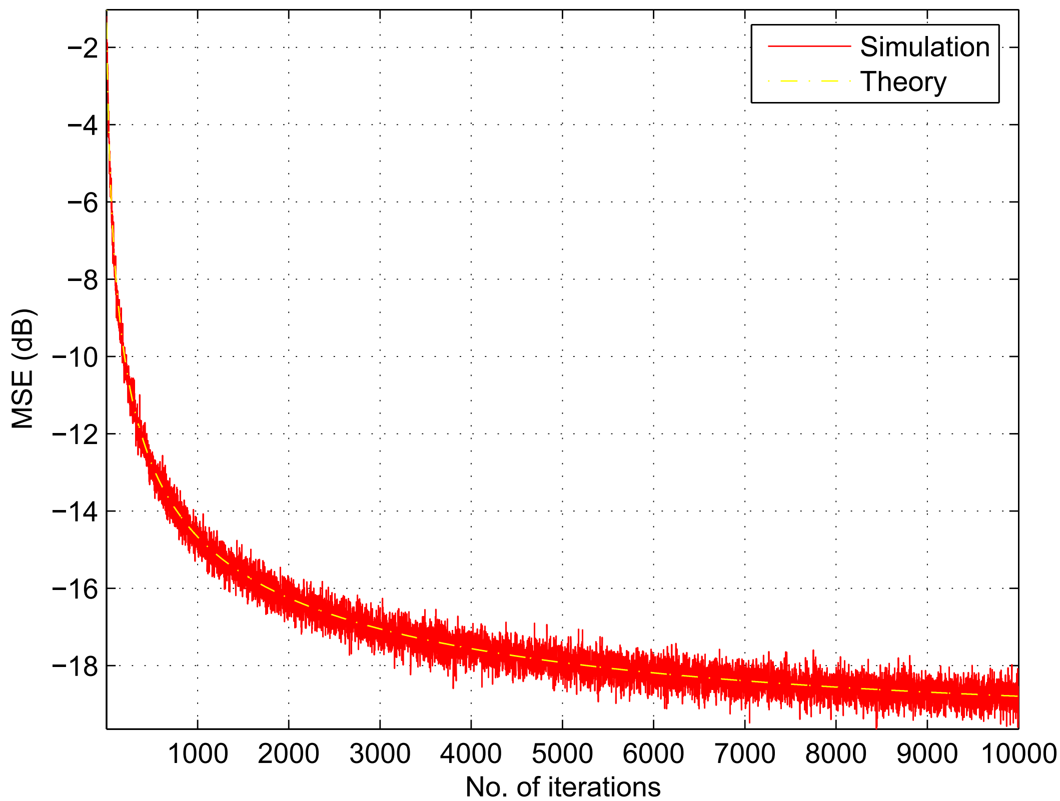

Figure 6.12: Leaky LMF MSE in Gaussian noise with correlated data and noise variance 0.01.

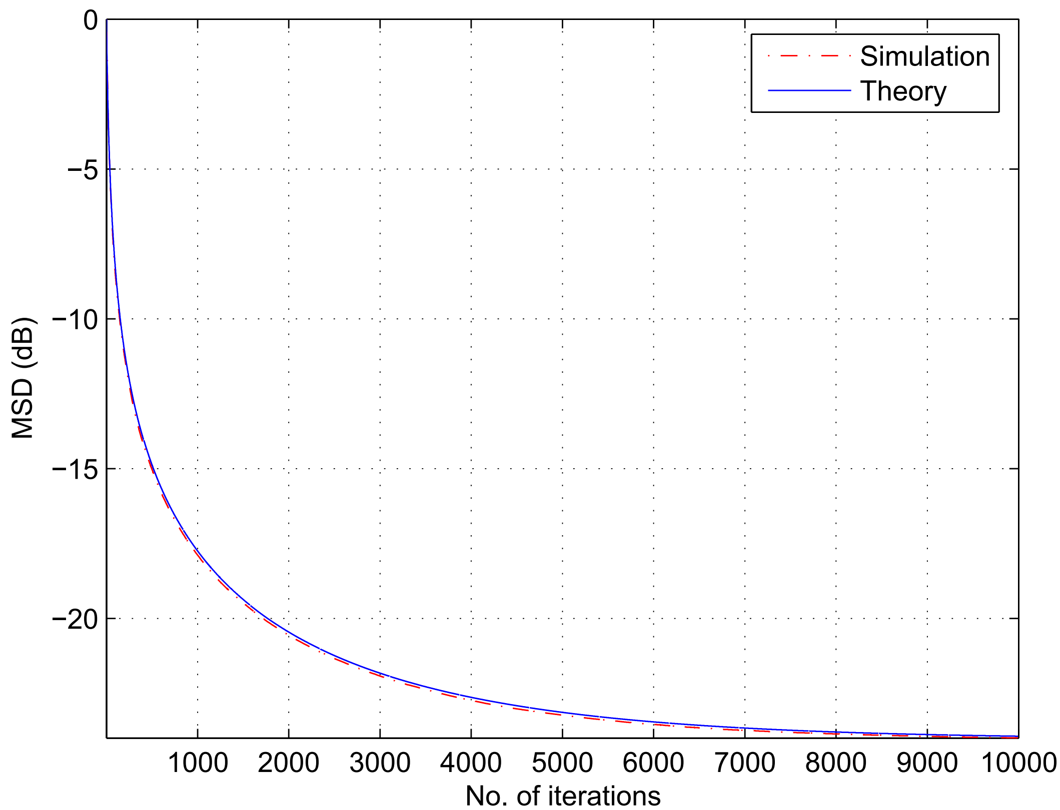

Figure 6.13: Leaky LMF MSD in Gaussian noise with white data and noise variance 0.01.

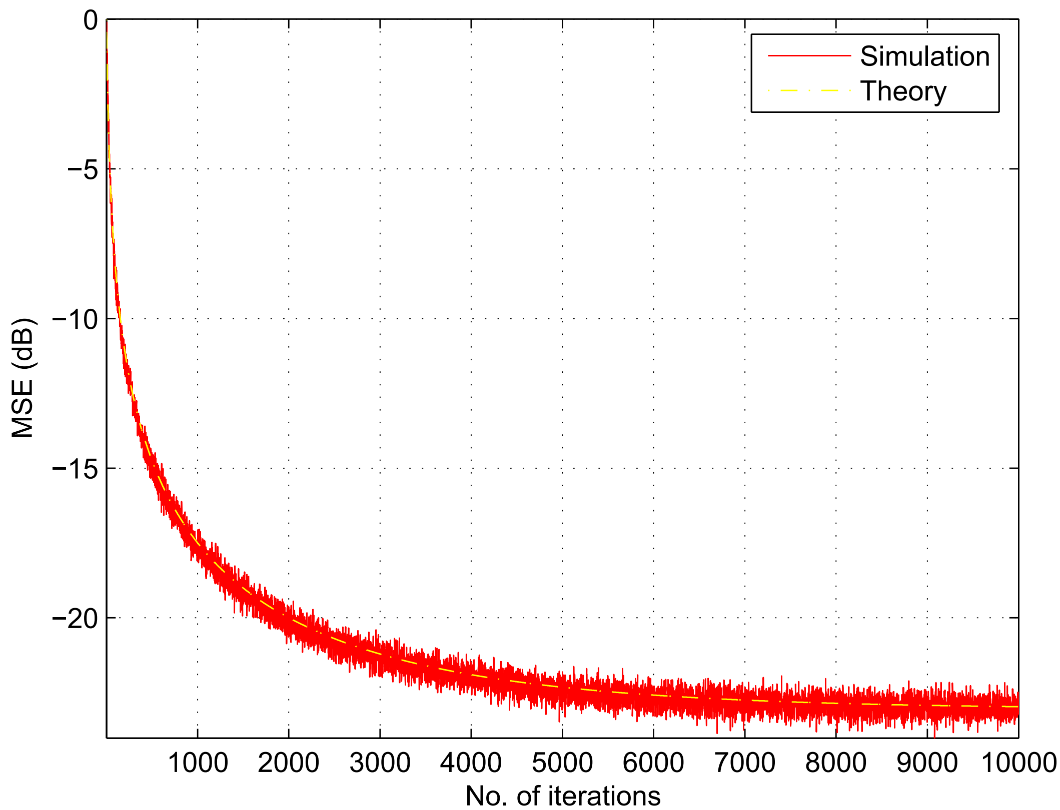

Figure 6.14: Leaky LMF MSE in Gaussian noise with white data and noise variance 0.01.

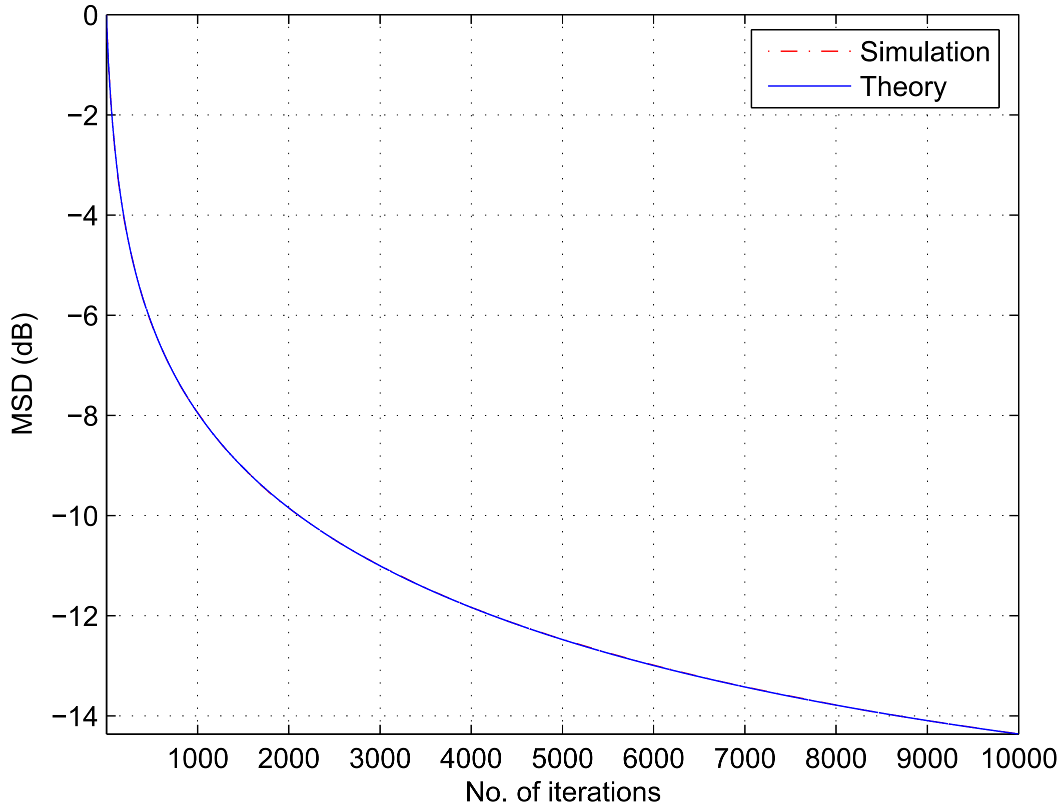

Figure 6.15: Leaky LMF MSD in Gaussian noise with correlated data and noise variance 0.01.

