LMS BASED SPARSE ADAPTIVE FILTERS: FROM 1D TO 2D REPRESENTATIONS

by

Gülden ELEYAN

Submitted to the Institute of Graduate Studies in Science and Engineering

in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

in

Electrical & Computer Engineering

Mevlana (Rumi) University

2016

LMS BASED SPARSE ADAPTIVE FILTERS: FROM 1D TO 2D REPRESENTATIONS

submitted by **Gülden ELEYAN** in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical & Computer Engineering at Mevlana (Rumi) University.

APPROVED BY:

Examining Committee Members:

Assist. Prof. Dr. Mohammad Shukri SALMAN (Thesis Supervisor)	. . . .
Prof. Dr. Bekir KARLIK	. . . .
Assoc. Prof. Dr. Halis ALTUN	. . . .
Assist. Prof. Dr. Ömer Kaan BAYKAN	. . . .
Assist. Prof. Dr. Hatem HADDAD	. . . .

Assist. Prof. Dr. Mehmet ARGIN . . . .
Head of Electrical & Computer Engineering Department

Prof. Dr. Ali SEBETCI . . . .
. Director, Institute of Graduate Studies in Science and Engineering

DATE OF APPROVAL ( )

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Gülden ELEYAN

Signature:

ABSTRACT

LMS BASED SPARSE ADAPTIVE FILTERS: FROM 1D TO 2D REPRESENTATIONS

Gülden ELEYAN

Ph.D. Dissertation, 2016

Thesis Supervisor: Assist. Prof. Dr. Mohammad Shukri SALMAN

Keywords: Adaptive filters, least mean square, sparse systems, zero attracting, *l*-norm

Adaptive filters are commonly and regularly used nowadays in different gadgets such as mobile phones and devices, medical equipment and digital cameras. This can be easily justified due to the increase of influence of digital signal processors. Example of the necessity for using adaptive filters is in the telecommunications field, where we can face many sources of echo caused by long distance communications or hands-free voice conversation. Another example is in computer vision applications, where the received images can be noisy and thus, need to be enhanced and filtered form such an unwanted noise.

The adaptive filter is a system that computationally models the relation between the input and output signal. The adaptive filter modifies or adjusts its coefficients iteratively based on some adaptive optimization algorithm. The cost function is the key or the criterion for optimizing the filter performance which will work on minimizing the error signal.

In this dissertation, new different algorithms for improving the performance of one-dimensional and two-dimensional least mean square (LMS) algorithm are proposed.

This study started by providing the derivation of the convergence analysis of the mixed-norm least mean square (MN-LMS) algorithm. Our first proposed algorithm is based on MN-

LMS algorithm. The proposed algorithm exploits the sparsity of the system by adding l_1-norm penalty term to the cost function of the original MN-LMS algorithm. The new term enables us to attract the zero and near-to-zero filter weights to the zero value in a faster way. However, when the targeted system is near or exactly non-sparse, the performance of this algorithm drops. To overcome the limitation of this algorithm when the system is near or exactly non-sparse another algorithm that uses an approximation of l_0-norm penalty term in the cost function of the original MN-LMS algorithm is proposed. This provides high performance even with completely non-sparse systems.

For improving the two-dimensional least mean square (2D-LMS) algorithm performance, a new two-dimensional zero-attracting least mean square (2D ZA-LMS) adaptive filter is proposed by imposing a sparsity aware l_1-norm penalty term in the cost function of the original 2D-LMS algorithm. The convergence analysis of the 2D ZA-LMS algorithm is presented and stability criterion is also derived.

Beside the mathematical derivation for the convergence analysis of the provided algorithms, all the algorithms are experimentally tested. In this study, extensive experiments are conducted using different parameters and scenarios such as signal-to-noise-ratio (SNR), sparsity level and filter tap length. Images corrupted with different noise types and different parameters are used to test the proposed 2D ZA-LMS algorithm against the 2D-LMS algorithm.

ÖZET

LMS BASED SPARSE ADAPTIVE FILTERS: FROM 1D TO 2D REPRESENTATIONS

Gülden ELEYAN

Doktora Tezi, 2016

Tez Danışmanı: Yrd. Doç. Dr. Mohammad Shukri SALMAN

Anahtar Kelimeler: Uyarlanır Filtre, en küçük kare ortalaması, seyrek sistem, sıfıra çeken, l-norm.

Sayısal sinyal işlemenin gücünün artışıyla, uyarlanır filtreme çok daha yaygın hale gelmiştir ve şu sıralar; mobil telefonları, diğer haberleşme cihazları ve tıbbi ekipmanlar ve dijital kameralar gibi elektronik cihazlarda sıklıkla kullanılmaktadır.

Uyarlanır filtre giriş ve çıkış sinyali arasındaki ilişkiyi hesaplamalı olarak modelleyen sistemdir. Uyarlanır filtreler katsayılarını bazı adaptif optimizasyon algoritmalarına dayanarak iteratif olarak ayarlar veya modifiye ederler. Hata sinyalini minimize etmeye yarayan cost fonksiyonu, süzgeç performansını optimize etmeye yarayan anahtar kriterdir.

Bu tezde bir ve iki boyutlu En Küçük Kare algoritmasının performansını iyileştirecek farklı yeni algoritmalar önermekteyiz.

Karışık normlu LMS algoritmasının çözümü ile başlamaktayız. İlk algoritmamız karışık norm'lu LMS algoritmasına dayalıdır. Sistem seyrekliği, MN-LMS algoritmasının cost fonksiyonuna l_1-norm ceza terimi eklenerek ortaya çıkarılmaktadır. Bu terim, sıfır veya sıfıra yakın filtre bileşenlerinin sıfıra çekilmesine hızlı bir şekilde olanak verir. Buna rağmen, sistem sparse'a yakın ya da tamamen sparse değilse çalışmamaktadır. Sistemin tamamen seyrek olmaması durumunun getirdiği sınırlamayı aşmak için MN-LMS algoritmasının cost

fonksiyonunda l_0-norm penaltısının bir yaklaşımını kullanan başka bir algoritma önermekteyiz. Bu yöntem, hiç seyrek olmayan sistemlerde bile yüksek performans sağlamaktadır.

İki boyutlu LMS algoritmasının performansını iyileştirmek için cost fonksiyonuna bir l_1-norm seyreklik farkındalık terimi ekleyerek yeni iki boyutlu (2D) sıfıra çeken en küçük kare algoritması (ZA-LMS) öneriyoruz. 2 boyutlu ZA-LMS algoritmasının yakınsama analizi ve denge kriteri yapılmıştır.

MN-LMS algoritmasının yakınsama analizi çıkarılmış tüm algoritmalar deneysel olarak test edilmiştir. Farklı parametreler, sinyalin gürültüye oranı, seyreklik ve süzgeç uzunluğu, kullanılarak çok miktarda deney gerçekleştirilmiştir. Bu çalışmada, farklı gürültü çeşitleri ve farklı parametreler kullanılarak bozulmuş görüntüler önerilen iki boyutlu sıfıra çeken LMS algoritmasının iki boyutlu standart LMS algoritmasıyla kıyaslanması için kullanılmıştır.

To My Family

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude to my PhD supervisor Assist. Prof. Dr. Mohammad Shukri SALMAN for his guidance, advice, encouragement and insight throughout the research. I appreciate the comments and suggestions of my thesis monitoring committee and my defense jury members: Prof. Dr. Bekir KARLIK, Assoc. Prof. Dr. Halis ALTUN, Assist. Prof. Dr. Ömer Kaan BAYKAN and Assist. Prof. Dr. Hattem HADDAD. Their experience helped to refine my dissertation and bring it to higher standards.

Special thanks goes to my dear parents, specially my mother Sema TARTANOĞLU for their unconditional support, prayers and love. I am also very thankful to my mother and father in law Isaaf and Adnan ELEYAN for their continuous support and prayers.

My deepest gratitude for my beloved husband Alaa ELEYAN and my children Münjid and Semanur. Without their endless patience, tolerance, support, prayers and love, nothing of this would be possible.

Finally, I would like to thank all the friends at Mevlana University especially at the engineering faculty for the lovely and valuable time I spent among them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACGN	Additive Correlated Gaussian Noise
AEC	Acoustic Echo Cancelers
ALE	Adaptive Line Enhancer
AWGN	Additive White Gaussian Noise
FIR	Finite Impulse Response
IIR	Infinite Impulse Response
LMS	Least Mean Square
LSE	Least Sum of Exponentials
MNLMS	Mixed-Norm Least Mean Square
MSE	Mean Square Error
MSD	Mean Square Deviation
NEC	Network Echo Cancelers
NLMS	Normalized Least Mean Square
PSNR	Peak Signal to Noise Ratio
RLS	Recursive Least Squares
RZA-LMS	Reweighted Zero Attracting Least Mean Square
SNR	Signal to Noise Ratio
ZA-LMS	Zero Attracting Least Mean Square
ZAMNLMS	Zero Attracting Mixed-Norm Least Mean Square

LIST OF SYMBOLS

$\mathbf{\Lambda}$	Diagonal Matrix
λ	Eigenvalue
μ	Convergence Rate
ρ	Zero-Attractor Controller
σ^2	Variance
d	Filter Output
e	Error of Estimation
$\mathbf{g_w}$	Gradient
$\mathbf{GI}$	Gradient (2D)
$\hat{\mathbf{g}}_\mathbf{w}$	Estimate of Gradient
$\mathbf{h}$	Unknown System Vector
$\mathbf{I}$	Identity Matrix
$J_\mathbf{w}$	Cost Function
N	Filter Length
$\mathbf{p}$	Cross-correlation Matrix
$\hat{\mathbf{p}}$	Estimate of Cross-correlation Matrix
$\mathbf{R}$	Autocorrelation Matrix
$\hat{\mathbf{R}}$	Estimate of Autocorrelation Matrix
$\mathbf{w}$	Tap Weights Vector
$\mathbf{x}$	Input Signal Vector

CHAPTER 1

INTRODUCTION

1.1. Adaptive Filters

An adaptive filter is system that relies for its updating process on a recursive algorithm. This is an important criterion in a situation where prior information about the relevant statistics is not presented. Adaptive filters can be categorized into two main categories: linear, and nonlinear filters. Adaptive filter that uses a linear combination of the signals which is applied to its input for estimating the desired response is called linear adaptive filters. Otherwise, it is nonlinear adaptive filter [1]. Adaptive filters may also be grouped into supervised and unsupervised filters:

1.2. Applications of Adaptive Filters

Since 1960, adaptive filters are popular because of their many advantages, such as the updating of filter weights over time, to acclimate the change in signal characteristics. Over the past thirty years, digital signal processors have made great improvements in terms of speed, complexity, and power consumption. As a result, real-time adaptive filters are rapidly becoming practical and crucial for the future of wired and wireless communications.

Adaptive linear combiners have been successfully used in the modeling of unknown systems [2] [6]–[8], linear prediction [2][9]–[11], adaptive noise cancellation [4][12], adaptive antenna systems [3][13]–[15], channel equalization systems[16]–[19], echo cancellation [20]–[23], instantaneous frequency estimation [24], adaptive line enhancer [4][25]–[30], adaptive control systems [31], and many other real-time applications.

The main adaptive filters configurations are:

1.2.1. System Identification

System identification configuration is shown in Fig 1.1. An input signal is applied both to the adaptive filter and to the unknown system. The unknown system delivers the desired response. Some applications for system identification are; control systems [5], seismic signal processing [32], and echo cancellation in some communication systems [33].

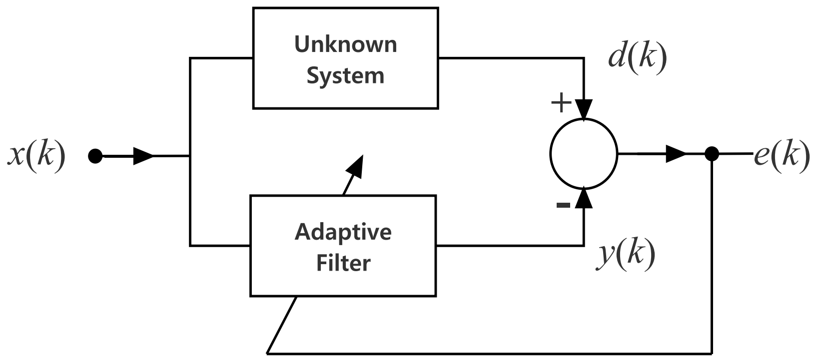


Figure 1.1 System identification configuration.

1.2.2. Noise Cancellation

In in noise cancellation configuration, the reference signal the desired signal which is corrupted by an additive noise $v_1(k)$. The input signal is a noise signal $v_2(k)$ that is correlated with the additive noise $v_1(k)$ and uncorrelated with $x(k)$. Fig. 1.2 shows the configuration of noise cancellation. Some real world examples are acoustic echo cancellation on telephone circuits [4], hydrophones noise cancellation [35], interference cancellation in electrocardiography ECG [5].

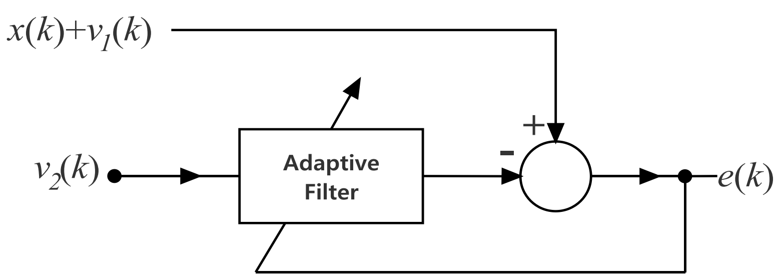


Figure 1.2 Noise cancellation configuration.

1.2.3. System Prediction

The adaptive filtering prediction process aim is developing a model of a signal of interest, instead of encoding it directly. The signal prediction provides the prediction of the best present value of the signal. The input values consist of an older version of the desired response as shown in Fig. 1.3. An important application is coding of speech signals [36]. Other applications are adaptive line enhancement (ALE) and suppression of interferences in signal.

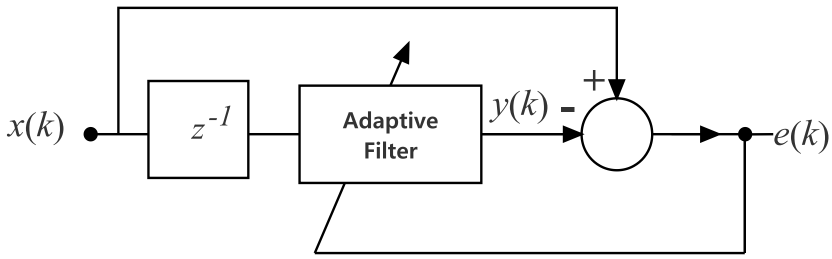


Figure 1.3 System prediction configuration.

1.2.4. Channel Equalization

Channel equalization or inverse modeling includes an estimation of an impulse response that is equal to the inverse of the impulse response of the unknown system, to equalize the linear distortion produced by the channel. The desired response for the adaptive filter is produced by a shifted version of the unknown system input (see Fig 1.4). In [37], decision-feedback equalization, decision-assisted inert-symbol interference ISI cancellation, and adaptive filtering for maximum-likelihood sequence estimation has been performed. A second interpretation is a control application in [5]. Inverse filtering basic idea is to produce signals that can be used in unknown system.

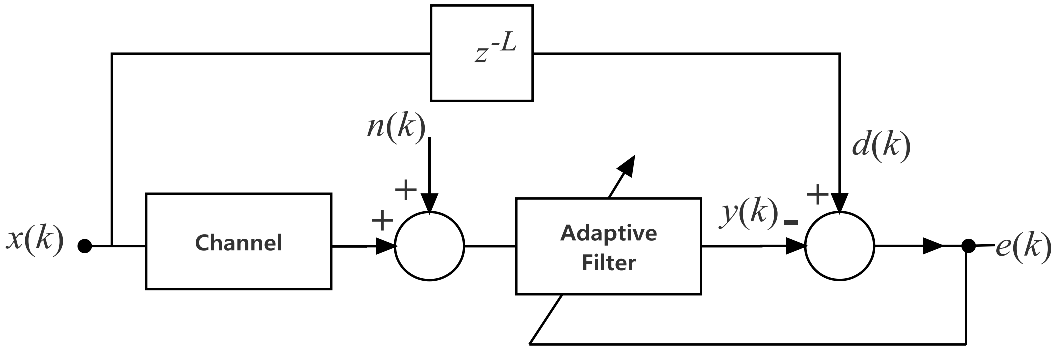


Figure 1.4 Channel equalization.

1.2.5. Echo Cancellation

Echo signal is defined as the delayed and attenuated version of the original signal produced by a certain source, such as a loudspeaker. A challenging problem is that the echo path is changing due to probable over time change of the channel characteristics, because of different parameters such as the distance between the loudspeaker and the microphone.

Fig. 1.5 shows the existence of acoustic echo. The received signal from the far end speaker, $r(k)$ is passing through the loudspeaker to the near end. A delayed version of $r(k)$ signal is received by the microphone, combined with the near end speech, forming the received signal from the acoustic channel, $r_A(k)$. The echo path is defined by functions $\boldsymbol{f}$ and $\boldsymbol{g}$ on both ends; these functions represent the linear or non-linear behavior of the echo path.

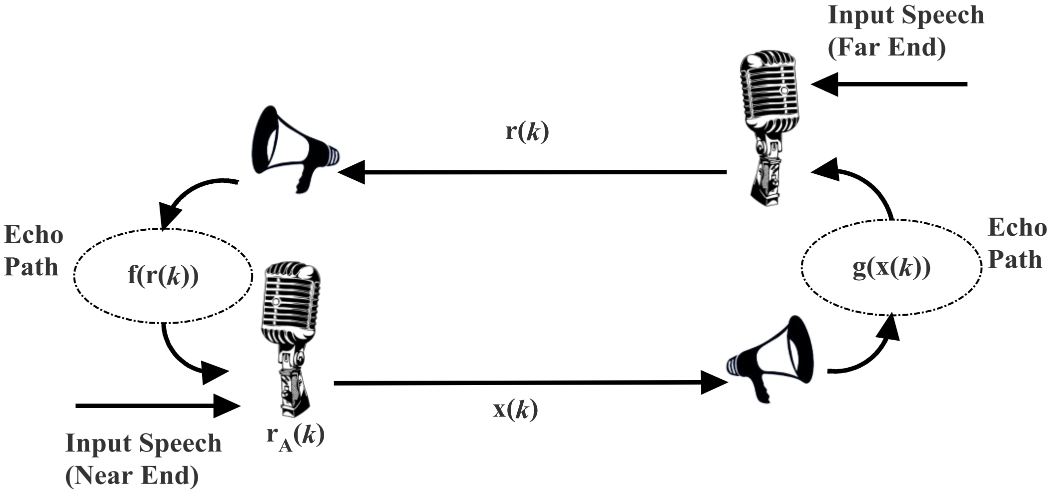


Figure 1.5 An acoustic echo scenario.

1.2.5.1. Sources of Echo

In the telecommunications field, we can find many sources of echo caused by long distance communications and/or the hands-free voice conversations [38]. For instance; on the hands-free voice over internet protocol (VoIP) setup in which we have a microphone and a loudspeaker on a mobile telephone connection.

Applications such as hands-free telephony, tele-conferencing and video-conferencing require the use of acoustic echo cancellation (AEC) techniques to eliminate acoustic feedback from the loudspeaker to the microphone [39]. To be able to construct a model for the mentioned time-changing echo characteristics and to cancel the unwanted effects on the conversation, adaptive filtering [5][40][41] has been widely used in the last three decades (see for instance, [42]-[47]). In Fig. 1.6, the acoustic echo canceller system is shown. The adaptive filter synthesizes a copy of the echo, which is deducted from the returned signal. This removes/minimizes the echo without interrupting the echo path. The adaptive filter compensates the undesired effects of the non-ideal hybrid circuit.

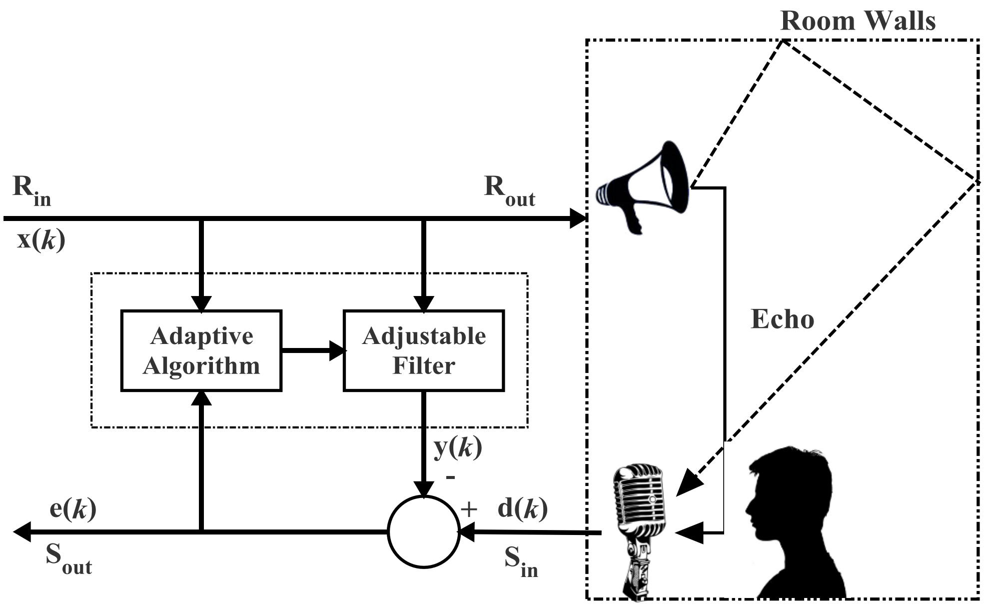


Figure 1.6 The acoustic echo canceller.

1.3. Adaptive Algorithms

1.3.1. Least Mean Square (LMS) Algorithm

The least mean square (LMS) algorithm was developed by Widrow and Hoff in a pattern-recognition machine research [48]. It is known as the adaptive linear element. It is used in many different applications due to its easy implementation. So far, the LMS algorithm is one of the most extensively used algorithms in adaptive filters. The main reasons are its low computational complexity, efficient convergence and stability behavior in stationary environments.

The LMS algorithm includes of two basic steps:

- A filtering step which involves calculating the filter output using tap inputs. Subtracting this output from desired response value will generate an estimation error.

- An adaptive step which involves the automatic updating of the weights of the adaptive filter based on the estimated error.

A summary of the algorithm is shown as following.

The desired output of the filter:

$$d(k) = \mathbf{h}^T \mathbf{x}(k), \tag{1.1}$$

where $\mathbf{h}$ is the optimum filter weights vector and $\mathbf{x}(k)$ is the input-tap vector.

The error of the estimation:

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k), \tag{1.2}$$

where $\mathbf{w}(k)$ is the estimated filter weights vector

The update equation:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k), \tag{1.3}$$

where μ is the step size or the convergence rate factor. μ should be less than the reciprocal of the maximum eigenvalue of the autocorrelation matrix of the input signal. The derivation of the LMS algorithm and its weight update equation are provided in details in Chapter 3.

1.3.2. Normalized LMS Algorithm

The LMS algorithm has two problems: the convergence rate is small and the performance is sensitive to the eigenvalue spread. In the standard LMS the adjustment applied is directly proportional to the tap-input vector, when it is large, the LMS algorithm suffers from a gradient noise amplification problem. To avoid of this problem, the normalized LMS was proposed in [49]. In deterministic systems [50], to avoid dividing by zero, a constant small value is added to the normalized least mean squares algorithm.

The usage of variable convergence rate μ_k in the update equation of NLMS helps to minimize the output error which, in turn, guarantees faster convergence compared to LMS algorithm. The drawback of such a variable convergence factor that it increases the misadjustment.

The desired output of the filter:

$$d(k) = \mathbf{h}^T \mathbf{x}(k). \tag{1.4}$$

The error of the estimation:

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k). \tag{1.5}$$

The update equation:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu_k}{\gamma + \mathbf{x}^T(k)\mathbf{x}(k)} e(k)\mathbf{x}(k), \tag{1.6}$$

where γ is a very small positive constant and $0 < \mu_k \leq 2$.

1.3.3. Quantized Error Algorithms

In echo cancellation or channel equalization adaptive filters supposed to run in high speed. This in turn, requires the minimization of the overall complexity of the used algorithm. Computational complexity of the LMS algorithm arises from the multiplications executed though the update process and from computing the output of adaptive filter. Applying quantization to the error signal of the LMS algorithm will generate the quantized-error algorithm for updating the filter weights.

1.3.3.1. Sign Error Algorithm

The simplest function for the quantization process is the sign (sgn) function defined by

$$sgn[b] = \begin{cases} \frac{b}{|b|} & , \ b \neq 0 \\ 0 & , \ b = 0 \end{cases}. \tag{1.7}$$

The sign-error algorithm utilizes the sign function as the error quantizer, where the weights vector updating is done by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu \, sgn[e(k)]\mathbf{x}(k). \tag{1.8}$$

The main drawback of the sign-error algorithm is its slow convergence. Fig. 1.7 shows an illustration of the sign-error algorithm for a delay line input $\mathbf{x}(k)$.

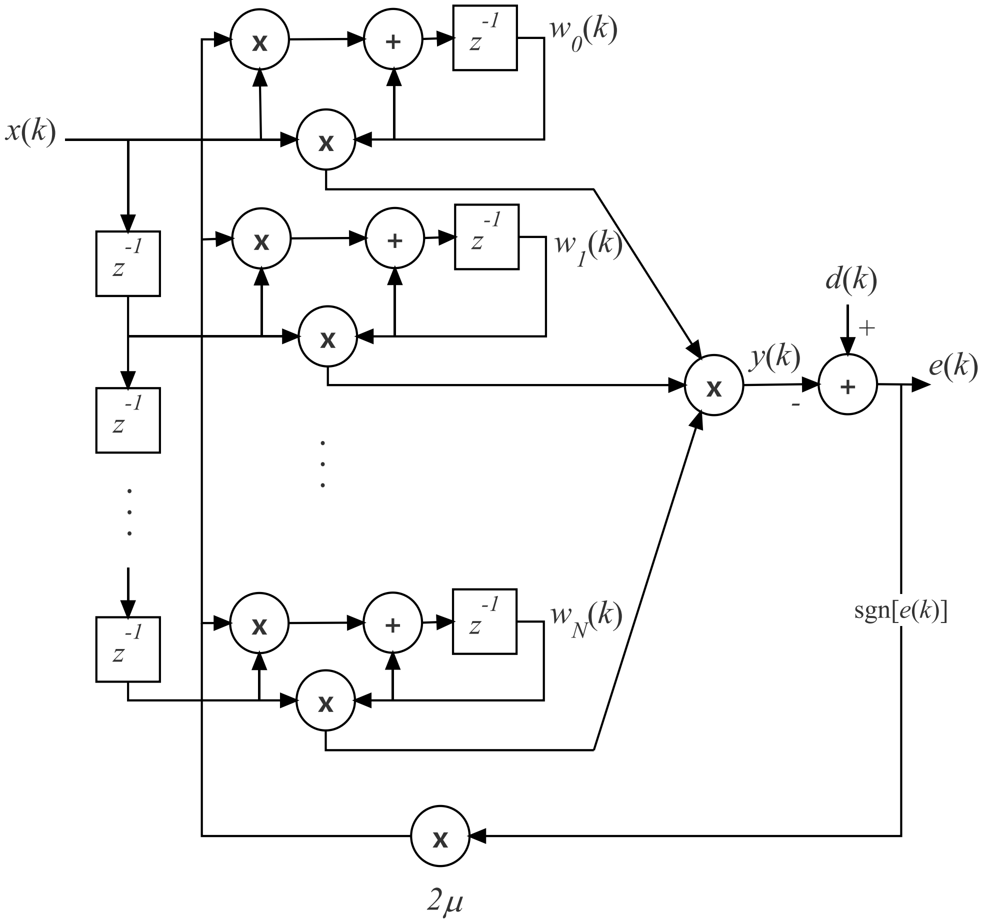


Figure 1.7 Sign-error adaptive filter [34]

1.3.3.2. Dual Sign Algorithm

The dual-sign algorithm performs large modifications to the weights vector when the modulus of the error signal is higher than a predefined level. The inspiration using the dual-sign algorithm is to speed up the slow convergence caused by replacing $e(k)$ by $sgn[e(k)]$ when $|e(k)|$ is value is high.

The quantization function for the dual-sign algorithm is given by

$$ds[b] = \begin{cases} \gamma\ sgn[b] & ,|b| > \rho \\ sgn[b] & ,|b| \leq \rho \end{cases}, \tag{1.9}$$

where $\gamma > 1$ is a power of two. The dual-sign algorithm uses this function as the error quantizer, and the weights updating is performed as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu \, ds[e(k)]\mathbf{x}(k). \tag{1.10}$$

1.3.3.3. Sign Data Algorithm

An alternative approach for reducing the computational complexity of the LMS algorithm is to apply quantization to the input signal $\mathbf{x}(k)$. One suggested quantization technique is to apply the sign function to the input signals. It is called the sign-data algorithm whose weight updating is performed as

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu \, e(k) sgn[\mathbf{x}(k)], \tag{1.11}$$

where the sign function is applied to each value in the input signal.

1.3.4. Newton-LMS Algorithm

The Newton-LMS algorithm incorporating estimates of the second-order statistics of the environment signals. The main idea of the algorithm is to get higher convergence rate than the LMS algorithm when the input signal is highly correlated. The drawback of the system is the increase of the computational complexity.

