

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2004

Adaptive noise cancellation using multichannel lattice structure.

Qing Xie

University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

Recommended Citation

Xie, Qing, "Adaptive noise cancellation using multichannel lattice structure." (2004). *Electronic Theses and Dissertations*. 891.

<https://scholar.uwindsor.ca/etd/891>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

Adaptive Noise Cancellation using Multichannel Lattice Structure

by
Qing Xie

A Thesis
Submitted to the Faculty of Graduate Studies and Research
through the Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada

2004

© 2004 Qing Xie



National Library
of Canada

Bibliothèque nationale
du Canada

Acquisitions and
Bibliographic Services

Acquisitons et
services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 0-612-92483-1
Our file *Notre référence*
ISBN: 0-612-92483-1

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this dissertation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de ce manuscrit.

While these forms may be included in the document page count, their removal does not represent any loss of content from the dissertation.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

ABSTRACT

This thesis presents a multichannel adaptive noise cancellation technique (MCLS) for cancelling the noise over nonlinear transmission channel. The technique applies to the situation in which the reference signal and noisy primary signal are collected simultaneously. The coefficients of the multichannel multiple regression transversal filter are modified adaptively according to the backward prediction error vector generated from the multichannel adaptive lattice predictor. This multichannel adaptive noise cancellation procedure involves the NLMS adaptive algorithm.

The performance of the new technique using different types of transmission channels, different types of reference inputs and different types of noise-free primary inputs are examined analytically. The new approach is experimentally shown to have better noise cancellation performance than the existing single-channel adaptive lattice noise cancellation algorithm (SCLS) over nonlinear transmission channel case, especially in low input SNR situation.

ACKNOWLEDGEMENT

I would like to acknowledge Professor H. K. Kwan for introducing me the field of lattice modeling and for suggesting me a study of multichannel lattice structure for adaptive noise cancellation as the theme of my master thesis. This thesis would not have been possible without the guidance of my advisor - Professor H. K. Kwan. I would like to thank him for providing me with the opportunity to work over the past two years as a Research Assistant in the Intelligent Signal Processing Laboratory, Department of Electrical and Computer Engineering. His insight, support and enthusiasm are very much appreciated.

I would like to thank my entire family and all my friends, especially my beloved wife – Mrs. Jian Yang, for supporting and standing by me through the years.

TABLE OF CONTENTS

Abstract.....	ii
Acknowledgement.....	iv
List of Tables	viii
List of Figures	ix
List of Symbols	xii
List of Acronyms.....	xiv
1. Introduction.....	1
1.1 Noise Cancellation.....	1
1.2 Adaptive Filter	3
1.2.1 Adaptive Filter Structure	5
1.2.1.1 Linear Structure.....	5
1.2.1.2 Nonlinear Structure	8
1.2.2 Adaptive Algorithm.....	15
1.2.2.1 LMS Algorithm	16
1.2.2.2 NLMS Algorithm	17
1.2.2.3 RLS Algorithm.....	18
1.3 Comparison.....	19
1.3.1 Adaptive Filtering vs. Fixed Filtering	19
1.3.2 Lattice Structure vs. FIR/IIR Structures.....	20
1.3.3 LMS/NLMS Algorithms vs. RLS Algorithm	21
1.3.4 Linear Structure vs. Nonlinear Structure.....	21
1.3.5 Volterra Nonlinear Structure vs. Multichannel Nonlinear Structure	22
1.4 Motivation	22
2 SCLS for ANC	27
2.1 Lattice Stage.....	27
2.2 Structure and Algorithm.....	28
3 MCLS for ANC.....	33

3.1	Problem Definition and Assumption	33
3.2	MCLS	35
3.2.1	SIMO Module	36
3.2.2	MALP Module	39
3.2.3	MMRTF Module	42
3.2.4	Summary of the Algorithm	45
3.3	Relationship between MCSL and SCLS	46
4	Simulation and Comparison	48
4.1	Definitions	49
4.1.1	Measurements	49
4.1.1.1	RE	49
4.1.1.2	MSE	50
4.1.1.3	SNR	50
4.1.2	Transmission Channels	51
4.1.3	Signals	53
4.1.3.1	Signal Type	53
4.1.3.2	Noise-free Primary Signal	56
4.1.3.3	Reference Signal	56
4.1.3.4	Noisy Primary Signal	57
4.1.4	Title	58
4.2	Method of Analysis	59
4.3	Simulations and Analysis	65
4.3.1	Second-order Nonlinear Transmission Channel	65
4.3.1.1	Simulation Scenarios	65
4.3.1.2	Analysis over Channel #1	68
4.3.1.3	Analysis over Channel #2	74
4.3.1.4	Analysis over Channel #3	78
4.3.1.5	Analysis over Channel #4	81
4.3.1.6	Summary on Second-order Nonlinear Transmission Channels	87
4.3.2	Linear Transmission Channel	87
4.3.3	Third-order Nonlinear Transmission Channel	91

5	Conclusions and Future Work.....	100
	Reference	102
	Vita Auctoris	106

LIST OF TABLES

Table 3-1: Total channels of different order (P) and different maximum delay ($M-1$)	39
Table 4-1: Description of the transmission channels used in the simulations	52
Table 4-2: Definition of voice signals	54
Table 4-3: Description of the noise-free primary inputs used in the simulations	57
Table 4-4: Description of the reference inputs used in the simulations	57
Table 4-5: Description of the points and input/output SNR levels in Figure 4-6	63
Table 4-6: Simulation titles using second-order nonlinear channel #1	66
Table 4-7: Simulation titles using Second-order nonlinear channel #2	66
Table 4-8: Simulation titles using Second-order nonlinear channel #3	67
Table 4-9: Simulation titles using Second-order nonlinear channel #4	68
Table 4-10: Simulation results using Second-Non channel #1	71
Table 4-11: Simulation results using Second-Non channel #2	77
Table 4-12: Simulation results using Second-Non channel #3	80
Table 4-13: Simulation results using Second-Non channel #4 with $L=4$	83
Table 4-14: Simulation results using Second-Non channel #4 with $L=13$	85
Table 4-15: Simulation titles using linear channel #1	88
Table 4-16: Simulation titles using linear channel #2	88
Table 4-17: Simulation results using Linear channels	90
Table 4-18: Simulation titles using Third-Non channel #1	92
Table 4-19: Simulation titles using Third-Non channel #2	92
Table 4-20: Simulation results using Third-Non channel #1 and #2	94

LIST OF FIGURES

Figure 1-1: Two-microphone ANC System	2
Figure 1-2: The general adaptive filtering problem	4
Figure 1-3: Structure of an FIR filter	6
Figure 1-4: Structure of a canonical-form IIR filter	6
Figure 1-5: Overall structure of lattice predictor and transversal filter	7
Figure 1-6: Structure of l -th lattice stage	8
Figure 1-7: Truncated Volterra system of order $P=2$ and maximum delay $M-1=2$	10
Figure 1-8: A block diagram of the Adaptive Volterra Filter	11
Figure 1-9: Block diagram of the Second-order Volterra Adaptive Lattice Filter	12
Figure 1-10: Single-input-multiple-output system	13
Figure 1-11: Multichannel adaptive lattice equalizer	15
Figure 1-12: Configuration of an adaptive filter	16
Figure 2-1: Detailed structure of l -th lattice stage	27
Figure 2-2: Lattice predictor of SCLS	28
Figure 2-3: Detailed structure of SCLS	31
Figure 3-1: Problem model for MCLS proposal	34
Figure 3-2: Block diagram of MCLS	36
Figure 3-3: Detailed structure of SIMO	37
Figure 3-4: Structure of j -th MALP stage	40
Figure 3-5: Structure of j -th MMRTF stage	43
Figure 4-1: Test model for the simulation and comparison	48
Figure 4-2: Frequency response of single/multiple-frequency signal	54

Figure 4-3: Frequency response of WGN and CGN	56
Figure 4-4: RE in Second-Non-CH1-O1-N1 when $\text{SNR}_{\text{input}} = 10\text{dB}$	59
Figure 4-5: MSE in Second-Non-CH1-O1-N1 when $\text{SNR}_{\text{input}} = 10\text{dB}$	60
Figure 4-6: Input/Output SNR in Second-Non-CH1-O1-N1	61
Figure 4-7: RE in Second-Non-CH1-O2-N2 when input $\text{SNR}_{\text{input}} = 0\text{dB}$	69
Figure 4-8: MSE in Second-Non-CH1-O2-N2 when $\text{SNR}_{\text{input}} = 0\text{dB}$	69
Figure 4-9: Input/Output SNR in Second-Non-CH1-O2-N2	70
Figure 4-10: SNR Improvement over Second-Non channel #1	72
Figure 4-11: Max output SNR Decrease over Second-Non channel #1	73
Figure 4-12: RE in Second-Non-CH2-O2-N2 when $\text{SNR}_{\text{input}} = 10\text{dB}$	74
Figure 4-13: MSE in Second-Non-CH2-O2-N2 when $\text{SNR}_{\text{input}} = 10\text{dB}$	75
Figure 4-14: Input/Output SNR in Second-Non-CH2-O2-N2	76
Figure 4-15: SNR Improvement over Second-Non channel #2	76
Figure 4-16: Max output SNR decrease over Second-Non channel #2	78
Figure 4-17: Input/output SNR in Second-Non-CH3-O2-N2	79
Figure 4-18: SNR improvement over Second-Non channel #3	79
Figure 4-19: Max output SNR decrease over Second-Non channel #3	81
Figure 4-20: Input/output SNR in Second-Non-CH4-O2-N2-L4	82
Figure 4-21: Input/output SNR in Second-Non-CH4-O2-N2-L13	84
Figure 4-22: SNR improvement over Second-Non channel #4	86
Figure 4-23: Max output SNR decrease over Second-Non channel #4	86
Figure 4-24: Input/Output SNR in Linear-CH1-O2-N2	89
Figure 4-25: Input/Output SNR in Third-Non-CH1-O3-N2	93

Figure 4-26: SNR improvement (TM) over Third-Non channel #1	97
Figure 4-27: Max output SNR decrease (TM) over Third-Non channel #1	97,
Figure 4-28: SNR improvement (TM) over Third-Non channel #2	98
Figure 4-29: Max output SNR decrease (TM) over Third-Non channel #2	98

LIST OF SYMBOLS

Symbol	Definition
t	continuous time index
$s(t)$	continuous-time analog noise-free primary signal
$x(t)$	continuous-time analog auxiliary or reference signal
$n(t)=h[x(t)]$	continuous-time analog additive noise signal, which is the auxiliary or reference signal $x(t)$ distorted by the transmission channel $h[.]$
$d(t)=s(t)+n(t)$	continuous-time analog noisy primary signal, which is the sum of noise-free primary signal $s(t)$ and the additive noise $n(t)$
m	discrete time index
$s(m)$	discrete-time digital noise-free primary signal
$x(m)$	discrete-time digital auxiliary or reference signal
$n(m)=h[x(m)]$	discrete-time digital additive noise signal, which is the auxiliary or reference signal $x(m)$ distorted by the transmission channel $h[.]$
$d(m)=s(m)+n(m)$	discrete-time digital noisy primary signal, which is the sum of noise-free primary signal $s(m)$ and the additive noise $n(m)$
$h[.]$	transfer function of transmission channel
$\hat{h}[.]$	transfer function estimate of transmission channel
$e(m)$	discrete-time digital error signal
$\hat{s}(m)$	discrete-time digital estimate of the noise-free primary signal $s(m)$, also be the output of adaptive signal processing filter
l	stage order index in lattice structure
$L-1$	total number of stages in lattice structure
$M-1$	maximum number of delays of the input signal
$f_l(m)$	forward prediction error of the l -th lattice stage in SCLS
$b_l(m)$	backward prediction error of the l -th lattice stage in SCLS
$k_l(m)$	reflection coefficient of the l -th lattice stage in SCLS
$w_l(m)$	filter coefficient on l -th multiplicative unit in SCLS
P	the order of nonlinearity in MCLS
$\underline{X}(m)$	vector output of SIMO module in MCLS, which is the multichannel version of reference input $x(m)$ including second-order nonlinearity
$f_{[i,j]}(m)$	the single channel forward prediction error output of the lattice

	module on the i -th channel at the j -th MALP stage in MCLS
$\underline{E}_l(m)$	multichannel forward prediction error vector output at the l -th MALP stage in MCLS
$b_{[i,j]}(m)$	the single channel backward prediction error output of the lattice module on the i -th channel at the j -th MALP stage in MCLS
$\underline{B}_l(m)$	multichannel backward prediction error vector output at the l -th MALP stage in MCLS
$E_l(m)$	the error output at the l -th MMRTF stage in MCLS
$k_{[i,j]}(m)$	the reflection coefficient of the lattice module on the i -th channel at the j -th MALP stage in MCLS
$\underline{K}_l(m)$	the reflection coefficient vector of the l -th MALP stage in MCLS
$\underline{W}_l(m)$	the filter coefficient vector of the l -th MMRTF stage in MCLS
μ	step size factor
α	smoothing factor
A^T	the transpose of matrix or vector A
$O(M)$	proportional to M
\hat{P}	power estimate of the forward prediction error and the backward prediction error
\hat{Q}	power estimate of the backward predictor error
$\#\#\#$	section index $\#\#\#$, where $\#$ represents an integer
$(\#\#\#)$	formula index $(\#\#\#)$, where $\#$ represents an integer

LIST OF ACRONYMS

Acronym	Definition
ADC	Analog-to-digital converter
ANC	Adaptive noise cancellation
CGN	Coloured Gaussian noise
FIR	Finite-impulse-response
IIR	Infinite-impulse-response
LMS	Least-mean-square
M-structure	MCLS with second-order SIMO module
MALP	Multichannel adaptive lattice predictor
MCLS	Multiple channels lattice structure
MMRTF	Multichannel multiple-regression transversal filter
MSE	Mean-square-error
NLMS	Normalized least-mean-square
RE	Residue error
RLS	Recursive-least-square
SCLS	Single channel lattice structure
SIMO	Single-input-multiple-output
SNR	Signal-to-noise ratio
S-structure	same as SCLS
T-structure	MCLS with third-order SIMO module
WGN	White Gaussian noise

1. INTRODUCTION

Our major aspect of current research in speech processing, such as recognition and coding, is to develop systems which retain good performance across a wide variety of acoustic environments. The need for such systems is well appreciated. One method of making robust speech systems is to include a pre-processing stage of noise reduction. The challenge here is to reduce the noise level, preserve or improve the intelligibility and, at the same time, introduce as little distortion as possible.