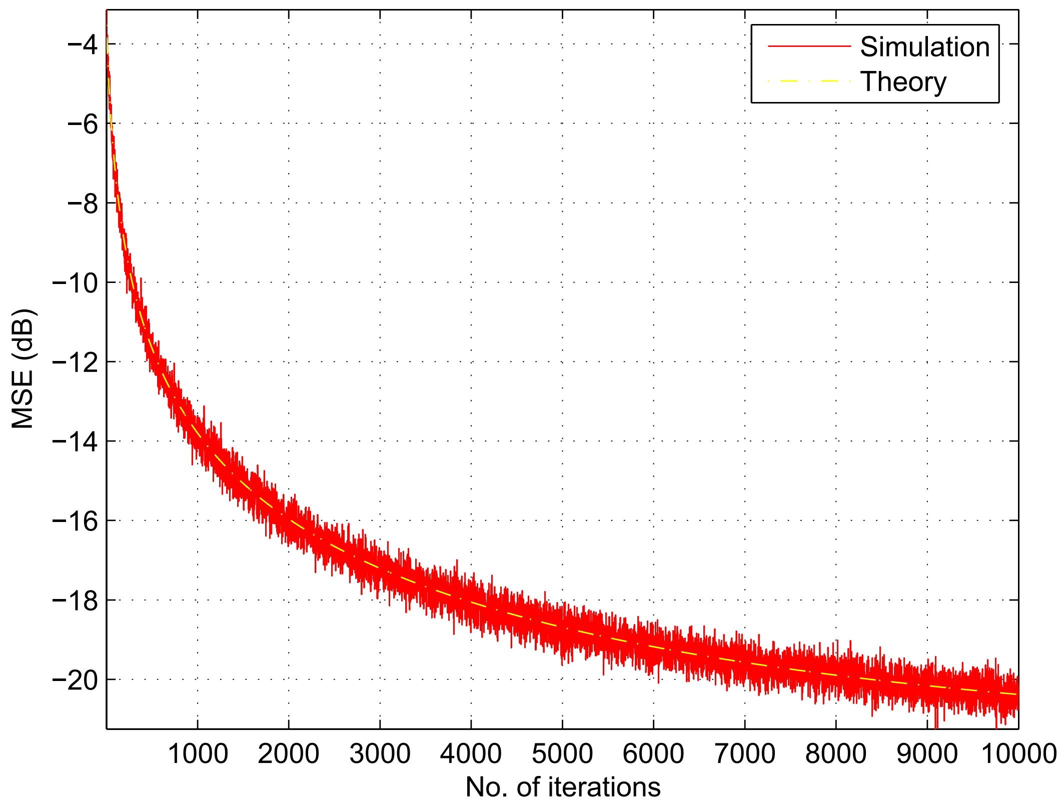

Figure 6.16: Leaky LMF MSE in Gaussian noise with correlated data and noise variance 0.01.

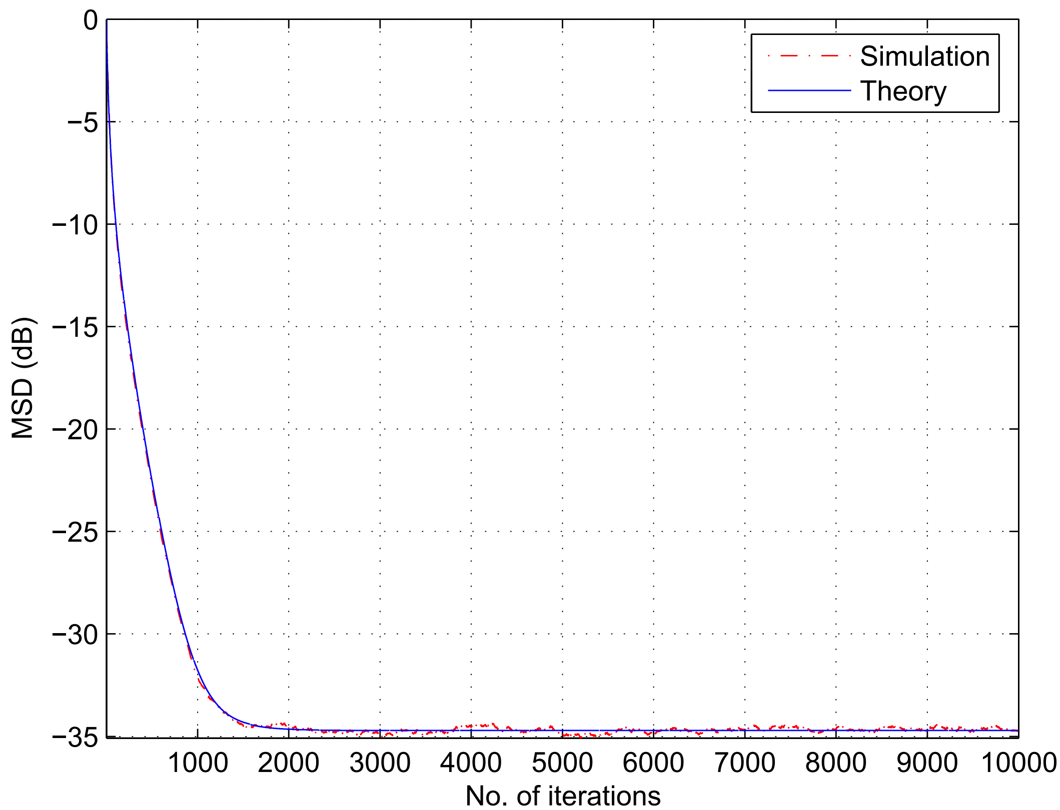

Figure 6.17: Leaky LMF MSD in uniform noise with white data and noise variance 0.1.

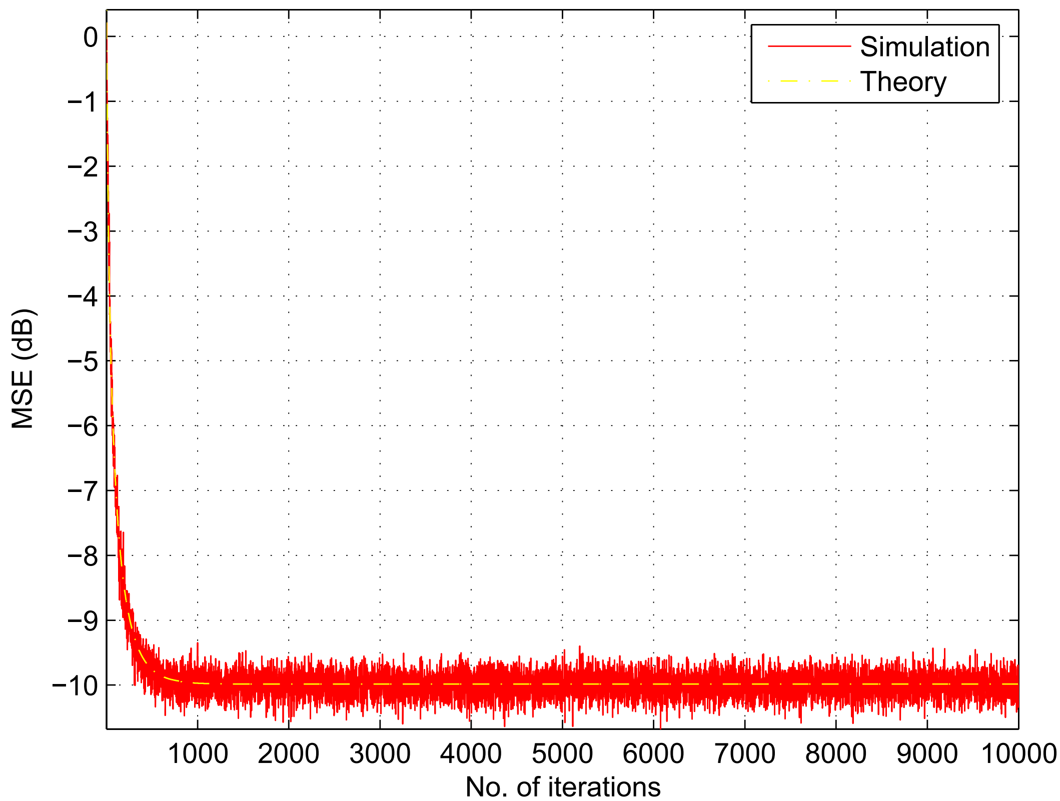

Figure 6.18: Leaky LMF MSE in uniform noise with white data and noise variance 0.1.

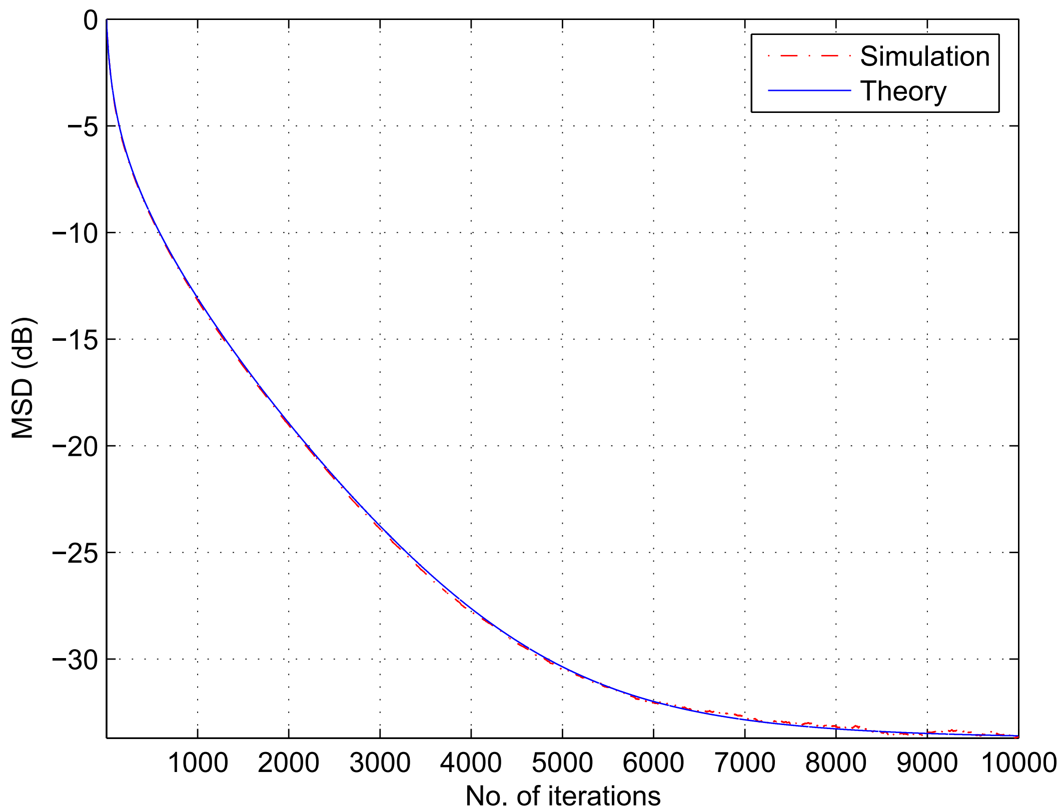

Figure 6.19: Leaky LMF MSD in uniform noise with correlated data and noise variance 0.1.