The desired output of the filter:

$$d(k) = \mathbf{h}^T \mathbf{x}(k). \tag{1.12}$$

The error of the estimation:

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k). \tag{1.13}$$

The update equation:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\widehat{\mathbf{R}}^{-1}(k)\mathbf{x}(k). \tag{1.14}$$

$$\widehat{\mathbf{R}}^{-1}(k) = \frac{1}{1-\alpha}\left[\widehat{\mathbf{R}}^{-1}(k-1) - \frac{\widehat{\mathbf{R}}^{-1}(k-1)\mathbf{x}(k)\mathbf{x}^T(k)\widehat{\mathbf{R}}^{-1}(k-1)}{\frac{1-\alpha}{\alpha} + \mathbf{x}^T(k)\widehat{\mathbf{R}}^{-1}(k-1)\mathbf{x}(k)}\right], \tag{1.15}$$

where $\widehat{\mathbf{R}}$ is the estimate of the autocorrelation matrix of the input-tap vector and α is a small value chosen as $0 < \alpha \leq 0.1$.

1.3.5. Transform-domain LMS algorithms

The transform-domain LMS algorithm is another approach used to enhance the performance of the LMS algorithm with highly correlated input signal. The objective behind this approach is to improve the conditioning number of the correlation matrix by transforming the input signal to be applied to the adaptive filter [34].

In the transform-domain LMS algorithm, the input signal $x(k)$ is transformed by using an orthonormal or unitary transform algorithm [51]-[53]. Fig. 1.8 shows an example of transform domain adaptive filter.

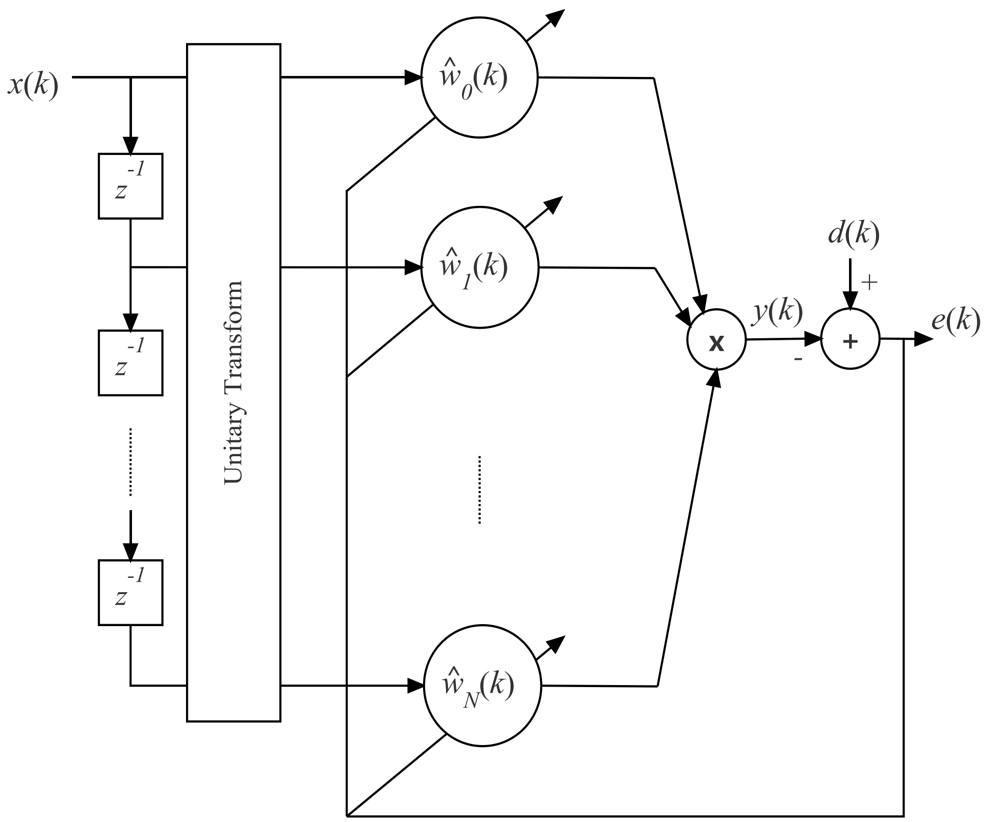


Figure 1.8 Transform domain adaptive filter

1.3.6. Recursive Least Square (RLS) Algorithm

The LMS algorithm is aiming to reduce the error between the output signal and the desired output of the filter [54]. In contrary, the RLS algorithm utilizes continuously updated prediction of the autocorrelation matrix of the input signal and the cross correlation vector at every iteration [1].

The recursive least squares adaptive filter (RLS) is recursively calculating the weights to reduce a weighted linear least squares cost function of the input signals. RLS algorithms have high convergence rate even when the eigenvalue spread of the input signal correlation matrix is large. These algorithms performances very well in time-varying environments [55].

The desired output of the filter:

$$d(k) = \mathbf{h}^T \mathbf{x}(k). \tag{1.16}$$

The error of the estimation:

$$e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1). \tag{1.17}$$

The update equation:

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \mathrm{e}(k)\mathbf{S}_{\boldsymbol{D}}(k)\mathbf{x}(k), \tag{1.18}$$

$$\mathbf{S}_D(k) = \frac{1}{\lambda}\left[\mathbf{S}_D(k-1) - \frac{\boldsymbol{\psi}(k)\boldsymbol{\psi}^{\boldsymbol{T}}(k)}{\lambda + \boldsymbol{\psi}^{\boldsymbol{T}}(k)\mathbf{x}(k)}\right], \tag{1.19}$$

$$\boldsymbol{\psi}(k) = \mathbf{S}_D(k-1)\mathbf{x}(k), \tag{1.20}$$

where λ is an exponential weighting factor usually in the range $0 \ll \lambda \leq 1$. $\mathbf{S}_D$ is the inverse of the deterministic autocorrelation matrix $\mathbf{R}_D$[34].

1.3.7. Mixed Norm Least Mean Square (MN-LMS) Algorithm

The mixed norm LMS algorithm proposed by Boukis *et al*. [56] is based on the least sum of exponentials (LSE). LSE provides a generalization of the class of weighted mixed norm algorithms. It is derived by minimizing a sum of error exponentials. The algorithm showed significant performance compared to that of the conventional LMS algorithm.

The cost function:

$$J_{\mathbf{w}}(k) = \Big(exp\big(e(k)\big) + exp(-e(k))\Big)^2. \tag{1.21}$$

The error of the estimation:

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k). \tag{1.22}$$

The update equation:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\, \mathbf{x}(k)\sinh\big(e(k)\big). \tag{1.23}$$

The derivation of the MN-LMS algorithm and its weight update equation is provided in details in Section 4.3.

1.3.8. Zero Attracting Least Mean Square (ZA-LMS) Algorithm

In the ZA-LMS, a new cost function $J_{\mathbf{w}}(k)$ is defined by adding l_1-norm penalty of the weight vector to the instantaneous square error.

The cost function is given as

$$J_{\mathbf{w}}(k) = \frac{1}{2}\, e^2(k) + \gamma\|\mathbf{w}(k)\|_1. \tag{1.24}$$

γ is a positive constant. Using the gradient descent update equation, the ZA-LMS filter update equation is defined as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\frac{\partial J_{\mathbf{w}}(k)}{\partial \mathbf{w}(k)}$$

$$= \mathbf{w}(k) - \rho\, sgn\,[\mathbf{w}(k)] + \mu\, e(k)\mathbf{x}(k), \tag{1.25}$$

where $\rho = \mu\gamma$ and $sgn[.]$ is the sign function defined in (1.4)

Comparing the ZA-LMS update equation (1.25) to the update equation of conventional LMS (1.3), the ZA-LMS algorithm has an additional term $-\rho\, sgn\,[\mathbf{w}(k)]$, which will attracts the vector coefficients to zero. It is called the zero attractor, whose power is managed by the shrinkage parameter ρ. Naturally, the zero attractor will increase convergence rate when most of coefficients of **w** are zero, i.e., the system is sparse.

1.3.9. Reweighted Zero Attracting Least Mean Square (RZA-LMS) Algorithm

This approach is inspired by the fact that the shrinkage parameter ρ in the ZA-LMS does not differentiate between zero and non-zero values. Since all the values are pushed to zero uniformly, its performance would decline for near non-sparse systems. Inspired by reweighting in compressive sampling [57], an approach to strengthen the zero attractor called the reweighted zero-attracting LMS (RZA-LMS) was proposed.

The cost function is given as

$$J_{\boldsymbol{w}}(k) = \frac{1}{2}e^2(k) + \gamma' \sum_{i=1}^{N} log\left(1 \; + \; \frac{|w_i|}{\varepsilon'}\right), \tag{1.26}$$

where γ' and ε' are positive constants. The log based penalty term $\sum_{i=1}^{N} log(1 \; + \; |w_i|/\varepsilon')$ is introduced as it has similar to the l_0-norm.

The weight vector can be updated as

$$w_{\boldsymbol{i}}(k+1) = w_i(k) - \rho\, \frac{sgn\,[w_i(k)]}{1+\varepsilon|w_i(k)|} + \mu e(k) x_i(k). \tag{1.27}$$

or equivalently, in the vector form

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \rho\, \frac{sgn\,[\mathbf{w}(k)]}{1+\varepsilon|\mathbf{w}(k)|} + \mu\, e(k)\mathbf{x}(k). \tag{1.28}$$

The RZA-LMS algorithm selectively shrinks elements with small magnitudes. The reweighted zero attractor has effect only on those elements for which magnitudes are close to $1/\varepsilon$; and there is small shrinkage employed on the elements whose $|w_i(k)| \gg 1/\varepsilon$. In this manner, the bias of the RZA-LMS algorithm is minimized.

1.4. SPARSE SYSTEMS

1.4.1. Definition of the sparse systems

In many engineering and mathematics problems, sparsity is a popular topic. In system identification and communication applications, the system may be in sparse nature such as acoustic echo cancellation [58] and network cancellation applications. Channel impulse response is frequently sparse due to high-speed data transmission, which will be dominated by a small number of high elements or taps [59][60]. A typical example of an finite impulse response FIR-based sparse channel is shown in Fig. 1.9. The channel dominant taps are at tap indices 3, 7, 11, and 15.

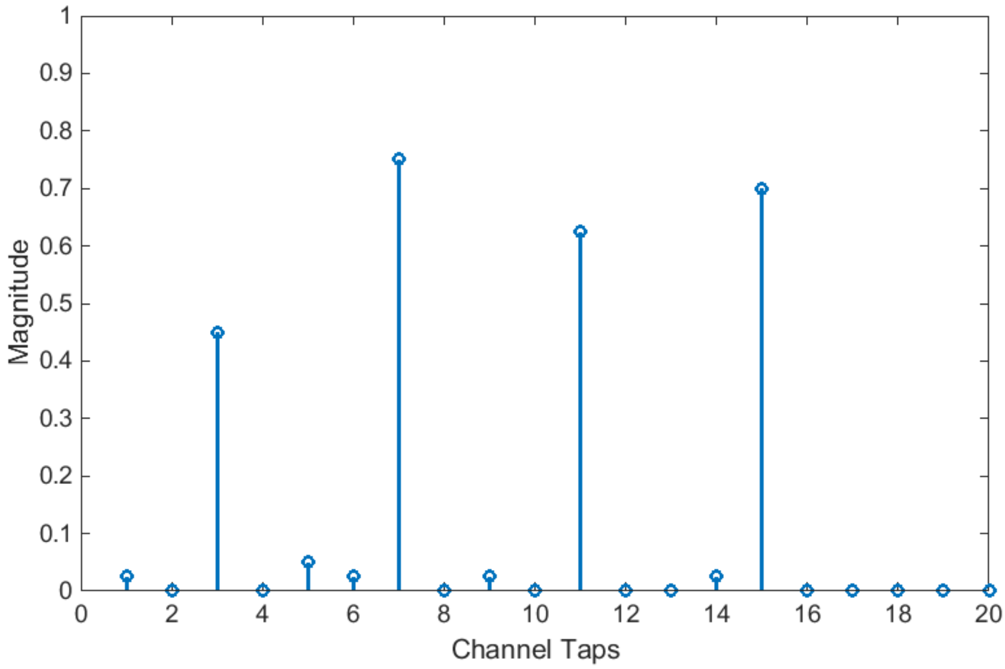


Figure 1.9 An example of an FIR based sparse channel with 4 dominant channel taps.

Sparse system identification is an essential requirement for fast converging adaptive filters in, for example, certain specific applications of echo cancellation. Recent developments based on proportionate update techniques have been used in network echo cancellation to overcome the problems of huge delays in IP network propagation for VoIP applications and the delay in the direct acoustic path.

1.4.2. Sparsity in Adaptive Filtering

In the last decade, many algorithms exploiting sparsity were based on applying a subset selection techniques during the filtering process, which was implemented via statistical detection of active taps [61]-[63] or sequential partial updating [64][65]. A general exploiting sparsity is done by updating a subset of the channel model or filter taps [66]. A derivation of the Exponentiated Gradient (EG) algorithm as an approximate gradient descent algorithm implemented in [67].

In [68] authors proposed a high convergence rate algorithm for sparse-tap adaptive filters to identify an unknown number of multiple dispersive areas by controlling simultaneously the weight values and tap-positions of the adaptive filter. A constrained area for new-tap positions is selected from equally sized subgroups of all possible tap-positions, and it hops from one subgroup to another to cover multiple dispersive regions. In [69], authors introduced an algorithm to exploit sparsity that is based on minimizing the cost function based on a time-dependent constraint on the norm of the filter update.

According to the research based on the least absolute shrinkage and selection operator (LASSO) [70], the sparsity is exploited by incorporating l-norm penalties to the general LMS algorithm cost function.

Many sparse LMS-type algorithms have been introduced lately to exploit sparsity [71][72]. In [71], zero-attracting LMS (ZA-LMS) has been proposed by Chen et al., using l_1-norm sparse penalty on mean square error (MSE). However, the ZA-LMS algorithm only exploits limited sparse information. Inspired by the reweight l_1-norm sparse signal recovery algorithm [73], Chen et al. proposed an improved version, which is called the reweighted zero-attracting LMS (RZA-LMS) algorithm [71], to further exploit the sparsity. Beside l_1-norm sparse penalty on LMS, Taheri and Vorobyov proposed l_p-norm LMS algorithm [74] to enhance the sparse channel estimation performance. Moreover, Gu et al. proposed l_0-norm LMS (l_0-LMS) algorithm in [75] and provided performance analysis in [76]. Later, Gui et al. applied the l_0-LMS algorithm on adaptive sparse channel estimation [72] to further enhance estimation performance. A method in [77] is introducing sparse representations in overcomplete transforms, based on minimization of the l_0-norm.

An approach is based on minimizing a regularized mean squared error criteria with sparsity being promoted by the regularizing term which consists of a diversity measure. new adaptive filters that directly exploit the sparsity of the filter are developed by using the scale mixture Gaussian distribution as the prior [78]

Other algorithms allocate proportional step sizes to different elements according to their magnitudes, such as the proportionate normalized LMS (PNLMS) algorithm and its variants [79]. In [80], the response of two independent adaptive filters are adaptively merged together and used to increase the improved proportionate normalized LMS (IPNLMS) algorithm robustness to sparse channels.

CHAPTER 2

PROBLEM STATEMENT & CONTRIBUTIONS

2.1. Overview

A filter is simply a device or process that removes some unwanted component or feature from a signal. Conventional filters have fixed parameters but, on the other hand, the adaptive filters have the power of being able to adjust their parameters iteratively or recursively depending on the environment.

In this chapter, we are emphasizing the contributions made in this dissertation to the field of adaptive filters, both in one-dimensional and two-dimensional aspects. We also outlined the materials and presented in each chapter of the dissertation briefly.

2.2. Contributions of the Dissertation

In this dissertation new different algorithms for improving the performance of the one-dimensional and two-dimensional least mean square (LMS) algorithm are proposed.

The first contribution of this dissertation is providing the derivation of the convergence analysis of the mixed-norm LMS algorithm.

In the second contribution, new algorithms based on mixed-norm LMS algorithm are proposed. The first algorithm exploits the sparsity of the system by adding l_1-norm penalty term to the cost function of the MN-LMS algorithm. This term enables us to attract the zero and/or near-to-zero filter coefficients to the zero value in a faster way. However, when the system is near or exactly non-sparse, the algorithm almost fails.

The third contribution was proposed to overcome this limitation of first algorithm when the system is near or exactly non-sparse, the second algorithm that uses an approximation of l_0-

norm penalty term in the cost function of the MN-LMS algorithm was proposed. This provides high performance even with completely non-sparse systems.

The fourth contribution was to improve the 2D-LMS algorithm performance by proposed a new two-dimensional (2D) zero-attracting least mean square (ZA-LMS) adaptive filter by imposing a sparsity aware l_1-norm penalty term in the cost function of the 2D-LMS algorithm. The convergence analysis of the 2D ZA-LMS algorithm are presented and stability criterion is derived.

In general, the objectives of the research conducted in this dissertation include:

1. Review of the adaptive filters and applications.
2. Review of the LMS algorithm and its variants.
3. Investigation of the sparse systems in adaptive filters.
4. Study of the zero attracting and reweighted zero attracting algorithms and their effect on the system performance.
5. Comprehensive and collective convergence analysis of the recently proposed mixed-norm LMS algorithm.
6. Implementation of the proposed algorithms for system identification and acoustic room problem.
7. Practical realization of the proposed 2D ZA-LMS algorithm for image de-noising problem.
8. Convergence analysis of the proposed 2D ZA-LMS algorithm.

2.3. Outlines of the Dissertation

Chapter one is intended to give general introduction about the adaptive filters. This chapter lists the main applications of adaptive filters. Sparsity systems are explained in the chapter together with their relations with adaptive filters.

Chapter two is talking about the problem statement of this dissertation. The contribution and the outlines of the dissertation are explained briefly in this chapter.

Chapter three is explaining the foundation algorithms of the proposed approaches. Least mean square (LMS) algorithm derivation is explained in details. Also, the main properties related to the convergence behavior of the LMS algorithm in a stationary environment are described. In this chapter, the convergence analysis of the mixed-norm least mean square (MN-LMS) algorithm is derived. MN-LMS algorithm shows outstanding performance compared to that of the conventional LMS algorithm. Update equations of the Zero attractive LMS and reweighted zero attractive LMS are listed in this chapter. At the end of the chapter, the two-dimensional LMS (2D-LMS) adaptive filter derivation is provided.

Chapter four explains the two proposed algorithms based on MN-LMS algorithm. Based on the fact that MN-LMS algorithm is performing better than LMS algorithm, two algorithms that exploit the sparsity of the system have been derived by adding l_1-norm penalty or an approximation of l_0-norm penalty to the cost function of the MN-LMS algorithm. Experiments with different settings are provided at the end of the chapter.

In Chapter five, the derivation of the proposed two-dimensional ZA-LMS algorithm is provided. Different data reuse patterns are explained. Experiments are conducted to compare the proposed algorithm to the conventional 2D-LMS algorithm.

At the end of the dissertation, conclusions and discussions on the introduced algorithms and simulation results are presented in Chapter six. Furthermore, possible future work to modify/ improve the proposed algorithms or other conventional algorithms is included and discussed.

CHAPTER 3

BASICS OF THE RELATED ADAPTIVE ALGORITHMS

3.1.Overview

This chapter gives the mathematical explanation of the related adaptive algorithms. These algorithms are the foundation for the proposed algorithms in Chapter 4 and Chapter 5. The steps of these algorithms together with some of their convergence analysis are given in the following sections.

3.2.Least Mean Square (LMS) Algorithm

The least mean square algorithm is a well-known adaptive algorithm in signal processing [5]. The optimal (Wiener) solution is given as

$$\boldsymbol{w_o} = \mathbf{R}^{-\mathbf{1}}\mathbf{p}, \tag{3.1}$$

where the autocorrelation matrix $\mathbf{R} = E[\mathbf{x}(k)\mathbf{x}^T(k)]$ and the cross correlation $\mathbf{p} = E[d(k)\mathbf{x}(k)]$ assuming that $d(k)$ and $\mathbf{x}(k)$ are jointly wide-sense stationary.

If the estimates of **R**, denoted by $\widehat{\mathbf{R}}(k)$, and of **p**, denoted by $\widehat{\mathbf{p}}(k)$, are available and good, a steepest-descent-based algorithm can be utilized to find the Wiener solution of (3.1) as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu\hat{\mathbf{g}}_{\boldsymbol{w}}(k)$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\left(\widehat{\mathbf{p}}(k) - \widehat{\mathbf{R}}(k)\mathbf{w}(k)\right), \tag{3.2}$$

for $k = 0,1,2, \ldots$, where $\hat{\mathbf{g}}_{\boldsymbol{w}}(k)$ is an estimate of the gradient vector of the cost function with respect to the filter weight vector.

A solution to estimate the gradient vector is by using instantaneous estimates for **R** and **p** as

$$\hat{\mathbf{R}}(k) = \mathbf{x}(k)\mathbf{x}^T(k),$$

$$\hat{\mathbf{p}}(k) = d(k)\mathbf{x}^T(k). \tag{3.3}$$

The gradient estimate will be given as

$$\begin{aligned}\hat{\mathbf{g}}_w(k) &= -2d(k)\mathbf{x}(k) + 2\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k) \\ &= 2\mathbf{x}(k)(-d(k) + \mathbf{x}^T(k)\mathbf{w}(k)) \\ &= -2e(k)\mathbf{x}(k).\end{aligned} \tag{3.4}$$

Replacing the cost function by the instantaneous square error $e^2(k)$, instead of the MSE, the above gradient estimate will become the true gradient vector since

$$\begin{aligned}\frac{\partial e^2(k)}{\partial \mathbf{w}} &= \begin{bmatrix} 2e(k)\dfrac{\partial e(k)}{\partial w_0(k)} \\ 2e(k)\dfrac{\partial e(k)}{\partial w_1(k)} \\ \vdots \\ 2e(k)\dfrac{\partial e(k)}{\partial w_{N-1}(k)} \end{bmatrix} \\ &= -2e(k)\mathbf{x}(k) \\ &= \hat{\mathbf{g}}_w(k).\end{aligned} \tag{3.5}$$

The gradient-based algorithm is known as the least mean square (LMS) algorithm, whose weight updating equation is

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu e(k)\mathbf{x}(k), \tag{3.6}$$

where the convergence rate μ should be selected carefully to ensure convergence. Fig. 3.1 shows the block diagram of the LMS algorithm for an input $x(k)$. One iteration of the LMS requires $N+2$ multiplications for the filter weight updating and $N+1$ multiplications for the error generation. The weight of the adaptive filter can be initialized with zeros.

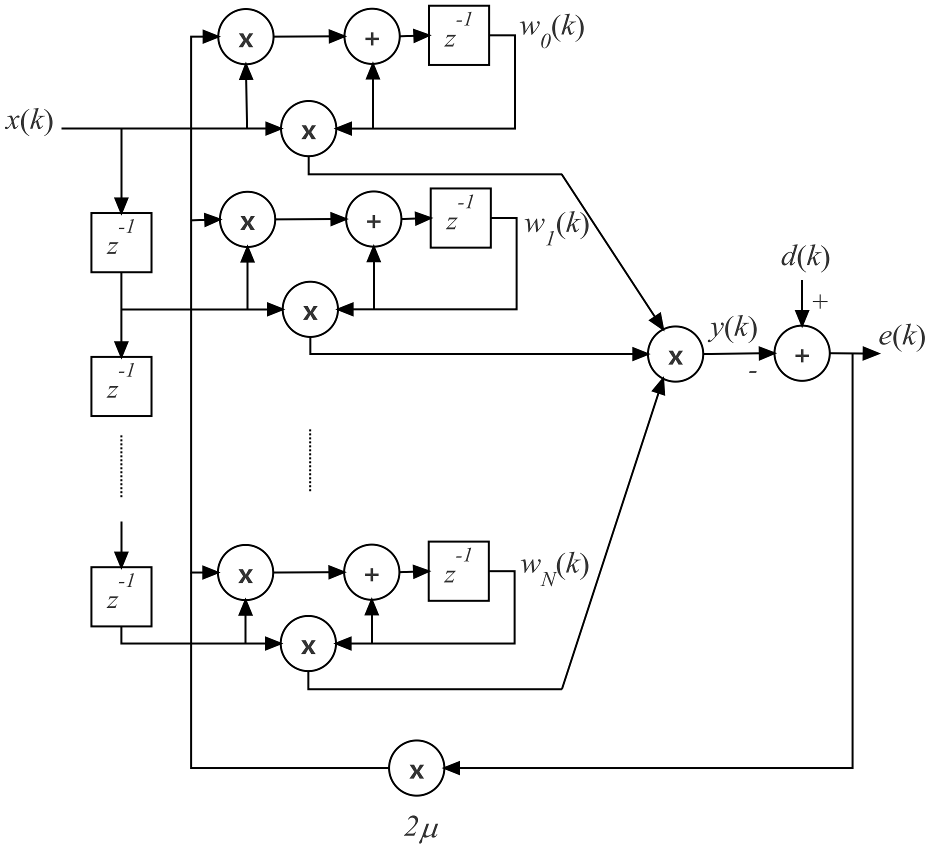


Figure 3.1 Block diagram of the LMS Adaptive FIR Filter

3.2.1. Some Properties of the LMS Algorithm

3.2.1.1. Gradient Behavior

The ideal gradient direction required to perform a search on the MSE surface for the optimum coefficient vector solution is

$$\mathbf{g}_{\boldsymbol{w}}(k) = 2E[\mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w}(k) - 2E[d(k)\mathbf{x}(k)]$$

$$= 2[\mathbf{R}\mathbf{w}(k) - \mathbf{p}]. \tag{3.7}$$

In the LMS algorithm, instantaneous estimates of **R** and **p** are used to determine the search direction, i.e.

$$\hat{\mathbf{g}}_{\boldsymbol{w}}(k) = 2[\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{w}(k) - d(k)\mathbf{x}(k)]. \tag{3.8}$$

As expected, the direction determined by (3.8) is totally different from that of (3.7). Therefore, by using the more computationally attractive gradient direction of the LMS algorithm, the convergence behavior is different from the steepest-descent algorithm. On average, it can be said that the LMS gradient direction tends to approach the ideal gradient direction since, for a fixed weight vector $\mathbf{w}$,

$$E[\hat{\mathbf{g}}_{\boldsymbol{w}}(k)\,] = 2\{E[\mathbf{x}(k)\mathbf{x}^T(k)]\mathbf{w} - E[d(k)\mathbf{x}(k)]\}$$

$$= \mathbf{g}_{\boldsymbol{w}}. \tag{3.9}$$

Hence, vector $\hat{\mathbf{g}}_{\boldsymbol{w}}(k)$ can be considered as an unbiased instantaneous estimate of $\mathbf{g}_{\boldsymbol{w}}$,. In an ergodic environment, if, for a fixed $\mathbf{w}$ vector, $\hat{\mathbf{g}}_{\boldsymbol{w}}(k)$ is calculated for a large number of inputs and reference signals, the average direction tends to $\mathbf{g}_{\boldsymbol{w}}$, i.e.,

$$\lim_{\mathrm{N}\to\infty} \frac{1}{N} \sum_{i=1}^{N} \hat{\mathbf{g}}_{\boldsymbol{w}}(k+i) \;\to\; \mathbf{g}_{\boldsymbol{w}}. \tag{3.10}$$

3.2.1.2. Convergence Behavior of the Weights Vector

Assume that an unknown FIR filter with weight vector $\mathbf{w}_{\boldsymbol{o}}$ is being identified by an adaptive FIR filter, using the LMS algorithm. Measurement white noise $v(k)$ with zero mean and variance σ^2 is added to the output of the unknown system. The error in the adaptive-filter weights as related to the ideal weights vector $\mathbf{w}_{\boldsymbol{o}}$, in each iteration, is described by the N length vector

$$\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_{\boldsymbol{o}}. \tag{3.11}$$

Here, the LMS algorithm can also be described as

$$\Delta\mathbf{w}(k+1) = \Delta\mathbf{w}(k) + 2\mu\, e(k)\mathbf{x}(k)$$

$$= \Delta\mathbf{w}(k) + 2\mu\,\mathbf{x}(k)\,[\mathbf{x}^T(k)\mathbf{w}_{\boldsymbol{o}} + n(k) - \mathbf{x}^T(k)\mathbf{w}(k)]$$

$$= \Delta\mathbf{w}(k) + 2\mu\,\mathbf{x}(k)\,[e_o(k) - \mathbf{x}^T(k)\Delta\mathbf{w}(k)]$$

$$= [\mathbf{I} - 2\mu\,\mathbf{x}(k)\mathbf{x}^T(k)]\Delta\mathbf{w}(k) + 2\mu\, e_o(k)\mathbf{x}(k), \tag{3.12}$$

where $e_o(k)$ is the optimum output error given as

$$\begin{aligned} e_o(k) &= d(k) - \mathbf{w}_o^T\mathbf{x}(k) \\ &= \mathbf{w}_o^T\mathbf{x}(k) + v(k) - \mathbf{w}_o^T\mathbf{x}(k) \\ &= v(k). \end{aligned} \tag{3.13}$$

The expected error in the weight vector is then given as

$$E[\Delta\mathbf{w}(k+1)] = E\{[\mathbf{I} - 2\mu\,\mathbf{x}(k)\mathbf{x}^T(k)]\Delta\mathbf{w}(k)\} + 2\mu\, e_o(k)\mathbf{x}(k). \tag{3.14}$$

If it is assumed that the elements of $\mathbf{x}(k)$ are statistically independent of the elements of $\Delta\mathbf{w}(k)$ and $e_o(k)$, (3.14) can be simplified as

$$\begin{aligned} E[\Delta\mathbf{w}(k+1)] &= \{\mathbf{I} - 2\mu\, E[\mathbf{x}(k)\mathbf{x}^T(k)]\}E[\Delta\mathbf{w}(k)] \\ &= \{\mathbf{I} - 2\mu\mathbf{R}\}E[\Delta\mathbf{w}(k)]. \end{aligned} \tag{3.15}$$

The first assumption is justified if we consider that the deviation in the parameters is reliant on previous input-tap vectors only, whereas in the second assumption we also assumed that the error signal at the optimal solution is orthogonal to the coefficients of the input-tap vector. The expression in (3.15) becomes

$$E[\Delta\mathbf{w}(k+1)] = (\mathbf{I} - 2\mu\mathbf{R})^{k+1}E[\Delta\mathbf{w}(0)]. \tag{3.16}$$

Premultiplying (3.16) by $\mathbf{Q}^T$, where $\mathbf{Q}$ is the unitary matrix that diagonalizes $\mathbf{R}$ through a similarity transformation, leads to

$$\begin{aligned} E[\mathbf{Q}^T\Delta\mathbf{w}(k+1)] &= (\mathbf{I} - 2\mu\,\mathbf{Q}^T\mathbf{R}\mathbf{Q})^{k+1}E[\mathbf{Q}^T\Delta\mathbf{w}(0)] \\ &= E[\Delta\mathbf{w}'(k+1)] \\ &= (\mathbf{I} - 2\mu\,\boldsymbol{\Lambda})E[\Delta\mathbf{w}'(k)] \\ &= \begin{bmatrix} 1-2\mu\lambda_0 & 0 & \cdots & 0 \\ 0 & 1-2\mu\lambda_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1-2\mu\lambda_{N-1} \end{bmatrix} E[\Delta\mathbf{w}'(k)], \end{aligned} \tag{3.17}$$

where $\mathbf{\Lambda}$ is the matrix of eigenvalues of $\mathbf{R}$ and $\Delta\mathbf{w}'(k+1) = \mathbf{Q}^T\Delta\mathbf{w}(k+1)$ is the rotated-weight error vector. The used rotation returned an equation where the driving matrix is diagonal, making it easier to analyze the dynamic behavior of the equation. Alternatively, the equation in (3.17) can be written as

$$E[\Delta\mathbf{w}'(k+1)] = (\mathbf{I} - 2\mu\,\mathbf{\Lambda})^{k+1}E[\Delta\mathbf{w}'(k)]$$

$$= \begin{bmatrix} (1-2\mu\lambda_0)^{k+1} & 0 & \cdots & 0 \\ 0 & (1-2\mu\lambda_1)^{k+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (1-2\mu\lambda_{N-1})^{k+1} \end{bmatrix} E[\Delta\mathbf{w}'(0)], \quad (3.18)$$

This equation shows that to ensure convergence of the weights in the mean, the convergence rate or the step-size of the LMS algorithm μ must be chosen as

$$0 < \mu < \frac{1}{\lambda_{max}}, \quad (3.19)$$

where λ_{max} represents the largest eigenvalue of $\mathbf{R}$. Values of μ in this range ensures that all elements of the diagonal matrix in (3.18) reach to zero as $k \to \infty$, since $-1 < (1 - 2\mu\lambda_i) < 1$, for $i = 0, 1, \ldots, N-1$. As a result $E[\Delta\mathbf{w}(k+1)]$ reaches to zero for large k. The selection of μ as above explained guarantees that the mean value of the coefficient vector approaches the optimum coefficient vector $\mathbf{w}_o$. It is worth mentioning that, if the matrix $\mathbf{R}$ has a large eigenvalue spread, it is recommended to select a value for μ much smaller than the upper bound. As a result, the convergence speed of the weights will be mainly dependent on the value of the smallest eigenvalue, responsible for the slowest mode in (3.18). The main assumption for the above analysis is the so-called independence theory, which assumes all vectors $\mathbf{x}(i)$ for $i = 0, 1, \ldots, k,$ statistically independent. This assumption allowed us to assume $\Delta\mathbf{w}(k)$ independent of $\mathbf{x}(k)\mathbf{x}^T(k)$ in equation (3.14). Such an assumption, regardless of not being strictly valid especially when $\mathbf{x}(k)$ contains the elements of a delay line, leads to theoretical results consistent with the experimental results.