1.1 Noise Cancellation

Noise Cancellation is a variation of optimal filtering that is highly advantageous in many applications. It makes use of an auxiliary or reference signal derived from one or more sensors located at points in the noise field where the noise-free primary signal is weak or undetectable. This input is filtered and subtracted from a noisy primary input containing both noise-free primary signal and additive noise signal (which is a filtered version of the auxiliary or reference signal). As a result, the additive noise is attenuated or eliminated by cancellation and the estimate of the desired noise-free primary signal becomes the output of the process.

The usual method of estimating a noise-free primary signal corrupted by additive noise signal is to pass the corrupted signal through a filter that tends to suppress the additive noise signal while leaving noise-free primary signal relatively unchanged. The filtering can be made fixed or adaptive according to the availability of *a priori* knowledge on the noise-free primary signal and the additive noise.

Adaptive noise cancellation (ANC) is performed when there is no *a priori* knowledge of the noise-free primary signal and the additive noise signal while the adaptive filters have the ability to adjust their own parameters automatically. Applications include hand-free mobile communication and teleconferencing in a noisy office or home environment.

Figure 1-1 shows a typical two-input ANC system for enhancement of noisy speech. In this system one directional microphone takes as the primary input, which is the noisy primary signal $d(t)$, and another directional microphone takes as the reference input, positioned some distance away where the noise-free primary signal $s(t)$ is weak or undetectable, measures the reference signal $x(t)$.

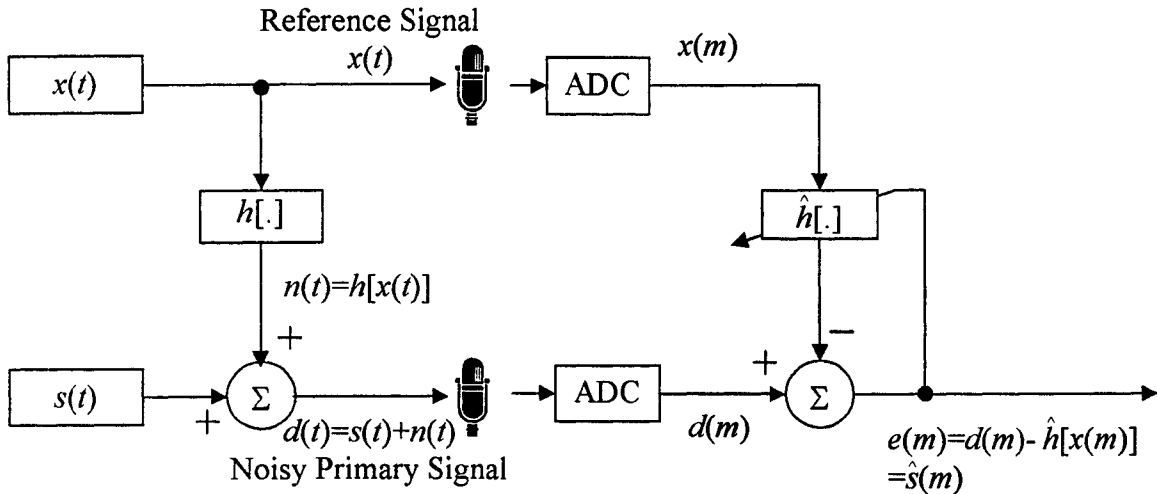


Figure 1-1: Two-microphone ANC System

The noisy observation $d(t)$, i.e. noisy primary signal, can be modelled using

$$d(t) = s(t) + h[x(t)] \quad (1.1.1)$$

where $s(t)$ and $x(t)$ are the noise-free primary signal and the reference signal, $h[\cdot]$ is a certain kind of transformation by converting the reference signal into the additive noise in the noisy primary signal $d(t)$, and t is the continuous-time index. After the directional microphones receive the reference signal and the noisy primary signal, separate ADCs (Analog-to-Digital Converters) sample and quantize the continuous-time analog signal $x(t)$ and $d(t)$ into the discrete-time digital signal $x(m)$ and $d(m)$, where m is the discrete-time index. The output signal $\hat{s}(m)$ may then be recovered by subtraction of an estimate of the additive noise $\hat{h}[x(m)]$ from the noisy primary signal $d(m)$, where $\hat{s}(m)$ is the digitized estimate corresponding to the analog noise-free primary signal $s(t)$. The $\hat{h}[\cdot]$ is an

digitized estimate corresponding to the analog transformation $h[.]$, which is optimized by the feedback from $e(m)$, i.e. $\hat{s}(m)$, adaptively.

1.2 Adaptive Filter

An adaptive filter is a computational device that attempts to model the relationship between two signals in real time in an iterative manner.

An adaptive filter is defined by the following four aspects:

- the *signals* being processed by the filter
- the *structure* that defines how the output signal of the filter is computed from its input signal
- the *parameters* within this structure that can be iteratively changed to alter the filter's input-output relationship
- the *adaptive algorithm* that describes how the parameters are adjusted from one time instant to the next

By choosing a particular adaptive filter structure, one specifies the number and type of parameters that can be adjusted. The adaptive algorithm used to update the parameter values of the system can take on a myriad of forms and is often derived as a form of *optimization procedure* that minimizes an *error criterion* that is useful for the task at hand.

Many computationally efficient algorithms for adaptive filtering have been developed. There are two algorithms most widely used. One is called LMS (least-mean-square) algorithm and its variation NLMS (normalized least-mean-square) algorithms, both are based on a gradient optimization for determining the coefficients. Another is called RLS (recursive-least-square) algorithm [33].

In general, any system with a *finite number* of parameters that affect how the output $y(m)$ is computed from the input $x(m)$ can be used for the adaptive filter in Figure 1-2. Define the parameter or coefficient vector $W(m)$ as

$$W(m) = [w_0(m) \quad w_1(m) \quad \cdots \quad w_{L-1}(m)]^T \quad (1.2.1)$$

where $\{w_i(m)\}$, $0 \leq i \leq L-1$ are the L parameters of the system at time index m . With this definition, a general input-output relationship for the adaptive filter is defined by

$$y(m) = f(W(m), y(m-1), \dots, y(m-N), x(m), \dots, x(m-M+1)) \quad (1.2.2)$$

where $f(\cdot)$ represents any well-defined linear or nonlinear function. M and N are positive integers. M is the maximum number of previous values on $x(m)$ used to calculate the output of $y(m)$. N is the maximum number of previous values on $y(m)$ used to calculate the output of $y(m)$. Implicit in this definition is the fact that the filter is *causal*, such that future values of the input $x(m)$ are not needed to compute the output $y(m)$. Noncausal filters can be handled in practice by suitably buffering or storing the input signal samples. In this thesis, this possibility is not considered.

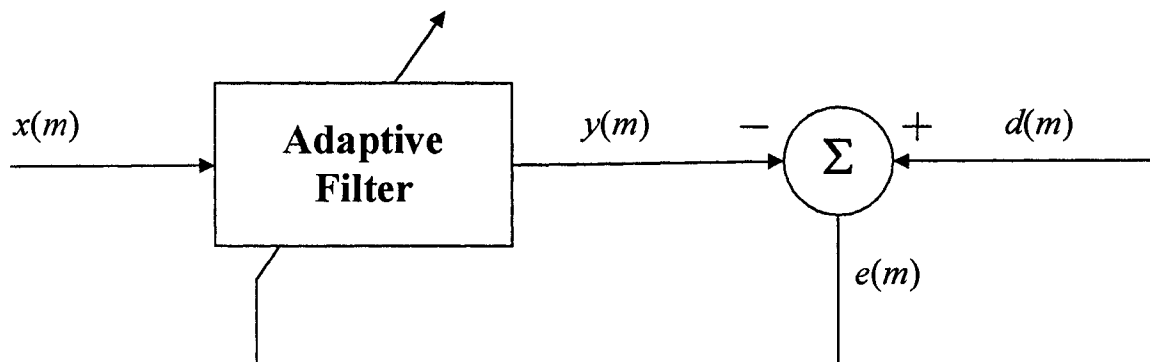


Figure 1-2: The general adaptive filtering problem

Adaptive filters have been widely used in communication systems, control systems and various other systems in which the statistical characteristics of the signals to be filtered are either no *a priori* knowledge or, in some cases, slowly time-variant (non-

stationary signals) [24]. Adaptive noise cancellation is one of the most noteworthy applications in which an adaptive filter is used to estimate and eliminate the noise component in some desired signals [1].

1.2.1 Adaptive Filter Structure

The structure of the adaptive filter defines how the output signal of the filter is computed from its input signals. The structure specifies the number and the type of parameters that can be adjusted. There are two types of structure discussed in this section. One is called linear structure. The other is called nonlinear structure.

1.2.1.1 Linear Structure

Linear structure has played a very crucial role in the development of various signal processing techniques. The obvious advantage of linear structure is its inherent simplicity. Design, analysis, and implementation of such structure are relatively straightforward tasks in many applications.

Although (1.2.2) is the most general description of an adaptive filter structure, in most time, we are interested in determining the best *linear relationship* between the input and the desired response for many problems. This relationship typically takes the form of a FIR or IIR filter structure.

Figure 1-3 shows the structure of a direct-form FIR filter, also known as a *tapped-delay-line* or *transversal filter*, where z^{-1} denotes the unit delay element and each $w_i(n)$ is a multiplicative gain within the system.

The input-output relationship is shown as

$$y(m) = \sum_{i=0}^{L-1} w_i(m) x(m-i) \quad (1.2.3)$$

where $L-1$ is the maximum number of the delays within the system.