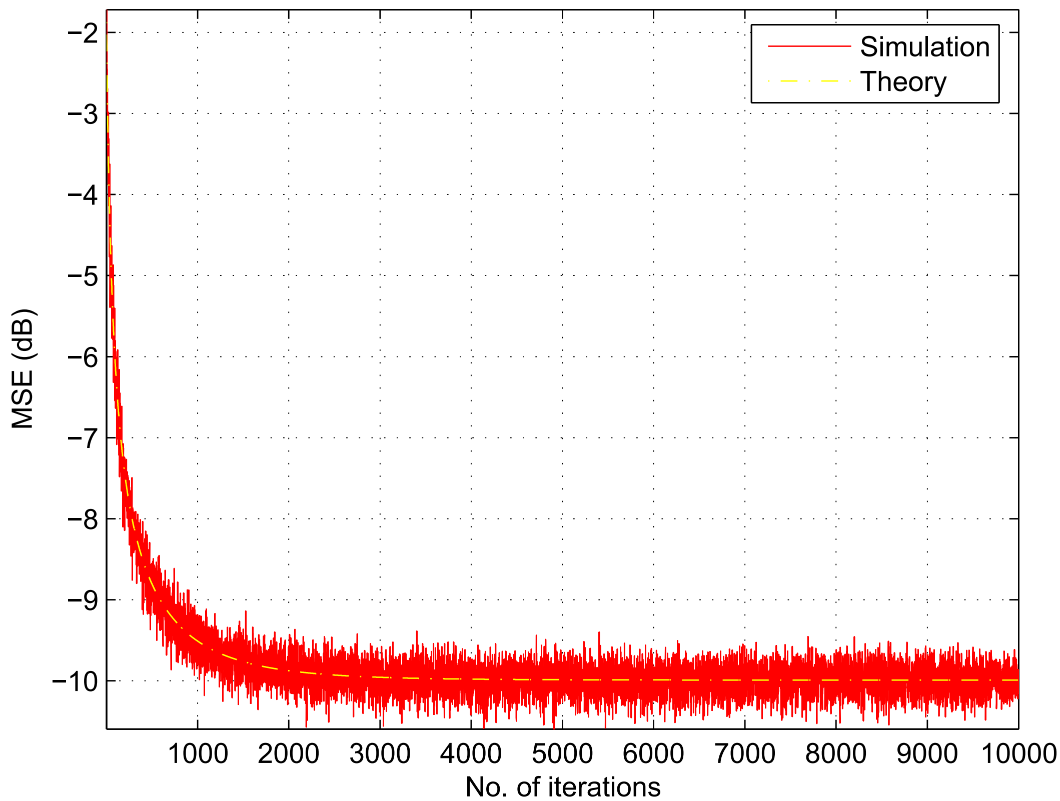

Figure 6.20: Leaky LMF MSE in uniform noise with correlated data and noise variance 0.1.

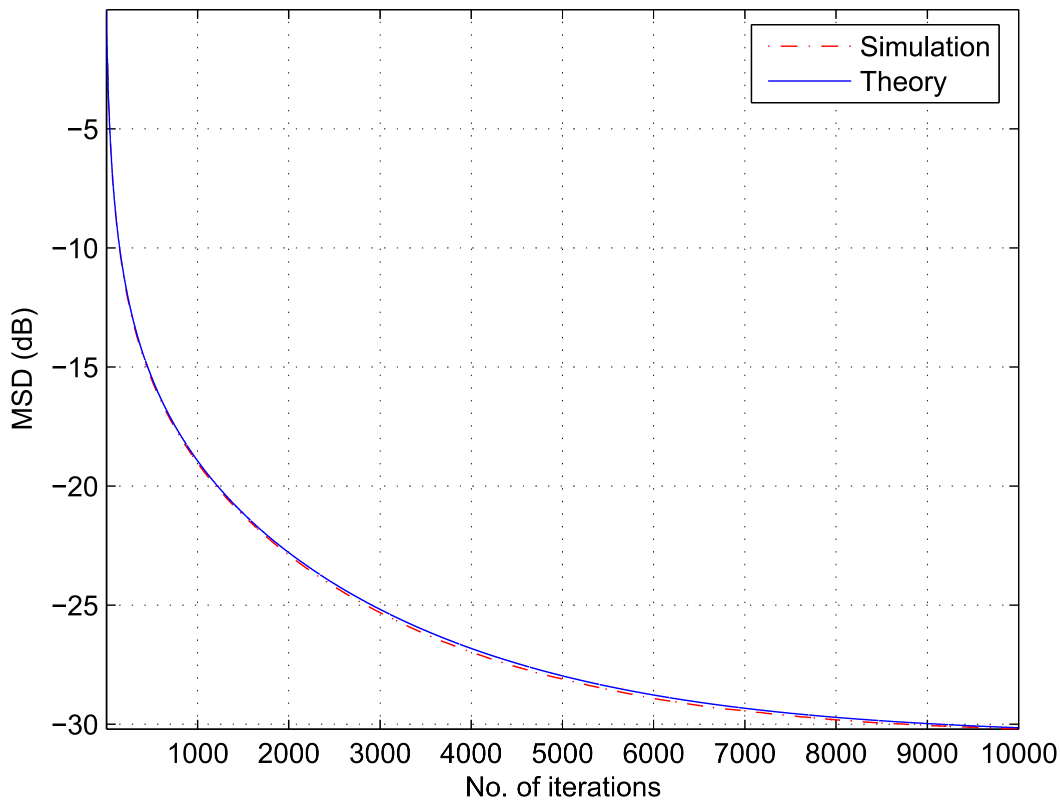

Figure 6.21: Leaky LMF MSD in uniform noise with white data and noise variance 0.01.

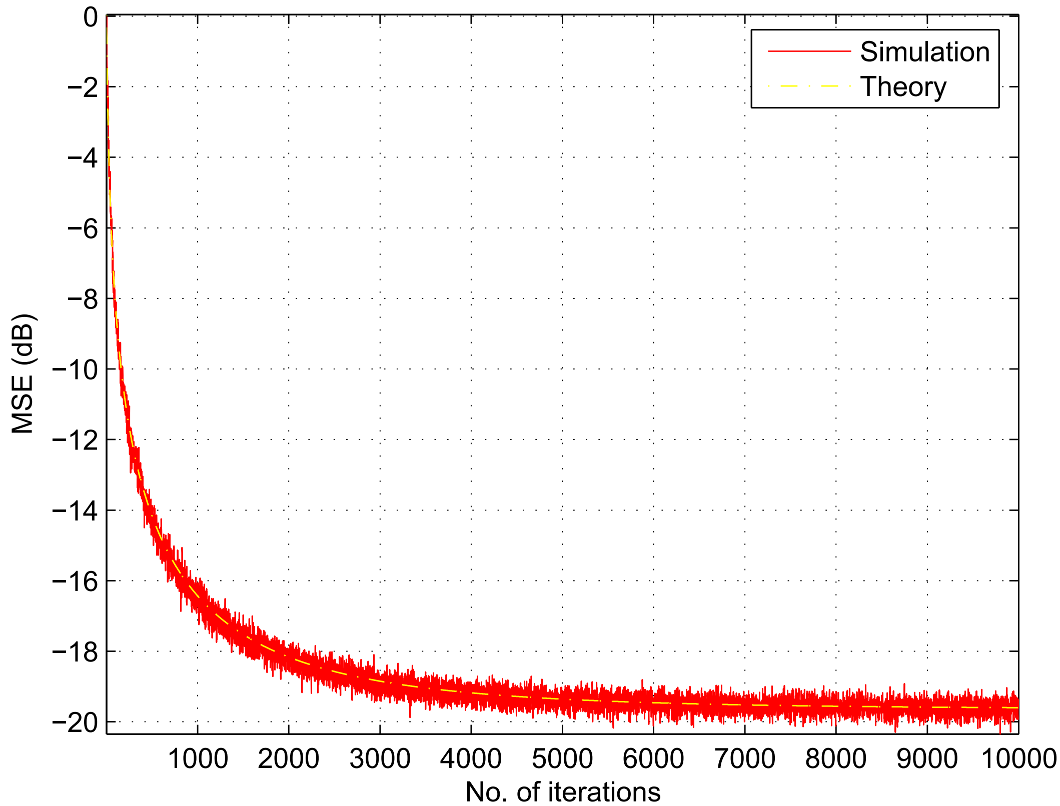

Figure 6.22: Leaky LMF MSE in uniform noise with white data and noise variance 0.01.

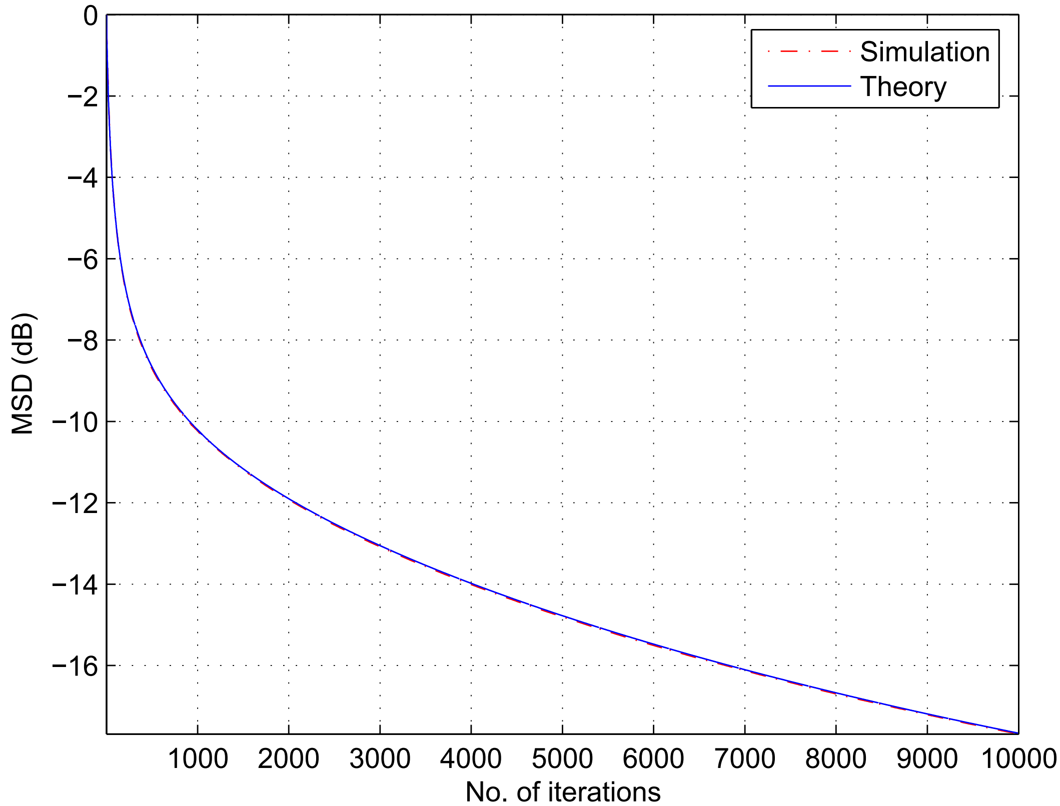

Figure 6.23: Leaky LMF MSD in uniform noise with correlated data and noise variance 0.01.