3.2 Zero Attracting Least Mean Square (ZA-LMS) Algorithm

Consider a linear system with its input signal $\mathbf{x}(k)$ and desired output $d(k)$ related by

$$d(k) = \mathbf{h}^T\mathbf{x}(k) + v(k), \tag{3.20}$$

where $\mathbf{h} = [h_0, \dots, h_{N-1}]^T$ is the unknown system with length N, $\mathbf{x}(k) = [x(k), \dots, x(k - N + 1)]^T$ is the system input vector, and $v(k)$ is the additive noise and independent with $\mathbf{x}(k)$. To adaptively estimate the unknown system coefficients vector $\mathbf{h}$ using the system input vector $\mathbf{x}(k)$ and output signal $d(k)$, the ZA-LMS algorithm [71] updates its estimate by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu\, e(k)\mathbf{x}(k) - \rho\, sgn[\mathbf{w}(k)], \tag{3.21}$$

where μ is the step size, ρ is the zero-attractor controller, $sgn[\cdot]$ is the sign function, the vector $\mathbf{w}(k) = [w_0(k), \dots, w_{N-1}(k)]^T$ denotes filter weights, and $e(k)$ is the instantaneous estimation error defined as

$$e(k) = d(k) - \mathbf{w}^T(k)\mathbf{x}(k) \tag{3.22}$$

3.2.1. Convergence Analysis of the ZA-LMS Algorithm

The mean square convergence analysis of the ZA-LMS algorithm [81] is shown in this section. Assuming an i.i.d. zero-mean Gaussian input signal $\mathbf{x}(k)$ and a zero-mean white noise $v(k)$, the second order stability condition was analyzed, the steady-state mean square deviation (MSD) based on the system sparsity, system response length, and filter parameters was derived and a criterion on filter parameter selection for the ZA-LMS to improve the standard LMS algorithm was proposed.

The filter misalignment vector can be defined as

$$\boldsymbol{\delta}(k) = \mathbf{w}(k) - \mathbf{h}. \tag{3.23}$$

Defining the mean $\boldsymbol{\varepsilon}(k)$ and auto-covariance matrix $\mathbf{S}(k)$ of $\boldsymbol{\delta}(k)$, respectively, as

$$\boldsymbol{\varepsilon}(k) = E[\boldsymbol{\delta}(k)], \tag{3.24}$$

$$\mathbf{S}(k) = E[\boldsymbol{z}(k)\boldsymbol{z}^T(k)], \tag{3.25}$$

where $E[\cdot]$ denotes the statistical expectation and $\boldsymbol{z}(k)$ is the zero-mean misalignment vector.

$$\boldsymbol{z}(k) = \boldsymbol{\delta}(k) - E[\boldsymbol{\delta}(k)]. \tag{3.26}$$

The MSD is used for performance evaluation. The instantaneous MSD is defined as

$$\mathrm{MSD}_{ins}(k) = E[\|\boldsymbol{\delta}(k)\|_2^2] = \sum_{i=0}^{N-1} \Gamma_i(k), \tag{3.27}$$

where $\Gamma_i(k)$ denotes the i^{th}-element MSD and is defined based on the i^{th} element of $\boldsymbol{\delta}(k)$ as

$$\Gamma_i(k) = E[\delta_i^2(k)] = S_{ii}(k) + \varepsilon_i^2(k), \quad i = 0, \dots, N-1. \tag{3.28}$$

$S_{ii}(k)$ is the i^{th} diagonal element of $\mathbf{S}(k)$, and $\varepsilon_i(k)$ is the i^{th} element of $\boldsymbol{\varepsilon}(k)$.

3.2.2. Mean Square Convergence

Combining (3.20), (3.21), (3.22) and (3.23), we obtain

$$\boldsymbol{\delta}(k+1) = \mathbf{A}(k)\boldsymbol{\delta}(k) + \mu\mathbf{x}(k)v(k) - \rho sgn[\mathbf{w}(k)], \tag{3.29}$$

where

$$\mathbf{A}(k) = \mathbf{I}_N - \mu\mathbf{x}(k)\mathbf{x}^T(k), \tag{3.30}$$

and $\mathbf{I}_N$ denotes the $N \times N$ identity matrix. The expectation of (3.29) will result in

$$\boldsymbol{\varepsilon}(k+1) = (1 - \mu\sigma_x^2)\boldsymbol{\varepsilon}(k) - \rho E\big[sgn[\mathbf{w}(k)]\big], \tag{3.31}$$

where σ_x^2 denotes the variance of $\mathbf{x}(k)$. Combining (3.26), (3.29) and (3.29), we obtain

$$\boldsymbol{z}(k+1) = \mathbf{A}(k)\boldsymbol{z}(k) + \mu\mathbf{B}(k)\boldsymbol{\varepsilon}(k) + \mu\mathbf{x}(k)v(k) + \rho\boldsymbol{p}(k), \tag{3.32}$$

where

$$\mathbf{B}(k) = E[\mathbf{x}(k)\mathbf{x}^T(k)] - \mathbf{x}(k)\mathbf{x}^T(k), \tag{3.33}$$

$$\boldsymbol{p}(k) = E[sgn[\mathbf{w}(k)]] - sgn[\mathbf{w}(k)]. \tag{3.34}$$

It is straightforward to verify that both $\mathbf{B}(k)$ and $\boldsymbol{p}(k)$ are of zero mean. Considering the independence assumption, i.e., $\boldsymbol{\delta}(k)$, $\mathbf{x}(k)$, and $v(k)$ are mutually independent, substituting (3.32) into (3.25) leads to

$$\mathbf{S}(k+1) = E[\mathbf{A}(k)\boldsymbol{z}(k)\boldsymbol{z}^T(k)\mathbf{A}^T(k)] + \mu^2 E[\mathbf{B}(k)\boldsymbol{\varepsilon}(k)\boldsymbol{\varepsilon}^T(k)\mathbf{B}^T(k)] + \mu^2\sigma_x^2\sigma_v^2\mathbf{I}_N + \rho^2 E[\boldsymbol{p}(k)\boldsymbol{p}^T(k)] + \rho E[\mathbf{A}(k)\boldsymbol{z}(k)\boldsymbol{p}^T(k)] + \rho E[\boldsymbol{p}(k)\boldsymbol{z}^T(k)\mathbf{A}^T(k)], \tag{3.35}$$

where σ_v^2 is the variance of $v(k)$. Using the facts that the fourth-order moment of a Gaussian variable is equal to three times the variance square and that $\mathbf{S}(k)$ is symmetric, we get

$$E[\mathbf{A}(k)\boldsymbol{z}(k)\boldsymbol{z}^T(k)\mathbf{A}^T(k)] = (1 - 2\mu\sigma_x^2 + 2\mu^2\sigma_x^4)\mathbf{S}(k) + \mu^2\sigma_x^4 tr[\mathbf{S}(k)]\mathbf{I}_N, \tag{3.36}$$

$$E[\mathbf{B}(k)\boldsymbol{\varepsilon}(k)\boldsymbol{\varepsilon}^T(k)\mathbf{B}^T(k)] = \sigma_x^4\{\boldsymbol{\varepsilon}(k)\boldsymbol{\varepsilon}^T(k) + tr[\boldsymbol{\varepsilon}(k)\boldsymbol{\varepsilon}^T(k)]\}\mathbf{I}_N, \tag{3.37}$$

where $tr[\cdot]$ represents the trace of a matrix. Also, with (3.23), (3.26) and (3.34) we obtain

$$E[\mathbf{A}(k)\boldsymbol{z}(k)\boldsymbol{p}^T(k)] = E[\boldsymbol{p}(k)\boldsymbol{z}^T(k)\mathbf{A}^T(k)] = (1 - 2\sigma_x^2)E[\mathbf{w}(k)\boldsymbol{p}^T(k)]. \tag{3.38}$$

Combining (3.35) - (3.38) we obtain

$$tr[\mathbf{S}(k+1)] = [1 - 2\mu\sigma_x^2 + (N+2)\mu^2\sigma_x^4]tr[\mathbf{S}(k)] + (N+1)\mu^2\sigma_x^4\boldsymbol{\varepsilon}^T(k)\boldsymbol{\varepsilon}(k) + N\mu^2\sigma_v^2\sigma_x^2 + \rho^2 E[\boldsymbol{p}^T(k)\boldsymbol{p}(k)] + 2\rho(1 - \mu\sigma_x^2)E[\mathbf{w}(k)\boldsymbol{p}^T(k)]. \tag{3.39}$$

$\boldsymbol{p}(k)$ is bounded and thus $E[\mathbf{w}^T(k)\boldsymbol{p}(k)]$ converges. Then, the adaptive filter is stable if, and only if,

$$|1 - 2\mu\sigma_x^2 + (N+2)\mu^2\sigma_x^4| < 1, \tag{3.40}$$

which can be simplified to

$$0 < \mu < \frac{2}{(N+2)\sigma_x^2}. \tag{3.41}$$

This shows that the ZA-LMS algorithm has the same stability constraint as the conventional LMS algorithm.

3.3 Reweighted Zero Attracting Least Mean Square (RZA-LMS) Algorithm

The reweighted l_0-norm minimization for sparse signal recovery has been studied in [82]. The resulting update equation is

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \rho\,\frac{sgn\,[\mathbf{w}(k)]}{1+\varepsilon|\mathbf{w}(k-1)|} + \mu e(k)\mathbf{x}(k), \tag{3.42}$$

where $\rho = \mu\gamma$ and $sgn[.]$ is the sign function defined in (1.7). The absolute value operator as well as the $sgn[.]$ and the division operator in the last term of (3.42) are all component-wise. Therefore, the i^{th} element of $\frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|}$ is $\frac{sgn\,[w_i(k)]}{\varepsilon+|w_i(k-1)|}$. Therefore, the reweighted l_0-norm penalized LMS algorithm is definite to converge to the global minimum under certain conditions.

3.3.1. Convergence Analysis

The update equation for the coefficient error vector of the l_0-norm penalized LMS $\boldsymbol{\delta}(k)$ can be written as

$$\boldsymbol{\delta}(k+1) = \boldsymbol{\delta}(k) - \mu\mathbf{x}(k)\mathbf{x}^T(k)\boldsymbol{\delta}(k) + \mu v(k)\mathbf{x}(k) - \rho\,\frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|}. \tag{3.43}$$

From (3.43) we can derive the evolution equation for $E[\boldsymbol{\delta}(k)]$. Since $v(k)$ and $\mathbf{x}(k)$ are independent and $n(k)$ is assumed to have zero mean, there is $E[\mu n(k)\mathbf{x}(k)] = 0$. Then the evolution equation is

$$E[\boldsymbol{\delta}(k+1)] = (\mathbf{I} - \mu\mathbf{R})E[\boldsymbol{\delta}(k)] - \rho\, E\left[\frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|}\right]. \tag{3.44}$$

It is easy to see that the term $\frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|}$ is bounded below and above element-wise as

$$-\frac{1}{\varepsilon} \leq \frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|} \leq \frac{1}{\varepsilon}, \tag{3.45}$$

where 1 is the vector with all of its entries set to one. Indeed, 1 is always less than or equal to $sgn\,[\mathbf{w}(k)]$, while 1 is always larger than or equal to $sgn\,[\mathbf{w}(k)]$. The equation of $\rho\, E\left[\frac{sgn\,[\mathbf{w}(k)]}{\varepsilon+|\mathbf{w}(k-1)|}\right]$ is bounded. After taking expectation,

$$\left[E[\mathbf{c}(k+N)]\right]_i = (1-\mu\lambda_i)^M\left[E[\mathbf{c}(k)]\right]_i - \sum_{m=0}^{N-1}(1-\mu\lambda_i)^m[\mathbf{w}'(k+N-m-1)]_i \tag{3.46}$$

Since the largest eigenvalue of $\mathbf{I} - \mu\mathbf{R}$ is smaller than 1, then all the diagonal elements $(1-\mu\lambda_i)$ are smaller than 1. Also, note that the i^{th} entry of the vector $\mathbf{w}'(k)$ is bounded. The other term on the right-hand side of (3.46) approaches zero as $N \to \infty$. As a result, $\left[E[\mathbf{c}(k+N)]\right]_i$ as well as the whole vector $E[\mathbf{c}(k+N)]$ are bounded when $N \to \infty$. Given that $E[\mathbf{c}(k)]$ is a rotated version of $E[\boldsymbol{\delta}(k)]$, then the coefficient error vector $\boldsymbol{\delta}(k)$ is also bounded in the mean sense. Therefore, given the largest eigenvalue of $\mathbf{I} - \mu\mathbf{R}$ is smaller than 1, then $E[\boldsymbol{\delta}(k)]$ is bounded as $k \to \infty$.

$$E[\boldsymbol{\delta}(k+1)] = (\mathbf{I} - \mu\mathbf{R})E[\boldsymbol{\delta}(k)]. \tag{3.47}$$

3.4 Two-Dimensional LMS (2D-LMS) Algorithm

The two-dimensional LMS (2D-LMS) adaptive filter operation is illustrated in Fig. 3.2 where the error e_k is used to update the filter weights [83]. The LMS algorithm is based on the steepest decent method. In this method, the update of next weight matrix is based on the present weight matrix and the negative gradient of the error power. This method will be used to find an approximate solution for the weights. According to this technique, the 2-dimensional weight adjustment algorithm is given in matrix form as follows

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \mu\,\widehat{\mathbf{G}}_k, \tag{3.48}$$

where $\mathbf{W}_{k+1}$ is the weight matrix after updating, $\mathbf{W}_k$ is the weight matrix before updating and $\widehat{\mathbf{G}}_k$ is estimate for the 2D instantaneous gradient of $E\{e_k^2\}$ with respect to $\mathbf{W}_k$.

$$\widehat{\mathbf{G}}_k = \frac{\partial E\{e_k^2\}}{\partial \mathbf{W}_k} = \frac{\partial E\left\{\left(\mathbf{D}(m,n) - {\mathbf{W}_k}^T \mathbf{X}_k(m-r,n-c)\right)^2\right\}}{\partial \mathbf{W}_k}. \tag{3.49}$$

Equation (3.49) which defines the true two-dimensional instantaneous gradient of the mean squared error during the k^{th} iteration, can be written in an explicit matrix form as

$$\widehat{\mathbf{G}}_k = \begin{bmatrix} \frac{\partial\{E[e_k^2]\}}{\partial w_k(0,0)} & \cdots & \frac{\partial\{E[e_k^2]\}}{\partial w_k(0,N-1)} \\ & \vdots & \\ \cdots & & \cdots \\ & \vdots & \\ \frac{\partial\{E[e_k^2]\}}{\partial w_k(N-1,0)} & \cdots & \frac{\partial\{E[e_k^2]\}}{\partial w_k(N-1,N-1)} \end{bmatrix} \tag{3.50}$$

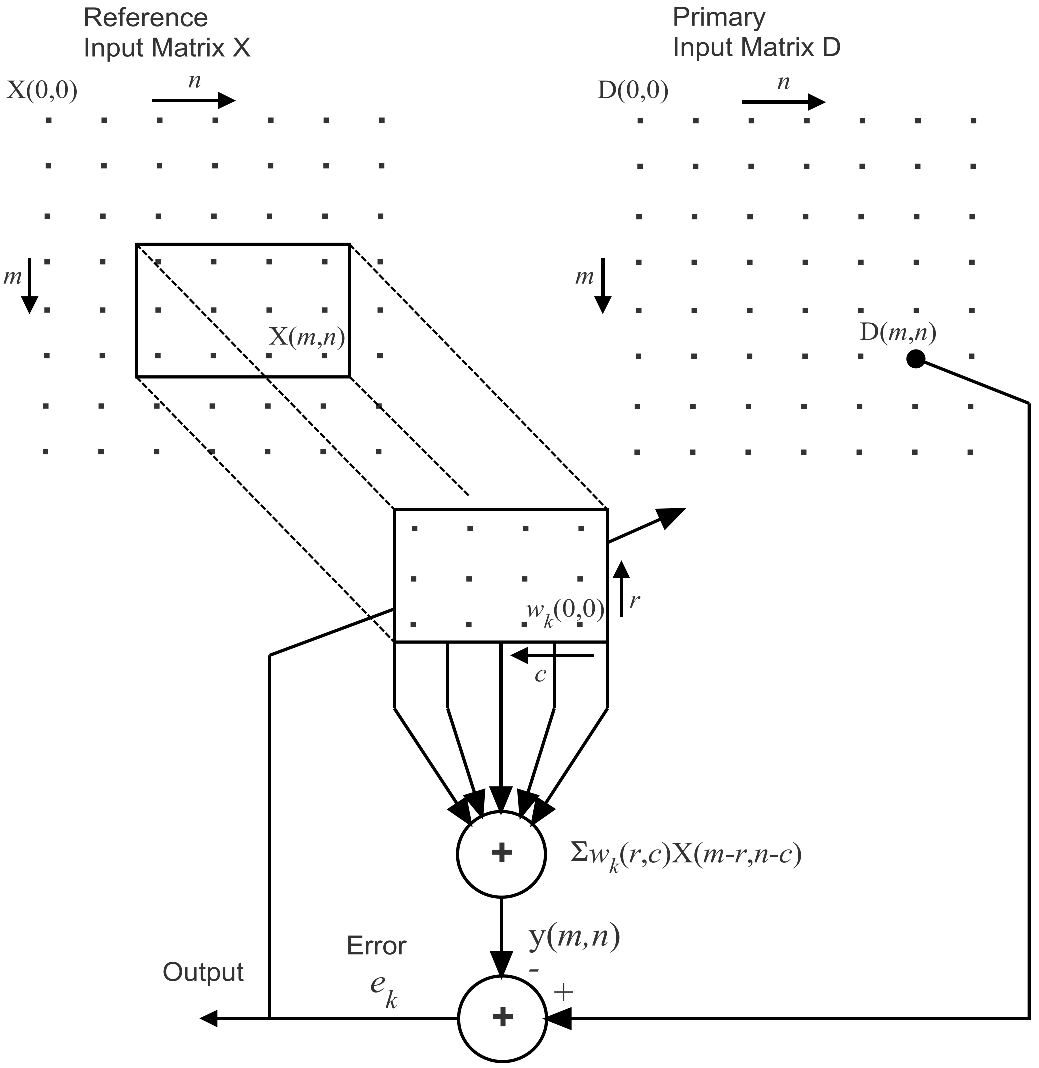


Figure 3.2 The two-dimensional LMS adaptive filter structure [83]

The LMS algorithm estimates an instantaneous gradient **GI** which is the gradient of the squared error of a single iteration, such that

$$\mathbf{GI}(r,c) = \frac{\partial e_k^2}{\partial \mathbf{W}_k} = 2e_k \frac{\partial e_k}{\partial \mathbf{W}_k}. \tag{3.51}$$

From (3.51) we obtain

$$\frac{\partial e_k^2}{\partial \mathbf{W}_k} = 0 - \mathbf{X}(m-r,n-c)$$

$$= -\mathbf{X}(m-r,n-c). \tag{3.52}$$

Then,

$$\mathbf{GI}(r,c) = -\mathbf{X}(m-r,n-c). \tag{3.52}$$

Or the estimated instantaneous gradient matrix $\mathbf{GI}$ is

$$\mathbf{GI} = -\mathbf{X}_k. \tag{3.53}$$

Substituting (3.51) and (3.53) into (3.48) we get

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu\, e_k\mathbf{X}_k, \tag{3.54}$$

which can be rewritten as

$$\mathbf{W}_{k+1}(r,c) = \mathbf{W}_k(r,c) + 2\mu\, e_k\mathbf{X}_k(m-r,n-c). \tag{3.55}$$

Equations (3.54) and (3.55) give the two-dimensional weight updating algorithm for the 2DLMS adaptive filter. The algorithm doesn't require any averaging, differentiation or any matrix operations. It delivers good nonstationary performance as its convergence is guaranteed regardless of the initial conditions.

CHAPTER 4

THE PROPOSED VARIANTS OF MIXED-NORM LMS ALGORITHM

4.1. Overview

The mixed-norm least mean square (MN-LMS) algorithm has shown outstanding performance compared to that of the conventional LMS algorithm [56]. In this chapter, the convergence analysis of the MN-LMS algorithm is derived. Based on that, two algorithms that exploit the sparsity of the system have been derived. The first algorithm is proposed by adding l_1-norm penalty to the cost function of the MN-LMS algorithms. This term enables us to attract the zero and/or near-to-zero filter coefficients to the zero value faster. However, when the system is near or exactly non-sparse, the algorithm almost fails. To overcome this limitation, we propose another algorithm that uses an approximation of l_0-norm penalty term in the cost function of the MN-LMS algorithm. This provides high performance even with completely non-sparse systems. The performances of the proposed algorithms are compared to those of the LMS and MN-LMS algorithms in an acoustic sparse system identification setting. These proposed algorithms provide significant performances compared to the other algorithms under different sparsity levels and signal-to-noise ratios (SNRs).

4.2. Related Work

Adaptive algorithms are usually used to update the filter weights by minimizing the error function; which is represented as the difference between the desired response and the real output of the adaptive filter. Adaptive filters have a wide range of applications such as; system identification [84][85], echo and/or adaptive noise cancelation [44][86], linear prediction [87][88], adaptive line enhancement [82][89], etc.

In the last decades, the least mean square (LMS) algorithm has gained much of interest because to its simplicity. The LMS algorithm is, basically, a stochastic gradient-based algorithm that iteratively updates the weights of the filter to converge to the optimum error. In many signal processing applications; namely the system identification or channel estimation problem, the impulse response of the system/channel may have sparse nature. Sparsity means that only small coefficients have relatively large magnitudes and the others are zero or near-zero. The sparsity can be handled by imposing the l_p-norm to the cost function of the classical LMS algorithm. In [90], authors performed theoretical performance analysis of l_0-LMS for white Gaussian input signals based on sparse systems. In [71], researchers introduced two new algorithms with l_1-relaxation to enhance the performance of the LMS algorithm. Their algorithms generated a zero-attractor in the LMS iteration which in turn helped to accelerate the convergence when identifying sparse systems. In [91], authors proposed adaptive algorithm which adapts to the degree of sparseness, using a convex combination based approach.

Introduction of *p*-norm-like penalty in the cost function of the LMS algorithm helped exerting zero attraction in LMS iterations in the work of Wu and Tong [92]. This in turn improved the performance of norm constraint based LMS algorithms. Penalized LMS algorithms were proposed in [93] using l_p-norm and l_1-norm. This work showed that such sparsity-aware LMS algorithms achieve better performance than the standard LMS algorithm when the system to be estimated is sparse or near-sparse.

The mixed norm LMS algorithm recently proposed by Boukis *et al.* [56] is based on the least sum of exponentials (LSE). LSE provides a generalization of the class of weighted mixed norm algorithms. It is derived by minimizing a sum of error exponentials. The algorithm has shown significant performance compared to that of the conventional LMS algorithm.

In this chapter, the convergence analyses of the MN-LMS algorithm in terms of mean and mean-square sense have been derived. We also propose two zero-attracting versions of the algorithm which have shown high performance in system identification setting, particularly, when the unknown system is a sparse or a near-sparse system.

4.3. The Mixed Norm Least Mean Square (MN-LMS) Algorithm

Before deriving the proposed algorithm, we introduce a review of the MN-LMS algorithm. In the system identification setting shown in Fig. 4.1, the output of a linear system with input signal $\mathbf{x}(k)$ is given by

$$d(k) = \mathbf{h}^{\boldsymbol{T}}\mathbf{x}(k) + v(k), \tag{4.1}$$

where $d(k)$ the desired response of the adaptive is filter, $\mathbf{h}$ is the impulse response of the unknown system, $\mathbf{x}(k)$ is the input-tap vector, $v(k)$ is the observation noise and $[.]^T$ is transposition operator.

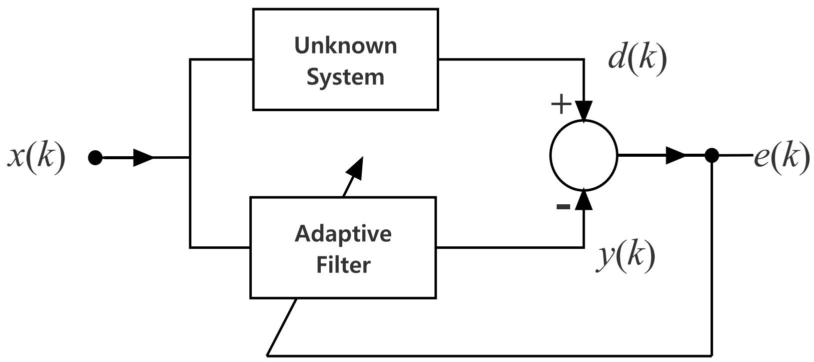


Figure 4.1 Block diagram of a system identification setting.

The cost function of the MN-LMS algorithm [56] is given by

$$J_{\boldsymbol{w}}(k) = \Big(exp\big(e(k)\big) + exp(-e(k))\Big)^{\mathbf{2}}, \tag{4.2}$$

where $\mathbf{w}(k)$ is the filter-tap weight vector with length N and $e(k)$ is the instantaneous error and defined by

$$e(k) = d(k) - \mathbf{w}^{\boldsymbol{T}}(k)\mathbf{x}(k). \tag{4.3}$$

Deriving (4.2) with respect to $\mathbf{w}(k)$ gives

$$\frac{\partial J_{(\mathrm{k})}}{\partial \mathbf{w}_{(k)}} = -4\mathbf{x}(k)sinh(e(k)). \quad (4.4)$$

The tap-update equation is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{\mu}{2}\frac{\partial J(k)}{\partial \mathbf{w}(k)}, \quad (4.5)$$

where μ is the step-size controlling convergence and the steady-state behaviors of the algorithm. Substituting (4.4) in (4.5) and rearranging, the update equation of the MN-LMS algorithm becomes

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\mathbf{x}(k)sinh\big(e(k)\big). \quad (4.6)$$

In order to test how the algorithm is successful and robust, convergence analyses of the algorithm in the mean sense and the mean-square sense are performed.