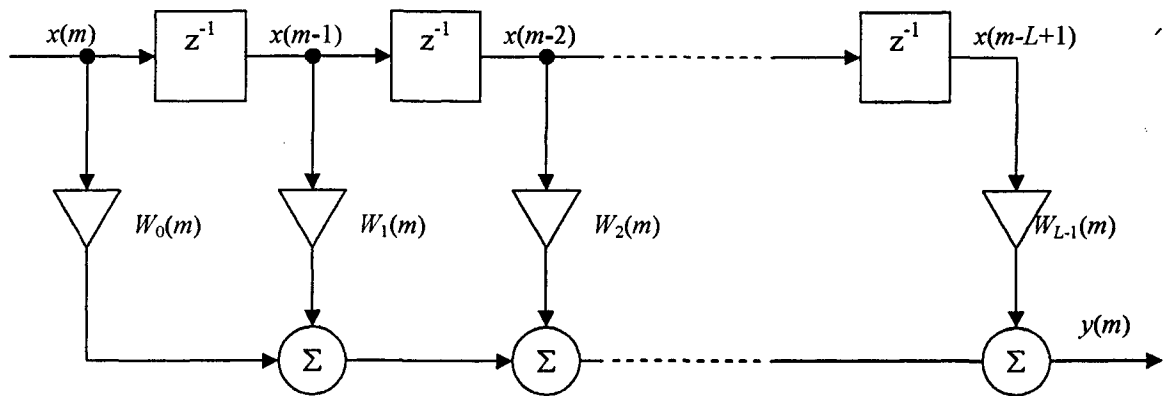


Figure 1-3: Structure of an FIR filter

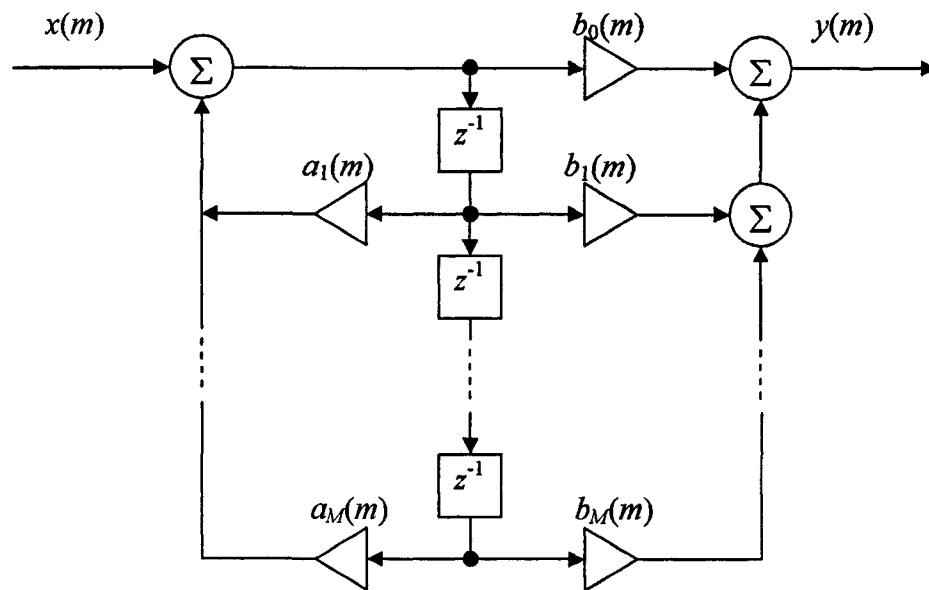


Figure 1-4: Structure of a canonical-form IIR filter

The structure of a canonical-form IIR filter is shown in Figure 1-4 [33]. In this case, the output of the system can be represented mathematically by

$$y(m) = \sum_{i=1}^M a_i(m) y(m-i) + \sum_{j=0}^M b_j(m) x(m-i) \quad (1.2.4)$$

where M is the maximum number of the previous values of $x(m)$ and $y(m)$ used to calculate the output $y(m)$.

A third structure that has proven useful for adaptive filtering tasks is the *lattice filter structure* [33]. A lattice filter is an FIR structure that employs $L-1$ stages of pre-processing to compute a set of auxiliary signals $\{b_i(m)\}$, $0 \leq i \leq L-1$, known as *backward prediction errors*. These signals have the special property that they are *uncorrelated*, and they represent the elements of $x(m)$ through a *linear transformation*. Thus, the backward prediction errors can be used in place of the delayed input signals in a structure similar to that in Figure 1-3, and the uncorrelated nature of the prediction errors can provide improved convergence performance of the adaptive filter coefficients with the proper choice of algorithm. The overall structure of adaptive lattice filter is shown in Figure 1-5.

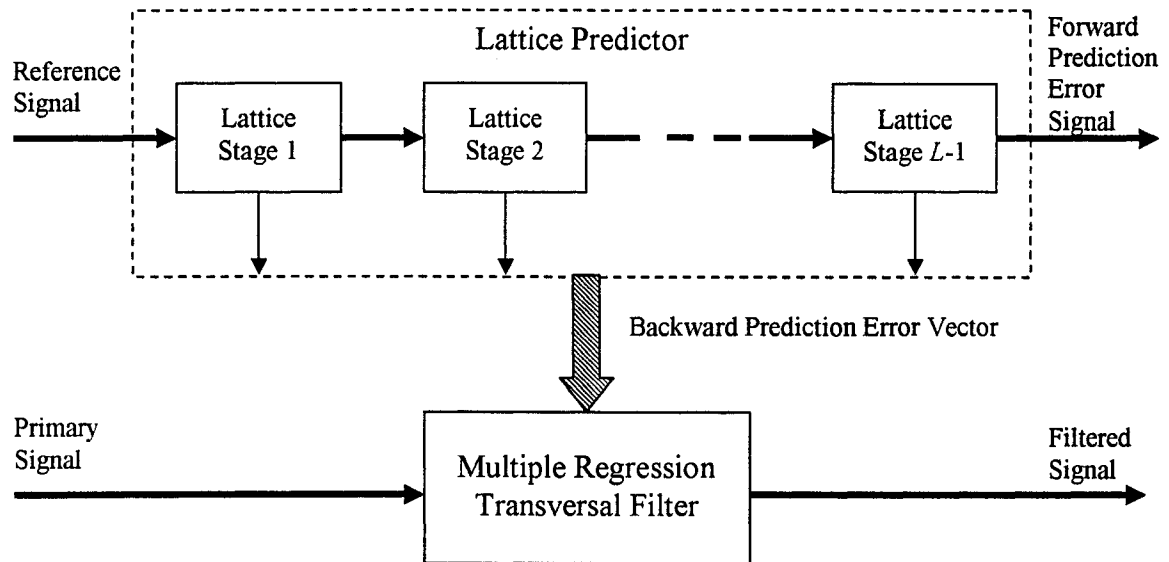


Figure 1-5: Overall structure of lattice predictor and transversal filter

The adaptive lattice predictor, shown in Figure 1-5, is a modular structure that consists of a number of cascaded lattice stages. A steepest descent or an exact least-squares algorithm is used to adjust the reflection coefficients independently at each lattice stage. The lattice structure enjoys the advantages of a simple test for filter stability, good performance in finite-wordlength hardware implementations, and greatly reduced

sensitivity to the eigenvalue spread¹ of the reference signal. The decorrelation of the signal at each stage allows the LMS algorithms to converge much faster than the conventional transversal filter, especially for a reference signal that has a large eigenvalue spread. The downside of the lattice filter is that improved convergence comes at the expense of increased computational complexity [33].

The structure of l -th lattice stage is illustrated in Figure 1-6.

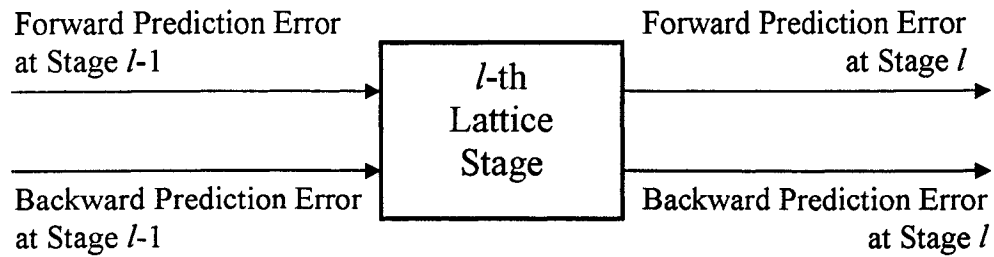


Figure 1-6: Structure of l -th lattice stage

A critical issue in choosing the structure of an adaptive filter is its computational complexity. Since the operation of the adaptive filter typically occurs in real time, all of the calculations for the system must occur during one sample time. The structures described above are all useful because the output $y(m)$ can be computed in a finite amount of time using simple arithmetical operations and finite amount of memory.

1.2.1.2 Nonlinear Structure

Although the linear structure benefits from its simplicity, however, there are several situations in which the performance of linear filters is unacceptable. Trying to identify these types of systems using linear models can often give misleading results.

¹ The ratio of the maximum to the minimum eigenvalue of a correlation matrix is called the eigenvalue spread of the correlation matrix. The spread in the speed of convergence of filter coefficients is proportional to the spread in eigenvalue of the autocorrelation matrix of the input signal [34].

When confronted with a nonlinear systems problem, the solutions are often difficult from an analytical and/or computational point of view. Those difficulties are much more magnified in the case of adaptive nonlinear systems [27].

Unlike the case of linear systems which are completely characterized by the system's unit impulse response function. It is impossible to find a unified framework for describing arbitrary nonlinear systems. Consequently, the researchers working on nonlinear filters are forced to restrict themselves to certain nonlinear system models that are less general. Nonlinear filters developed using such models include order statistics filters [7][12][14], homomorphic filters [1], morphological filters [21][22], and filters based on Volterra and other polynomial descriptions of the nonlinearity involved [27].

The polynomial models of nonlinearity description are more general than most of the other models that were mentioned above. The Volterra system model is extremely popular in adaptive nonlinear filtering [27].

1.2.1.2.1 Volterra Series Expansion

Let $x(m)$ and $y(m)$ represent the input and output signals, respectively, of a discrete-time and causal nonlinear system. The Volterra series expansion [7][9][10][12][13] for $y(m)$ using $x(m)$ is given by

$$\begin{aligned}
 y(m) = & h_0 + \sum_{k_1=0}^{\infty} h_1(k_1)x(m-k_1) + \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} h_2(k_1, k_2)x(m-k_1)x(m-k_2) + \\
 & \cdots + \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} h_p(k_1, k_2, \dots, k_p)x(m-k_1)x(m-k_2) \cdots x(m-k_p) + \cdots
 \end{aligned} \tag{1.2.5}$$

In (1.2.5), $h_p(k_1, k_2, \dots, k_p)$ is known as the p -th order Volterra kernel of the system. The limitation of Volterra series is that the expansion does not do well when there are discontinuities in the system description [27]. Even though clearly not applicable in all situations, Volterra system models have been successfully employed in a wide variety of applications, and such models continue to be popular with researchers in this area.

Since an infinite series expansion like (1.2.5) is not useful in filter applications, one must work with the truncated Volterra series expansion of the form

$$y(m) = h_0 + \sum_{k_1=0}^{M-1} h_1(k_1) x(m-k_1) + \sum_{k_1=0}^{M-1} \sum_{k_2=0}^{M-1} h_2(k_1, k_2) x(m-k_1) x(m-k_2) + \dots + \sum_{k_1=0}^{M-1} \sum_{k_2=0}^{M-1} \dots \sum_{k_p=0}^{M-1} h_p(k_1, k_2, \dots, k_p) x(m-k_1) x(m-k_2) \dots x(m-k_p) \quad (1.2.6)$$

where P is the order of the truncated Volterra series expansion, and $M-1$ is the maximum delay of the input $x(m)$ in the calculation. Both P and M are integers.

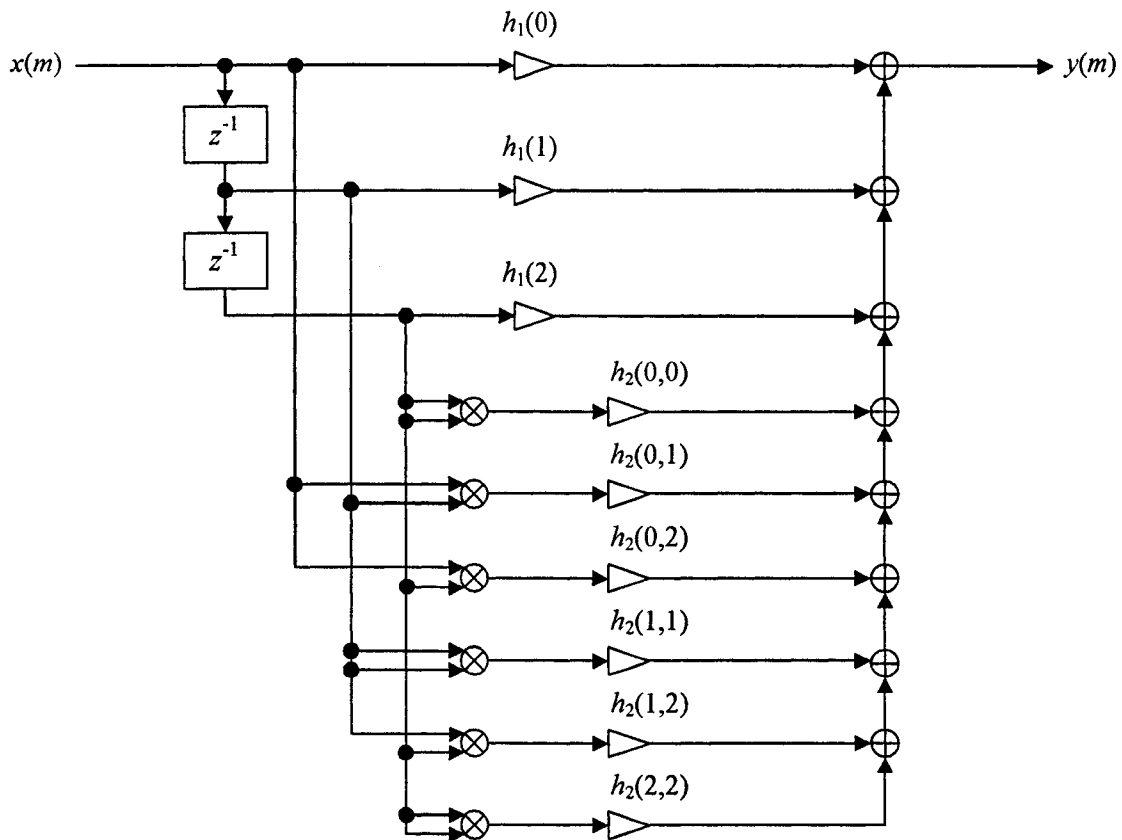


Figure 1-7: Truncated Volterra system of order $P=2$ and maximum delay $M-1=2$.