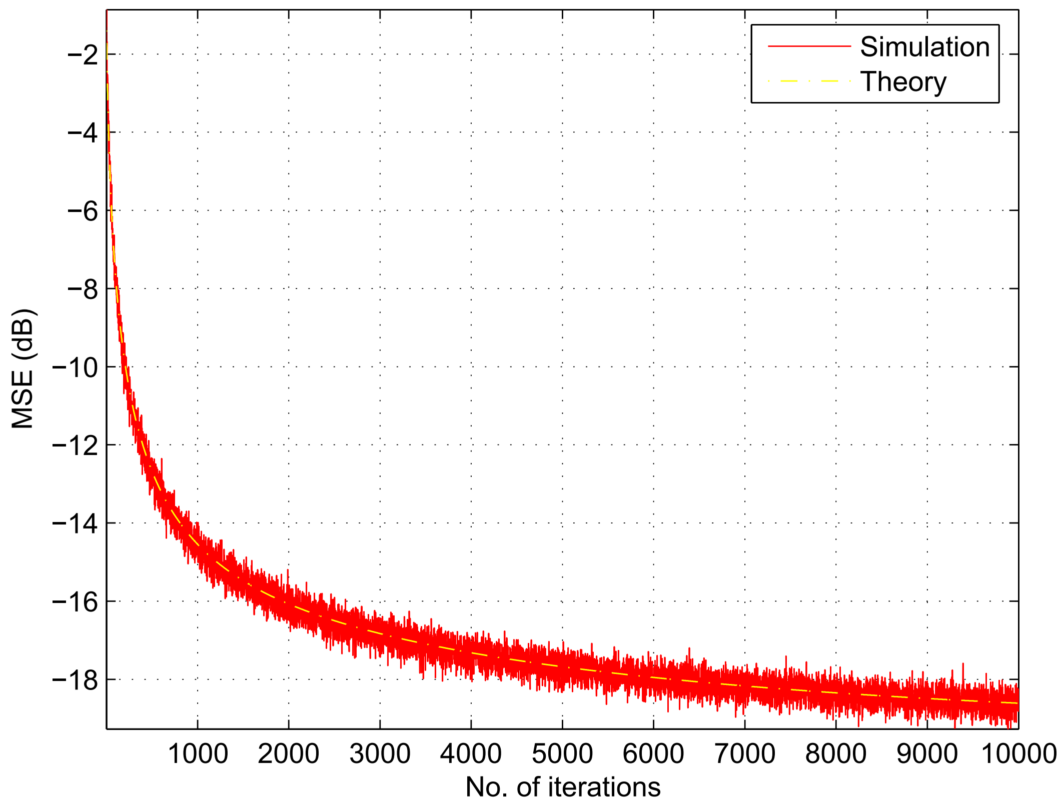

Figure 6.24: Leaky LMF MSE in uniform noise with correlated data and noise variance 0.01.

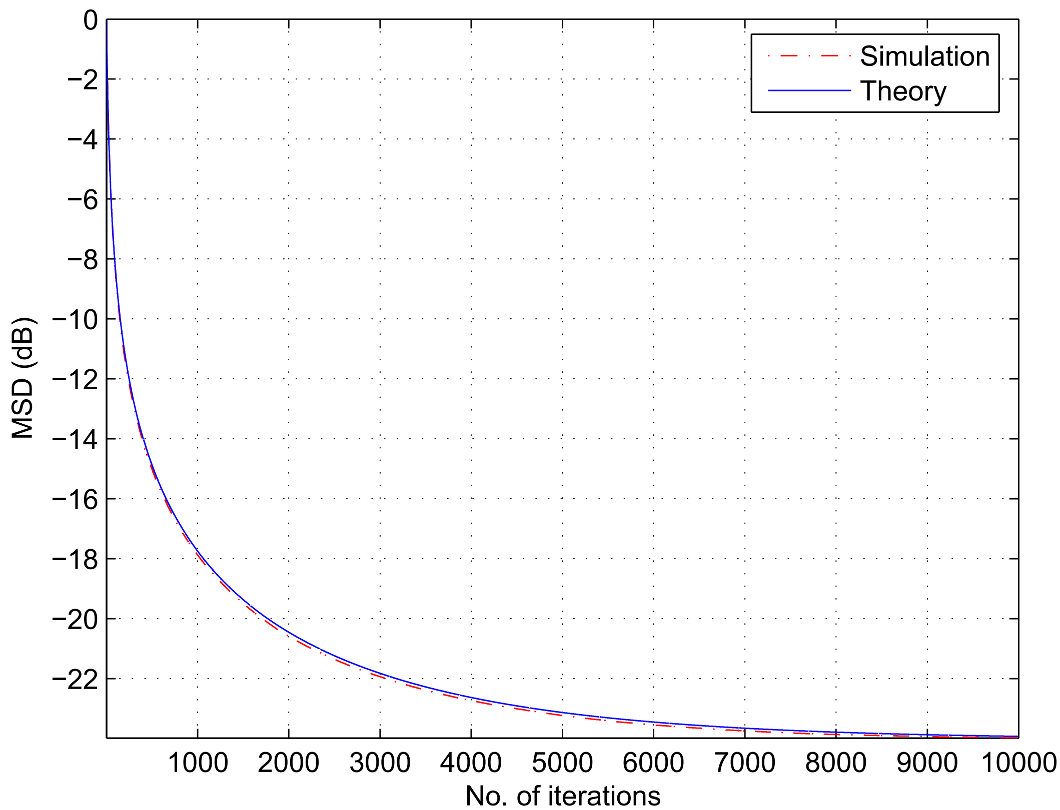

Figure 6.25: Leaky LMF MSD in uniform noise with white data and noise variance 0.01.

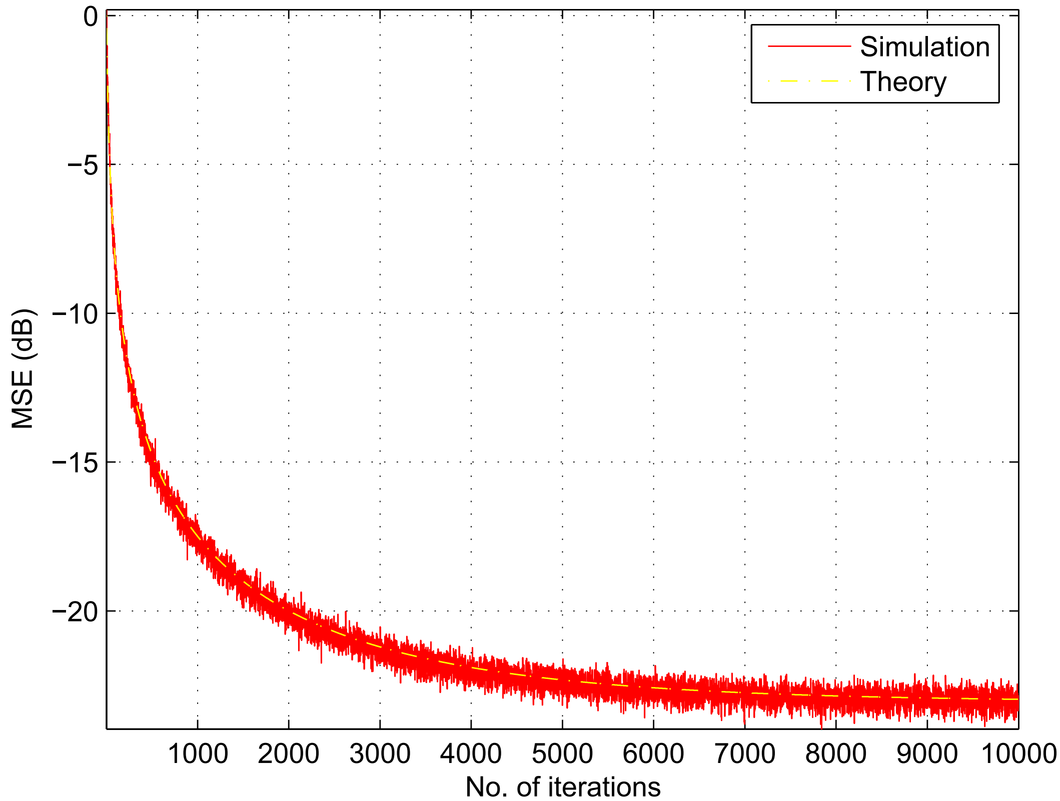

Figure 6.26: Leaky LMF MSE in uniform noise with white data and noise variance 0.01.

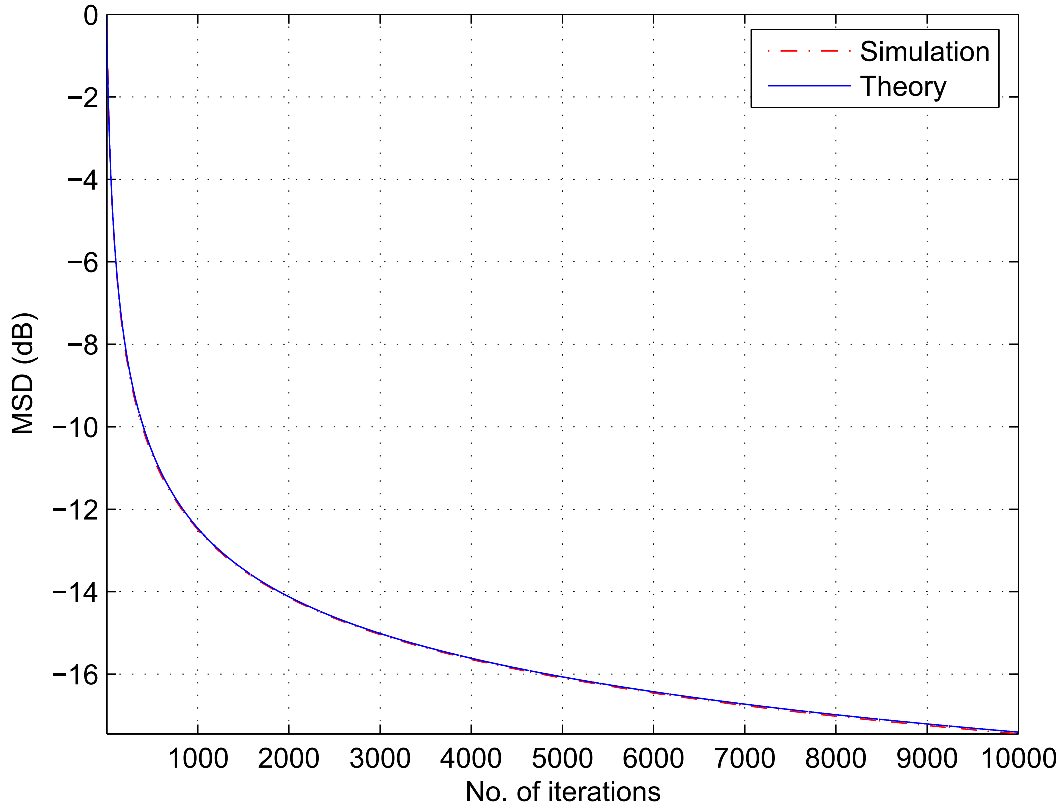

Figure 6.27: Leaky LMF MSD in uniform noise with correlated data and noise variance 0.01.