4.4. Convergence Analysis of the MN-LMS Algorithm

The convergence analysis of the MN-LMS algorithm is performed in this section. Through the analysis, the input signal $\mathbf{x}(k)$ is assumed to be Gaussian with zero-mean and variance σ_x^2. The misalignment vector is defined as

$$\boldsymbol{\delta}(\mathrm{k}) = \mathbf{w}(\mathrm{k}) - \mathbf{h}, \quad (4.7)$$

and hence, the instantaneous error becomes

$$e(k) = \boldsymbol{\delta}^T(k)\mathbf{x}(k) + \eta(k). \quad (4.8)$$

Combining (4.6) and (4.7), we obtain

$$\boldsymbol{\delta}(k+1) = \boldsymbol{\delta}(k) + 2\mu\mathbf{x}(k)\sinh\big(e(k)\big). \quad (4.9)$$

4.4.1. Convergence in the Mean Sense

Let us define the mean-square-error (MSE) as

$$\boldsymbol{\varepsilon}(k) = E\{\boldsymbol{\delta}(k)\boldsymbol{\delta}(k)^T\}. \tag{4.10}$$

Using the relation, $\sinh\big(e(k)\big) = \frac{(e(k))^{2i+1}}{(2i+1)!}$ and substituting (4.1) and (4.3) in (4.9) we obtain

$$\boldsymbol{\delta}(k+1) = \boldsymbol{\delta}(k) + 2\mu\mathbf{x}(k)\sum_{i=0}^{\infty}\frac{[\mathbf{x}(k)^T\boldsymbol{\delta}(k)+\eta(k)]^{2i}}{(2i+1)!}. \tag{4.11}$$

Taking the expectation of both sides of (4.11) yields

$$E\{\boldsymbol{\delta}(k+1)\} \approx E\{\boldsymbol{\delta}(k)\} - 2\mu\mathbf{x}(k)\sum_{i=0}^{\infty}\frac{(2i+1)^{2i}}{(2i+1)!}E\{{\eta_k}^{2i}\}\,\mathbf{R}\boldsymbol{\delta}(k)$$

$$= \left\{\mathbf{I} - 2\mu\sum_{i=0}^{\infty}\frac{(2i+1)^{2i}}{(2i+1)!}E\{{\eta_k}^{2i}\}\mathbf{R}\right\}\boldsymbol{\delta}(k), \tag{4.12}$$

where is the autocorrelation matrix defined as $\mathbf{R} = E\{\mathbf{x}(k)\mathbf{x}^{\boldsymbol{T}}(k)\}$. From (4.12) it is noted that, $E\{\boldsymbol{\delta}(k+1)\} \rightarrow$ constant value, if the maximum eigenvalue of $\Big(\mathbf{I} - 2\mu\sum_{i=0}^{\infty}\frac{(2i+1)^{2i}}{(2i+1)!}E\{{\eta_k}^{2i}\}\mathbf{R}\Big)$ should be less than unity, and hence, it shows that the filter weights converge to their optimum solution in the mean sense.

4.4.2. Convergence in the Mean Square Sense

In this section, convergence in the mean-square sense is derived. The convergence condition is obtained by multiplying both sides of (4.11) by $\boldsymbol{\delta}(k+1)^T$, taking the expectation and using (4.10) it provides

$$\boldsymbol{\varepsilon}(k+1) = \boldsymbol{\varepsilon}(k) - E\left\{\mu\sum_{i=0}^{\infty}\frac{(2i+1)^{2i}}{(2i+1)!}E\{{\eta_k}^{2i}\}\mathbf{R}\boldsymbol{\delta}(k)\boldsymbol{\delta}(k)^T\right\}$$
$$-\left\{\mu\sum_{i=0}^{\infty}\frac{(2i+1)^{2i}}{(2i+1)!}E\{{\eta_k}^{2i}\}\boldsymbol{\delta}(k)\boldsymbol{\delta}(k)^T\mathbf{R}\right\}$$

$$+ E\left\{\mu^2 \sum_{i=0}^{\infty} \frac{(2i+1)^{4i}}{[(2i+1)!]^2} \left[E\left\{{\eta_k}^{2i}\right\}^2\right] \mathbf{R}\boldsymbol{\delta}(k)\boldsymbol{\delta}(k)^T\mathbf{R}\right\}. \quad (4.13)$$

calculating the trace the fourth term of (4.13)

$$tr(E\{\mu^2 \sum_{i=0}^{\infty} \frac{(2i+1)^{4i}}{[(2i+1)!]^2} \left[E\left\{{\eta_k}^{2i}\right\}^2\right] \mathbf{R}\boldsymbol{\delta}(k)\boldsymbol{\delta}(k)^T\mathbf{R}\}$$

$$= \alpha\mu^2 \sum_{i=0}^{\infty} ({\sigma_\eta}^{2i})^2 \, tr\{\mathbf{R}\boldsymbol{\varepsilon}(k)\}, \quad (4.14)$$

From [1], $\{\mathbf{R}\,\boldsymbol{\varepsilon}(k)\} = \sum_{i=1}^{N} \lambda_i x_i(k) = \sum_{j=1}^{N} \lambda_j E\left\{\left|\boldsymbol{\delta}_j(k)\right|^2\right\}$. The matrices $\mathbf{R}$ and $\boldsymbol{\varepsilon}(k)$ are symmetric, and hence, $tr(\mathbf{R}\boldsymbol{\varepsilon}(k)\mathbf{R}) = tr\left(\mathbf{R}^2\boldsymbol{\varepsilon}(k)\right)$. Substituting these in (4.13) gives

$$\boldsymbol{\varepsilon}(k+1) = (1 - 2\gamma\mu{\sigma_x}^2 + \mu^2\alpha{\sigma_x}^4)\boldsymbol{\varepsilon}(k). \quad (4.15)$$

where

$$\alpha = \sum_{i=0}^{\infty} \frac{(2i+1)^{4i}}{\left((2i+1)!\right)^2},$$

$$\gamma = \sum_{i=0}^{\infty} \frac{(2i+1)^{2i}}{\left((2i+1)!\right)} \text{ and}$$

$$E\{{\eta_k}^2\} = {\sigma_\eta}^2.$$

In order for (4.15) to converge, the term $|1 - 2\gamma\mu{\sigma_x}^2 + \mu^2\alpha{\sigma_x}^4|$ should be less than unity. Solving this term provides

$$0 < \mu < \frac{2\gamma}{\alpha{\sigma_x}^2}. \quad (4.16)$$

The result of (4.16) provides a tighter bound on μ than that of the conventional LMS algorithm. Based on this, we propose the following sparse adaptive algorithms.

4.5. The Proposed Algorithms

The performance of the MN-LMS can be further improved in system identification settings if the system is sparse. This improvement can be achieved by imposing a penalty term in its cost function. In this chapter, we derive two sparse algorithms by imposing the l_1 and l_0-norms to the cost function of the MN-LMS algorithm.

4.5.1. Zero-Attracting Mixed Norm LMS (ZA-MN-LMS) Algorithm

The improvement in the performance of the MN-LMS algorithm, when the system is sparse, can be achieved by modifying the cost function in (4.2) to become

$$J_{1,\mathbf{w}}(k) = \big(exp(e(k)) + exp(-e(k))\big)^2 + \xi\|\mathbf{w}(k)\|_1, \tag{4.17}$$

where $\|.\|_1$ denotes the l_1-norm of the weight vector and ξ is a positive constant. Deriving (4.17) with respect to $\mathbf{w}(k)$ and substituting in (4.5) yields

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\mathbf{x}(k)sinh\big(e(k)\big) - \rho\, sgn[\mathbf{w}(k)]. \tag{4.18}$$

where $\rho = \mu\xi$ and $sgn(.)$is the sign function and defined as

$$sgn[b] = \begin{cases} \frac{b}{|b|}, & b \neq 0 \\ 0, & b = 0 \end{cases}.$$

The term $\rho\, sgn[\mathbf{w}(k)]$ in (4.18) imposes an attraction to zero on small weights (zero or near-to-zero coefficients). Particularly, if the filter weight value is positive, it will decrease and if it is negative, it will increase. However, if the system is non-sparse or near to non-sparse, the performance of the algorithm may deteriorate due to this term. Thereby, a reweighted zero attraction term is introduced next.

4.5.2. Reweighted Zero-Attracting Mixed Norm LMS (RZA-MN-LMS) Algorithm

The cost function, here, may be written as

$$J_{2,\mathbf{w}}(k) = \left(exp(e(k)) + exp(-e(k))\right)^2 + \beta\|\mathbf{w}(k)\|_0, \tag{4.19}$$

where $\|.\|_0$ is the l_0-norm of the weight vector and β is a positive constant. The main problem in the cost function given in (4.19), is taking the derivative of the l_0-norm term with respect to $\mathbf{w}(k)$. Therefore, the well-known approximation of l_0-norm maybe used, which is given as

$$\|\mathbf{w}(k)\|_0 = \sum_{i=1}^{N} log\left(1 + \frac{|w_i(k)|}{\eta}\right), \tag{4.20}$$

where η is a positive parameter. Taking derivative of (4.19) with respect to $\mathbf{w}(k)$ and substituting in (4.5) results in

$$\mathbf{w}(k+1) = \mathbf{w}(k) + 2\mu\mathbf{x}(k)\sinh\left(e(k)\right) - \rho\,\frac{sgn[\mathbf{w}(k)]}{1+\varepsilon|\mathbf{w}(k)|} \tag{4.21}$$

where $\rho = \frac{\mu\beta}{\eta}$ and $\varepsilon = \frac{1}{\eta}$. The term $\rho\,\frac{sgn[\mathbf{w}(k)]}{1+\varepsilon|\mathbf{w}(k)|}$ in (4.21) imposes a reweighted attraction to zero on small weights (zero or near-to-zero weights). Particularly, if the filter weight value is positive, it will decrease and if it is negative, it will increase. The reweighted zero attractor has effect only on those elements for which magnitudes are comparable to $\frac{1}{\eta}$ and there will be a small shrinkage employed on the taps whose $|w_i(k)| < \frac{1}{\eta}$.

4.6. Simulation Results and Discussions

In this section, the performances of the ZA-MN-LMS and RZA-MN-LMS algorithms are compared to those of the LMS and MN-LMS algorithms in the system identification setting. The performance measures are the convergence rate and the mean square deviation (MSD) which is calculated as

$$\text{MSD}(k) = E\left\{\|\mathbf{w}(k) - \mathbf{h}\|^2\right\}. \tag{4.22}$$

The observation noise is assumed to be an additive white Gaussian noise (AWGN) with zero mean and variance $\sigma_v^2 = 0.001$. The simulations of all algorithms are performed for 100 independent runs. For all the algorithms μ= 0.005.

In the first experiment, we investigate the MSD of the proposed algorithms and MN-LMS versus different values of ρ with 90% and 50% sparsity levels. The filter length is assumed to be 20 taps. The SNR is 10 dB and for the RZA-MN-LMS algorithm ε = 10.

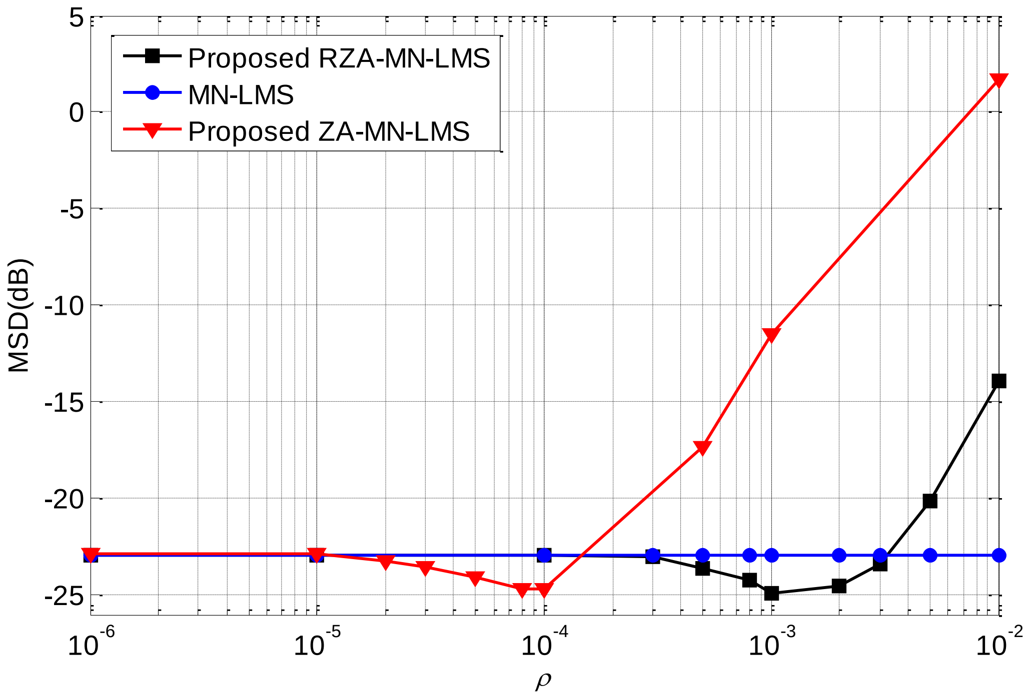


Figure 4.2 MSD vs. ρ for the RZA-MN-LMS, ZA-MN-LMS and MN-LMS algorithms in AWGN with 90% sparsity.

From Fig. 4.2, the proposed algorithms reach MSDs lower than that of the MN-LMS algorithm by 2.1 dB with 90% sparsity. Also, even though both the ZA-MN-LMS and the RZA-MN-LMS algorithms provide almost the same optimum MSD (-25 dB), the minimum MSD of the RZA-MN-LMS algorithm can be achieved for a wider range of ρ.

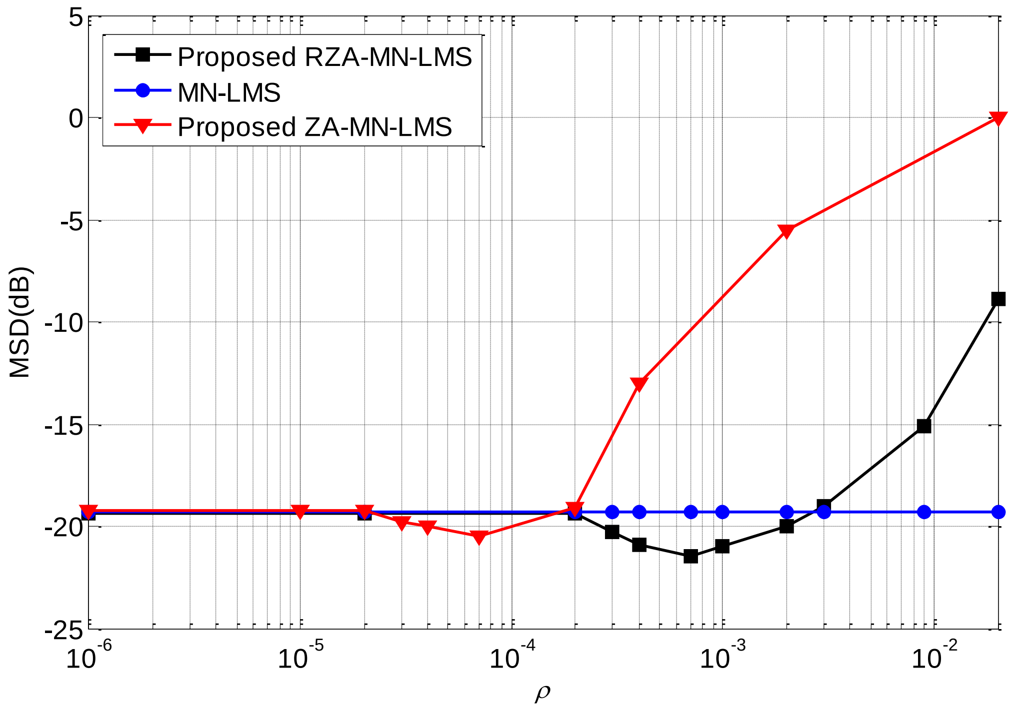


Figure 4.3 MSD vs. ρ for the RZA-MN-LMS, ZA-MN-LMS and MN-LMS algorithms in AWGN with 50% sparsity.

Fig. 4.3 shows the MSD vs ρ with 50% sparsity. The proposed algorithms still reach MSDs lower than that of the MN-LMS algorithm. The optimum MSD for the ZA-MN-LMS and the RZA-MN-LMS algorithms are -20.5 dB and -21.3 dB, respectively. With less sparsity of the system the performance of ZA-MN-LMS starts to deteriorate as it perform best at high sparsity levels.

In the second experiment, the convergence behaviors of the proposed algorithms, MN-LMS and LMS are investigated in AWGN environment. The parameters are the same as the first experiment. The ρ values of the ZA-MN-LMS algorithm is chosen according to the results in Fig. 4.2 as ρ=2×10^{-4} while for the RZA-MN-LMS algorithm is $\rho = 10^{-3}$. From Fig. 4.4, we clearly see that RZA-MN-LMS algorithm has a faster convergence with a lower MSD value than the other algorithms.

In the third experiment, we investigate the MSD behaviors of the RZA-MN-LMS, ZA-MN-LMS and MN-LMS algorithms for different sparsity ratios. The filter length here is N=10 taps. For the ZA-MN-LMS algorithm $\rho = 5 \times 10^{-5}$. For the RZA-MN-LMS algorithm $\rho =$

10^{-3} (again these ρ values are found in the same way as in the first experiment). Fig. 4.5 shows that even though the performance of the ZA-MN-LMS is good at high sparsity levels, it deteriorates when the system tends to be non-sparse system. However, the performance of the RZA-MN-LMS is always better than all algorithms or the same (in the worst case when the system is completely non-sparse).

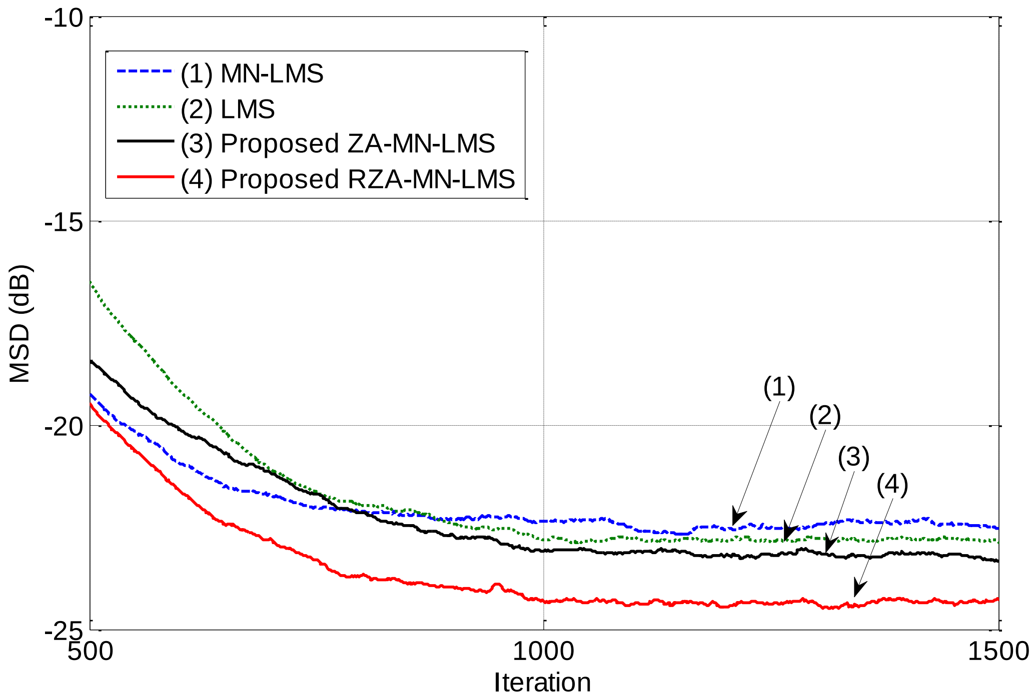


Figure 4.4 Convergence behaviors of the proposed algorithms, MN-LMS and LMS algorithms in AWGN environment.

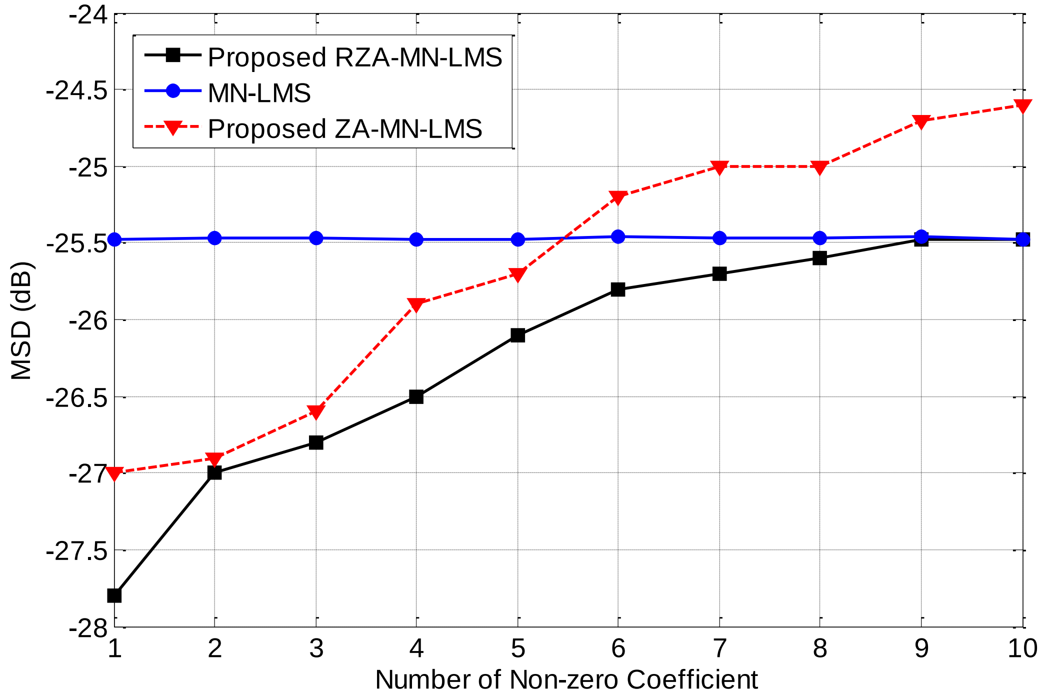


Figure 4.5 MSD curves of the RZA-MN-LMS, ZA-MN-LMS and MN-LMS algorithms for different sparsity ratios.

In the fourth experiment, the input signal and the observation noise is considered same as previous settings. The unknown system is considered to be a clustered sparse system with 256 taps as shown in Fig. 4.6.

The algorithms are simulated with the following parameters (see Fig. 4.7):

For the ZA-MN-LMS algorithm : $\mu = 0.005$ and $\rho = 1.5\times10-5$.

For the RZA-MN-LMS algorithm : $\mu = 0.005$, $\rho = 1.5\times10-5$ and $\varepsilon = 10$.

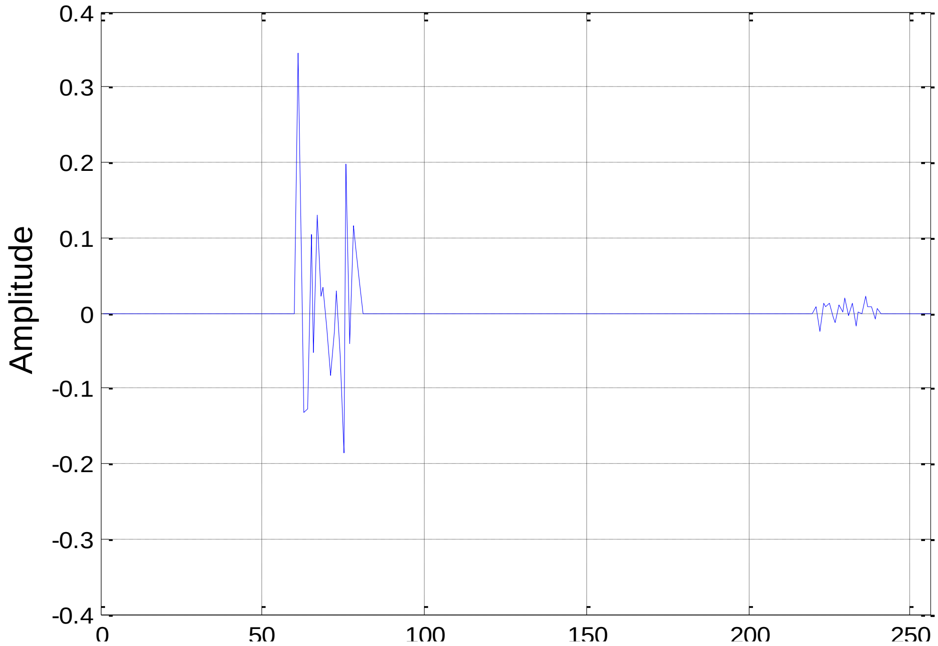


Figure 4.6 Impulse function of the acoustic system.

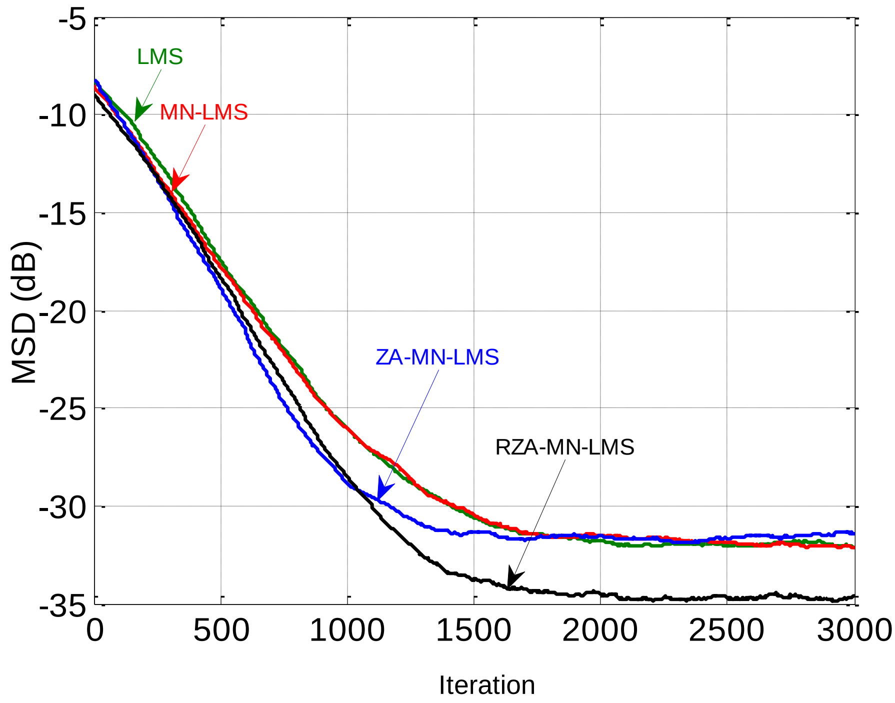


Figure 4.7 MSD performances of the algorithms in AWGN environment.

In the fifth experiment, the input signal and the impulse response of the unknown system are considered to be as those of experiment 4. In order to study the effect of using different noise on the performance of the proposed algorithms, the observation noise is assumed to be an additive correlated Gaussian noise (ACGN) with 90 % sparsity. The correlated noise is created using an AR(1) process $z(k) = 0.8z(k-1) + v(k)$ where $v(k)$ is a white Gaussian process with zero mean and variance $\sigma_v^2 = 0.001$. In Figure 4.8, even though ZA-MN-LMS is showing faster convergence rate, the RZA-MN-LMS is the one with the lowest MSD value (≈ -25 dB).

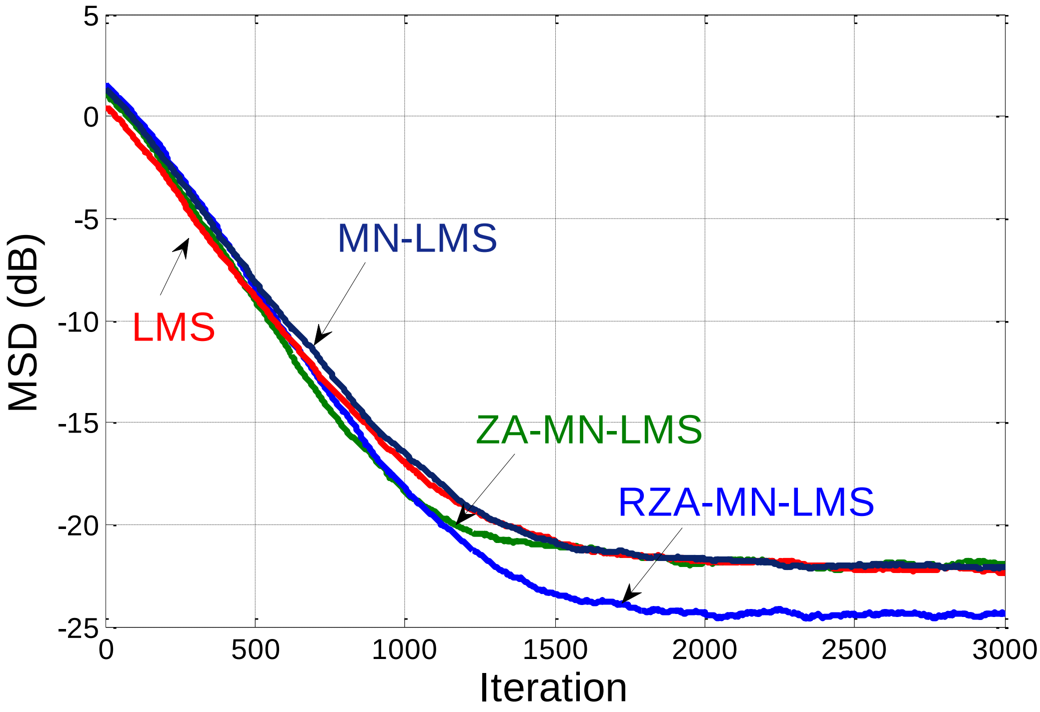


Figure 4.8 MSD performances of the algorithms in ACGN environment.