An example of nonlinear system characterized using truncated Volterra series expansion with order $P=2$ and maximum delay $M-1=2$ is illustrated in Figure 1-7. Note that this system is linear in the input signal to each coefficient. This fact highly simplifies

the design problems involving Volterra series representations. On the other hand, even for moderately large values of M and P , the number of coefficients becomes very large. Consequently, the truncated Volterra series representation is most useful in applications where the values of M and P are relatively small.

1.2.1.2.2 Adaptive Volterra Filter

An adaptive filter structure using truncated Volterra series expansions is called an adaptive Volterra filter. The block diagram of an adaptive Volterra filter is illustrated in Figure 1-8, where $d(m)$ is the noisy primary signal shown in (1.2.7) and $x(m)$ is the reference signal. $s(m)$ is the noise-free primary signal, and $n(m)$ is the additive noise signal. $\hat{s}(m)$, the estimate of the noise-free primary signal $s(m)$, is computed using the truncated Volterra series expansion on $x(m)$.

$$d(m) = s(m) + n(m) \quad (1.2.7)$$

The adaptive Volterra filter structure tries to estimate the noise-free primary signal $s(m)$ using a truncated Volterra series expansion on the input signal $x(m)$ as (1.2.6).

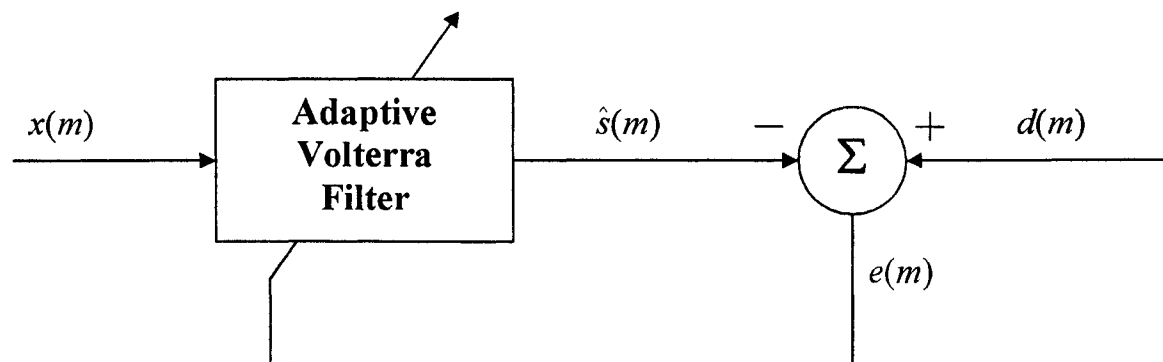


Figure 1-8: A block diagram of the Adaptive Volterra Filter

The objective, as in most adaptive filtering problems, is to choose the coefficients of the adaptive filter, $h_p(k_1, k_2, \dots, k_p)$, so that an appropriate cost function of the error signal $e(m)$ is minimized. The adaptive algorithm depends on the choice of the above cost function. Among the most commonly used algorithms are LMS-type algorithm, including

LMS and NLMS, and RLS algorithms. RLS algorithm for adaptive Volterra filtering is at least an order of magnitude more complex than LMS-type algorithms [27].

There are $O(M^P)$ coefficients in the polynomial expansion (1.2.6). One big disadvantage for the Volterra system model is that the complexity of implementing filters using this model can be very high even for moderately values of M and P . Consequently, most of the practical applications of systems employing Volterra series expansions only involve low-order models.

V.J. Mathews proposed an adaptive lattice structure for Volterra systems, i.e. Second-order Volterra lattice filter. The block diagram with $M=3$ and $P=2$ is shown in Figure 1-9.

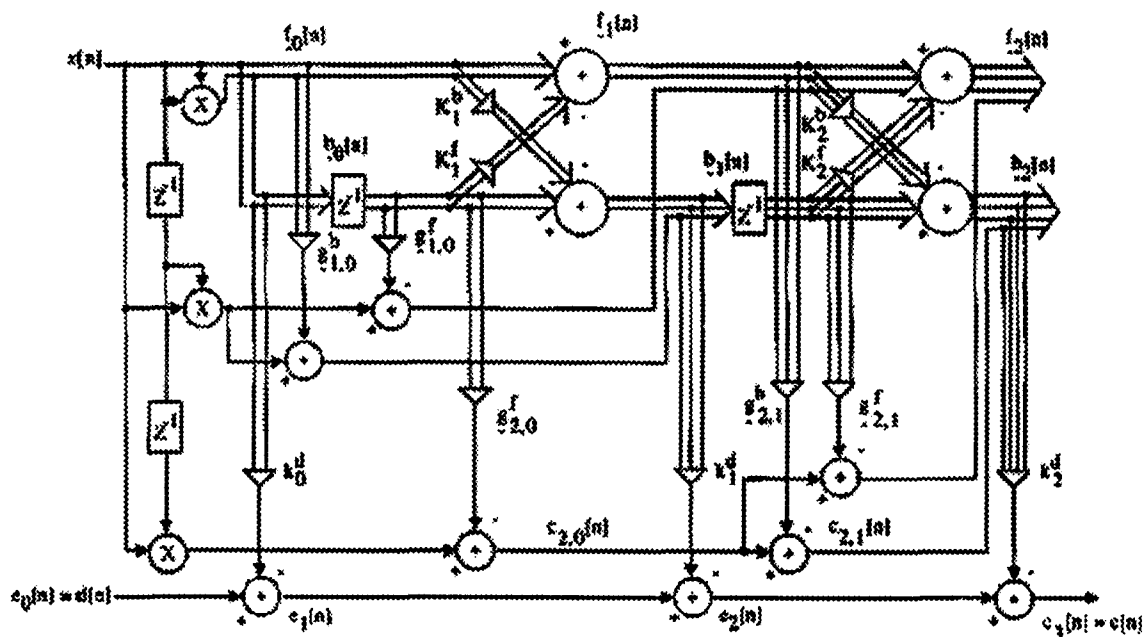


Figure 1-9: Block diagram of the Second-order Volterra Adaptive Lattice Filter

The number of lines going into and out of a system component indicates the number of input and output signals, respectively, of that component. The backward prediction error vector \underline{b}_0 , \underline{b}_1 and \underline{b}_2 are orthogonal to each other, and the components of these vectors span the whole space by the elements of

$$\underline{X}(m) = \begin{bmatrix} x(m) & x(m-1) & x(m-2) \\ x^2(m) & x^2(m-1) & x^2(m-2) \\ & x(m)x(m-1) & x(m)x(m-2) \\ & & x(m-1)x(m-2) \end{bmatrix} \quad (1.2.8)$$

At each stage of the lattice, the prediction error vector has one more element than previous stage. This prediction error signal, that corresponds to estimating $x(m)x(m-i)$ for the i -th stage, must be computed outside the basic lattice structure. The coefficients denoted using the letter g , are used to compute these additional prediction-error signal. Efficient computation of the backward prediction-error vectors requires computation of the forward prediction-error vectors f_0 , f_1 , and f_2 also. A gradient and least-squares adaptive algorithm based on this lattice structure is straightforward [27].

1.2.1.2.3 Adaptive Multichannel Structure

Another approach in adaptive nonlinear filter is proposed by S. Ozgunel, A. H. Kayran and E. Panayirci [26] in the application of nonlinear channel equalization. Based on the truncated second-order Volterra series expansion, a single-input-multiple-output system (shown in Figure 1-10) is introduced. This system has the second-order nonlinearity. An algorithm, either LMS or RLS, is used for updating the linear and quadratic weights of the second-order nonlinear filter.

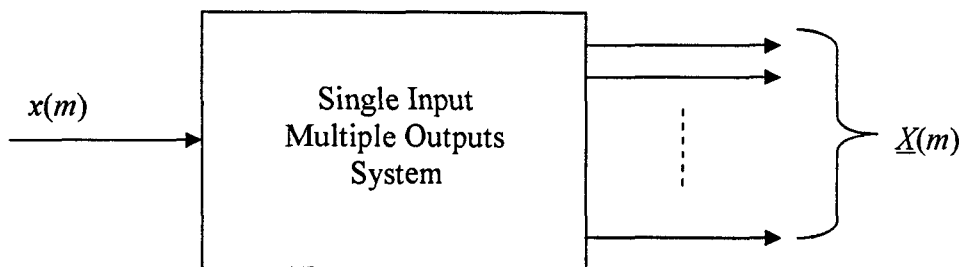


Figure 1-10: Single-input-multiple-output system

The output $\underline{X}(m)$ of this single-input-multiple-output system is defined by

$$\underline{X}(m) = \begin{bmatrix} x_1(m) \\ x_2(m) \\ x_3(m) \\ \vdots \\ x_{M-1}(m) \\ x_M(m) \\ x_{M+1}(m) \end{bmatrix} \quad (1.2.9)$$

where

$$\begin{cases} x_1(m) = F_1[x(m)] \\ x_2(m) = F_2[x^2(m)] \\ x_3(m) = F_3[x(m)x(m-1)] \\ \vdots \\ x_{M-1}(m) = F_{M-1}[x(m)x(m-M+3)] \\ x_M(m) = F_M[x(m)x(m-M+2)] \\ x_{M+1}(m) = F_{M+1}[x(m)x(m-M+1)] \end{cases} \quad (1.2.10)$$

$\underline{X}(m)$ is an $(M+1) \times 1$ vector and its dimension is determined by the channel length, or the maximum delay of input, $M-1$. This vector is used as the input to the multichannel adaptive lattice predictor. $x_1(m)$ is the input for the linear part of the lattice and $x_2(m)$, $x_3(m)$, ..., $x_{M+1}(m)$ are the inputs for the nonlinear parts. Figure 1-11 shows the structure of the multichannel adaptive lattice equalizer.

This multichannel adaptive lattice equalizer involves the concept of the second-order truncated Volterra series expansion. But not all the elements in the second-order truncated Volterra series expansion are included. Only elements relating to the reference input $x(m)$ are included.

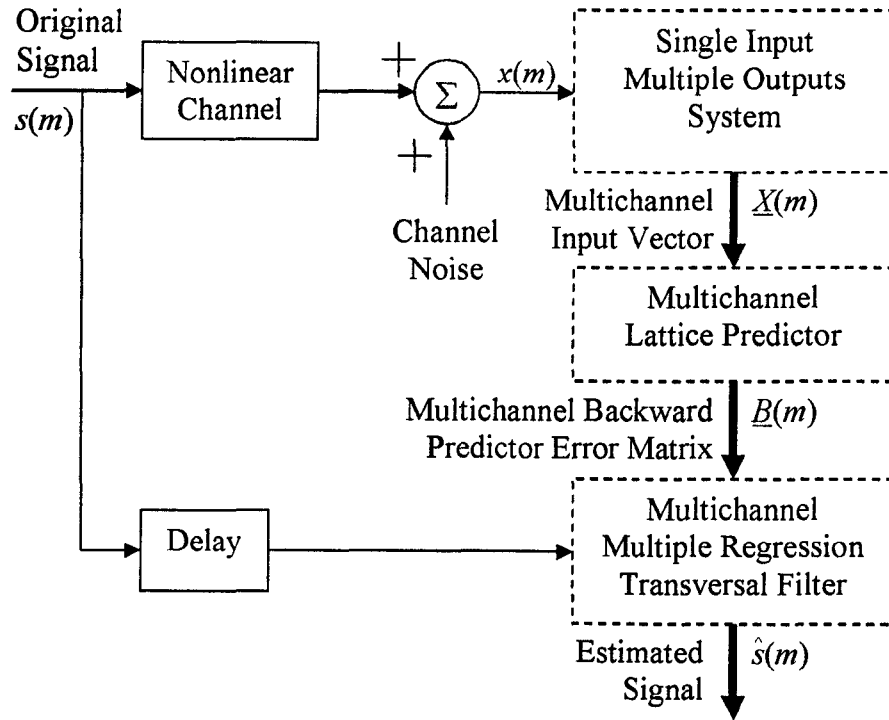


Figure 1-11: Multichannel adaptive lattice equalizer

1.2.2 Adaptive Algorithm

The configuration of an adaptive filter is illustrated in Figure 1-12. The output of the filter is defined by

$$\hat{x}(m) = \underline{W}^T(m) \underline{Y}(m) \quad (1.2.11)$$

where $\underline{Y}(m) = [y(m), y(m-1), \dots, y(m-P+1)]^T$, $\underline{W}(m) = [w_0(m), w_1(m), \dots, w_{P-1}(m)]^T$ and $x(m)$ denote the filter input, the filter coefficient vector and the desired signal respectively [34]. $\hat{x}(m)$ is an estimate of the desired signal $x(m)$.