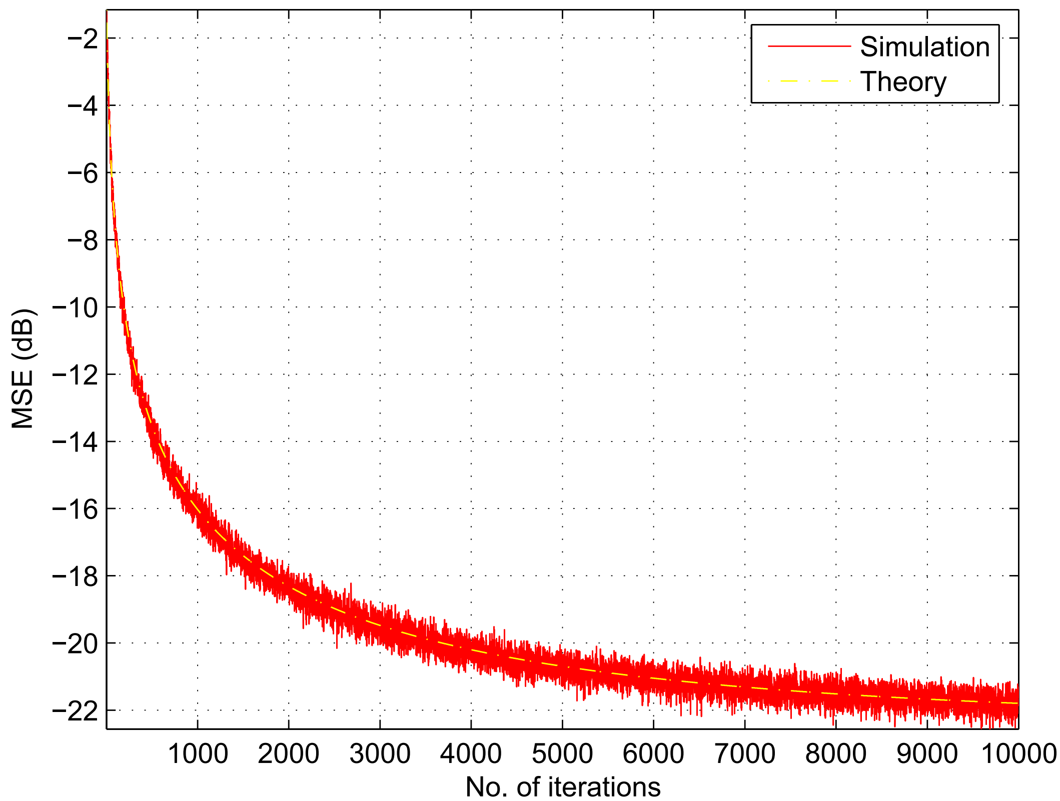

Figure 6.28: Leaky LMF MSE in uniform noise with correlated data and noise variance 0.01.

6.4 Tracking Analysis of Leaky LMF

In this section, we will look at the behavior of the Proposed Leaky LMF algorithm in a non-stationary environment for which we will use the random walk model. The step size, leakage factor and the noise variance were set at 0.01, 0.001 and 0.001, respectively. The number of samples used was 10^4 and the number of trials was set at 800. The mean vector of the varying true system weight was randomly generated and normalized and the number of taps was set to 5. The elements of the weight vector are independent and identically distributed. The simulations were carried out for both Uniform and Gaussian noise and algorithm was tested with the variances of true weight vector elements set at 10^{-5},10^{-6} and 10^{-7}. Theoretical results were generated using (5.13). We see from the Fig. 6.29-6.40 that the theoretical and the simulation results match.

Moreover, as expected, it is observed that as the variance of the true weight vector decreases from 10^{-5} to 10^{-7}, the MSE performance of the Leaky LMF improves.

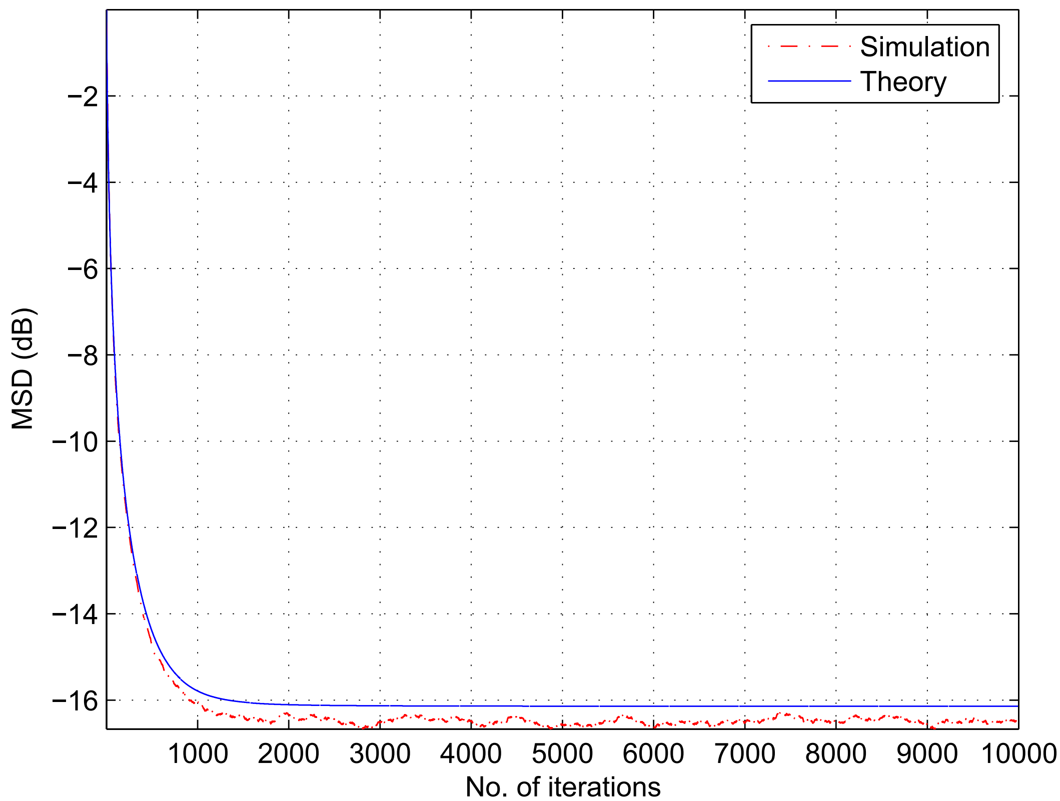

Figure 6.29: Tracking MSD of leaky LMF in Gaussian noise with weight variance 10^{-5}.

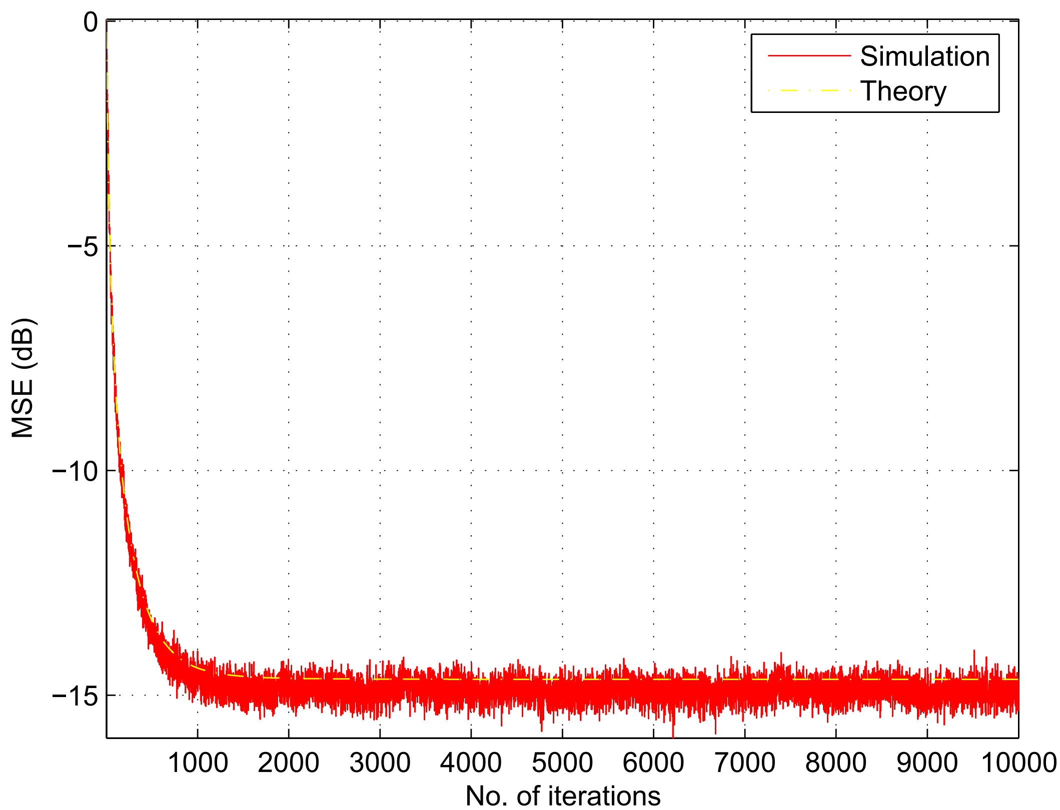

Figure 6.30: Tracking MSE of leaky LMF in Gaussian noise with weight variance 10^{-5}.