In the last experiment, the performances of the RZA-MN-LMS and MN-LMS algorithms with different SNR values was investigated (here the ZA-MN-LMS algorithm is neglected because it diverges (goes to infinity) at low SNR values). The sparsity ratio is assumed to be 90% and $\rho = 10^{-3}$. Fig. 4.9 shows that the RZA-MNLMS algorithm is always better than the MN-LMS algorithm under the given conditions.

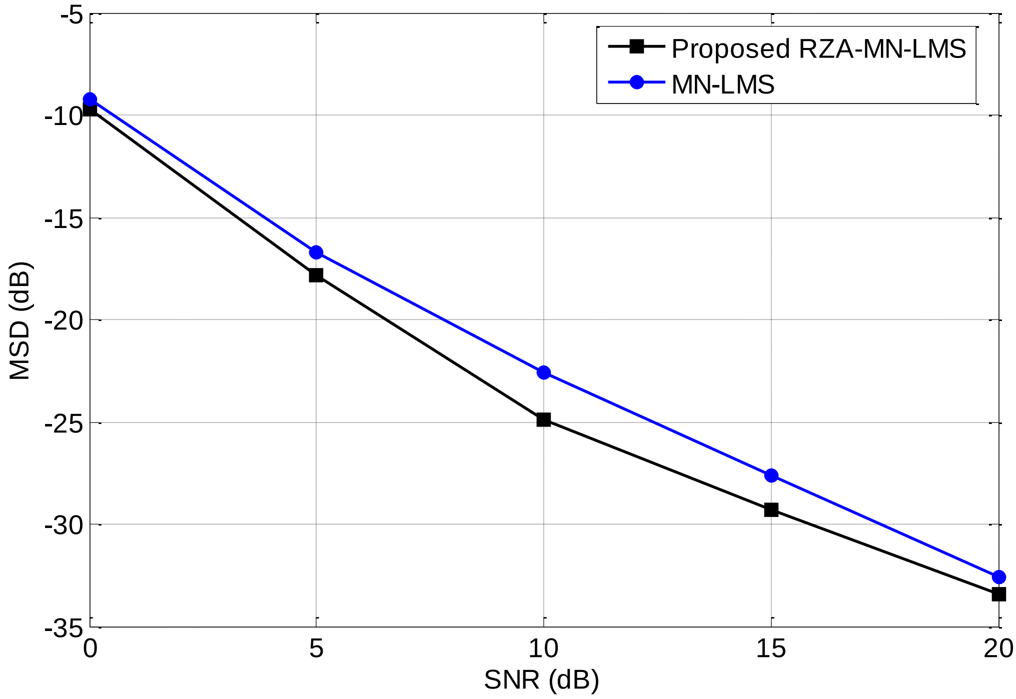


Figure 4.9 MSD curves of the RZA-MN-LMS and MN-LMS algorithms for different SNR values.

4.7. Conclusion

In this chapter, convergence analyses in the mean and mean-square sense of the MN-LMS algorithm were presented. In addition to this, two algorithms that exploit the sparsity of the system were proposed, namely: ZA-MN-LMS and RZA-MN-LMS Algorithms. The performances of the proposed ZA-MN-LMS and the RZA-MN-LMS algorithms were investigated under different filter lengths, sparsity ratios and SNRs. The proposed algorithms always showed high performance compared to the original MN-LMS algorithm under the given conditions.

CHAPTER 5

THE PROPOSED 2D ZERO ATTRACTING LMS ALGORITHM

5.1. Overview

In this chapter, a new two-dimensional zero-attracting least mean square (2D ZA-LMS) adaptive filter is proposed. The filter is imposing a sparsity aware l_1-norm penalty term into the cost function of the 2D-LMS algorithm. The convergence analysis of the 2D ZA-LMS algorithm is presented and stability criterion is derived. The performance of the proposed algorithm has been compared to that of the 2D-LMS algorithm in adaptive line enhancer (ALE) problem on sparse and non-sparse images. Simulation results have shown that the proposed algorithm has good capabilities in updating the filter coefficients along both horizontal and vertical directions, and its performance is always the same/better than that of the 2D-LMS algorithm with lower computational complexity.

5.2. Related Work

The earliest studies on gradient-based adaptive algorithms may be traced back to more than six decades [48]. In late 1950's, the least mean square (LMS) algorithm was devised by Widrow and Hoff [48] in their study of the adaptive linear element machine. It is a stochastic gradient algorithm in that it iterates each tap weight of the filter [1]. In [83], the first two dimensional adaptive filtering algorithm which can be applied as an adaptive line enhancer is presented. It is a direct extension of the one-dimensional (1-D) LMS algorithm. Two-Dimensional (2D) adaptive filters are applied to the problems of image denoising as in [94]. The performance of the algorithms is improved by changing the step-size and updating the filter coefficients partially. In [95], a system identification application has been developed. It has linear adaptive filters, whose coefficients are updated based on the normalized LMS algorithm. In [96], another 2D adaptive filtering application on equalization that employs the optimum (minimum mean-square-error (MSE)) was proposed.

LMS algorithms for sparse system identification recently appeared as a popular research topic as the unknown systems are sparse in most physical paths. One of the first trials is the zero-attracting LMS (ZA-LMS) algorithm [71]. The algorithm is derived by imposing the l_1-norm penalty into the quadratic cost function of the standard LMS algorithm. Its mean square error analysis has been provided in [81]. In [90], an l_0 norm constraint LMS algorithm modifies the cost function of the standard algorithm and the parameter selection rule is performed. In [90], the sparsity is exploited by using l_1-norm penalty in the cost function of the standard algorithm.

5.3. The 2D ZA-LMS Algorithm

Before deriving the proposed algorithm, we try to highlight the main role of the zero-attracting term in the 1-D ZA-LMS algorithm.

5.3.1 Review of the 1-D ZA-LMS Algorithm

In the 1-D ZA-LMS algorithm, the cost function $J_{\mathbf{w}}$ is defined by adding the l_1 norm penalty of the weights vector to the instantaneous squared error

$$J_{\mathbf{w}} = \frac{1}{2} e_k^2 + \gamma \|\mathbf{w}_k\|_1, \tag{5.1}$$

where e_k is the instantaneous error given by $e_k = d_k - \mathbf{w}_k \mathbf{x}_k$, d_k is the desired response, γ is a small positive parameter and $\mathbf{w}_k$ is the tap-weight vector of the algorithm of length N. The ZA-LMS filter weight update equation is defined as

$$\begin{aligned} \mathbf{w}_{k+1} &= \mathbf{w}_k - \mu \frac{\partial J_{\mathbf{w}}}{\partial \mathbf{w}_k} \\ &= \mathbf{w}_k - \rho\, sgn[\mathbf{w}_k] + \mu\, e_k \mathbf{x}_k, \end{aligned} \tag{5.2}$$

where $\rho = \gamma\mu$ which controls the zero-attraction term and $sgn[.]$ is a sign function defined as

$$sgn[b] = \begin{cases} \frac{b}{|b|} & , b \neq 0 \\ 0 & , b = 0 \end{cases}. \tag{5.3}$$

The term $\rho\ sgn[\mathbf{w}_k]$ imposes zero attraction on small weights (zero or near-to-zero weights). Particularly, if the filter weight value is positive, it will decrease and if it is negative, it will increase.

5.3.2. Extending to the 2D Case

The update equation of the 2D ZA-LMS adaptive filter weight matrix can be given as:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \mathbf{G}_k, \tag{5.4}$$

where $\mathbf{W}_k$ is the updated weight matrix of size $N\times N$. The estimate of the true 2D instantaneous gradient of $E\{e_k^2\}$, with respect to $\mathbf{W}_k$ is

$$\mathbf{G}_k = \frac{\partial E\{e_k^2\}}{\partial \mathbf{W}_k} + \lambda \|\mathbf{w}_k\|_1. \tag{5.5}$$

$\mathbf{G}_k$ can be further expended to

$$\mathbf{G}_k = \begin{bmatrix} -2e_k\mathbf{X}_k(0,0) + \lambda\, sgn[\mathbf{W}_k(0,0)] & \dots - 2e_k\mathbf{X}_k(0, N-1) + \lambda\, sgn[\mathbf{W}_k(0, N-1)] \\ \vdots & \ddots & \vdots \\ -2e_k\mathbf{X}_k(N-1,0) + \lambda\, sgn[\mathbf{W}_k(N-1,0)] & \dots\ -2e_k\mathbf{X}_k(0,0) + \lambda\, sgn[\mathbf{W}_k(N-1, N-1)] \end{bmatrix}. \tag{5.6}$$

The ZA-LMS algorithm estimates the instantaneous gradient; the gradient of the squared error of an iteration.

$$\mathbf{GI}(r,c) = \frac{\partial e_k^2 + \lambda\, \partial \|\mathbf{W}_k(r,c)\|_1}{\partial \mathbf{W}_k}$$

$$= -2e_k\mathbf{X}_k(r,c) + \lambda\, sgn[\mathbf{W}_k(r,c)], \tag{5.7}$$

where r and c denote the row and column indices, respectively. Substituting (5.7) in (5.4) and simplifying yields the weight update equation as

$$\mathbf{W}_{k+1}(r,c) = \mathbf{W}_k(r,c) + \mu\, \mathbf{X}_k(l-r, m-c)d(l,m)$$

$$-\mu\, \mathbf{X}_k(l-r,m-c) \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} \mathbf{W}_k(a,b)\, \mathbf{X}_k(l-a,m-b)$$

$$-\lambda\, sgn[\mathbf{W}_k(r,c)], \tag{5.8}$$

where the filter output is defined as

$$y(r,c) = \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} \mathbf{W}_i(a,b)\, \mathbf{X}_k(l-a,m-b). \tag{5.9}$$

The filter weights $\mathbf{w}_{vk}$ and the input data matrix $\mathbf{x}_{vk}$ can be reshaped into one dimensional form. During the k^{th} iteration, the column vectors are respectively

$$\mathbf{w}_{vk} = \begin{bmatrix} w_k(0,0) \\ \vdots \\ w_k(0,N-1) \\ \vdots \\ w_k(N-1,0) \\ \vdots \\ w_k(N-1,N-1) \end{bmatrix} \tag{5.10}$$

and

$$\mathbf{x}_{vk} = \begin{bmatrix} x_k(0,0) \\ \vdots \\ x_k(0,N-1) \\ \vdots \\ x_k(N-1,0) \\ \vdots \\ x_k(N-1,N-1) \end{bmatrix}. \tag{5.11}$$

According to these, (5.8) can be written as

$$\mathbf{w}_{v(k+1)} = \mathbf{w}_{vk} + \mu\, d(l,m)\mathbf{x}_{vk} - \mu\, \mathbf{x}_{vk}\, \mathbf{x}_{vk}^T \mathbf{w}_{vk} - \lambda\, sgn[\mathbf{w}_{vk}]. \tag{5.12}$$

Hence, using (5.10) and (5.11) the 1-D ZA-LMS can be used as a 2D ZA-LMS algorithm in (5.12) and the update equation becomes

$$\mathbf{w}_{v(k+1)} = (\mathbf{I} - \mu\, \mathbf{x}_{vk}\, \mathbf{x}_{vk}^T)\mathbf{w}_{vk} + \mu\, d(l,m)\mathbf{x}_{vk} - \lambda\, sgn[\mathbf{w}_{vk}], \tag{5.13}$$

where **I** is an $N\times N$ identity matrix.

5.3.3. Convergence Analysis of the Proposed Algorithm

In this section we derive the convergence analysis of the 2D ZA-LMS algorithm. Through the analysis, the input signal $\mathbf{x}_{vk}$ is considered to be Gaussian with zero-mean and variance σ_x^2. The misalignment vector is defined as;

$$\boldsymbol{\delta}_{vk} = \mathbf{w}_{vk} - \mathbf{h}, \tag{5.14}$$

where $\mathbf{h}$ is the vector reshaping the matrix of the blurring function. And hence, the instantaneous error becomes

$$e_{vk} = \boldsymbol{\delta}_{vk}^T \mathbf{x}_{vk} + \eta_k, \tag{5.15}$$

where η_k represents the noise. Combining (5.13) and (5.14) and rearranging, we obtain

$$\boldsymbol{\delta}_{v(k+1)} = (\mathbf{I} - \mu\, \mathbf{x}_{vk}\, \mathbf{x}_{vk}^T)\boldsymbol{\delta}_{vk} + \mu\, \mathbf{x}_{vk}\eta_k - \lambda\, sgn[\mathbf{w}_{vk}]. \tag{5.16}$$

In (5.16), the expected value of the term $\lambda\, sgn[\mathbf{w}_{vk}]$ is bounded. Hence, the expected value of (5.16) yields

$$E\{\boldsymbol{\delta}_{vk}\} = (\mathbf{I} - \mu\mathbf{R})^{k+1}\boldsymbol{\delta}(0), \tag{5.17}$$

where $\mathbf{R}$ is the autocorrelation matrix of $\mathbf{x}_k$ and defined as $\mathbf{R} = E\{\, \mathbf{x}_{vk}\, \mathbf{x}_{vk}^T\}$.

In order for (5.17) to converge, the maximum eigenvalue of the term $(\mathbf{I} - \mu\mathbf{R})$ should be less than unity and, hence, the step-size selection criteria of the proposed algorithm will be the same as that of the LMS algorithm.

5.4. The Data Reuse Patterns

For 2D applications, the update of the filter is done along the horizontal and vertical directions by moving a mask of the filter size horizontally to the right by one column at a time until the end of each row. Afterward, the same process is repeated with the next row below until the last pixels of the image are reached.

Some possible ways of data reuse are shown in Fig. 5.1. In the scheme shown in Fig. 5.1 (a), all the data should be used [10]. However, with the same considered mask size, another pattern as in Fig. 5.1 (b) with almost 63% of the pixels may be used. And in case of sparse images (where most of the pixels are zeros) the performance of this pattern is very comparable to that in Fig. 5.1 (a) but with lower computational complexity.

The filter tap weights vector update equation of the ZA-LMS algorithm can be generalized into its 2D form as

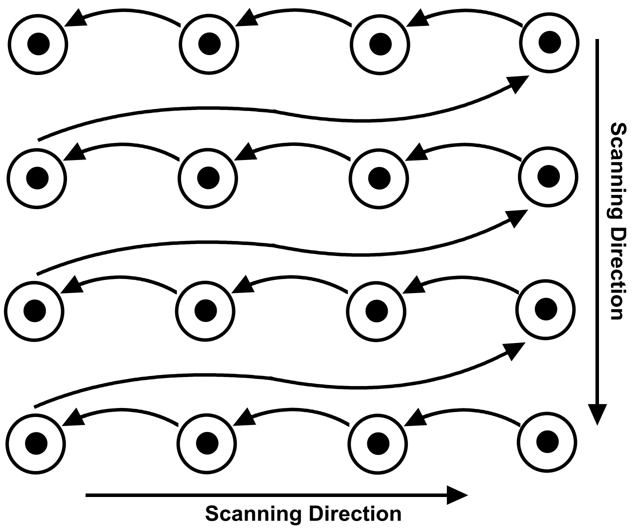
$$\mathbf{w}_{k+1}(m_1, m_2) = \mathbf{w}_k(m_1, m_2) - \rho\, sgn[\mathbf{w}_k(m_1, m_2)] + \mu\, e(k)\mathbf{x}(n_1, n_2), \quad (5.18)$$

where $\mathbf{w}_k(m_1, m_2)$ is the 2D weights matrix with dimensions $N \times N$, $m_1 = 0,1,\cdots,N-1$ and $m_2 = 0,1,\cdots,N-1$. The filter weights and the input data can be reshaped into 1D case, respectively, by

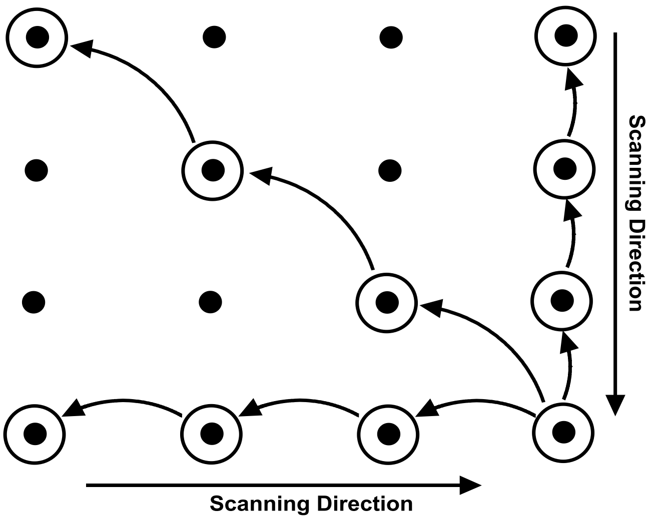
$$\mathbf{w}_k(m_1, m_2) = \begin{bmatrix} w_k(0,0) \\ \vdots \\ w_k(0, N-1) \\ \vdots \\ w_k(N-1, 0) \\ \vdots \\ w_k(N-1, N-1) \end{bmatrix}, \quad (5.19)$$

and

$$\mathbf{x}(n_1, n_2) = \begin{bmatrix} \mathrm{x}(n_1, n_2) \\ \vdots \\ \mathrm{x}(n_1, n_2 - N + 1) \\ \vdots \\ \mathrm{x}(n_1 - N + 1, n_2) \\ \vdots \\ \mathrm{x}(n_1 - N + 1, n_2 - N + 1) \end{bmatrix}. \quad (5.20)$$



(a)



(b)

Figure 5.1 (a) Rectangular configuration of data-reusing in 2D and (b) Axial configuration of data-reusing in 2D [10].

Two data reuse configurations in 2D applications are shown in Fig. 5.1 for a mask of size 4×4. The rectangular configuration is simple and straight forward. First of all, the mask will process all pixels of the image by moving from left top corner horizontally and vertically to cover all the rows and columns until it reaches the bottom right corner. For each processed pixel, the mask will be convolved with the corresponding sub-image after reshaping the sub-image pixels into a vector following the directions shown in Fig 5.1a. Same process will be followed for the axial configuration. The difference will be that we will just use 10 pixels instead of 16 pixels as shown in Fig 1b. Reshaping the 10 pixels into a vector is shown in Fig. 5.1b.

The first configuration provides a good performance but it needs higher number of computations because it uses all the pixels in the filter mask. The second configuration provides a lower performance with a lower computational complexity. However, in sparse images (many zero pixels), the lower performance can be enhanced due to the sparsity of the image. The filter output can be given by the following 2D convolution

$$y(n_1, n_2) = \sum_{m_1=0}^{N-1} \sum_{m_2=0}^{N-1} w(m_1, m_2)\, x(n_1 - m_1, n_2 - m_2). \qquad (5.21)$$

5.5. Simulation Results and Discussions

5.5.1. Subjective Experiments

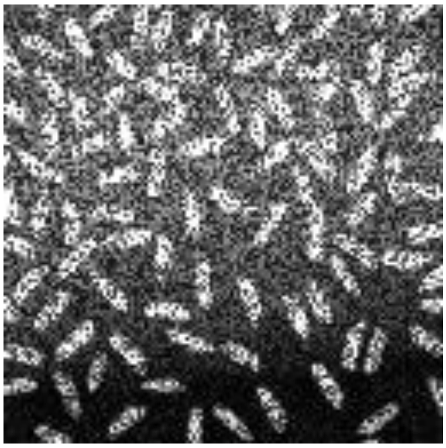
In all the experiments we consider different 8-bit 256×256 grayscale sparse and non-sparse images where the number of pixels with zero/near zero values is changing. Different noises were applied to the images namely; additive white Gaussian noise (AWGN), salt & pepper, and sparkle noise. The performance of proposed 2D ZA-LMS algorithm is compared to that of the 2D-LMS algorithm for image denoising.

In the first experiment, Lena image was degraded with additive white Gaussian noise AWGN with zero mean and 0.01 variance. Fig 5.1 shows the restored image using the 2D-LMS algorithm (Fig 5.1c) and the restored image using the proposed algorithm (Fig 5.1d).

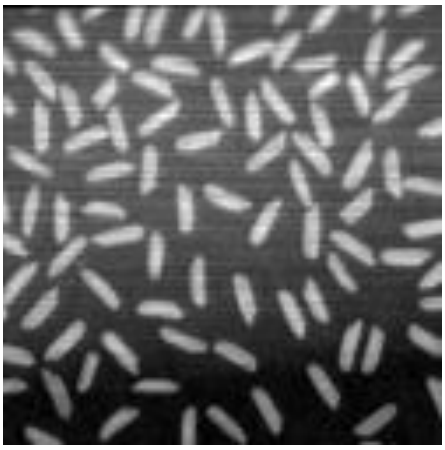
Figure 5.2 (a) Lena image, (b) Noisy image (AWGN), (c) Restored image using the 2D-LMS algorithm (PSNR: 56.98) (d) Restored image using the proposed 2D ZA-LMS algorithm (PSNR: 56.05).

In order to test the performance of the proposed 2D ZA-LMS algorithm with respect to the noise type, the first experiment is repeated with the same parameters and a speckle noise with zero mean and 0.04 variance and μ=0.001 for both algorithms. Figure 5.3 shows that the proposed algorithm still performs the same as the 2D-LMS algorithm but with lower computational complexity.

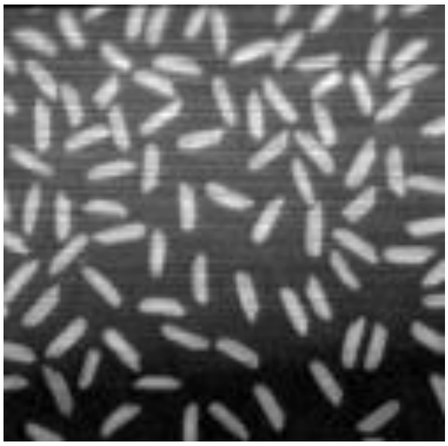
(a)



(b)



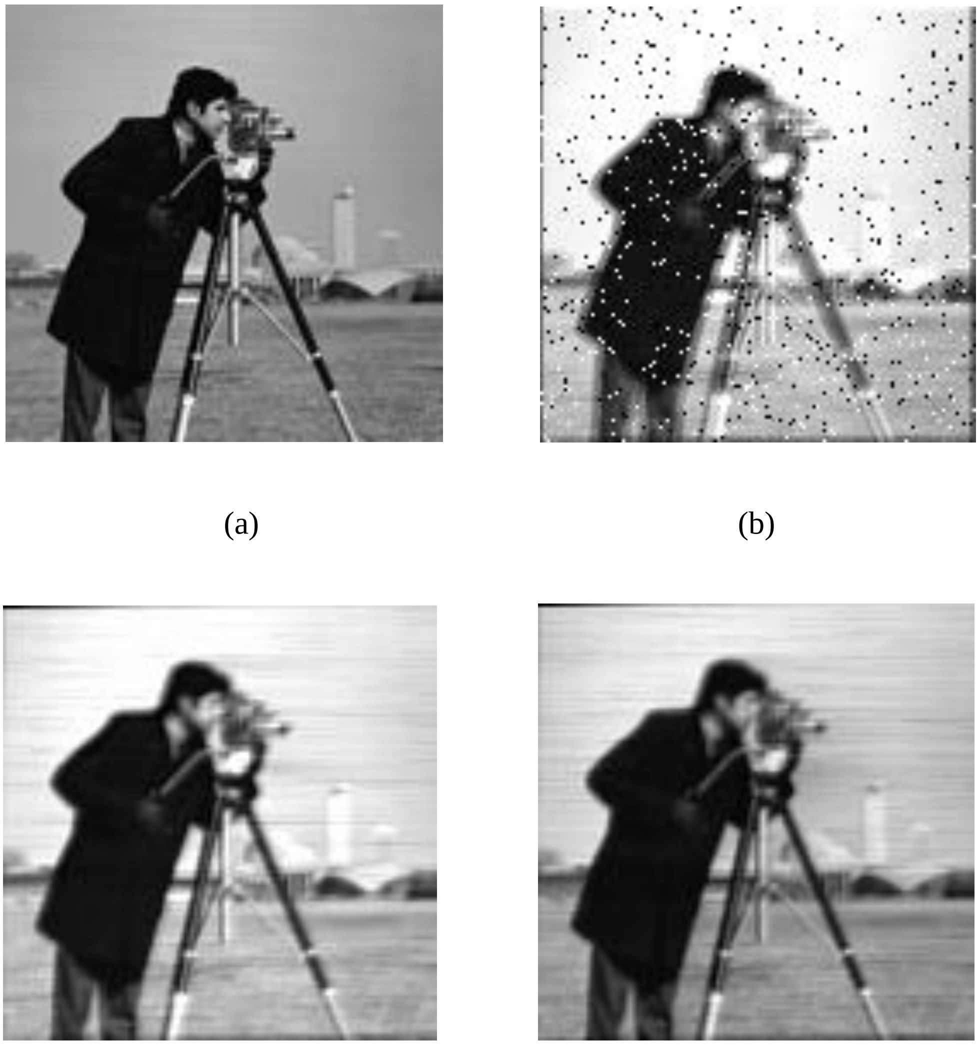
(c)



(d)

Figure 5.3 (a) Rice image (b) Noisy image (Speckle noise), (c) Restored image using the 2D-LMS algorithm (PSNR: 54.89) (d) Restored image using the proposed 2D ZA-LMS algorithm (PSNR: 55.07)

In order to test the performance of the algorithms with different filter sizes, the first experiment is again repeated with the same parameters. A 3×3 filter size and salt and pepper noise with 0.05 density were used with μ=0.001 for both algorithms. Fig. 5.4 shows that the proposed 2D ZA-LMS algorithm is better than the 2D-LMS algorithm with almost 22% less computational complexity.



(a) (b)

(c) (d)

Figure 5.4 (a) Cameraman image, (b) Noisy image (Salt&Pepper noise), (c) Restored image using the 2D-LMS algorithm (PSNR: 54.87) (d) Restored image using the proposed 2D ZA-LMS algorithm(PSNR: 55.34).

In Fig. 5.5, the 2D-LMS algorithm is simulated with the parameters: μ=0.001 and 3×3 filter size are used in the rectangular configuration of data-reusing. For the proposed 2D ZA-LMS algorithm, axial configuration was applied. The parameters for 2D ZA-LMS are μ=0.001, ε=10, and $\rho=10^{-4}$.