The error signal is defined by

$$e(m) = x(m) - \hat{x}(m) = x(m) - \underline{W}^T(m) \underline{Y}(m) \quad (1.2.12)$$

The adaptive process is based on the minimization of the mean square error criterion defined by

$$\begin{aligned}
 E[e^2(m)] &= E\left\{\left[x(m) - \underline{W}^T(m)\underline{Y}(m)\right]^2\right\} \\
 &= E\left[x^2(m)\right] - 2\underline{W}^T(m)E\left[\underline{Y}(m)x(m)\right] + \underline{W}^T(m)\left[\underline{Y}(m)\underline{Y}^T(m)\right]\underline{W}(m)
 \end{aligned}
 \tag{1.2.13}$$

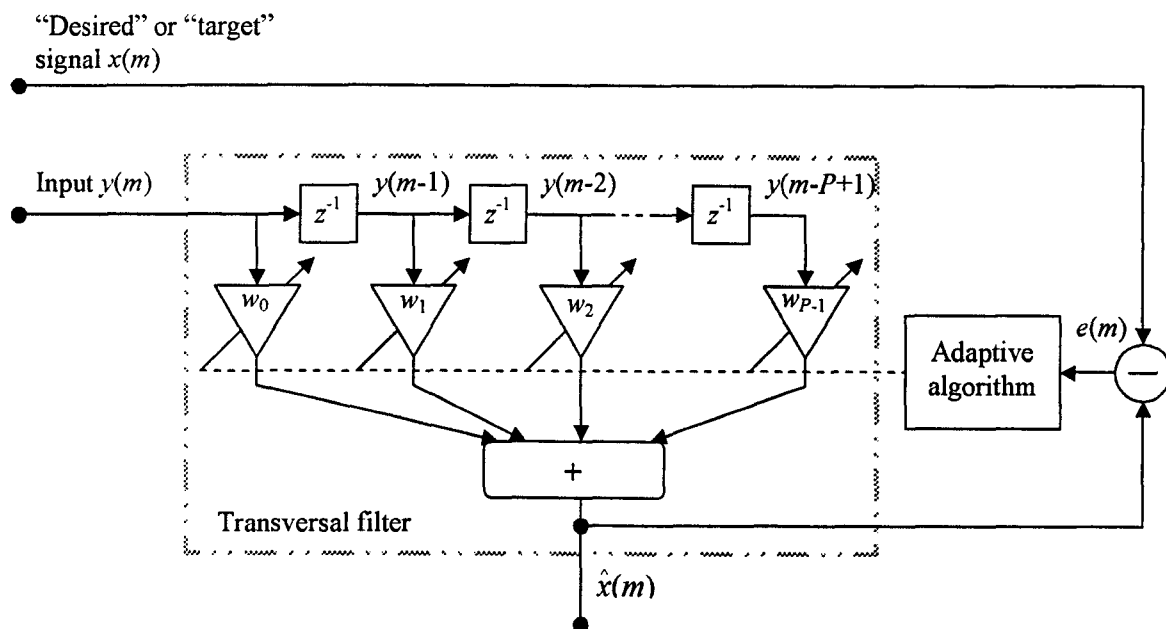


Figure 1-12: Configuration of an adaptive filter

1.2.2.1 LMS Algorithm

The steepest-descent method employs the gradient of the averaged square-error to search for the filter coefficients over least square-error. A computationally simple version of the gradient search method is the LMS (least-mean-square) algorithm, in which the gradient of the MSE (mean-square-error) is substituted with the gradient of the instantaneous square-error function [34].

The LMS adaptation method is defined by

$$\underline{W}(m+1) = \underline{W}(m) + \frac{\mu}{2} \left(-\frac{\partial e^2(m)}{\partial \underline{W}(m)} \right) \quad (1.2.14)$$

where the error signal $e(m)$ is given by (1.2.12), and μ is the update step size. The instantaneous gradient of the square-error can be re-expressed by

$$\begin{aligned} \frac{\partial e^2(m)}{\partial \underline{W}(m)} &= \frac{\partial}{\partial \underline{W}(m)} \left[x(m) - \underline{W}^T(m) \underline{Y}(m) \right]^2 \\ &= -2\underline{Y}(m) \left[x(m) - \underline{W}^T(m) \underline{Y}(m) \right] = -2\underline{Y}(m) e(m) \end{aligned} \quad (1.2.15)$$

Substituting (1.2.15) into the recursive coefficient update equation (1.2.14) yields the LMS adaptation equation:

$$\underline{W}(m+1) = \underline{W}(m) + \mu \left[\underline{Y}(m) e(m) \right] \quad (1.2.16)$$

It can be seen that the filter update equation is very simple. The LMS filter is widely used in adaptive filter applications. The main advantage of the LMS algorithm is its simplicity in terms of both the memory requirement and the computational complexity, which is $O(P)$, where P is the filter length [34].

1.2.2.2 NLMS Algorithm

In practice, the exact statistics of $x(m)$ and $y(m)$ are unknown and varied over time. A time-varying step size $\mu(m)$, if properly computed, can provide stable, robust and accurate convergence behaviour for the LMS adaptive algorithm in these situations. This introduces the NLMS algorithm.

The normalized step size $\mu(m)$ can be calculated by

$$\mu(m) = \frac{\bar{\mu}}{\delta + L \hat{\sigma}_x^2(m)} \quad (1.2.17)$$

where $\hat{\sigma}_x^2(m)$ is an estimate of the input signal power, $\bar{\mu}$ is a constant to keep the system stable, and δ is a small non-zero constant to avoid the divide-by-zero problem, when $\hat{\sigma}_x^2(m)$ approaches zero.

Typical estimators include:

- Exponentially weighted estimate:

$$\hat{\sigma}_x^2(m) = (1-c)\hat{\sigma}_x^2(m-1) + cx^2(m) \quad (1.2.18)$$

- Sliding-window estimate:

$$\hat{\sigma}_x^2(m) = \frac{1}{M} \sum_{i=0}^{M-1} x^2(m-i) \quad (1.2.19)$$

where the parameters c ($0 < c \ll 1$) and M ($M \geq L$) control the effective memories of the two estimators respectively [33].

1.2.2.3 RLS Algorithm

In the RLS (recursive-least-square) algorithm, the adaptation starts with some initial filter states. And successive samples of the input signals are used to adapt the filter coefficients. The following steps show the details of the RLS algorithm [34]:

- Input Signals: $y(m)$ and $x(m)$
- Initial values: $\Phi_{yy}(m) = \delta I$, $\underline{w}(0) = \underline{w}_I$
- For $m = 1, 2, \dots$
 - a. Filter gain vector:

$$\underline{K}(m) = \frac{\lambda^{-1} \Phi_{yy}(m-1) \underline{Y}(m)}{1 + \lambda^{-1} \underline{Y}^T(m) \Phi_{yy}(m-1) \underline{Y}(m)} \quad (1.2.20)$$

b. Error signal equation:

$$e(m) = x(m) - \underline{W}^T(m-1)\underline{Y}(m) \quad (1.2.21)$$

c. Filter coefficients:

$$\underline{W}(m) = \underline{W}(m-1) - \underline{K}(m)e(m) \quad (1.2.22)$$

d. Inverse correlation matrix update:

$$\Phi_{yy}(m) = \lambda^{-1}\Phi_{yy}(m-1) - \lambda^{-1}\underline{K}(m)\underline{Y}^T(m)\Phi_{yy}(m-1) \quad (1.2.23)$$

1.3 Comparison

Various aspects in the area of noise cancellation are discussed in the previous sections. From those descriptions, the advantages and disadvantages of different methods, different structures and different algorithms can be found. After comparing the differences, the appropriate method, structure and algorithm under certain scenario can be selected.

1.3.1 Adaptive Filtering vs. Fixed Filtering

According to the availability of *a priori* information on the reference signal, the noise-free primary signal and the transmission channel, fixed or adaptive filtering method can be used. Fixed filtering method on noise cancellation is performed when *a priori* information is available. The filter's coefficients are fixed. On the contrary, adaptive filtering method on noise cancellation is performed when there is no *a priori* information available. The adaptive filters have the ability to adjust their own parameters, i.e. coefficients, automatically.

Adaptive filtering method has to update the coefficients from time to time. Apparently it is more complex and less computational efficient than the fixed filtering method. As in the most circumstances, *a priori* information is unavailable. Adaptive

filtering method is more popular than fixed filtering method in the area of noise cancellation.

Sometimes, for a relatively stable system, the coefficients can be trained using adaptive filter. After the coefficients are stable at a certain point of time, a fixed filtering can be used to improve the system performance.

1.3.2 Lattice Structure vs. FIR/IIR Structures

Adaptive lattice filters, adaptive FIR filters and adaptive IIR filters are all linear structures of the adaptive filter.

Adaptive lattice filters try to orthogonalize the input signals to the filter and then estimate the desired response signal as a linear combination of the transformed signals that are hopefully orthogonal to each other. The advantages of lattice structure in adaptive filtering applications comparing with FIR and IIR structure are shown as below:

- Adaptive lattice structure equipped with LMS-type adaptation algorithm tends to show faster and less input signal-dependent convergence behaviour than their direct form counterparts, such as FIR and IIR structures.
- Adaptive lattice structure tends to have better numerical properties than their direct form counterparts.
- Adaptation of the parameters using the adaptive lattice structure can be done independently in each stage without the involvement of other stages.
- The adaptive lattice structure is highly modular; therefore they are very suitable for VLSI hardware implementation.

From the above description, the lattice structure among the linear adaptive filter structures enjoys the advantages of a simple test for filter stability, good performance in finite-wordlength hardware implementations, and greatly reduced sensitivity to the eigenvalue spread of the reference signal. The decorrelation of the signal at each stage

allows the LMS algorithms to converge much faster than the conventional transversal filter, especially for a reference signal that has a large eigenvalue spread [33].

The downside of the lattice filter is that the computational complexity is higher than that of FIR and IIR [33].

1.3.3 LMS/NLMS Algorithms vs. RLS Algorithm

From the previous description of the adaptive algorithms, the major advantage of the LMS algorithm is the relative simplicity of the algorithm. NLMS is slightly more complex than LMS, because of the time-varying step size. But NLMS has better performance in the situation of the unknown statistics and time-varying attributes of the input.

However, for signals with a large spectral dynamic range, or equivalently a large eigenvalue spread, the LMS or NLMS has an uneven and slow rate of convergence. If in addition to having a large eigenvalue spread a signal is also non-stationary, then the LMS/NLMS can be unsuitable adaptation method. And the RLS method, with its better convergence rate and less sensitivity to the eigenvalue spread, becomes a more attractive alternative in such situations [34].

RLS algorithm is much more complex than LMS/NLMS algorithms [27].

1.3.4 Linear Structure vs. Nonlinear Structure

The obvious advantage of linear structure is their inherent simplicity. Nonlinear structure is much more complicated and higher computational requirement than the linear structure. However, if the process is nonlinear, the nonlinear structure has dominant performance advantage over the linear structure, including higher noise reduction rate and better convergence properties [27].

1.3.5 Volterra Nonlinear Structure vs. Multichannel Nonlinear Structure

By comparing the two different nonlinear adaptive filter structures, the structure of adaptive Volterra filter is more complicated than that of multichannel adaptive lattice filter [26][27]. In adaptive Volterra filter, the nonlinearity of the system is represented by truncated Volterra series expansion. However, the multichannel adaptive lattice filter, which includes part of the truncated second-order Volterra series expansion, introduces a single-input-multiple-outputs system between the input signal and the lattice predictor, to cover the second-order nonlinearity [26].

The total number of elements for the second-order Volterra adaptive filter is $O(M^2)$ [27], where $M-1$ is the maximum delay of the input signal. The total number of elements for the multichannel adaptive lattice filter is $O(M)$ [26]. Apparently, the multichannel structure is less in computational complexity than the Volterra structure.

Both the nonlinear adaptive filters are implemented by using the multichannel adaptive lattice predictor. From Figure 1-9, the adaptive Volterra structure has different elements on each channel. However, multichannel adaptive lattice structure has the same number of elements on each channel. The multichannel nonlinear structure is less complex for the hardware implementation than the Volterra nonlinear structure.