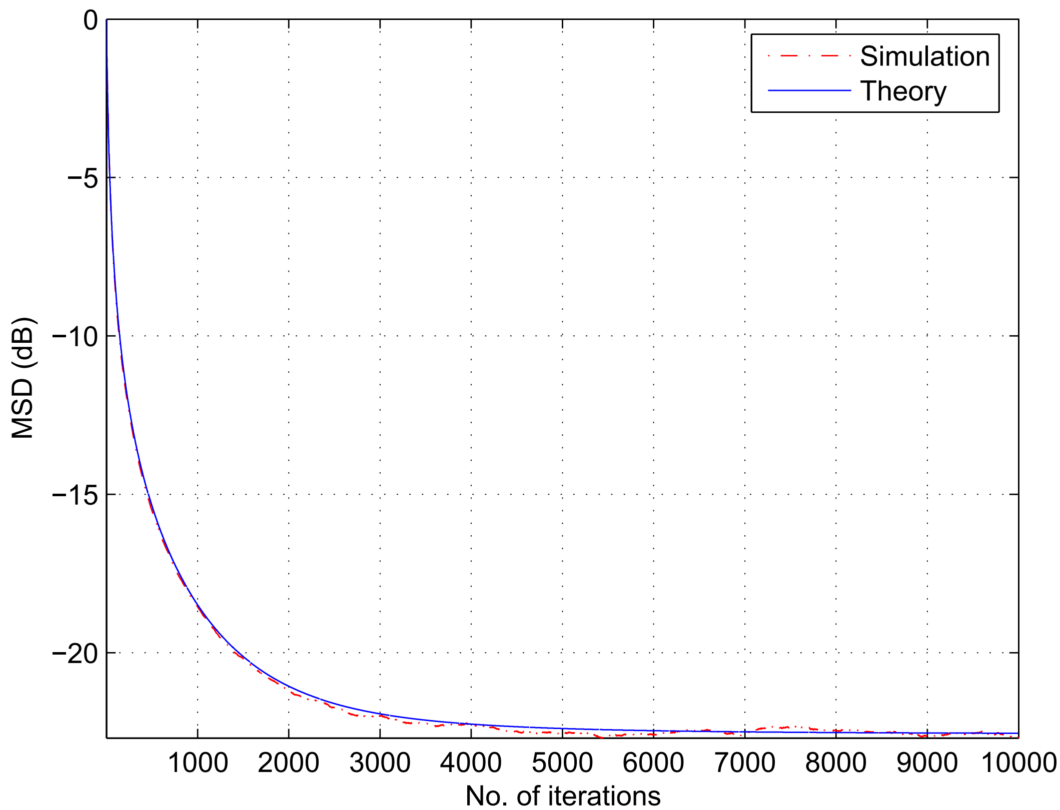

Figure 6.31: Tracking MSD of Leaky LMF in Gaussian noise with weight variance 10^{-6}.

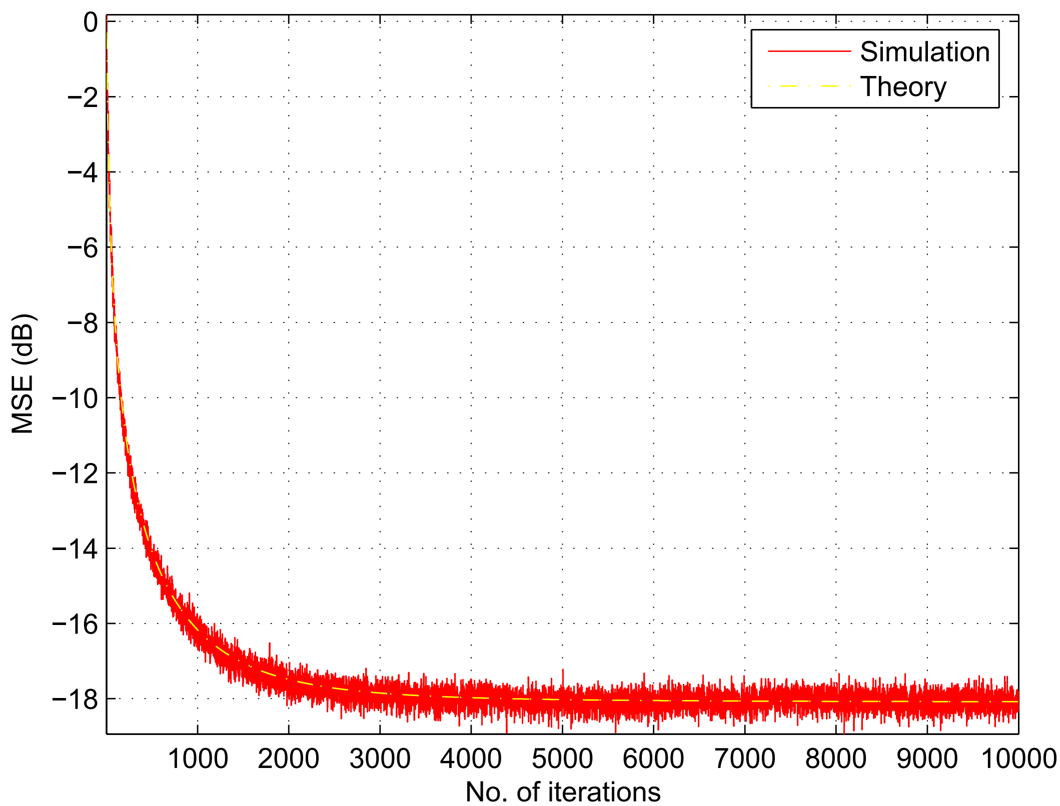

Figure 6.32: Tracking MSE of Leaky LMF in Gaussian Noise with Weight variance 10^{-6}.

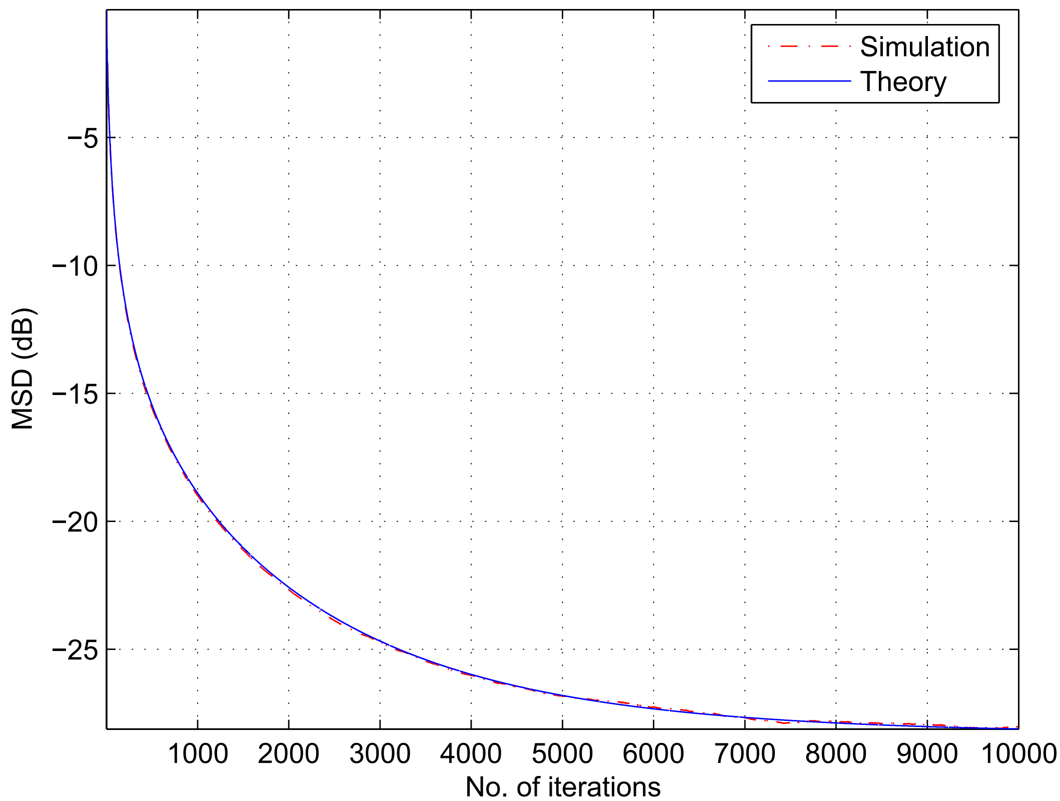

Figure 6.33: Tracking MSD of leaky LMF in Gaussian noise with weight variance 10^{-7}.

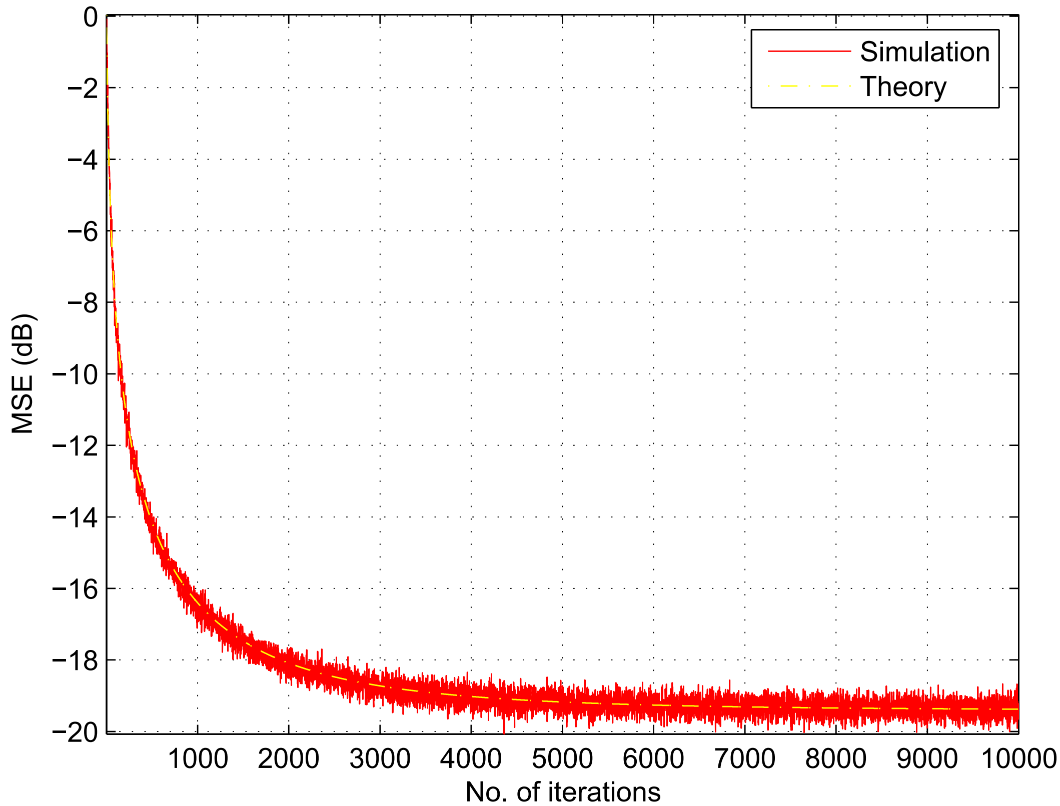

Figure 6.34: Tracking MSE of leaky LMF in Gaussian noise with weight variance 10^{-7}.