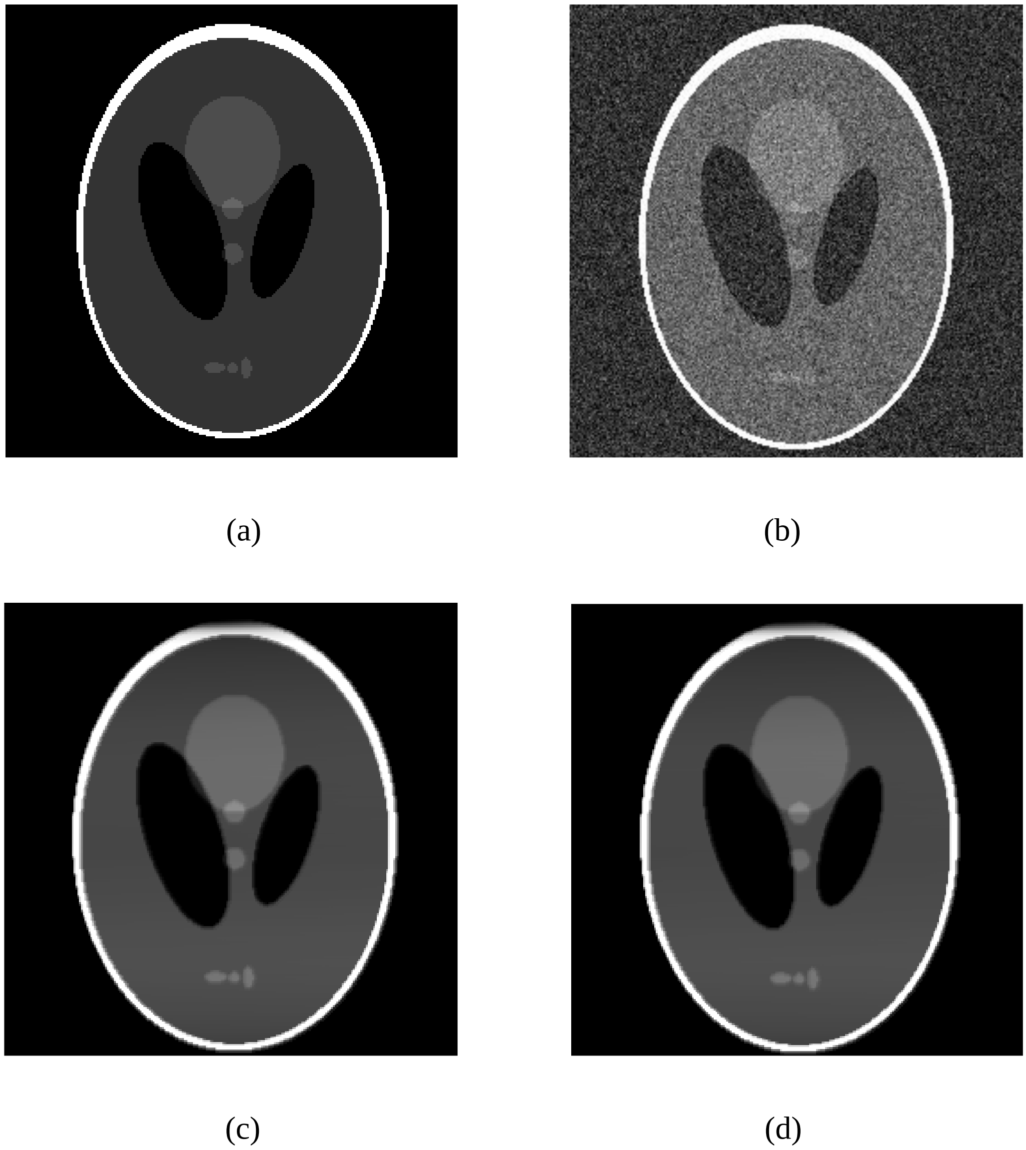
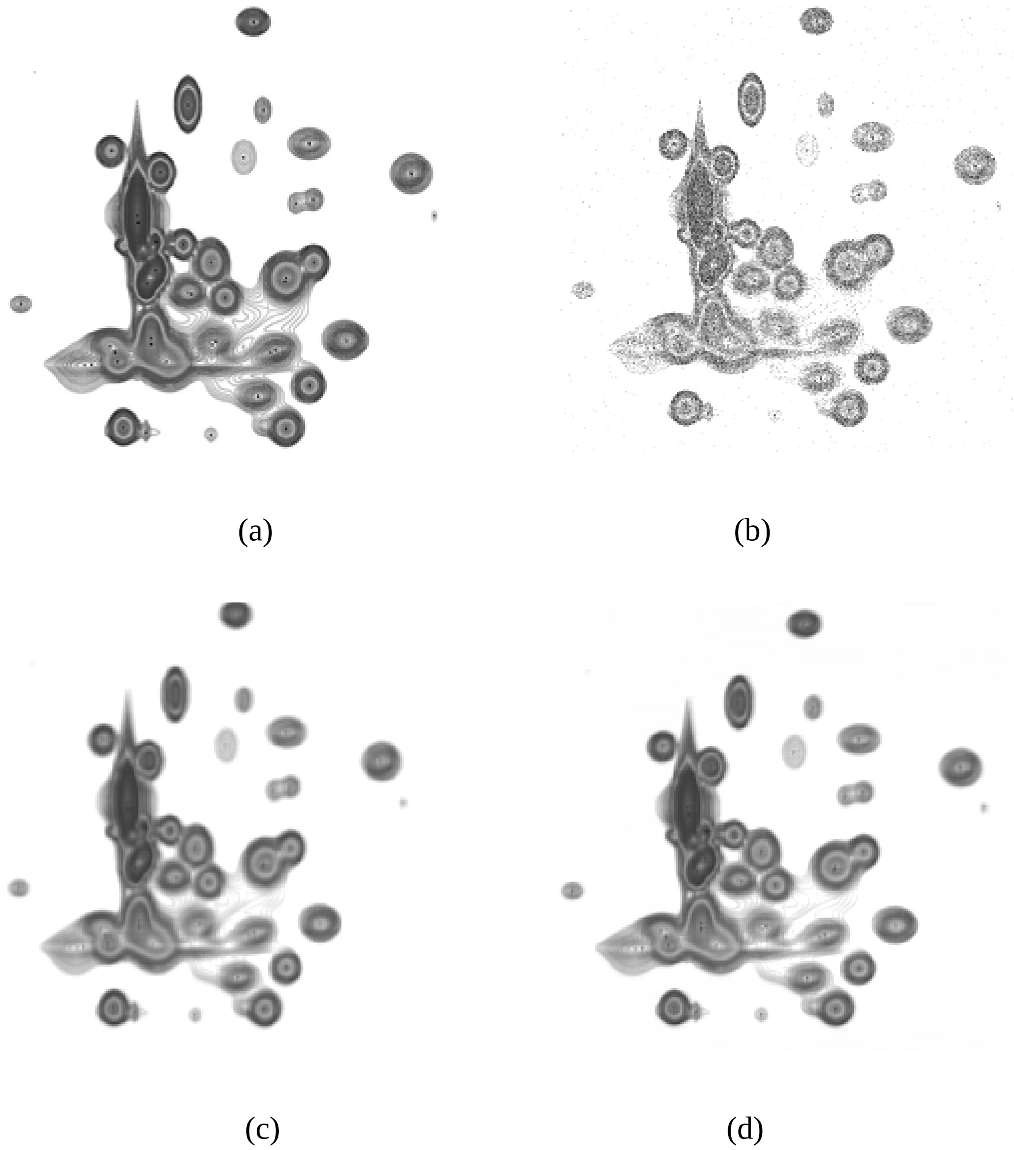


Figure 5.5 (a) Phantom image, (b) Noisy image (AWGN), (c) Restored image using the 2D-LMS algorithm (PSNR: 34.31) (d) Restored image using the proposed 2D ZA-LMS algorithm (PSNR: 34.65).

In Fig. 5.5b, the test phantom image is degraded by an AWGN with zero mean and 0.2 variance. Fig. 5.5c shows the restored image by the 2D-LMS algorithm, and Fig. 5.5d shows the restored image by the proposed 2D ZA-LMS algorithm.



(a) (b) (c) (d)

Figure 5.6 (a) Splash image, (b) Noisy image (AWGN), (c) Restored image using the 2D-LMS algorithm (PSNR: 45.87) (d) Restored image using the proposed 2D ZA-LMS algorithm (PSNR: 45.74).

Fig. 5.6b shows the splash image again degraded by the same AWGN in Fig. 5.6. The 3×3 Kernel here is replaced by a 4×4 kernel for both rectangular and axial configurations with μ=0.001 for both algorithms. Fig. 5.6c, the restored image by the 2D-LMS algorithm is shown while Fig. 5.6d shows the restored image by the proposed 2D ZA-LMS algorithm.

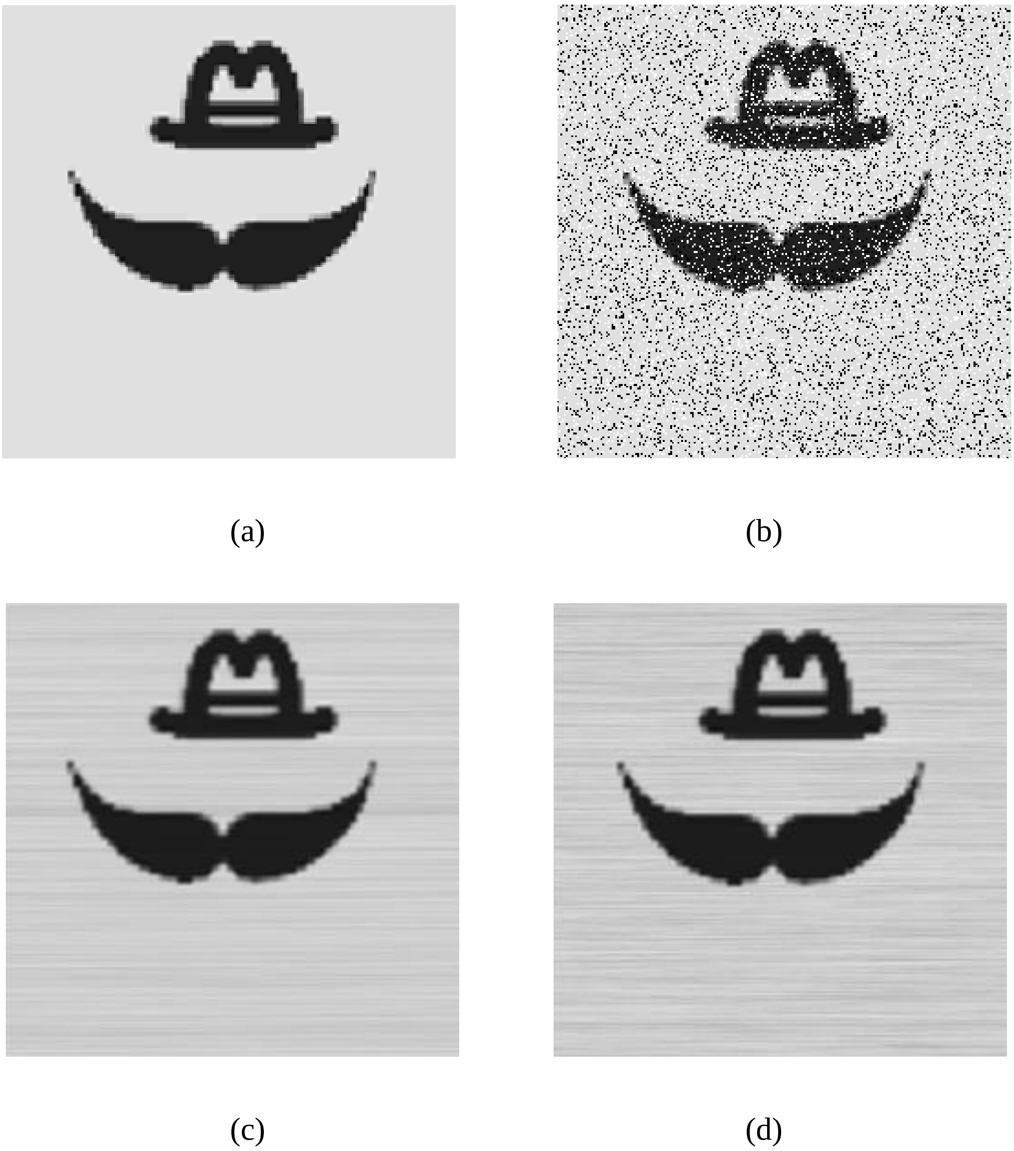
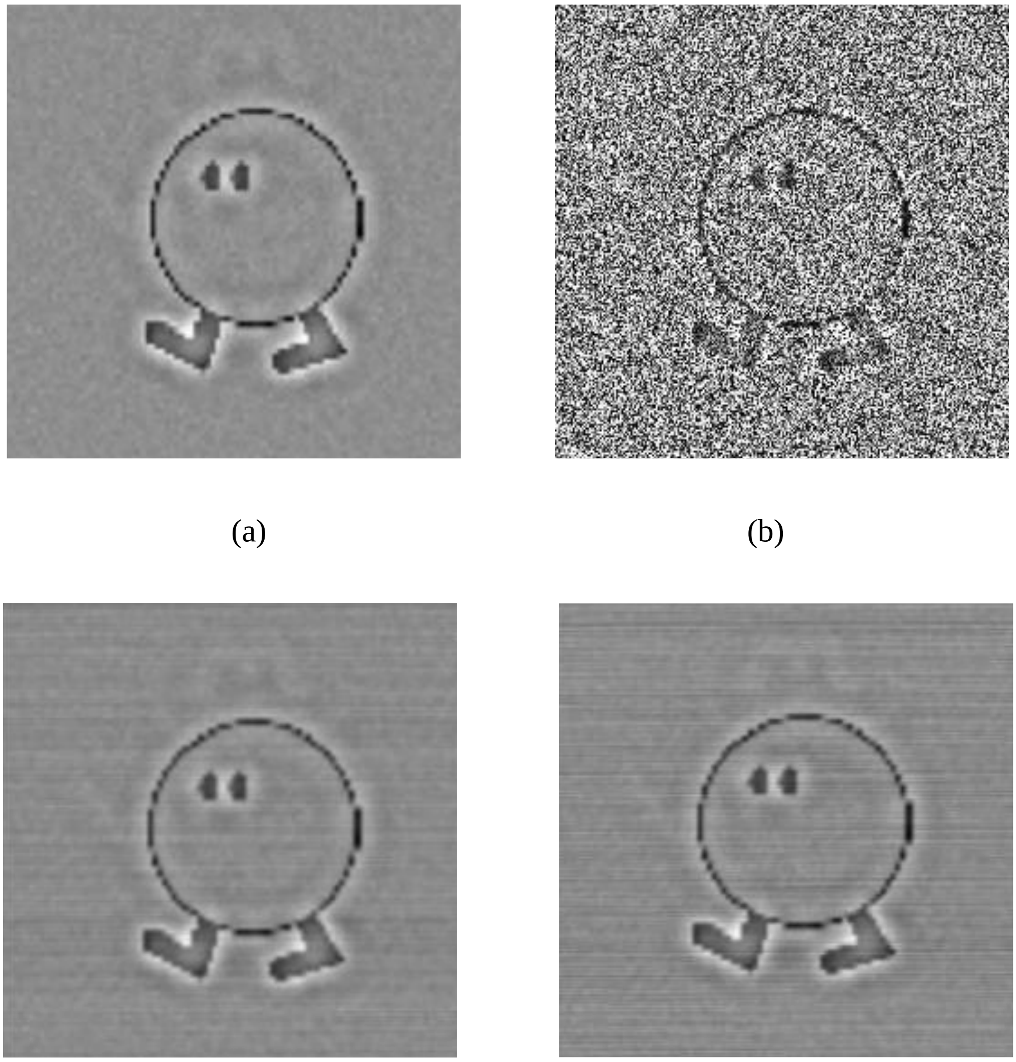


Figure 5.7 (a) Mustache image, (b) Noisy image (salt & pepper noise), (c) Restored image using the 2D-LMS algorithm (PSNR: 67.47) (d) Restored image using the proposed 2D ZA-LMS algorithm(PSNR: 67.33) .

The image in Fig. 5.7 shows the mustache image degraded by salt & pepper noise with probability of 0.2 and μ=0.001 for both algorithms. Figs. 5.7c and 5.7d show the restored images using 2D-LMS and the proposed algorithms, respectively.



(a) (b) (c) (d)

Figure 5.8 (a) Man image, (b) Noisy image (speckle noise), (c) Restored image using the 2D-LMS algorithm (PSNR: 57.8) (d) Restored image using the proposed 2D ZA-LMS algorithm (PSNR: 58.14) .

Fig. 5.8 shows the man image degraded by speckle noise (multiplicative noise) with zero mean and 0.4 variance with μ=0.001 for both algorithms. Figs. 5.8c and 5.8d show the restored images using 2D-LMS and the proposed 2D ZA-LMS algorithm, respectively.

To test the effect of using AWGN noise different variance values, we performed an experiment as shown in Fig 5.9 to Fig. 5.12. The AWGN is chosen with zero mean and variance equal to 0.0001, 0.001, 0.01, 0.02, 0.05, 0.07, 0.1, 0.2 and 0.4 as shown in Fig 5.10.

Performance of both algorithms looks similar (see Fig 5.11 and Fig 5.12) with the advantage of faster computation of the proposed 2D ZA-LMS algorithm.

Figure 5.9 The Maze image

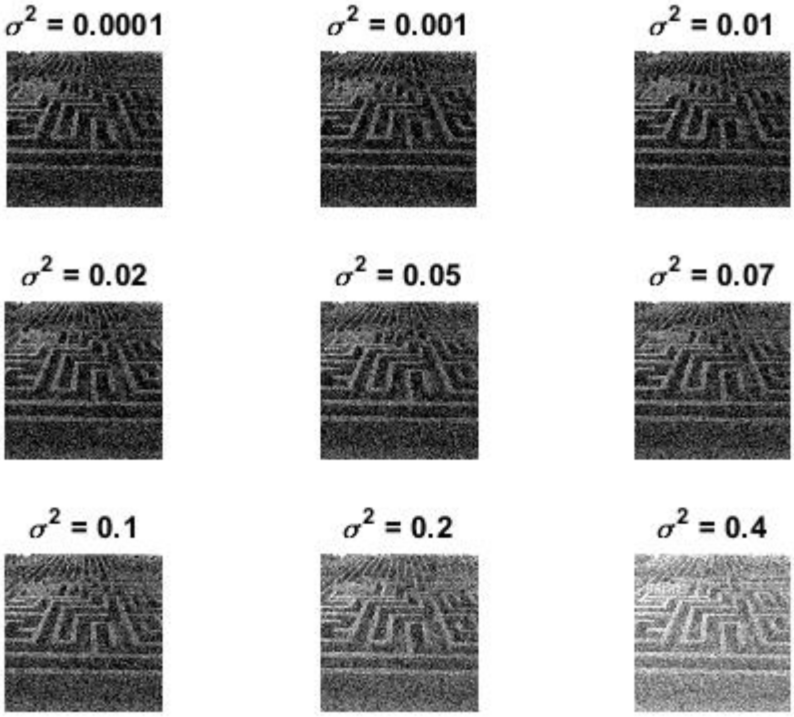


Figure 5.10 Maze image corrupted by AWGN with different variance values and zero mean

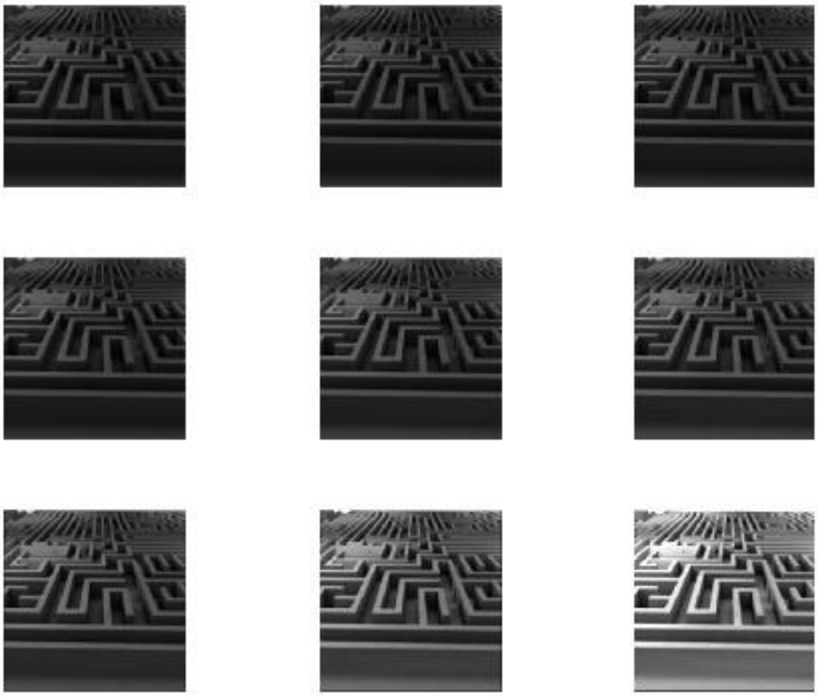


Figure 5.11 Restored images corresponding to the noisy maze images in previous figure using the proposed 2D ZA-LMS algorithm ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$)

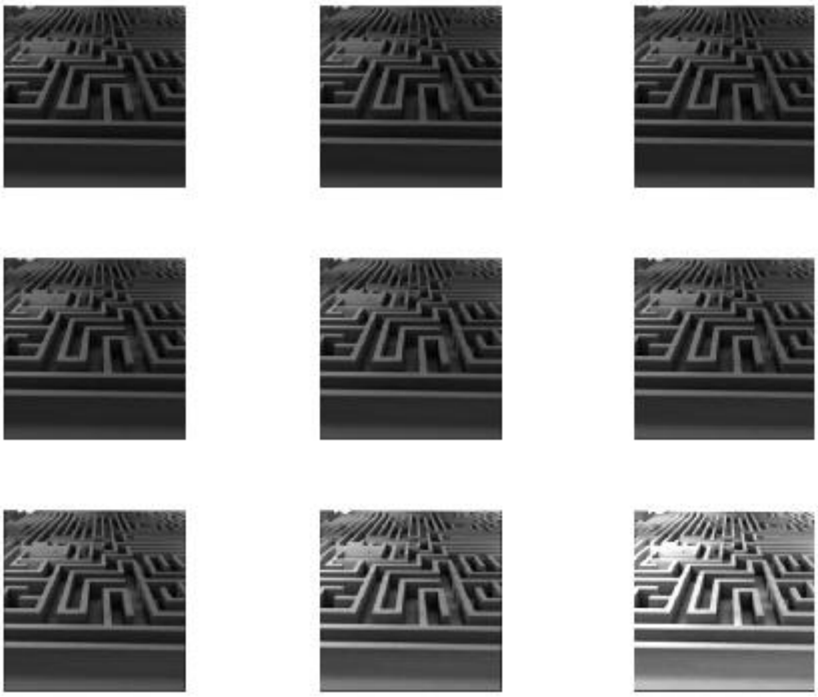


Figure 5.12 Restored images corresponding to noisy maze images in previous figure using 2D-LMS algorithm ($\mu = 0.001$)

From all the test figures generated from different experiments, we can easily observe that a successful image denoising is achieved by the proposed 2D ZA-LMS algorithm in different noise environments with a less time consumption due to the usage of the axial configuration of the data mask.

5.5.2. Objective Experiments

To further emphasize the time consumption criteria of the proposed 2D ZA-LMS and 2D-LMS, a comparison between the two algorithms is conducted under AWGN, speckle and salt&pepper noise using different images with 2D ZA-LMS using axial configuration. The results are shown in Table 5.1, Table 5.2 and Table 5.3, respectively. On average, the proposed 2D ZA-LMS algorithm is 15.96, 17.1 and 17.45 times faster than the 2D-LMS algorithm under the three noise types, respectively. This can be justified by the use of the axial configuration for the filter in 2D ZA-LMS algorithm instead of the rectangular configuration which is used by the filter of the 2D-LMS algorithm.

Table 5.1 Time consumption comparison in seconds between the 2D-LMS and Proposed 2D ZA-LMS algorithms using different images under AWGN ($\sigma^2 = 0.01$).

Image 256×256	**Proposed 2D ZA-LMS** $\mu = 0.001, \rho = 10^{-4}$, $\varepsilon = 10$ (sec)	**2D-LMS** $\mu = 0.001$ (sec)	**Gain Factor**
Man	2.23	36.91	16.55
Splash	2.21	36.56	16.54
Lena	2.43	37.02	15.23
Maze	2.07	37.21	17.98
Moustache	2.56	36.89	14.41
Rice	2.59	38.82	14.98
Phantom	2.33	37.30	16.01
Average Gain Factor			**15.96**

Table 5.2 Time consumption comparison in seconds between the 2D-LMS and Proposed 2D ZA-LMS algorithms using different images under Speckle noise ($\sigma^2 = 0.1$).

Image 256×256	**Proposed 2D ZA-LMS** $\mu = 0.001, \rho = 10^{-4}$, $\varepsilon = 10$ (sec)	**2D-LMS** $\mu = 0.001$ (sec)	**Gain Factor**
Man	2.24	37.55	16.76
Splash	2.16	36.43	16.87
Lena	2.16	38.47	17.81
Maze	2.13	37.42	17.57
Moustache	2.13	36.54	17.16
Rice	2.12	36.16	17.06
Phantom	2.19	36.23	16.54
Average Gain Factor			**17.1**

Table 5.3 Time consumption comparison in seconds between the 2D-LMS and Proposed 2D ZA-LMS algorithms using different images under salt&pepper noise ($Pr = 0.1$).

Image 256×256	**Proposed 2D ZA-LMS** $\mu = 0.001, \rho = 10^{-4}$, $\varepsilon = 10$ (sec)	**2D-LMS** $\mu = 0.001$ (sec)	**Gain Factor**
Man	2.10	36.53	17.40
Splash	2.14	37.30	17.43
Lena	2.14	39.40	18.41
Maze	2.11	36.95	17.51
Moustache	2.17	36.42	16.78
Rice	2.10	36.10	17.19
Phantom	2.15	37.70	17.54
Average Gain Factor			**17.47**

Another objective test was conducted using the signal to noise ratio SNR and the peak signal to noise ratio PSNR. PSNR and SNR are metrics for measuring the Restored/denoised image quality and usually is shown in the logarithmic decibel scale.

Given a restored $N \times N$ 8-bit image I_1 and its noisy version I_2. The formula for SNR and PSNR metrics are given, respectively, as

$$\text{SNR} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} [I_2(i,j)]^2}{\sum_{i=1}^{N} \sum_{j=1}^{N} [I_1(i,j) - I_2(i,j)]^2}, \tag{5.22}$$

$$\text{PSNR} = 10 log_{10} \left(\frac{255^2}{MSE} \right), \tag{5.23}$$

where mean square error MSE is calculated by

$$\text{MSE} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} [I_1(i,j) - I_2(i,j)]^2 \tag{5.24}$$

Generally, the PSNR values for 8-bit image compression are between 30 and 50 dB. Typically, higher values indicate a better quality. Comparisons between the 2D-LMS and the proposed 2D ZA-LMS algorithms for different images and different noise with variable parameters are provided in Table 5.4, Table 5.5, and Table 5.6.

Results on theses tables indicate that, regardless of the noise type, with the increase in the noise level, the PSNR and SNR values between the restored image and the noisy image is decreasing. This is due to the increase in the MSE value which is inversely proportional to SNR and PSNR values. Generally, the proposed 2D ZA-LMS algorithm is performing very close to the 2D-LMS algorithm in terms of PSNR, SNR and MSE values for sparse images.

Table 5.4 PSNR, SNR and MSE comparisons between the 2D-LMS and 2D ZA-LMS algorithms under AWGN ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$).

	Algorithm	**Proposed 2D ZA-LMS**			**2D-LMS**		
Image	**variance**	**PSNR**	**SNR**	**MSE**	**PSNR**	**SNR**	**MSE**
Man	**0.1**	68.1933	20.0548	0.0068	65.2747	4.8842	0.0034
	0.2	64.2734	16.0658	0.0320	59.0973	1.7037	0.0060
	0.4	59.4061	10.9253	0.1218	52.1742	0.6437	0.0254
Splash	**0.1**	61.6492	13.2644	0.0018	65.5002	5.0320	0.0034
	0.2	60.3176	12.0678	0.0021	59.1464	1.7033	0.0060
	0.4	59.4588	11.3213	0.0043	52.1544	0.6328	0.0253
Lena	**0.1**	56.8468	6.7796	0.0047	57.1048	7.0515	0.0070
	0.2	56.5810	7.2754	0.0213	56.8409	7.5438	0.0265
	0.4	55.0042	6.6438	0.0772	55.1941	6.8223	0.0873
Maze	**0.1**	58.3278	0.4670	0.0020	58.4537	0.6140	0.0063
	0.2	55.6026	1.4369	0.0144	55.8054	1.6436	0.0266
	0.4	52.0478	2.6537	0.0887	52.3157	2.9300	0.1163
Moustache	**0.1**	69.7538	21.1240	0.0037	69.8414	21.2107	0.0045
	0.2	67.4704	18.8778	0.0091	67.0322	18.4466	0.0109
	0.4	59.7060	11.3933	0.0130	59.7031	11.3916	0.0152
Rice	**0.1**	54.8899	3.6526	0.0023	55.0708	3.8311	0.0034
	0.2	53.5688	3.6429	0.0117	53.9141	3.9913	0.0157
	0.4	54.3875	5.7774	0.0545	54.7201	6.1185	0.0639
Phantom	**0.1**	65.2781	4.7872	0.0046	65.6510	5.1372	0.0034
	0.2	59.0758	1.7173	0.0061	58.9741	1.6377	0.0060
	0.4	51.9434	0.3933	0.0160	52.1374	0.6290	0.0255

Table 5.5 PSNR, SNR and MSE comparisons between the 2D-LMS and 2D ZA-LMS algorithms under Speckle noise ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$).

	Algorithm	**Proposed 2D ZA-LMS**			**2D-LMS**		
Image	**variance**	**PSNR**	**SNR**	**MSE**	**PSNR**	**SNR**	**MSE**
Man	**0.1**	52.200	1.7663	0.0013	52.202	1.7727	0.0009
	0.2	51.833	1.1629	0.0014	51.858	1.2058	0.0010
	0.4	51.641	0.8644	0.0020	51.657	0.8690	0.0014
Splash	**0.1**	58.562	9.7752	0.0196	58.501	9.7148	0.0167
	0.2	54.916	5.5583	0.0351	54.875	5.5031	0.0317
	0.4	53.402	3.5919	0.0651	53.388	3.5563	0.0605
Lena	**0.1**	54.309	2.5002	0.0032	54.428 0	2.5814	0.0021
	0.2	53.528	1.5913	0.0043	53.626	1.6644	0.0027
	0.4	53.014	1.0144	0.0069	53.067	1.0843	0.0045
Maze	**0.1**	61.990	0.0417	0.0058	62.114	0.0821	0.0021
	0.2	60.390	0.0234	0.0059	60.411	0.0490	0.0022
	0.4	58.865	0.0157	0.0057	58.929	0.0278	0.0022
Moustache	**0.1**	58.079	8.9320	0.0074	58.128	8.9838	0.0055
	0.2	55.014	5.2752	0.0162	55.042	5.3037	0.0134
	0.4	53.721	3.5244	0.0346	53.722	3.5126	0.0308
Rice	**0.1**	56.003	3.0531	0.0068	56.110	3.1811	0.0044
	0.2	54.699	1.8527	0.0089	54.782	1.9433	0.0060
	0.4	53.787	1.1265	0.0129	53.899	1.1778	0.0097
Phantom	**0.1**	64.884	2.9439	0.0149	70.073	8.1570	0.0076
	0.2	63.875	1.6416	0.0164	66.222	3.9614	0.0088
	0.4	62.969	0.9273	0.0181	64.163	2.1245	0.0102

Table 5.6 PSNR, SNR and MSE comparisons between the 2D-LMS and 2D ZA-LMS algorithms under Salt&Pepper noise ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$).