1.4 Motivation

In recent years, many applications are developed and implemented in the area of speech processing, such as speech recognition, coding, etc. The noise-cancellation pre-processing stage shows high demand in the real-world environment. Meanwhile, more and more techniques are developed to provide the system with higher computational capability and smaller size.

Several filtering techniques have been proposed over the years. Among them are linear processing techniques, whose mathematical simplicity and existence of a unifying theory make their design and implementation easy. Their simplicity, in addition to their

satisfactory performance in a variety of practical applications, has made them methods of choice for many years. However, most of these techniques operate assuming a Gaussian model for the statistical characteristics of the underlying process, and thus they try to optimize the parameters of a system suitable for such a model. Many signal processing problems can not be efficiently solved by using linear techniques. To fit into the real-world environment using the advanced technologies, some more realistic cases (nonlinear system models) of the noise environment need to be investigated in the area of adaptive noise cancellation.

An approach of adaptive noise cancellation using the method of independent component analysis (ICA) is presented in [42], where the signals are contaminated with high-level additive noise and/or outliers. A prewhitening technique is used to reduce the power of additive noise, the dimensionality and the correlation among sources. A cross-validation technique is introduced to estimate the number of sources. After that, the nonlinear function is derived using the parameterized t-distribution density model [42]. This approach shows good performance in blind separation of independent sources in the fields of neural networks and statistical signal processing. However, as the approach makes a lot of assumptions on the properties and distributions of signals and system models, it can only apply to some specific scenarios. The computational cost of this approach is high.

Another approach on adaptive noise cancellation using radial-basis-function-network (RBFN) is presented in [44]. First, identification (modeling) of the additive noise process using RBF networks and its learning algorithm. Secondly, neural control is put on neural model obtained in the first stage. This approach uses the online structure learning and parameters determination algorithm. The minimum firing strength (MFS) criterion ensures that the RBF neurons can represent the input space in sufficient degree. The self-organizing algorithm makes the RBFN covers the input space well [44]. As the approach uses the neural networks technology, the performance will be limited upon successful training of the parameters. The computational cost of this approach is high. Similar

approach in the area of active noise control is presented in [39], which has the same limitation.

Under some general continuity requirements, i.e. the function and its n first derivatives are continuous and differentiable, the output of a nonlinear system can be expanded into a Volterra series [40]. The Volterra series can be seen as being the sum of the responses of a first-order operator, a second-order operator, etc. The usefulness of the Volterra series hinges on the ability to model a very wide class of nonlinear systems and the capability of presenting the solution in terms of generalized frequency response functions [1][7]. However, there are some limitations associated with the applications of Volterra series to nonlinear problems. The major drawback of this approach is the convergence of the series especially when strong nonlinearities such as saturating elements are to be modeled or when the inputs are large [14][15][17]. Consequently, the present work only relates to the study of weakly nonlinear systems which can properly and adequately be characterized by means of a Volterra series. However in these cases, the need to determine a large number of Volterra kernels increases the complexity of identifying Volterra series models, it is often time consuming and computational cost may not be realistic. Thus building a parsimonious nonlinear model with a minimum number of Volterra kernels at a reasonable computational cost, while retaining the Volterra structure (in terms of the generalized impulse/frequency response function), is therefore an important practical problem [41].

The nonlinear models based on Volterra series expansion are more general than most of the other models that were discussed in 1.2.1.2. Two specific cases were presented in [27]. One is adaptive filters employing truncated Volterra series representation of nonlinear systems. The other is to use recursive nonlinear difference equations to relate the input and output signals of the system. It is possible to treat the truncated Volterra series representation as a special case of the recursive nonlinear system representation and consider a unified framework for polynomial system representation [27]. The Volterra system model is extremely popular in adaptive nonlinear filtering, which uses the truncated Volterra series expansion based adaptive Volterra filtering technique to

process the signal under nonlinear environment [27]. A second-order Volterra filter with rapid convergence is presented in [23]. The RLS adaptive second-order Volterra filter is presented in [27] to improve the performance of [23] using RLS adaptive algorithm. The adaptive Volterra filter shows better performance in modeling the nonlinear system than the conventional linear adaptive filter, with higher computational cost and more difficult to converge.

An approach on acoustic echo cancellation using Volterra technique is presented in [43]. This approach includes a nonlinear upstream module based on polynomial Volterra filters identifies the loudspeaker impulse response, and a linear downstream module identifies the global linear response. The tracking of the overall system model is achieved by a modified NLMS algorithm [43]. As this structure includes the second-order and third-order nonlinearity, the computational complexity is high.

Another approach to control the nonlinear noise processes using adaptive Volterra filters is presented in [38]. This approach uses a Volterra filtered-X LMS algorithm based on a multichannel structure for feedforward active noise control. RLS algorithm is used to update the coefficients of the filter. The adaptive Volterra filter uses the truncated Volterra series expansion to include the nonlinearity [38]. Although this approach shows good performance over nonlinear noise processes, the structure is very complicated and it is difficult to converge.

The availability of a wide set of multichannel information sources in application areas have stimulated a renewed interest in developing efficient and cost effective processing techniques for multichannel signals. For the past two decades, multichannel adaptive signal processing has been an extensive research area. Many different methodologies and techniques like block-type multichannel schemes or scalar-only operations schemes have been used, and many different concepts like multidimensional signal processing, least-squares type algorithms have been combined with multichannel applications.

An approach of building a multichannel structure using the nonlinear model with a minimum number of Volterra kernels at a reasonable computational cost, while retaining

the Volterra structure (in terms of the generalized impulse/frequency response function), is presented in [26] regarding nonlinear channel equalization. Only some basic elements within the Volterra series expansion are used, which shows good performance in nonlinear channel equalization and less computational complexity than the adaptive Volterra filter. The method on transferring the nonlinear problem into several linear problems can be used onto the nonlinear adaptive noise cancellation applications [26].

Another approach using multichannel feedforward adaptive systems for noise cancellation is presented in [34]. It alters the reference-signal compensation filtering from the conventional choice, ways to force systems to converge to arbitrary solutions of possible interest other than the standard Wiener solution. It collects reference signals from different sources. Then it uses the multichannel feedforward adaptive structure to cancel the noise adaptively. If there is a very large number of measurement points, which, conventionally, the noise cancellation would require a correspondingly complex, possibly prohibitively large, structure. This approach enables very efficient usage of error signals, such that systems with large numbers of disturbance-cancellation points need employ only a relatively small number of error signals in the actual control-system implementation [34]. However, this approach does not consider the possibility of nonlinear property of the transmission channel (propagation path) between the reference signal and the primary signal.

Inspired by the research of [23][26][27][28][34][38][39][40][41][42][43][44][40][34], this thesis will focus the research in the area of adaptive noise cancellation under nonlinear transmission channel using multichannel adaptive lattice structure. A two-microphone system will be built. The truncated Volterra series expansion will be used to include the second-order or higher-order nonlinearity. The lattice technique will be used to reduce the sensitivity to the eigenvalue spread of the reference signal, with better convergence behaviour and better numerical properties. NLMS algorithm will be used to update the coefficients within the filter, with better performance in unknown statistics and time-varying input over LMS algorithm. The multichannel lattice structure should show positive results when the transmission channel has nonlinear properties.

2 SCLS FOR ANC

According to the lattice structure proposed by S. M. Kuo and D. R. Morgan [30], the traditional lattice structure for adaptive noise cancellation including the adaptive lattice predictor and the multiple regression transversal filter, as shown in Figure 1-5. Because its reference input is a single-channel input, it is also called SCLS (single-channel lattice structure) in comparison with MCLS (multi-channel lattice structure), which will be discussed in the next chapter.

The adaptive lattice predictor is a modular structure that contains a number of cascaded lattice stages. A steepest descent or an exact least-squares algorithm is used to adjust the filter coefficients independently at each stage [30].

2.1 Lattice Stage

The lattice stage, as illustrated in Figure 2-1, is the basic component with two input and two output channels within the lattice predictor [30].

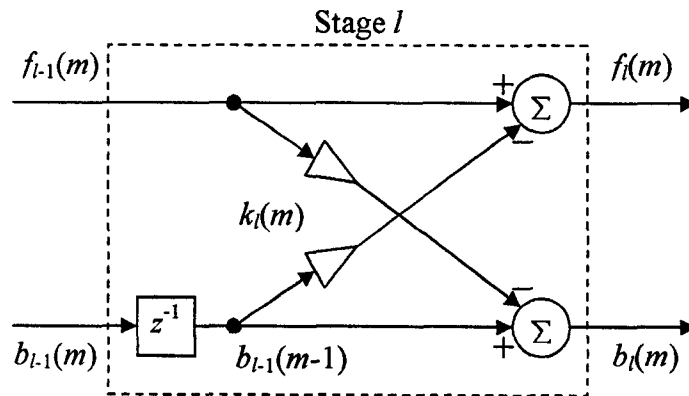


Figure 2-1: Detailed structure of l -th lattice stage

The recursive equations that describe the lattice structure are expressed by

$$f_l(m) = f_{l-1}(m) - k_l(m)b_{l-1}(m-1), \quad l = 1, 2, \dots, L-1 \quad (2.1.1)$$

and

$$b_l(m) = b_{l-1}(m-1) - k_l(m) f_{l-1}(m), \quad l = 1, 2, \dots, L-1 \quad (2.1.2)$$

where $f_l(m)$ is the forward prediction error, $b_l(m)$ is the backward prediction error, $k_l(m)$ is the reflection coefficient, l is the stage (order) index, and $L-1$ is the total number of cascaded stages [30].

2.2 Structure and Algorithm

The overall SCLS for ANC, as illustrated in Figure 1-5, is composed of two major parts: the adaptive lattice predictor and the multiple-regression transversal filter.

The adaptive lattice predictor consists of a number of cascaded lattice stages, as illustrated in Figure 2-1. The reference signal $x(m)$ is used as the input signal for stage 1, as shown in Figure 2-2 and expressed by

$$f_0(m) = b_0(m) = x(m) \quad (2.2.1)$$

Given these initial conditions and a set of reflection coefficients $k_1(m), \dots, k_{L-1}(m)$, the pair of outputs $f_l(m)$ and $b_l(m)$ can be produced by moving through the lattice filter, stage by stage, as illustrated in Figure 2-2.

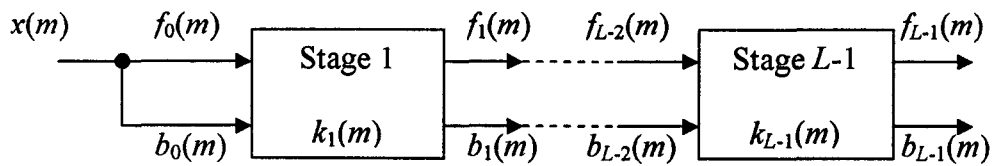


Figure 2-2: Lattice predictor of SCLS

To complete the specification of the adaptive lattice filter, a means of determining the reflection coefficients $\{k_l(m)\}$, $1 \leq l \leq L-1$, is required. In [30], a steepest-descent method

designed to minimize the sum of the MSEs of forward and backward prediction errors at each stage, expressed by

$$\xi_l(m) = E[f_l^2(m) + b_l^2(m)], \quad l = 1, 2, \dots, L-1 \quad (2.2.2)$$

where $\xi_l(m)$ is the cost function of stage l at time m .

Similar to the derivation of LMS algorithm [30], the cost function defined in (2.2.2) can be approximated by a sum of the instantaneous squared forward and backward prediction errors:

$$\hat{\xi}_l(m) = f_l^2(m) + b_l^2(m), \quad l = 1, 2, \dots, L-1 \quad (2.2.3)$$

From (2.1.1) and (2.1.2), the gradient of the $\hat{\xi}_l(m)$ with respect to the reflection coefficient $k_l(m)$ becomes

$$\begin{aligned} \nabla_{\hat{\xi}_l(m)} &= \frac{\partial \hat{\xi}_l(m)}{\partial k_l(m)} = \frac{\partial [f_l^2(m) + b_l^2(m)]}{\partial k_l(m)} \\ &= \frac{\partial \left\{ [f_{l-1}(m) - k_l(m)b_{l-1}(m-1)]^2 + [b_{l-1}(m-1) - k_l(m)f_{l-1}(m)]^2 \right\}}{\partial k_l(m)} \\ &= -2[f_l(m)b_l(m-1) + b_l(m)f_{l-1}(m)] \end{aligned} \quad (2.2.4)$$

Substituting this gradient estimate into the steepest-descent method, the reflection coefficient $k_l(m+1)$ can be recursively updated by

$$\begin{aligned} k_l(m+1) &= k_l(m) - \frac{\mu_l}{2} \nabla_{\hat{\xi}_l(m)} \\ &= k_l(m) + \mu_l [f_l(m)b_l(m-1) + b_l(m)f_{l-1}(m)], \quad l = 1, 2, \dots, L-1 \end{aligned} \quad (2.2.5)$$

where μ_l is the step size at the l -th stage.