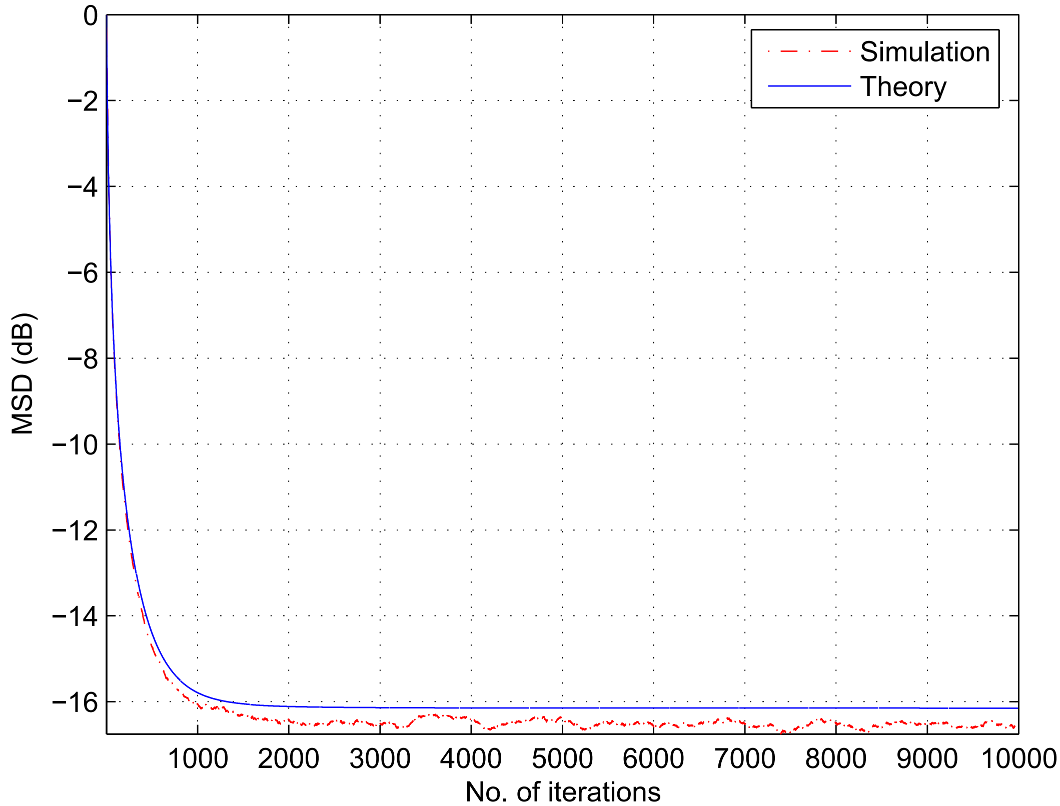

Figure 6.35: Tracking MSD of leaky LMF in uniform noise with weight variance 10^{-5}.

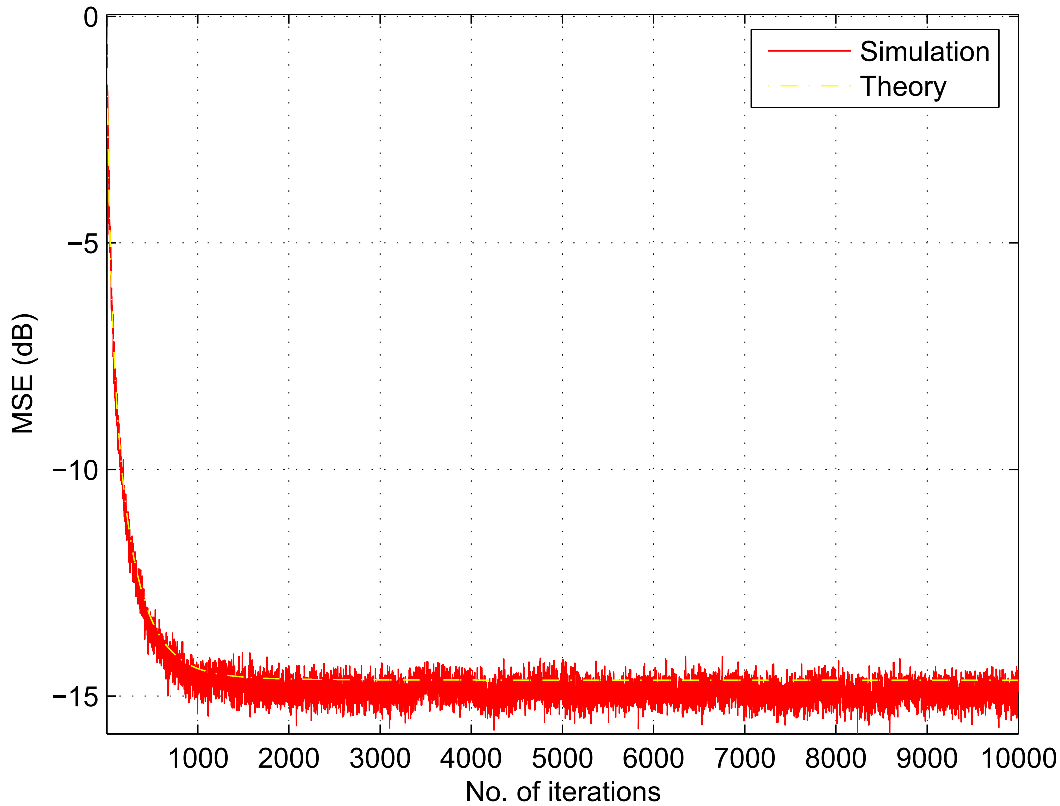

Figure 6.36: Tracking MSE of leaky LMF in uniform noise with weight variance 10^{-5}.

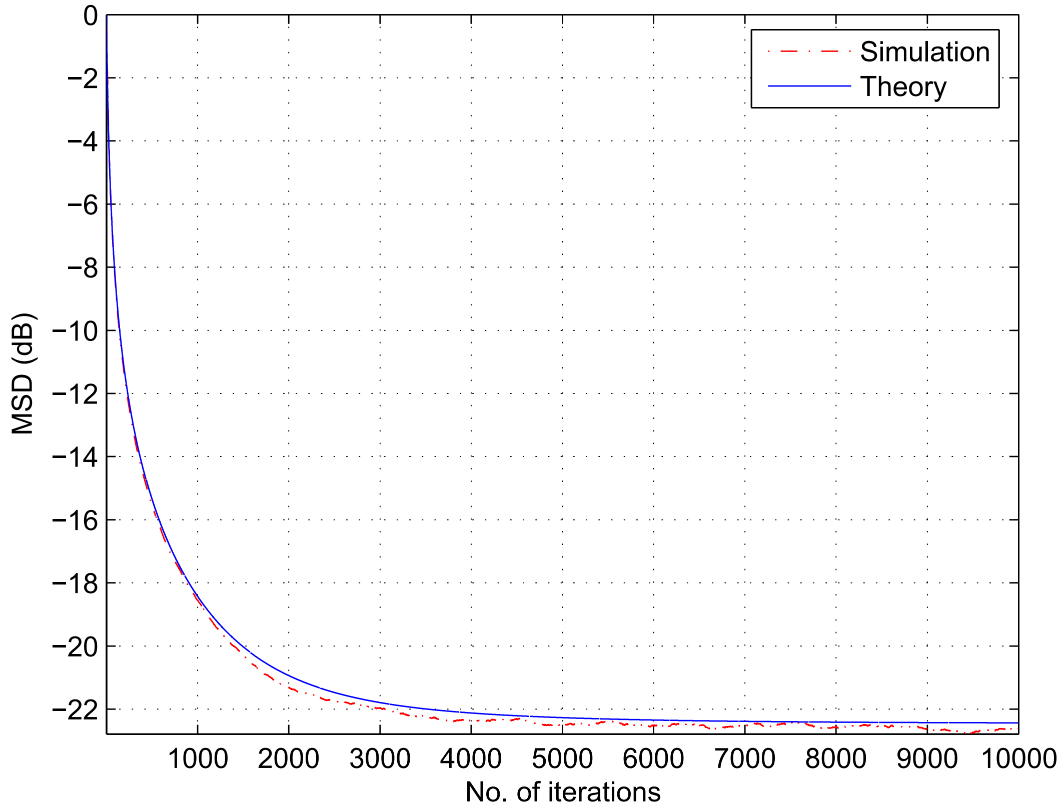

Figure 6.37: Tracking MSD of leaky LMF in uniform noise with weight variance 10^{-6}.

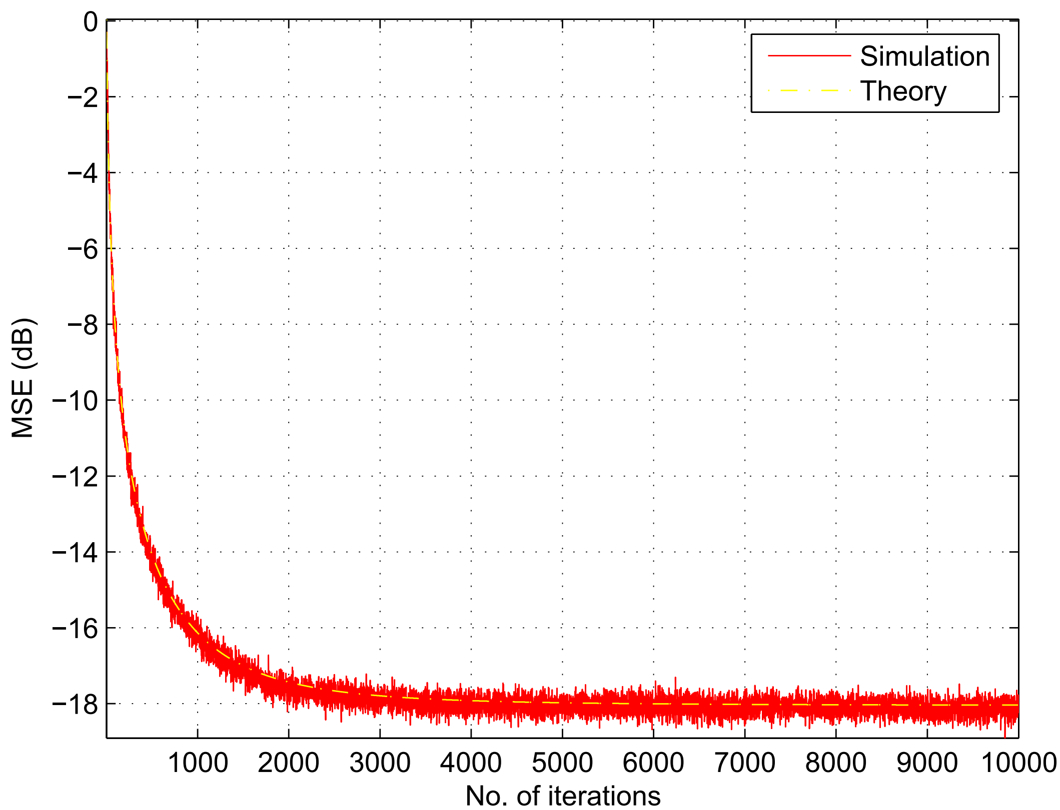

Figure 6.38: Tracking MSE of Leaky LMF in uniform noise with weight variance 10^{-6}.