	Algorithm	**Proposed 2D ZA-LMS**			**2D-LMS**		
Image	**probability**	**PSNR**	**SNR**	**MSE**	**PSNR**	**SNR**	**MSE**
Man	**0.1**	58.5863	10.0654	0.0015	59.5168	10.9968	0.0010
	0.2	56.3878	7.6516	0.0018	57.0559	8.3117	0.0013
	0.4	53.2251	4.0069	0.0029	54.4518	5.2335	0.0018
Splash	**0.1**	59.8710	11.1168	0.0051	59.9508	11.1960	0.0038
	0.2	57.2349	8.2705	0.0122	57.3663	8.4036	0.0100
	0.4	54.4521	5.0183	0.0392	54.6136	5.2047	0.0334
Lena	**0.1**	56.9295	5.8639	0.0031	57.9482	6.8987	0.0020
	0.2	55.5507	4.4740	0.0036	56.2083	5.1556	0.0024
	0.4	53.8208	2.7469	0.0049	54.1100	3.0486	0.0035
Maze	**0.1**	60.2200	0.0818	0.0032	60.3983	0.2115	0.0019
	0.2	57.8298	0.0711	0.0021	57.7247	0.0882	0.0028
	0.4	54.9573	0.0083	0.0032	54.4086	0.0127	0.0090
Moustache	**0.1**	60.2200	0.0818	0.0032	60.8911	12.0460	0.0025
	0.2	57.8298	0.0711	0.0021	57.9229	8.8618	0.0067
	0.4	54.9573	0.0083	0.0032	54.9325	5.4308	0.0219
Rice	**0.1**	58.4494	5.1920	0.0050	59.0308	5.8078	0.0029
	0.2	56.8061	3.8359	0.0056	57.1316	4.1602	0.0036
	0.4	54.6283	2.1981	0.0073	54.6637	2.2640	0.0052
Phantom	**0.1**	60.7193	2.0332	0.0104	60.9592	2.3314	0.0049
	0.2	57.8902	1.0546	0.0095	58.0407	1.1958	0.0048
	0.4	55.0493	0.4386	0.0083	55.0395	0.4743	0.0054

For testing the effect of the filter size on the performance of the algorithm, another experiment with different filter sizes was conducted. Experiment was applied on rice image with different noise types. The results are shown in Table 5.7 below. The results indicate that the larger the filter size, the higher the MSE is going to reach, which in turn, will decrease the SNR and PSNR values as they are inversely proportional with each other. From results in Table 5.7 it is clear that the best values (highest PSNR & SNR values and lowest MSE value) for both algorithms were recorded with the use of 3×3 filter size.

Table 5.7 PSNR, SNR and MSE comparisons between the 2D-LMS and the proposed 2D ZA-LMS algorithms under different filter sizes ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$).

	Algorithm	**Proposed 2D ZA-LMS**			**2D-LMS**		
Noise	**Filter Size**	**PSNR**	**SNR**	**MSE**	**PSNR**	**SNR**	**MSE**
AWGN (σ^2 =0.1)	**3×3**	**55.1070**	**3.8601**	**0.0023**	**54.9167**	**3.6711**	**0.0034**
	4×4	54.9522	3.7183	0.0029	54.7383	3.4839	0.0039
	5×5	54.6774	3.4478	0.0066	54.4606	3.2345	0.0091
Speckle (σ^2 =0.1)	**3×3**	**56.0540**	**3.0993**	**0.0044**	**55.9747**	**3.0180**	**0.0070**
	4×4	55.9779	3.0359	0.0072	55.8840	2.9446	0.0099
	5×5	55.5332	2.5938	0.0132	55.6514	2.7134	0.0148
Salt&Pepper (Pr =0.1)	**3×3**	**59.0660**	**5.8127**	**0.0029**	**58.3780**	**5.1385**	**0.0049**
	4×4	58.3928	5.1440	0.0055	58.0439	4.7919	0.0077
	5×5	56.9049	3.6802	0.0112	57.3320	4.0964	0.0125

5.6. Conclusion

In this chapter, a new 2D zero-attracting LMS (2D ZA-LMS) algorithm was proposed. The convergence analysis of the proposed 2D ZA-LMS algorithm is derived. The performance of the proposed algorithm was compared to that of the 2D-LMS algorithm with different sparse and non-sparse images. Experiments on different images showed that the proposed 2D ZA-LMS algorithm has shown high capabilities in updating the filter weights along the horizontal and vertical directions with less computational complexity, less time consumption and comparable performance with the 2D-LMS algorithm.

CHAPTER 6

CONCLUSIONS AND FUTURE WORKS

6.1. Conclusions

The adaptive filter is a system that models the relation between the input and output signal by adjusting its coefficients iteratively or recursively based on some adaptive optimization algorithms.

In this dissertation, new and novel adaptive filter algorithms on the light of the available adaptive filter algorithms were introduced. Three different adaptive algorithms based on the least mean square (LMS) algorithm and/or its variants were successfully introduced. These algorithms are: zero attracting mixed norm least mean square (ZA-MN-LMS) algorithm, reweighted zero attracting least mean square (RZA-MN-LMS) algorithm and (2D ZA-LMS) algorithm.

This study started by providing the detailed derivation of the convergence analysis of the mixed-norm least mean square (MN-LMS) algorithm. MN-LMS convergence analysis was derived in both mean and mean square senses.

Given that the system is sparse, the performance of the MN-LMS algorithm can easily be further improved in system identification settings. This improvement can be achieved by imposing a penalty term in its cost function. Based on this observation, new algorithm named zero attracting mixed norm LMS (ZA-MN-LMS) algorithm was proposed in this dissertation. It exploits the sparsity of the system has by adding l_1-norm penalty term to the cost function of the MN-LMS algorithm. This term enables us to attract the zero and/or near-to-zero filter weights to the zero value in a faster manner. However, when the system is near or exactly non-sparse, the performance of the algorithm drops.

To overcome the limitation of this algorithm when the system is near of exactly non-sparse another algorithm named reweighted zero attracting mixed norm LMS (RZA-MN-LMS) was proposed in this dissertation that uses an approximation of l_0-norm penalty term in the cost function of the MN-LMS algorithm. This provides high performance even with completely non-sparse systems. The performances of the proposed ZA-MN-LMS and the RZA-MN-LMS algorithms were investigated experimentally under different filter lengths, sparsity ratios and signal to noise ratios SNRs. The proposed algorithms always showed high performance compared to the original MN-LMS algorithm as shown in results presented in Chapter 4.

The new two-dimensional algorithm presented in Chapter 5 was introduced for improving the 2D-LMS algorithm performance. The new 2D zero-attracting least mean square (2D ZA-LMS) adaptive filter is improving the performance by imposing a sparsity aware l_1-norm penalty term into the cost function of the conventional 2D-LMS algorithm. Moreover, two data reuse configurations namely; rectangular and axial configurations were explained and utilized to boost the filtering process. The convergence analysis of the 2D ZA-LMS algorithm and stability criterion was also derived.

Beside mathematical derivation for the convergence analysis of the proposed algorithms, both algorithms were tested experimentally. Images corrupted by different noise types with different parameters such as variance were used. Both algorithms showed a comparable results both subjectively by human eye inspection and/or objectively using PSNR metric. Still the results showed that the proposed 2D ZA-LMS algorithm is faster than the 2D-LMS algorithm by a high gain factor under different noise types and parameters with image size = 256×256, $\mu = 0.001, \rho = 10^{-4}$, and $\varepsilon = 10$.

Moreover, another experiment regarding the effect of changing the filter size was conducted. The results indicated that filter size will have inverse effect on the SNR and PSNR values. According to the carried out experiment, 3×3 filter size recorded the best results for PSNR, SNR and MSE for both algorithms.

The rest of the obtained results for the comparison of the 2D-LMS and 2D ZA-LMS under different scenarios are shown in the Appendix at the end of the dissertation.

6.2. Future works

Through the proposed algorithms, convergence rate or step size μ was chosen to be fixed. Our proposed algorithms can be investigated with variable step size where it will be updated at each iteration.

Another research direction can be the introduction of two dimensional reweighted zero attracting LMS algorithm. An approximation of l_0-norm penalty term can be imposed in the cost function of the conventional 2D-LMS algorithm. Convergence analysis derivation can also be provided for the new suggested algorithms.

For sparse images restoration purposes, extension of the MN-LMS algorithm to two dimensional case can be studied using zero attracting and/or reweighted zero attracting techniques.

The proposed adaptive filtering algorithms can be useful for enhancing sparse medical images such as X-Ray, ultrasound and magnetic resonance imaging (MRI).

For instance if the breast cancer is in the beginning stage, with image restoration it will be clearer so the tumor region can be detected more efficiently.

The ECG signal of the baby is usually affected by the mother’s heart beats. Considering the mother’s heart beats as a noise, adaptive filters can be utilized for noise cancellation which will help for better detection of any abnormality in the ECG of the baby.

REFERENCES

[1] S. Haykin, "*Adaptive filter theory*", Prentice Hall, Upper Saddle River, NJ, 3rd Edition. 2008.

[2] B. Widrow, "*Adaptive filters*", in Aspects of Network and System Theory, R. Kalman and N. DeClaris, eds., pp. 563–587, Holt, Rinehart, and Winston, New York, 1971.

[3] B. Widrow, P. Mantey, L. Griffiths, and B. Goode, "Adaptive antenna systems", *Proceeding of IEEE,* vol. 55, no. 12, pp. 2143–2159, Dec. 1967.

[4] B. Widrow, J. R. Glover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, Jr., and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications", *Proceeding of IEEE*, vol. 63, no. 12, pp. 1692–1716, Dec. 1975.

[5] B. Widrow and S. D. Stearns, "*Adaptive signal processing*", Prentice-Hall, Upper Saddle River, NJ, 1985.

[6] B. Farhang-Boroujeny, "On statistical efficiency of the LMS algorithm in system modeling," *IEEE Transactions on Signal Processing*, vol. 41, no. 5, pp. 1947–1951, May 1993.

[7] M. Mboup, M. Bonnet, and N. Bershad, "LMS coupled adaptive prediction and system identification: A statistical model and transient mean analysis," *IEEE Transactions on Signal Processing*, vol. 42, no. 10, pp. 2607–2615, Oct. 1994.

[8] S. Theodoridis, "Adaptive filtering algorithms," *18th IEEE Instruments and Measurements Technology Conference,* Budapest, Hungary, vol. 3, pp. 1497–1501. May 2001.

[9] J. Makhoul, "Linear prediction: A tutorial review," *Proceeding of IEEE,* vol. 63, no. 4, pp.561–580, Apr. 1975.

[10] B. Widrow, M. Lehr, F. Beaufays, E. Wan, and M. Bilello, "Learning algorithms for adaptive processing and control," *IEEE Int. Conference on Neural Networks*, San Francisco, CA, , vol. 1, pp. 1–8. Mar./Apr. 1993

[11] P. Prandoni and M. Vetterli, "An FIR cascade structure for adaptive linear prediction," *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2566–2571, Sep. 1998.

[12] S. M. Kuo and D. R. Morgan, "Active noise control: A tutorial review," *Proceeding of IEEE*, vol. 87, no. 6, pp. 943–973, June 1999.

[13] L. C. Godara, "Improved LMS algorithm for adaptive beamforming," *IEEE Transactions on Antennas Propagation*, vol. 38, no. 10, pp. 1631–1635, Oct. 1990.

[14] C. C. Ko, "A simple, fast adaptive algorithm for broad-band null steering arrays," *IEEE Transactions on Antennas Propagation*, vol. 39, no. 1, pp. 122–125, Jan. 1991.

[15] S. Affes, S. Gazor, and Y. Grenier, "An algorithm for multisource beamforming and multitarget tracking," *IEEE Transactions on Signal Processing*, vol. 44, no. 6, pp. 1512–1522, June 1996.

[16] R. W. Lucky, "Automatic equalization for digital communication," *Bell System Technical Journal*, vol. 44, no. 4, pp. 547–588, Apr. 1965.

[17] R. D. Gitlin, E. Y. Ho, and J. E. Mazo, "Passband equalization of differentially phase-modulated data signals," *Bell System Technical Journal*, vol. 52, no. 2, pp. 219–238, Feb. 1973.

[18] S. Qureshi, "Adaptive equalization (data transmission)," *IEEE Communications Magazine*, vol. 20, no. 2, pp. 9–16, Mar. 1982.

[19] J. G. Proakis, "*Digital communications*", 4th edition, chapter 11, McGraw-Hill, New York, 2001.

[20] M. M. Sondhi and A. J. Presti, "A self-adaptive echo canceller," *Bell System Technical Journal*, vol. 46, no. 3, pp. 497–511, Mar. 1967.

[21] V. G. Koll and S. B. Weinstein, “Simultaneous two-way data transmission over a twowire circuit,” *IEEE Transactions on Communications*, vol. COM-21, no. 2, pp. 143–147, Feb. 1973.

[22] D. L. Duttweiler, “A twelve-channel digital echo canceler,” *IEEE Transactions on Communications*, vol. COM-26, no. 5, pp. 647–653, May 1978.

[23] K. C. Ho, “Performance of multiple LMS adaptive filters in tandem,” *IEEE Transactions on Signal Processing*, vol. 49, no. 11, pp. 2762–2773, Nov. 2001.

[24] L. J. Griffiths, “Rapid measurement of digital instantaneous frequency,” *IEEE Transactions on Acoustic. Speech Signal Processing*, vol. ASSP-23, no. 2, pp. 207–222, Apr. 1975.

[25] J. R. Zeidler, E. H. Satorius, D. M. Chabries, and H. T. Wexler, “Adaptive enhancement of multiple sinusoids in uncorrelated noise,” *IEEE Transactions on Acoustic. Speech Signal Processing*, vol. ASSP-26, no. 3, pp. 240–254, June 1978.

[26] J. R. Treichler, “Transient and convergent behavior of the adaptive line enhancer,” *IEEE Transactions on Acoustic. Speech Signal Processing*, vol. ASSP-27, no. 1, pp. 53–62, Feb. 1979.

[27] J. T. Rickard, J. R. Zeidler, M. J. Dentino, and M. Shensa, “A performance analysis of adaptive line enhancer-augmented spectral detectors,” *IEEE Transactions on Circuits & Systems*, vol. CAS-28, no. 6, pp. 534–541, June 1981.

[28] N. J. Bershad and O. M. Macchi, “Adaptive recovery of a chirped sinusoid in noise, Pt. II: Performance of the LMS algorithm,” *IEEE Transactions on Signal Processing*, vol. 39, no. 3, pp. 595–602, Mar. 1991.

[29] M. Ghogho, M. Ibnkahla, and N. J. Bershad, “Analytic behavior of the LMS adaptive line enhancer for sinusoids corrupted by multiplicative and additive noise,” *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2386–2393, Sept. 1998.

[30] R. L. Campbell, N. H. Younan, and J. Gu, “Performance analysis of the adaptive line enhancer with multiple sinusoids in noisy environment,” *Signal Processing,* vol. 82, pp. 93–101, Jan. 2002.

[31] B. Widrow and E. Walach, “*Adaptive inverse control*”, Prentice-Hall, Upper Saddle River, NJ, 1996.

[32] L.C. Wood & S. Treitel. “Seismic signal processing,” in *Proceedings of IEEE* , vol.63, no.4, pp.649-661, April 1975

[33] M. L. Honig & D. G. Messerschmitt, “*Adaptive filters: Structures, algorithms, and applications,*” The Kluwer International Series in Engineering and Computer Science; SECS 1, Boston: Kluwer, 1984

[34] P.S.R. Diniz. “*Adaptive filters: Algorithms and practical implementation,*” 3rd edition, springer. 2008.

[35] B. D. Van Veen and K. M. Buckley, “Beamforming: a versatile approach to spatial filtering,” *IEEE Acoustic, Speech, Signal Processing Magazine*, vol. 37, pp. 4-24, April 1988.

[36] L.H. Rosenthal, R.W. Schafer, L.R. Rabiner. “An algorithm for locating the beginning and end of an utterance using adpcm coded speech,” *Bell System Technical Journal,* The , vol.53, no.6, pp.1127-1135, July-Aug. 1974

[37] M.V. Eyuboglu & S.U.H. Qureshi. “Reduced-state sequence estimation with set partitioning and decision feedback,” *IEEE Transactions on Communications* , vol.36, no.1, pp.13-20, Jan 1988

[38] M. Sondhi. “The history of echo cancellation”, *IEEE Signal Processing Magazine,* vol. 23, no.5, pp. 95 – 102. 2006

[39] S. Gay & J. Benesty. “*Acoustic signal processing for telecommunications*”, Kluwer Academic Publishers. 2000

[40] S. Haykin. “*Adaptive filter theory,*” 4th edn, Prentice-Hall. 2002

[41] A. Sayed. "*Fundamentals of adaptive filtering*," Wiley-IEEE Press. 2003

[42] J. Benesty, T. Gaensler, D. Morgan, M. Sondhi & S. Gay. "*Advances in Network and Acoustic Echo Cancellation*," Springer-Verlag. 2001

[43] J. Greenberg. "Modified LMS algorithms for speech processing with an adaptive noise canceller," *IEEE Transactions on Signal Processing*, vol. 6, no. 4, pp. 338–351. 1998

[44] J. Ni & F. Li. "Adaptive combination of subband adaptive filters for acoustic echo cancellation", *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1549 – 1555. 2010

[45] E. Krishna, M. Raghuram, K. Madhav & K. Reddy. "Acoustic echo cancellation using a computationally efficient transform domain lms adaptive filter", *10th International Conference on Information Sciences Signal Processing and their Applications* (ISSPA), Kuala Lumpur, Malaysia, pp. 409–4, 2010

[46] P. Marques. "*Long distance echo cancellation using centered short length transversal filters*," Master's thesis, Instituto Superior Técnico, Lisbon, Portugal. 1997

[47] M. Sondhi & D. Berkeley. "Silencing echoes on the telephone network", *Proceedings of IEEE*, vol. 68, no. 8, pp. 948–963. 1980

[48] B. Widrow and M.E. Hoff, "Associative storage and retrieval of digital information in networks of adaptive 'neurons'," *Biological Prototypes and Synthetic Systems*, vol. 1, pp.160, 1962.

[49] N.J. Bershad, "Analysis of the normalized LMS algorithm with Gaussian inputs," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 4, pp. 793-806, Aug 1986

[50] G.C. Goodwin, Teoh, E.K.; Elliott, H., "Deterministic convergence of a self-tuning regulator with covariance resetting," *IEE Proceedings in Control Theory and Applications*, D , vol. 130, no. 1, pp. 6-8, January 1983

[51] S. H. Cho and V. J. Mathews, "Tracking analysis of the sign algorithm in nonstationary environments," *IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. 38, pp. 2046-2057, Dec. 1990.

[52] J. C. M. Bermudez and N. J. Bershad, "A nonlinear analytical model for the quantized LMS algorithm: The arbitrary step size case," *IEEE Transaction on Signal Processing*, vol. 44, pp. 1175-1183, May 1996.

[53] S. S. Narayan, A. M. Peterson, and M. J. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP-31, pp. 609-615, June 1983.

[54] G. Goodwin, M. Zarrop and R., Payne. "Coupled design of test signals, sampling intervals, and filters for system identification," *IEEE Transactions on Automatic Control*, vol.19, no.6, pp.748-752, Dec 1974

[55] J. Benesty, C. Paleologu, S. Ciochin, "Regularization of the RLS algorithm", *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 94, no. 8, pp. 1628-1629, 2011

[56] C. Boukis, D.P. Mandic, A.G. Constantinides, "A Generalised Mixed Norm Stochastic Gradient Algorithm," *15th International Conference on Digital Signal Processing*, 2007, vol., no., pp.27-30, 1-4 July 2007

[57] Y. Chen, Y. Gu, and A. O. Hero. "Sparse LMS for system identification". *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 3125–3128. 2009

[58] J.Yang, "*Adaptive filter design for sparse signal estimation*", a dissertation submitted to the faculty of the graduate school of the university of Minnesota. 2011.

[59] N. Czink, X. Yin, H. OZcelik, M. Herdin, E. Bonek, and B. Fleury. "Cluster characteristics in a MIMO indoor propagation environment". *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1465–1475. 2007

[60] L. Vuokko, V.M. Kolmonen, J. Salo, and P. Vainikainen. “Measurement of large-scale cluster power characteristics for geometric channel models”. *IEEE Transactions on Antennas and Propagation*; vol. 55, no. 11, pp. 3361–3365, 2007

[61] S. Kawamura and M. Hatori, “A tap selection algorithm for adaptive filters,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 11, pp. 2979– 2982, 1986

[62] J. Homer, I. Mareels, R.R. Bitmead, B. Wahlberg, and A. Gustafsson, “LMS estimation via structural detection,” *IEEE Transactions on Signal Processing*, vol. 46, pp. 2651–2663, Oct. 1998.

[63] Y. Li, Y. Gu, and K. Tang, “Parallel NLMS filters with stochastic active taps and step-sizes for sparse system identification,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 109–112, 2006

[64] D.M. Etter, “Identification of sparse impulse response systems using an adaptive delay filter,” *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1169–1172. 1985

[65] M. Godavarti and A. O. Hero, “Partial update LMS algorithms,” *IEEE Transactions on Signal Processing*, vol. 53, pp. 2382–2399, 2005

[66] R.K; Martin, W.A. Sethares, R.C. Williamson and C.R. Jr. Johnson. “Exploiting sparsity in adaptive filters,” *IEEE Transactions on Signal Processing*, vol.50, no.8, pp.1883-1894, Aug 2002

[67] A. Cichocki, S.Amari “*Adaptive blind signal and image processing: learning algorithms and applications*,” John Wiley & Son, 2002.

[68] A. Sugiyama, H. Sato, A. Hirano and S. Ikeda. “A fast convergence algorithm for adaptive FIR filters under computational constraint for adaptive tap-position control,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 43, no. 9, pp. 629-636, Sep 1996

[69] L. Rey Vega, H. Rey, J. Benesty and S. Tressens. “A Family of robust algorithms exploiting sparsity in adaptive filters,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol.17, no.4, pp.572-581, May 2009

[70] R. Tibshirani, “Regression shrinkage and selection via the lasso*,” Journal of the Royal Statistical Society: Series B*, vol. 58, pp. 267–288, 1996

[71] Y. Chen, Y. Gu, and A. O. Hero. “Sparse LMS for system identification”. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, pp. 3125–3128. 2009

[72] G. Gui, W. Peng and F. Adachi. “Improved adaptive sparse channel estimation based on the least mean square algorithm”. *IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, Shanghai, China, April 2013

[73] E.J. Candes, M.B. Wakin and S.P. Boyd. “Enhancing sparsity by reweighted l_1 minimization”. *Journal of Fourier Analysis and Applications* 2008; vol. 14, no. (5–6), pp. 877–905

[74] O. Taheri and S.A. Vorobyov. “Sparse channel estimation with *lp*-norm and reweighted L1-norm penalized least mean square”. *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*; pp. 2864–2867, 2011

[75] Y. Gu, J. Jin, and S. Mei. “l_0-Norm constraint LMS algorithm for sparse system identification”. *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 774–777, 2009

[76] G. Su, J. Jin, Y. Gu, and J. Wang. “Performance analysis of l_0-norm constraint least mean square algorithm”. *IEEE Transactions on Signal Proecessing*, vol. 60, no. 5, pp. 2223–2235. 2012

[77] L. Mancera, J. Portilla, “l_0-Norm-Based Sparse Representation Through Alternate Projections,” *2006 IEEE International Conference on Image Processing*, , vol., no., pp.2089-2092, Oct. 2006

[78] G. Deng. “Partial update and sparse adaptive filters,” *IET Signal Processing*, , vol.1, no.1, pp.9-17, March 2007

[79] Y. Li, Y. Gu, K. Tang, "Parallel NLMS Filters with Stochastic Active Taps and Step-Sizes for Sparse System Identification," *2006 IEEE International Conference on Acoustics, Speech and Signal Processing*, vol.3, no., pp.III-III, 14-19 May 2006

[80] J. Arenas-Garcia, A.R. Figueiras-Vidal, "Adaptive Combination of Proportionate Filters for Sparse Echo Cancellation," *IEEE Transactions on Audio, Speech, and Language Processing*, , vol.17, no.6, pp.1087-1098, Aug. 2009

[81] K. Shi, P. Shi, "Convergence analysis of sparse LMS algorithms with l_1-norm penalty based on white input signal", *Signal Processing*, vol. 90, no. 12, December 2010, pp. 3289-3293.

[82] O. Taheri and S. A. Vorobyov, "Reweighted l_1-norm penalized LMS for sparse channel estimation and its analysis," *Signal Processing*, vol. 104, pp. 70-79, 2014.

[83] M. Hadhoud and D. Thomas, "The two-dimensional adaptive LMS (TDLMS) algorithm", *IEEE International Symposium on Circuits and Systems - ISCAS* , vol. 35, pp. 485-494, 1988.

[84] J. Chambers and A. Avlonitis, "A robust mixed-norm adaptive filter algorithm," *IEEE Signal Processing Letters*, vol. 4, no. 2, pp .46-48, 1997.

[85] Y. S. Choi, "A new subband adaptive filtering algorithm for sparse system identification with impulsive noise," *Journal of Applied Mathematics*, vol. 2014, pp. 1-7, 2014.

[86] G. Rombouts, "*Adaptive filtering algorithms for acoustic echo and noise cancellation*," Katholieke Universiteit Leuven Ph.D. Thesis, 2003.

[87] D.R. Morgan, S.Craig, "Real-time adaptive linear prediction using the least mean square gradient algorithm," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.24, no.6, pp.494-507, 1976.

[88] A. Mader, H. Puder and G. U. Schmidt, "Step-size control for acoustic echo cancellation filters-an overview," *Signal Processing*, vol. 80, pp. 1697–1719, 2000.

[89] W. Tianhui, M. Liyuan, L. Yongjun an D. Yonggang, "An improved variable step size LMS adaptive spectral-line enhancement algorithm and its simulation," *Proceedings*

of the International Conference on Pervasive Computing, Signal Processing and Applications (PCSPA 2010), pp. 727-730, 2010.

[90] G. Su, J. Jin, Y. Gu and J. Wang, "Performance analysis of l_0-norm constraint least mean square algorithm," *IEEE Transaction on Signal Processing*, vol. 60, no. 5, pp. 2223-2235, 2012.

[91] B. K. Das, M. Chakraborty and S. Banerjee, "Adaptive identification of sparse systems with variable sparsity," *IEEE International Symposium on Circuits and Systems* (ISCAS2011), pp. 1267-1270, 2011.

[92] F. Y. Wu and F. Tong, "Gradient optimization p-norm-like constraint LMS algorithm for sparse system estimation," *Signal Processing*, vol. 93, no. 4, pp. 967–971, 2013.

[93] J. Jin, Q. Qu and Y. Gu, "Robust zero-point attraction least mean square algorithm on near sparse system identification," *IET Signal Processing*, vol. 7, no. 3, pp. 210-218, 2013.

[94] M.S.E. Abadi, and S. Nikbakht, "Image denoising with two-dimensional adaptive filter algorithms." *Iranian Journal of Electrical & Electronic Engineering*, vol.7, no.2, pp. 84-105, 2011.

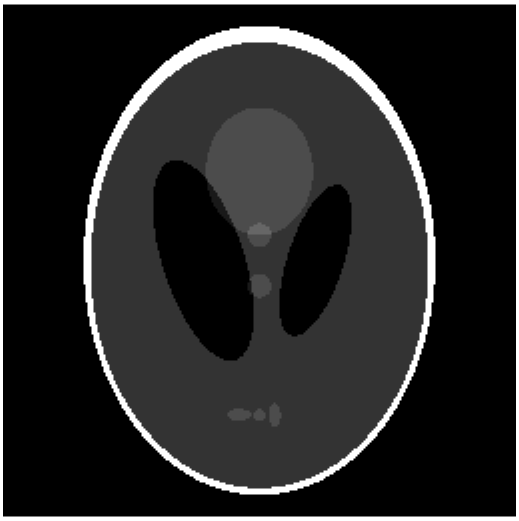
[95] G. Masui, K. Nishikawa and H. Kiya, "2D blind system identification using adaptive algorithms," In Proceedings: *TENCON 2000*, Kuala Lumpur, vol. 2, no.11, pp.77-79, 2000.

[96] P. S. Kumar, and R. Sumit. "Two-dimensional equalization: Theory and applications to high density magnetic recording," *IEEE Transactions on Communications*, vol. 42, no.234, pp. 386-395, 1994.

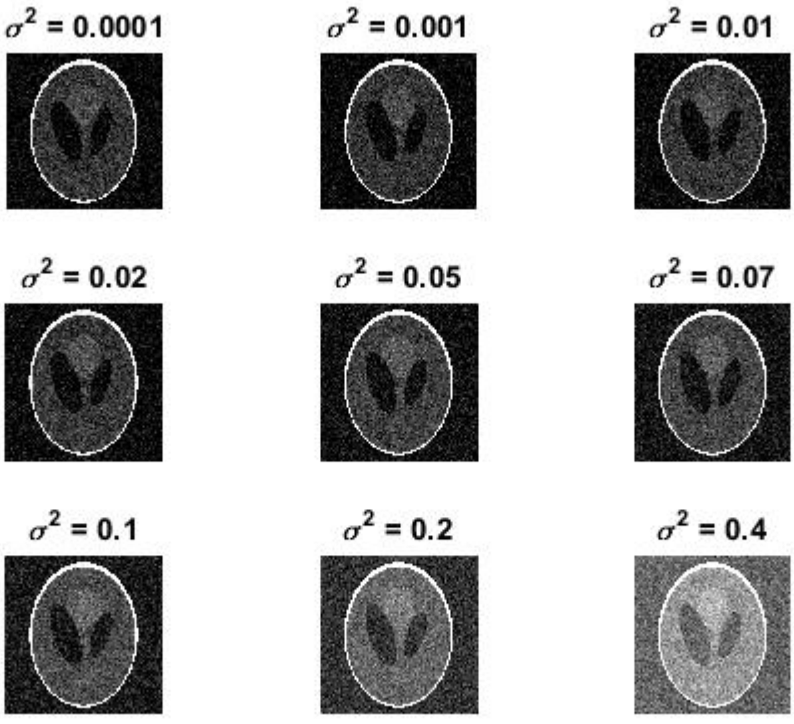
APPENDIX A

A.1. Extra Simulation Results

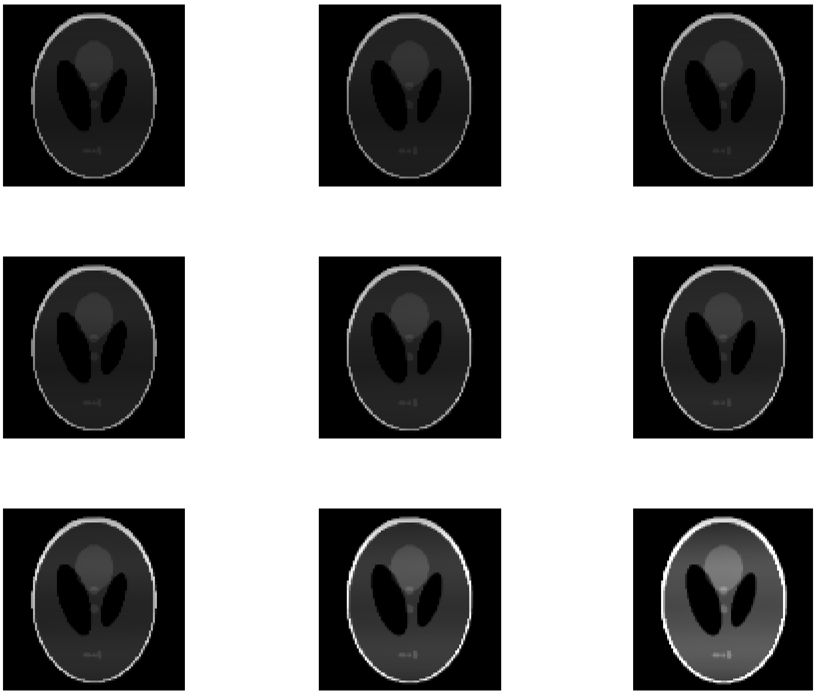
Comparison between 2D-LMS & 2D ZA-LMS algorithms on different images using AWGN noise with different variance values and zero mean.