As with the NLMS algorithm [30], the constant step size μ_l can be replaced by a time-varying step size $\mu_l(m)$ which is normalized to the signal power at the l -th stage. This time-varying step size has the advantage of responding to the changing input signal power, which can be estimated by recursively averaging the sum of the squared forward and backward prediction errors. Thus, let

$$\mu_l(m) = \frac{\mu}{\hat{P}_l(m)} \quad (2.2.6)$$

where μ is a constant, and

$$\hat{P}_l(m) = (1-\alpha)\hat{P}_l(m-1) + \alpha[f_{l-1}^2(m) + b_{l-1}^2(m-1)] \quad (2.2.7)$$

is the power estimate of the sum of the forward and backward prediction errors at the input of stage l , where $0 < \alpha \ll 1$ is a smoothing factor.

Thus the update formula in (2.2.5) becomes

$$k_l(m+1) = k_l(m) + \mu_l(m)[f_l(m)b_{l-1}(m-1) + b_l(m)f_{l-1}(m)] \quad (2.2.8)$$

where $\mu_l(m)$ is calculated by (2.2.6) and (2.2.7), $l = 1, 2, \dots, L-1$.

The major function of adaptive lattice predictor is to transform the correlated reference signals $\{x(m) \ x(m-1) \ \dots \ x(m-L+1)\}$ into a corresponding sequence of uncorrelated backward prediction errors $\{b_0(m) \ b_1(m) \ \dots \ b_{L-1}(m)\}$.

The detailed SCLS is illustrated in Figure 2-3.

The multiple regression transversal filter with coefficients $\{w_0(m) \ w_1(m) \ \dots \ w_{L-1}(m)\}$ then operates on the backward prediction errors $\{b_0(m) \ b_1(m) \ \dots \ b_{L-1}(m)\}$ to produce a

filter output $\hat{s}(m)$. As shown in Figure 2-3, the initial step of the multiple-regression transversal filter is formed as

$$e_0(m) = d(m) - w_0(m)b_0(m) \quad (2.2.9)$$

where $d(m)$ is the noisy primary signal, and the following steps of the multiple-regression transversal filter are formed as

$$e_i(m) = e_{i-1}(m) - w_i(m)b_i(m), \quad i = 1, 2, \dots, L-1 \quad (2.2.10)$$

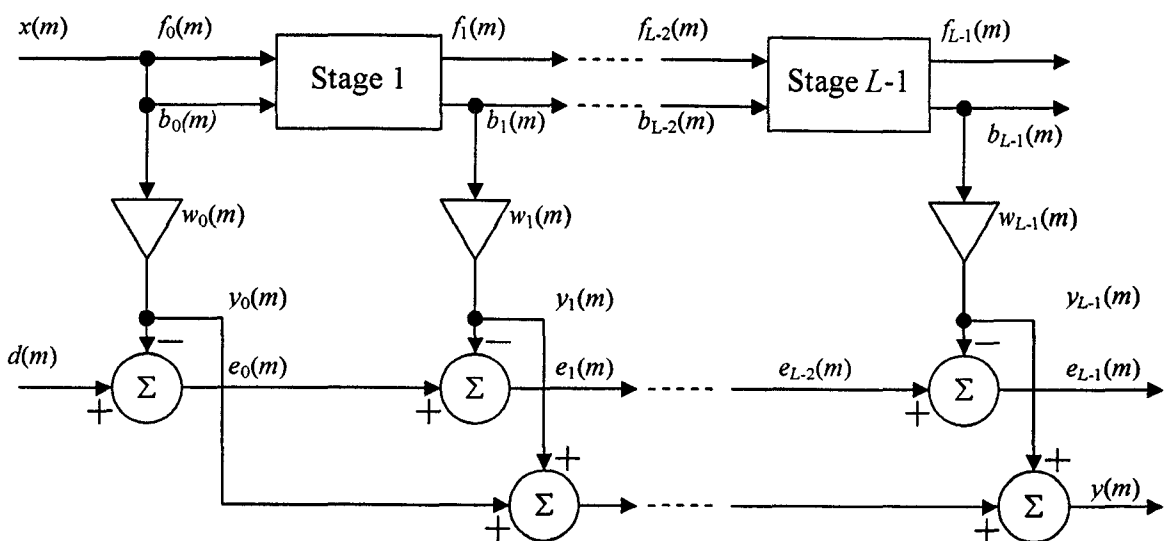


Figure 2-3: Detailed structure of SCLS

Finally, the output signal is formed by

$$y(m) = \sum_{i=0}^{L-1} w_i(m)b_i(m) \quad (2.2.11)$$

and $\hat{s}(m)$, i.e. the estimate of the noise-free primary signal $s(m)$, is formed by

$$\hat{s}(m) = e_{L-1}(m) = d(m) - y(m) = d(m) - \sum_{i=0}^{L-1} w_i(m)b_i(m) \quad (2.2.12)$$

The coefficients of the multiple regression transversal filter are updated by the NLMS algorithm, expressed by

$$w_i(m+1) = w_i(m) + \frac{\mu}{\hat{Q}_i(m)} b_i(m) e_i(m) \quad (2.2.13)$$

where μ is a constant and $\hat{Q}_i(m)$ is an estimate of the backward prediction error power at stage i , which can be obtained using

$$\hat{Q}_i(m) = (1 - \alpha) \hat{Q}_i(m-1) + \alpha b_i^2(m) \quad (2.2.14)$$

where α is a smoothing factor and $0 < \alpha \ll 1$.

The overall SCLS can be viewed as a lattice pre-processor followed by a transversal filter. The pre-processor (adaptive lattice predictor) decorrelates (whitens) the reference signal to produce uncorrelated backward prediction error signal. The transversal filter operates on these uncorrelated signals; thus the convergence of the adaptive transversal filter does not suffer from eigenvalue disparity problem [30].

3 MCLS FOR ANC

This chapter introduces the MCLS (multichannel lattice structure) for ANC under nonlinear transmission channel. A specific definition of the problem and the basic assumption are provided. The detailed description of its structure and algorithm is illustrated.

3.1 Problem Definition and Assumption

The objective of ANC is to eliminate the noise from the noisy primary signal and recover the noise-free primary signal adaptively. ANC uses the reference signal, noisy primary signal and the adaptive structure to cancel the additive noise signal from the noisy primary signal adaptively, and recover the noise-free primary signal eventually.

As illustrated in Figure 1-1, the continuous-time analog impulse response of the transmission channel between the reference signal and the noisy primary signal, $h[\cdot]$, can be linear or nonlinear. And in most of the cases, it should be nonlinear. If the transmission channel is nonlinear, i.e. nonlinear impulse response, the existing SCLS normally can not recover the noise-free primary signal successfully. SCLS has bad performance and poor convergence properties over the nonlinear cases.

By using MCLS over the nonlinear cases, the noise-free primary signal can be recovered successfully. Some experiments show the positive results using this structure over nonlinear cases.

To simplify the simulation, the problem model, as illustrated in Figure 3-1, is used in this thesis.

The problem is stated as follows:

Given a reference signal $x(m)$ and a noisy primary signal $d(m)$, which can be expressed by

$$d(m) = s(m) + n(m) = s(m) + h[x(m)] \quad (3.1.1)$$

where $s(m)$ is the noise-free primary signal, $n(m)$ is the additive noise signal within the noisy primary signal $d(m)$. According to (3.1.1), the additive noise $n(m)$ is the reference signal $x(m)$ degraded by the nonlinear transmission channel, where h is the discrete-time digital impulse response of the nonlinear filter equivalent to the nonlinear transmission channel, and m is the discrete-time index.

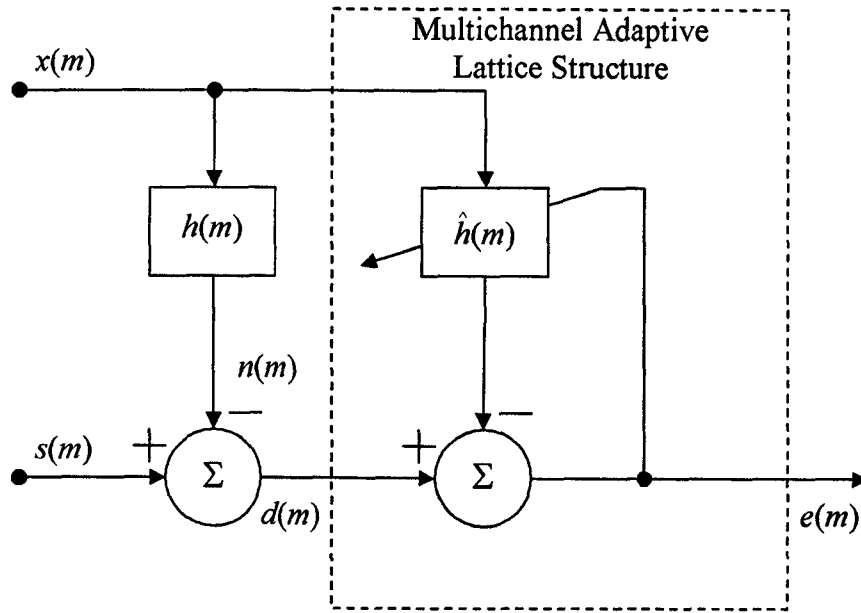


Figure 3-1: Problem model for MCLS proposal

Assume the impulse response of the nonlinear transmission channel can be expressed by the truncated Volterra series expansion, i.e. $n(m)$ can be simulated using the Volterra series expansion, as shown in (1.2.5) or the truncated version as shown in (1.2.6) to make it realizable. To simplify the structure and the calculation, $n(m)$ is assumed that it can be expressed by a second-order Volterra series expansion as

$$n(m) = h_0 + \sum_{k_1=0}^{M-1} h_1(k_1)x(m-k_1) + \sum_{k_1=0}^{M-1} \sum_{k_2=0}^{M-1} h_2(k_1, k_2)x(m-k_1)x(m-k_2) \quad (3.1.2)$$

where h_0 , $h_1(k_1)$, $h_2(k_1, k_2)$ is known as the 0th, 1st and 2nd order Volterra kernel of the system. The 2nd order Volterra kernels can characterize the second-order nonlinearity of the system.

The goal of MCLS is to eliminate the additive noise signal $n(m)$ and recover the noise-free primary signal $s(m)$ from the noisy primary signal $d(m)$ when the transmission channel is nonlinear, while the nonlinear transmission channel can be characterized using the second-order truncated Volterra series expansion.

3.2 MCLS

As described in the previous chapter, the existing SCLS exhibits bad performance and poor convergence property under nonlinear cases. They are, therefore, unsuitable for practical ANC under nonlinear transmission channel.

The objective of MCLS is to produce satisfied results under nonlinear transmission channel case. The algorithm should exhibit improved convergence property over existing SCLS and should provide meaningful solutions in the presence of nonlinear property on the transmission channel.

MCLS uses the Volterra series expansion of the nonlinear system, and provides the significant improvement on ANC performance and convergence property under nonlinear transmission channel.

The block diagram of MCLS is shown in Figure 3-2. The single channel reference signal $x(m)$ is fed into the SIMO module, generating the multichannel output vector $\underline{X}(m)$. The multichannel vector $\underline{X}(m)$ is then fed into the first MALP stage and the initial stage of MMRTF. Each MALP stage produces the forward prediction error vector $\underline{F}(m)$ and backward prediction error vector $\underline{B}(m)$. The generated $\underline{F}(m)$ and $\underline{B}(m)$ are then fed into the next MALP stage. Meanwhile, the initial MMRTF stage uses $\underline{X}(m)$ and $d(m)$ to generate the initial error output $E_0(m)$, the generated error output $E(m)$ and backward prediction error vector $\underline{B}(m)$ are fed into the next MMRTF module. The signal is

processed stage by stage. The error output of the last MMRTF stage, $\underline{E}_{L-1}(m)$, is the output of MCLS, i.e. $\hat{s}(m)$, which is the estimate the noise-free primary signal $s(m)$.