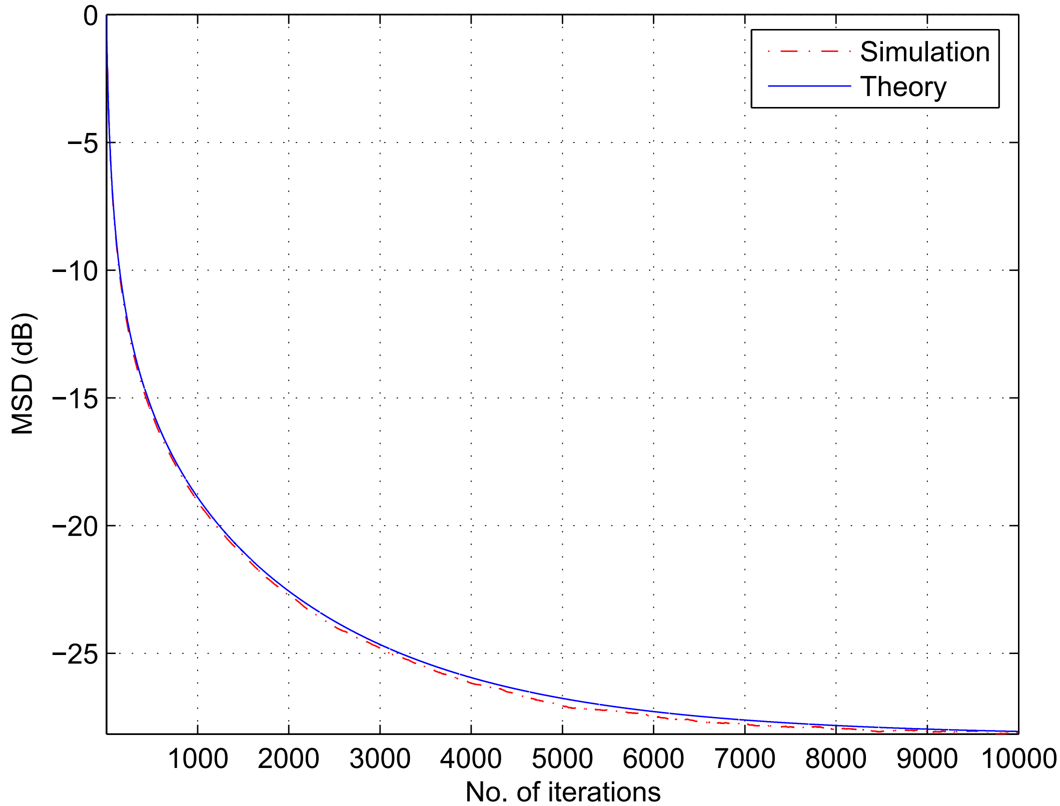

Figure 6.39: Tracking MSD of leaky LMF in uniform noise with weight variance 10^{-7}.

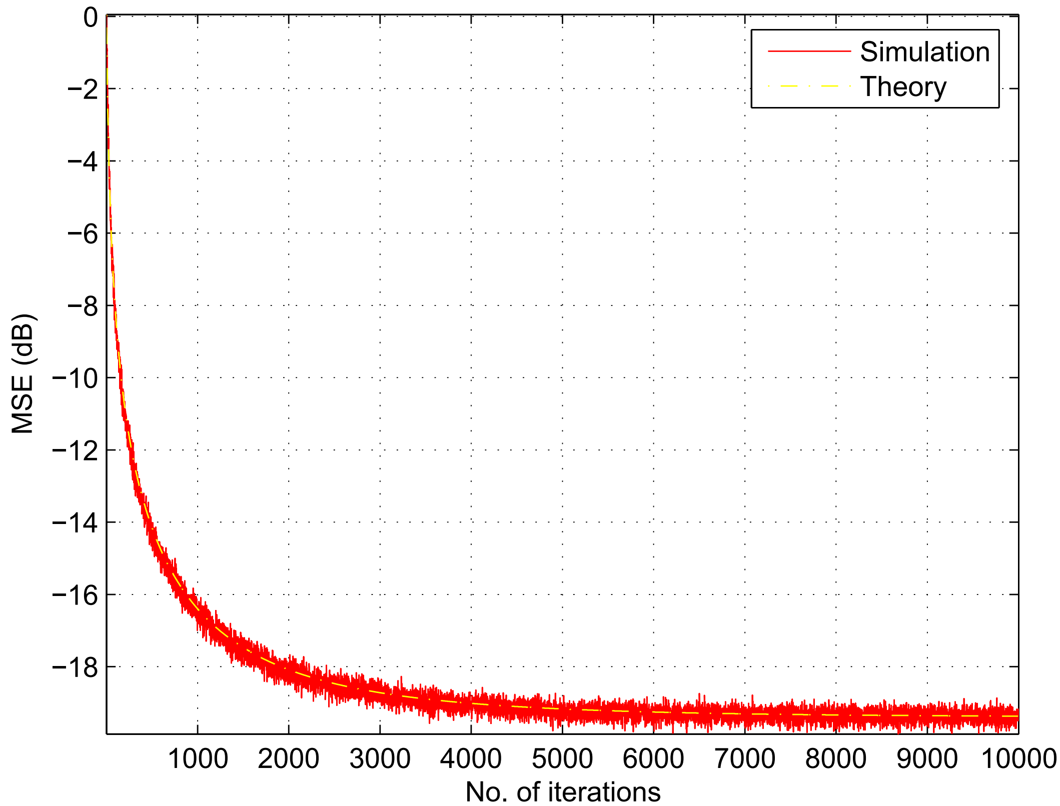

Figure 6.40: Tracking MSE of leaky LMF in uniform noise with weight variance 10^{-7}.

CHAPTER 7

THESIS CONTRIBUTIONS AND RECOMMENDATIONS FOR FUTURE WORK

7.1 Thesis Contributions

This work successfully presented the Leaky LMF algorithm. This algorithm was analyzed in terms of its convergence properties, steady-state and tracking performances and transient behavior. The performance of the proposed algorithm has been supported by presenting the simulation scenarios. the major contributions of this thesis work are as follows:

1. A new LMF variant with a leakage factor which mitigates weight drift.

2. The convergence analysis of the proposed algorithm derived in terms of the mean and mean square sense and as well as a model for estimating the time

evolution of the mean square error and the mean square deviation for the algorithm.

3. The steady state analysis of the algorithm carried as the limiting case of the transient behavior of the algorithm.

4. Tracking ability of the algorithm analyzed and the model for the time evolution of the algorithm in a non-stationary environment derived.

5. Finally, the analytical results compared with the experimental results which support the analysis.

7.2 Recommendations for Future Work

There are a few suggestions regarding future work. In this thesis, a constant leakage factor was used which caused bias in the mean square error. However, by using the various techniques used for removing the bias in the case of Leaky LMS, we can find even better variants of the Leaky LMF that mitigate the weight drift problem without causing a bias. Furthermore, these variants are expected to perform better than their LMF counterparts in terms of steady state misadjustment.

REFERENCES

[1] S. Haykin, Adaptive Filter Theory. Upper Saddle River, NJ:Prentice-Hall, 1996.

[2] A.H. Sayed, Fundamentals of Adaptive Filtering. New York, NY:Wiley Interscience,2003.

[3] J.G. Proakis, Digital Communications, McGraw-Hill, 2001.

[4] E. Wallach and B. Widrow, "The Least Mean Fourth (LMF) Adaptive Algorithm and its Family," IEEE Trans. Information Theory, vol. IT-30, no. 2, March 1984.

[5] A. Zerguine, "Convergence Behaviour of the Normalized Least Mean Fourth Algorithm," Proceedings of the 34th Annual Asilomar Conf. on Signals, Systems, and Computers, pp. 275-278, Oct.29-Nov.1, 2000.

[6] S. Haykin,Unsupervised Adaptive Filtering Volume I:Blind Source Separation. New York, NY:Wiley Interscience, 2000.

[7] B. Widrow and M.E. Hoff,Jr., "Adaptive Switching Circuits", IRE WESCON Conv.Rec., Pt.4, pp. 96-104.

[8] J.I. Nagumo and A. Noda, "A Learning Method for System Identification", IEEE Trans. on Automatic Control,vol 4C-12, no.3, pp. 282-287.

[9] A.E. Albert and L.S. Gardner,Jr., Stochastic Approximations and Nonlinear Regression, MIT Press, Cambridge, MA.

[10] W. Sethares et al, "Parameter Drift in LMS Adaptive Filters", IEEE Trans. Accoustics, Speech and Signal Processing, Vol. ASSP-34, No.4, August 1986

[11] D. A. Lawrence and W. Sethares, " Parameter Drift Instability in Disturbance-Free Adaptive Systems", IEEE Trans. Automatic Control, Vol.38, No.4, April 1993.

[12] M. Hovd and R. R. Bitmead, " Directional Leakage and Parameter Drift", International Journal of Adaptive Control & Signal Processing, 20:27-39, 2006

[13] J. Cioffo, "Limited Precision Effects in Adaptive Filtering", IEEE Trans. on Signals & Systems, CAS-34, no.7, pp. 821-833, July 1987.

[14] V.H. Nascimento and A.H. Sayed, "Unbiased and Stable Leakage-Based Adaptive Filters", IEEE Trans. Signal Processing, vol. 47, No.12, pp. 3261-3276, December 1999.

[15] B.D. Rigling and P. Scniter, "Subspace Leaky LMS", IEEE Trans. Signal Processing, Vol. 11, no.2, pp. 136-139, February 2004.

[16] T. Y. Al-Naffouri and A. H. Sayed, "Transient Analysis of Adaptive FIlters with Error Non-Linearities", IEEE Trans. Signal Processing, Vol.51, No.3, pp. 653-663, March 2003.

[17] T. Y. Al-Naffouri and A. H. Sayed, "Transient Analysis of Data-Normalized Adaptive FIlters", IEEE Trans. Signal Processing, Vol.51, No.3, pp. 639-652 March 2003.

[18] T. Y. Al-Naffouri and A. H. Sayed, "Mean Square Analysis of Normalized Leaky Adaptive FIlters", Proceedings of ICASSP, Salt Lake City, Utah, May 2001.

[19] P. I. Hubscher and J. C. Bermudez, " An improved Statistical Analysis of the Least Mean Fourth Algorithm", IEEE Trans. Signal Proc., Vol. 51, No.3, pp. 664-671, March 2003.

[20] A. Zerguine et al, " Convergence and Tracking Analysis of a Variable Normalized LMF (XE-NLMF) Algorithm", Signal Processing, Vol. 89, No. 5, pp. 778-790, May 2009.

[21] A.D. Poularikas, The Handbook of Formulas and Tables for Signal Processing. Boca Raton, FL:CRC Press,1999.

Vitae

- Obaid ur Rehman Khattak.
- Pakistani
- Born in Karak, Pakistan on August 2nd, 1984.
- King Fahd University of Petroleum & Minerals, P.O.Box# 1854, Dhahran 31261, Kingdom of Saudi Arabia.
- House# 344, Street# 14, Sector F-10/2, Islamabad, Pakistan.
- Received Bachelor of Science in Computer Engineering from King Fahd University of Petroleum & Minerals, Dhahran, KSA in June 2006.
- Joined King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia as a Research Assistant in September 2006.
- Completed Master of Science (M.S.) in Electrical Engineering in June 2009.
- Email: *obaidkhattak@gmail.com.*