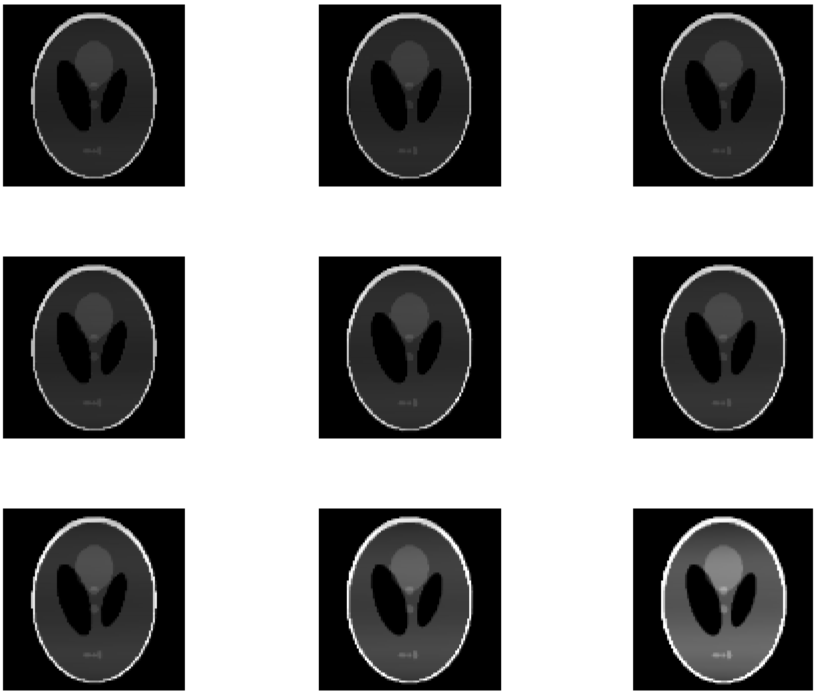
Phantom Image



Phantom Image corrupted by AWGN with different variance values and zero mean

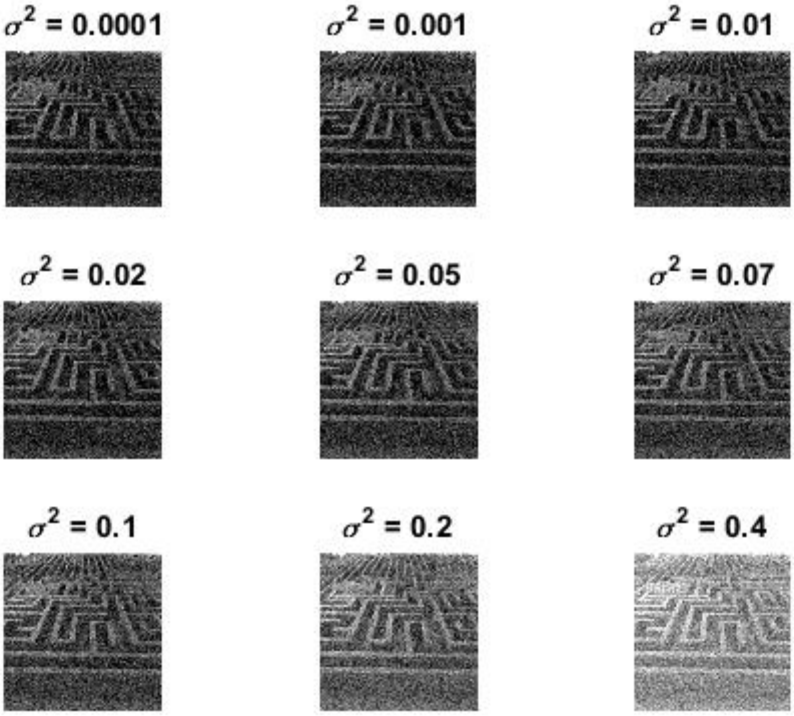


Denoised images corresponding to noisy Phantom images in previous figure using 2D ZA-LMS
$(\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10)$

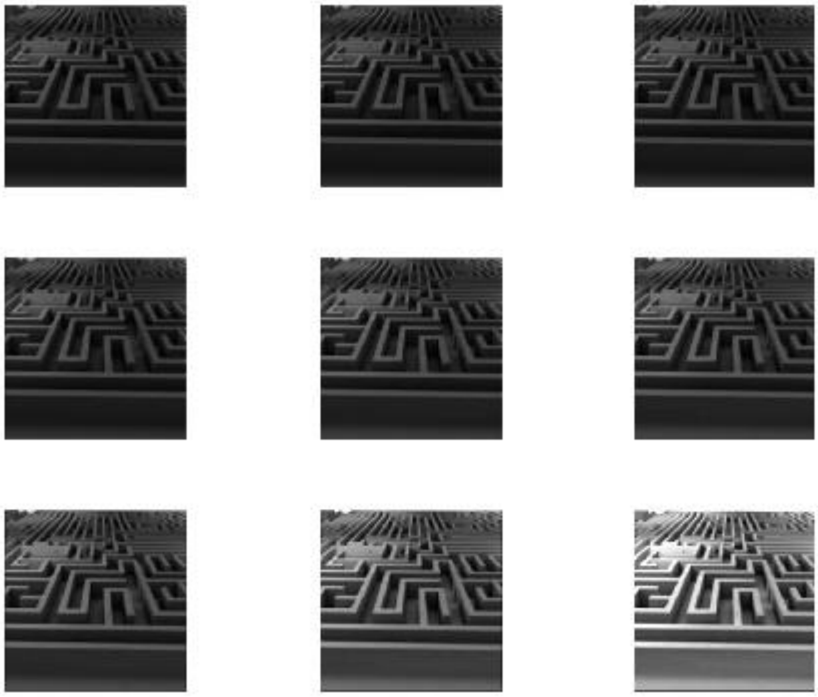


Denoised images corresponding to noisy Phantom images in previous figure using 2D-LMS
$(\mu = 0.001)$

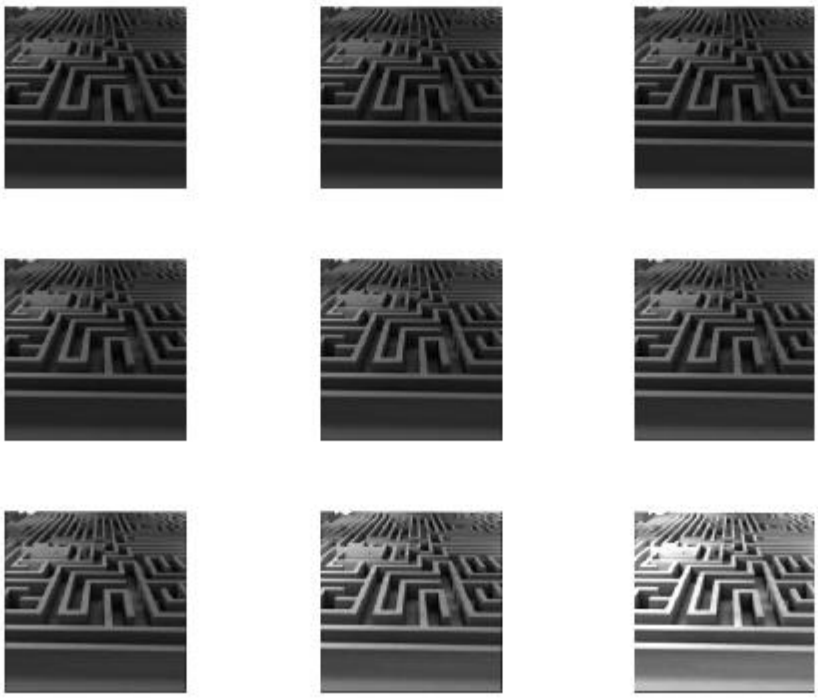
Maze Image



Maze Image corrupted by AWGN with different variance values and zero mean

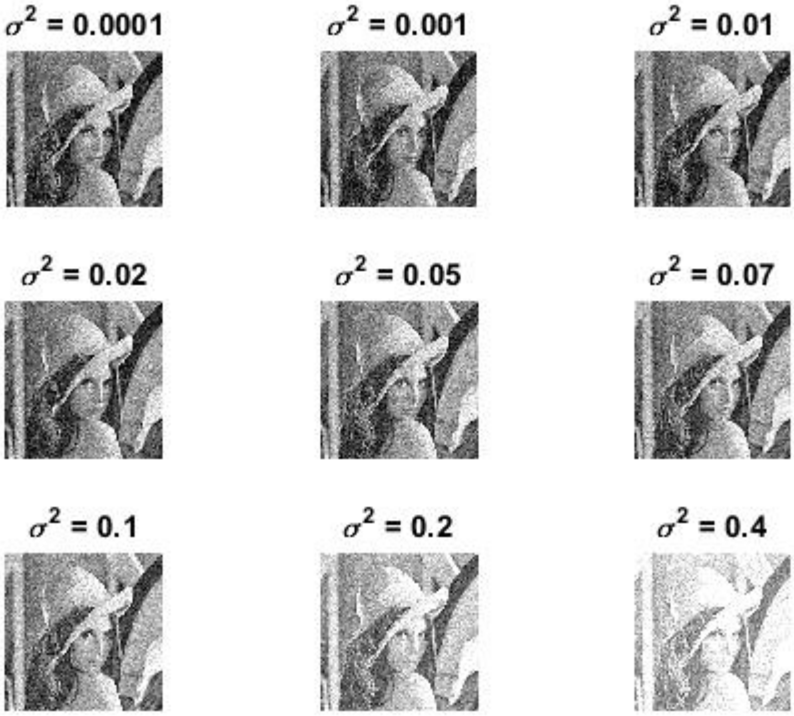


Denoised images corresponding to noisy maze images in previous figure using 2D ZA-LMS ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$)



Denoised images corresponding to noisy maze images in previous figure using 2D-LMS ($\mu = 0.001$)

Lena Image

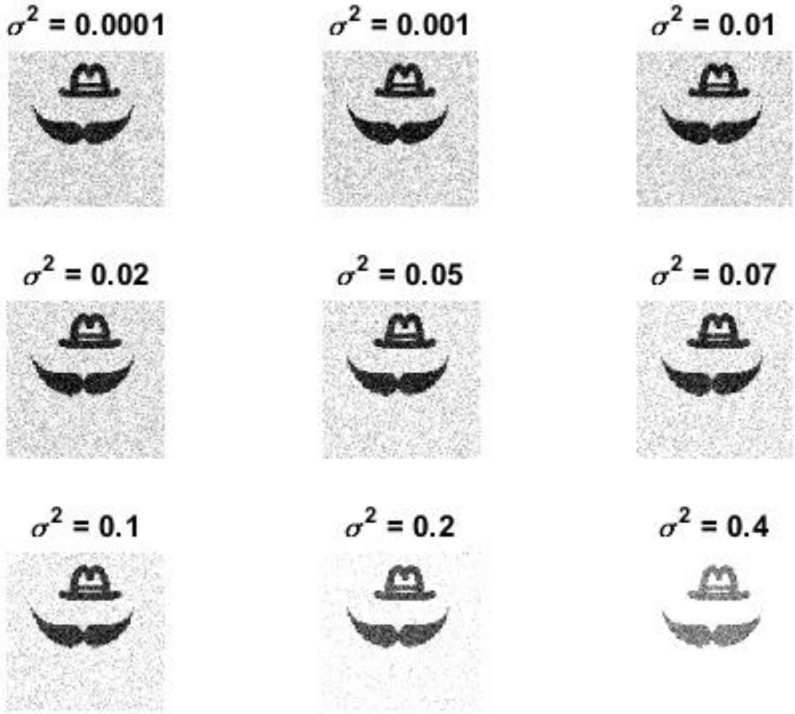


Lena Image corrupted by AWGN with different variance values and zero mean

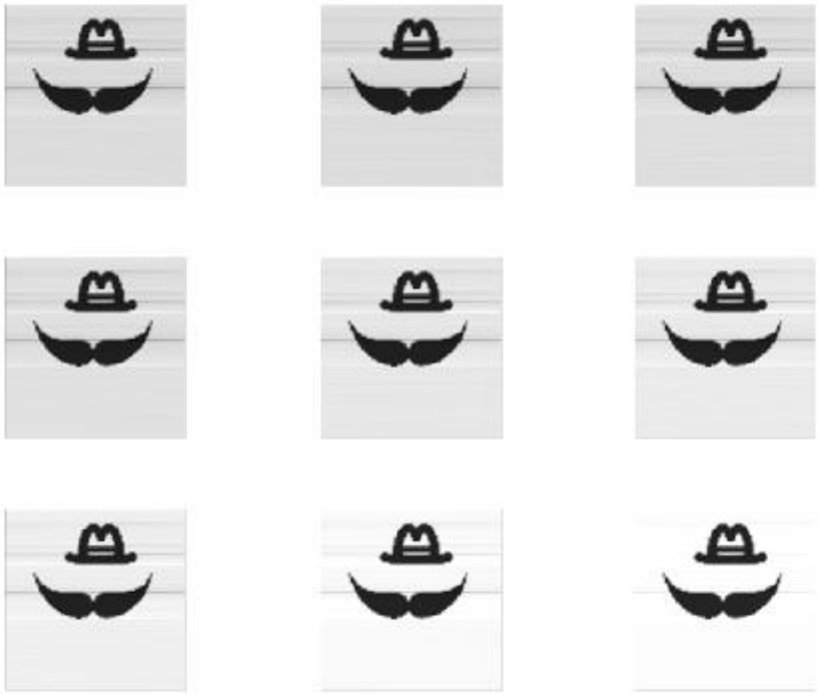
Restored images corresponding to noisy Lena images in previous figure using 2D ZA-LMS
$(\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10)$

Restored images corresponding to noisy Lena images in previous figure using 2D-LMS
$(\mu = 0.001)$

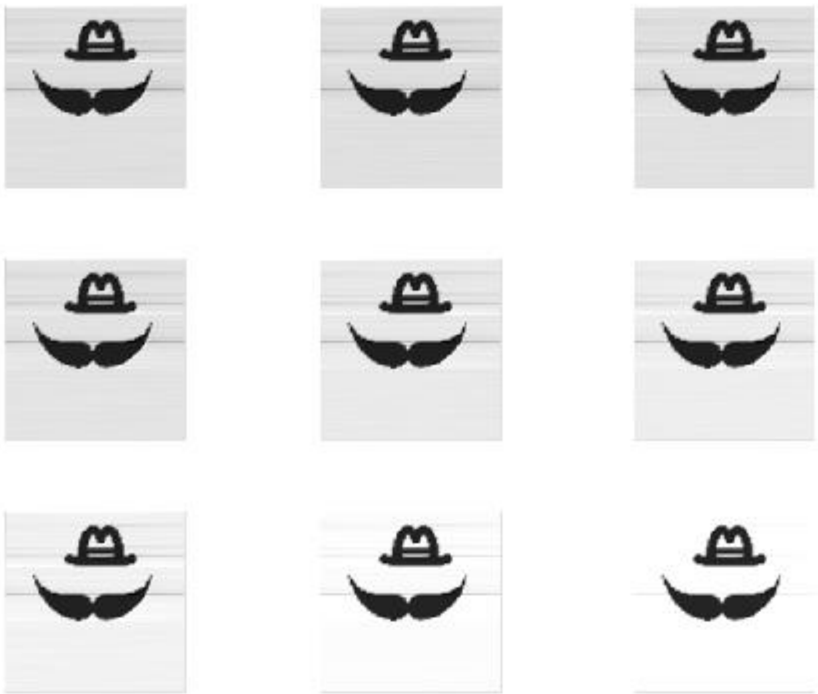
Moustache Image



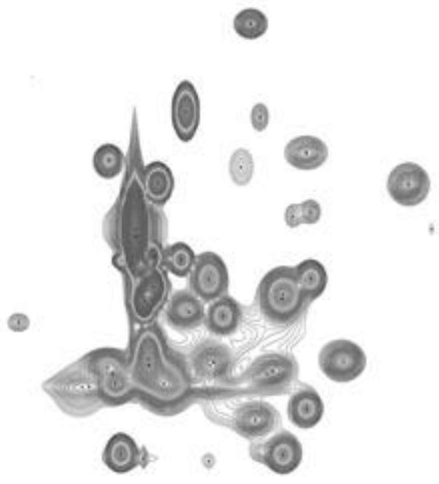
Moustache Image corrupted by AWGN with different variance values and zero mean



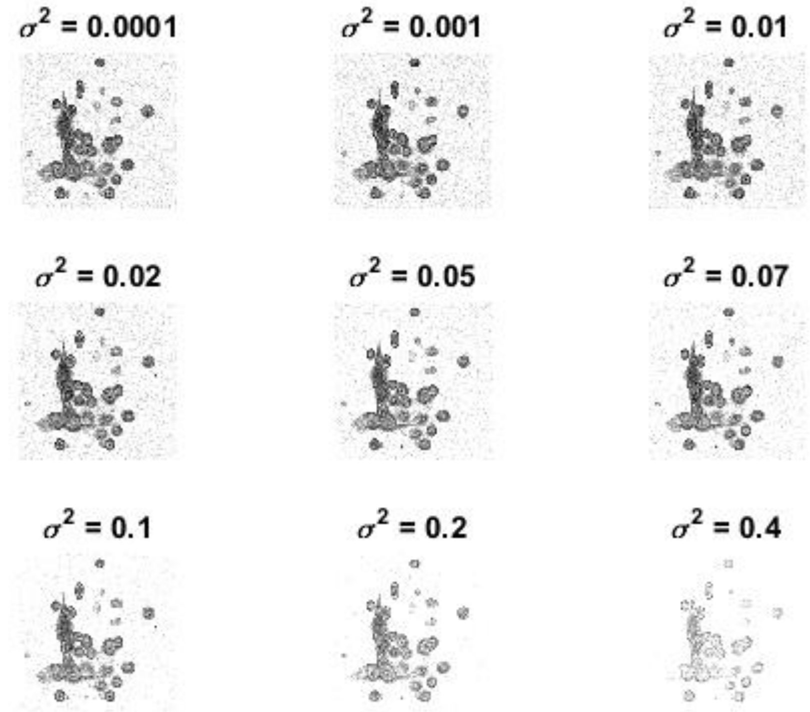
Restored images corresponding to noisy moustache images in previous figure using 2D ZA-LMS ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$)



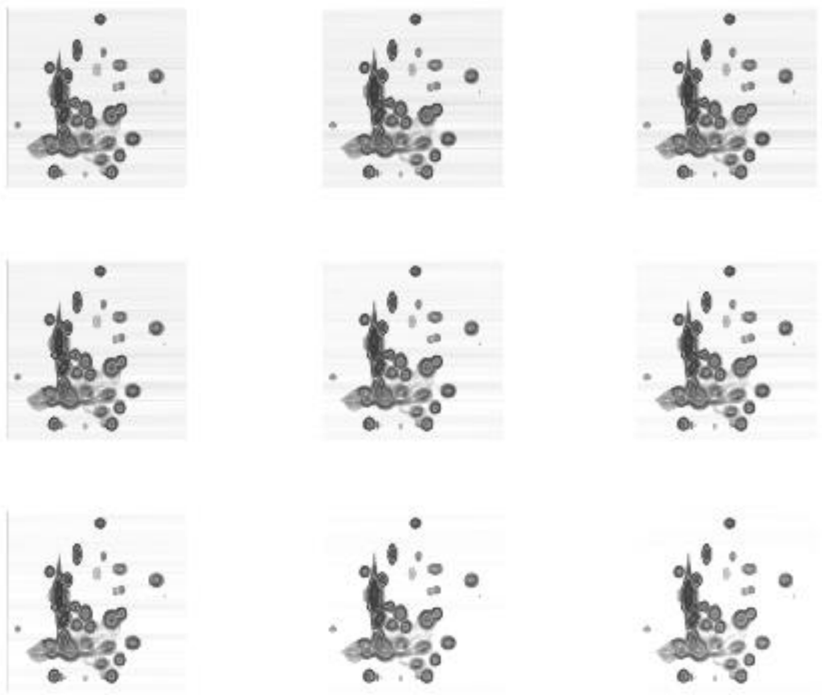
Restored images corresponding to noisy moustache images in previous figure using 2D-LMS ($\mu = 0.001$)



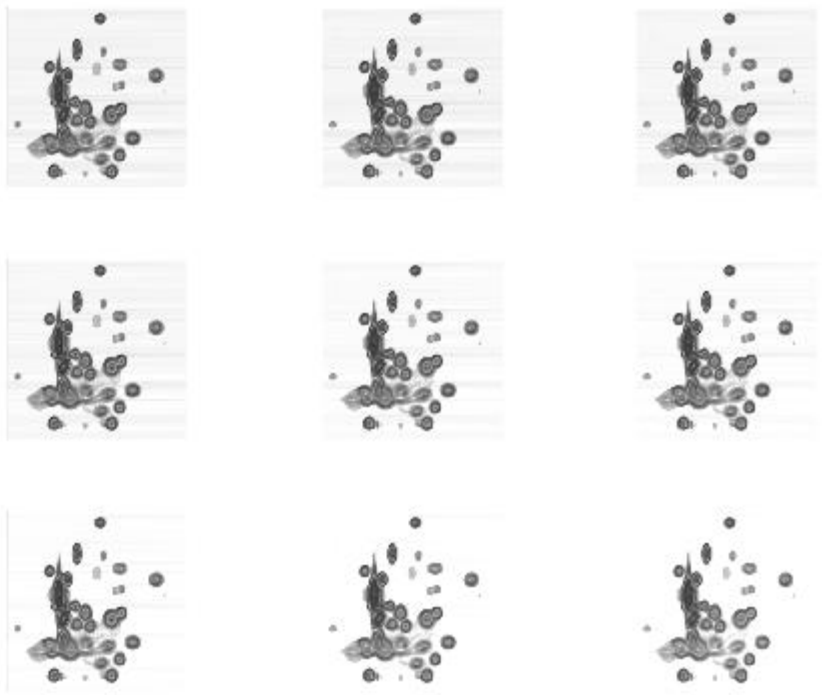
Splash Image



Splash Image corrupted by AWGN with different variance values and zero mean

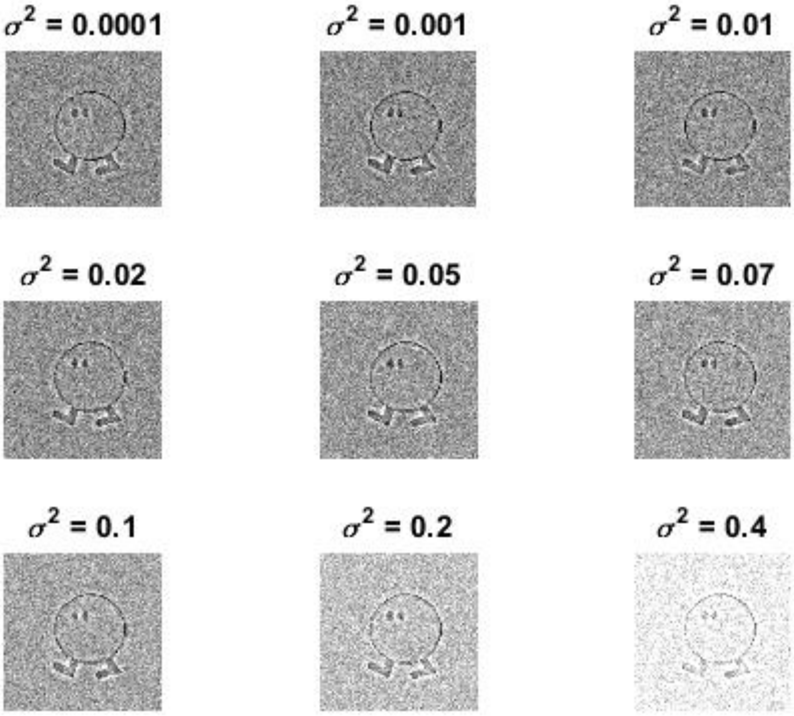


Restored images corresponding to noisy splash images in previous figure using 2D ZA-LMS ($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$)

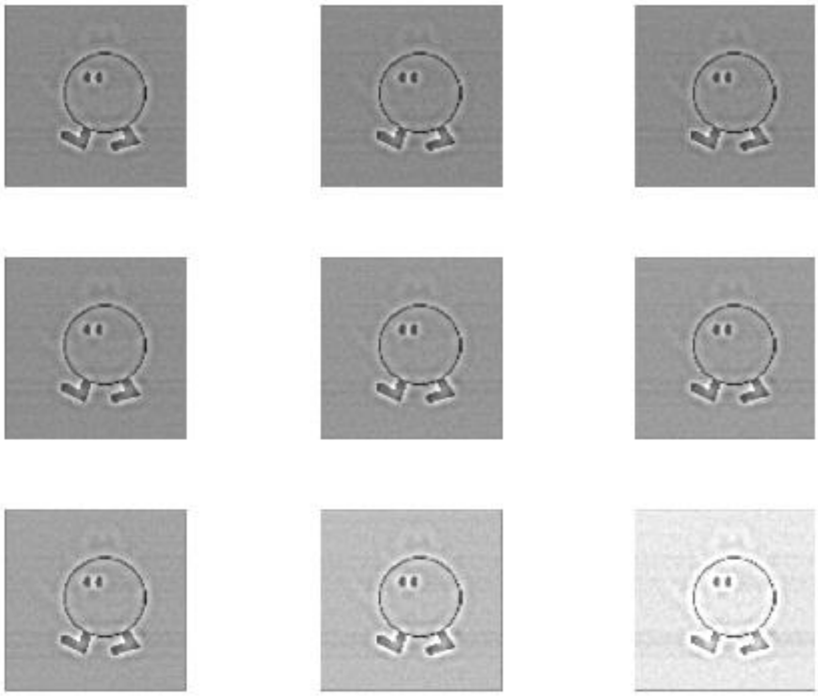


Restored images corresponding to noisy splash images in previous figure using 2D-LMS ($\mu = 0.001$)

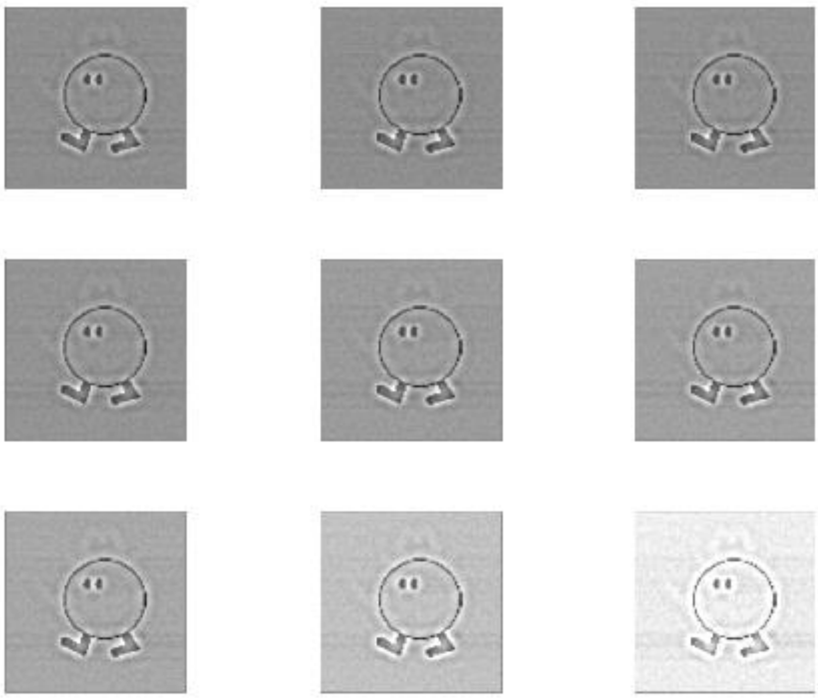
Man Image



Man Image corrupted by AWGN with different variance values and zero mean



Restored images corresponding to noisy man images in previous figure using 2D ZA-LMS
($\mu = 0.001, \rho = 10^{-4}, \varepsilon = 10$)



Restored images corresponding to noisy man images in previous figure using 2D-LMS
($\mu = 0.001$)