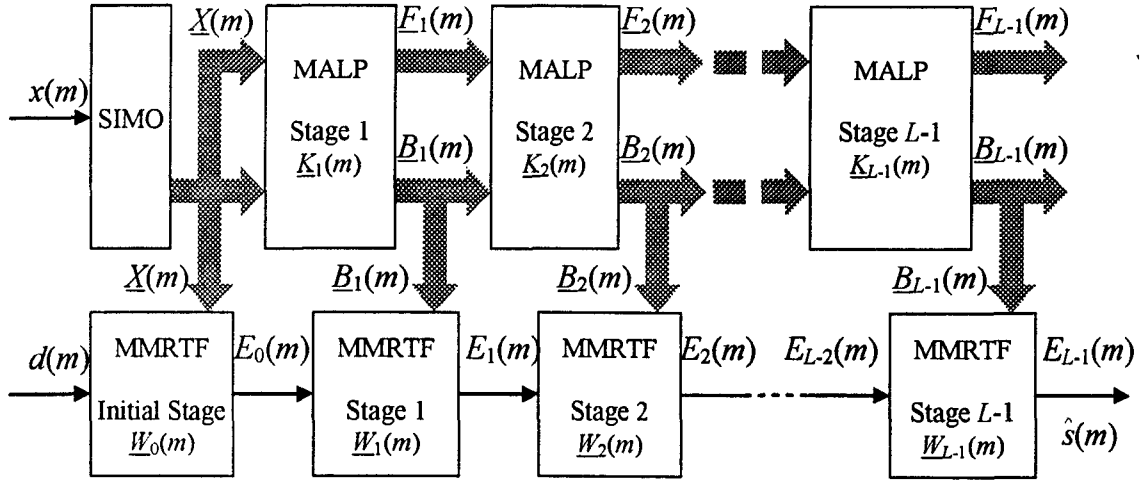


Figure 3-2: Block diagram of MCLS

The reflection coefficient vector $\underline{K}(m)$ of each MALP stage and the coefficient vector $\underline{W}(m)$ of each MMRTF stage are generated adaptively using adaptive algorithm, such as LMS/NLMS or RLS. In this thesis, NLMS algorithm is used.

In the following sections, the detailed structure inside each module will be described extensively. The process of how the noisy primary signal is filtered by using the reference signal and MCLS will be illustrated.

3.2.1 SIMO Module

Using the concept of second-order truncated Volterra series expansion, as (3.1.2), it involves all the second-order nonlinearity. The linear part is shown as

$$\begin{bmatrix} h_1(0)x(m) \\ h_1(1)x(m-1) \\ \vdots \\ h_1(M-1)x(m-M+1) \end{bmatrix} \quad (3.2.1)$$

and the nonlinear part is shown as

$$\begin{bmatrix} h_2^{(0,0)} x^2(m) & h_2^{(0,1)} x(m)x(m-1) & \dots & h_2^{(0,M-1)} x(m)x(m-M+1) \\ h_2^{(1,1)} x^2(m-1) & h_2^{(1,2)} x(m-1)x(m-2) & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ h_2^{(M-1,M-1)} x^2(m-M+1) & 0 & \dots & 0 \end{bmatrix} \quad (3.2.2)$$

where

$$h_2^{(i,j)} = \begin{cases} h_2(i,j), & i = j \\ h_2(i,j) + h_2(j,i), & i \neq j \end{cases} \quad (3.2.3)$$

From (3.2.1) and (3.2.2), the elements in the first row of each matrix or vector are called *basic elements*, which can be used to characterize the linearity and second-order nonlinearity of the system. The other elements are only those basic elements with simple delays, called *non-basic elements*. In MCLS, only basic elements are used. Some results in later simulations prove that MCLS can provide effective ANC even if the non-basic elements exist.

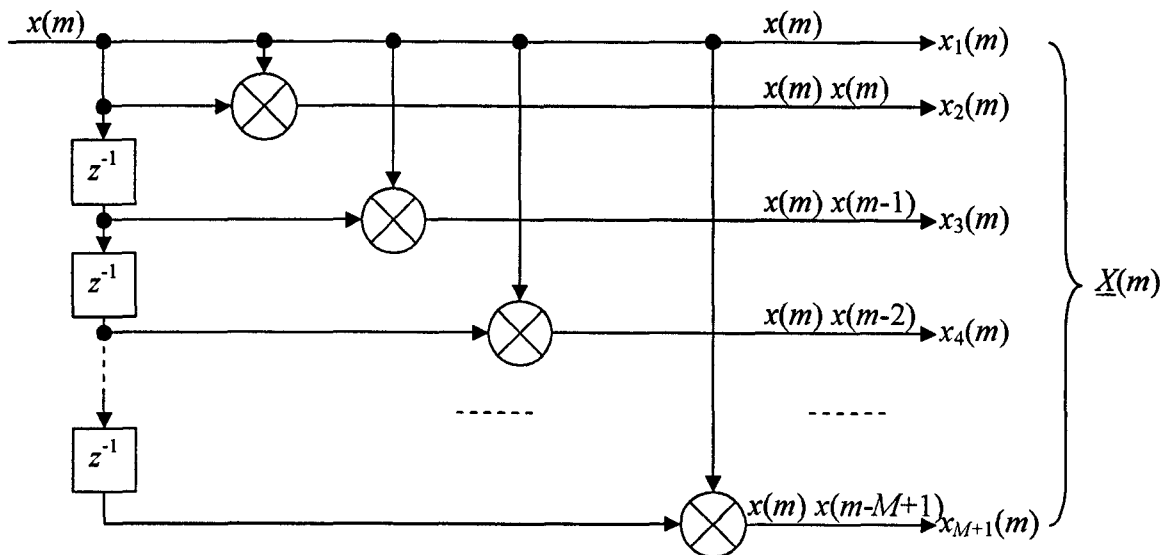


Figure 3-3: Detailed structure of SIMO

In this thesis, SIMO (single-input-multiple-output) module, as shown in Figure 3-3, is used to produce the linearity and the second-order nonlinearity of the nonlinear system by generating the multichannel reference signal vector $\underline{X}(m)$.

The multichannel reference signal vector $\underline{X}(m)$ is defined by

$$\underline{X}(m) \triangleq \begin{cases} x_1(m) = x(m) \\ x_2(m) = x^2(m) \\ x_3(m) = x(m)x(m-1) \\ \vdots \\ x_M(m) = x(m)x(m-M+2) \\ x_{M+1}(m) = x(m)x(m-M+1) \end{cases} \quad (3.2.4)$$

where $M-1$ is the maximum delay, and $\underline{X}(m)$ has $M+1$ channels.

It is easy to expand the second-order nonlinearity into the arbitrary-order nonlinearity using the similar SIMO structure. For 3rd-order with maximum delay $M-1=2$, the multichannel reference signal vector $\underline{X}(m)$ is defined by

$$\underline{X}(m) \triangleq \begin{cases} x_1(m) = x(m) \\ x_2(m) = x^2(m) \\ x_3(m) = x(m)x(m-1) \\ x_4(m) = x(m)x(m-2) \\ x_5(m) = x^3(m) \\ x_6(m) = x^2(m)x(m-1) \\ x_7(m) = x^2(m)x(m-2) \\ x_8(m) = x(m)x^2(m-1) \\ x_9(m) = x(m)x^2(m-2) \\ x_{10}(m) = x(m)x(m-1)x(m-2) \end{cases} \quad (3.2.5)$$

For 2nd order SIMO with $M-1$ maximum delays, the number of basic element is $M+1$. For P^{th} -order SIMO with $M-1$ maximum delays, the total number of basic element is calculated by

$$1 + C_M^1 + C_M^2 + \dots + C_M^{P-1} = 1 + \sum_{p=0}^{P-1} \left[\prod_{q=0}^p (M-p) \right] \quad (3.2.6)$$

From (3.2.6), the total number of SIMO output channels can be calculated for arbitrary number of order and arbitrary number of maximum delays. When the order of nonlinearity and maximum delay are increased, the total output channel number of SIMO module will be increased exponentially. Some calculation results are shown in Table 3-1.

Table 3-1: Total channels of different order (P) and different maximum delay ($M-1$)

Channels	$M-1=1$	$M-1=2$	$M-1=3$	$M-1=4$	$M-1=5$
$P=2$	3	4	5	6	7
$P=3$	5	10	17	26	37
$P=4$	7	16	41	86	157
$P=5$	9	22	65	206	517
$P=6$	11	28	89	326	1237

In this thesis, the research will be concentrated on the second-order nonlinear model, and some experiments will be made on the third-order nonlinear model with small value of maximum delay $M-1$ to verify whether MCLS can show positive results on higher-order cases.

3.2.2 MALP Module

The major function of MALP (multichannel-adaptive-lattice-predictor) module is to transform a sequence of correlated reference signal vectors $\{\underline{X}(m), \underline{X}(m-1), \dots, \underline{X}(m-L+1)\}$ into a corresponding sequence of uncorrelated backward prediction error vectors $\{\underline{B}_0(m), \underline{B}_1(m), \dots, \underline{B}_{L-1}(m)\}$. Each MALP module includes $M+1$ lattice stages, as illustrated in 2.1. The structure of the j -th MALP stage is illustrated in Figure 3-4.

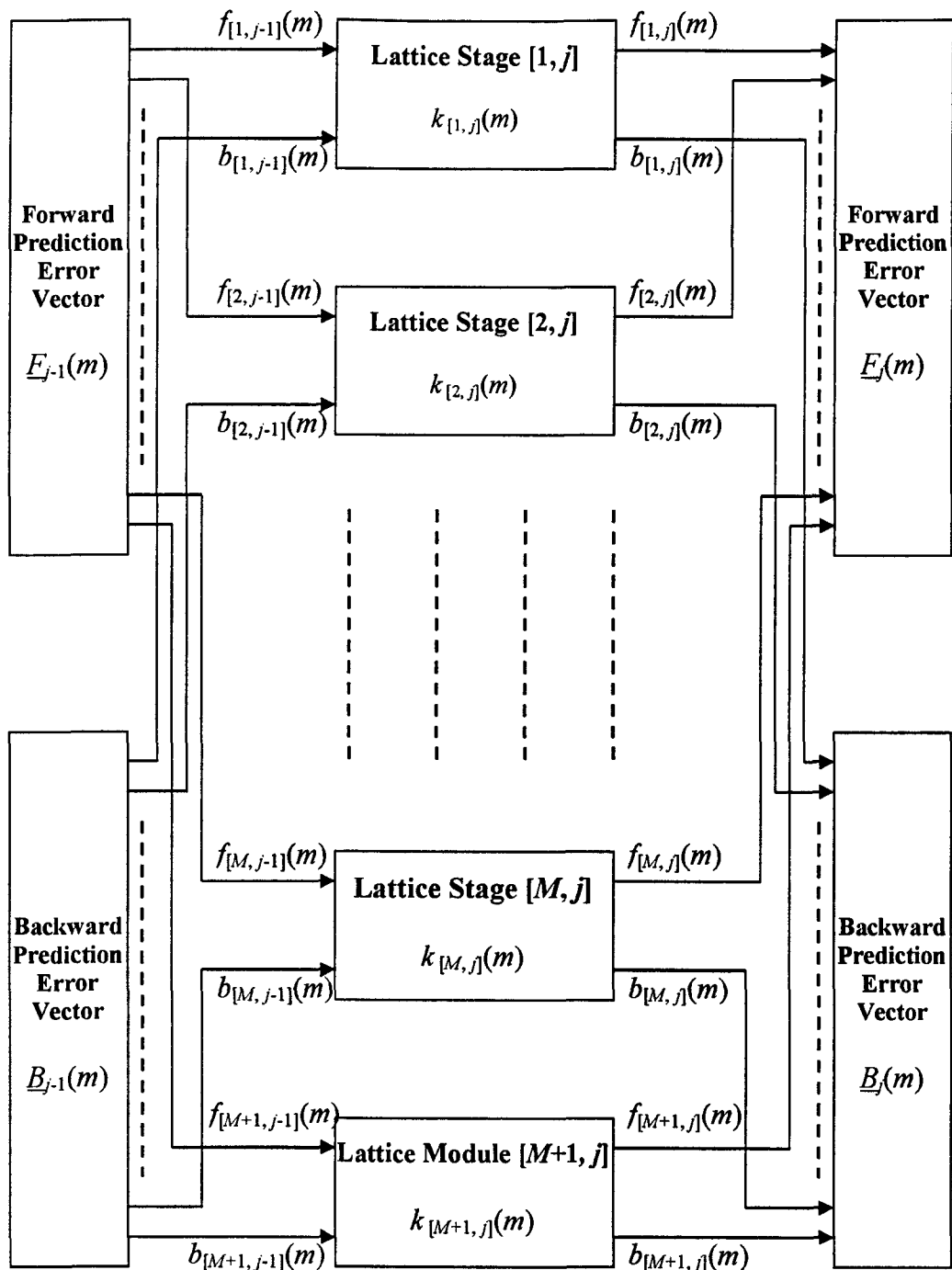


Figure 3-4: Structure of j -th MALP stage

After feeding the reference signal $x(m)$ into the SIMO module, the multichannel reference signal vector $\underline{X}(m)$ is generated, which contains $M+1$ elements. $\underline{X}(m)$ is fed into the first MALP stages, as

$$\underline{E}_0(m) = \underline{B}_0(m) = \underline{X}(m) \quad (3.2.7)$$

All the reflection coefficients, $k_{[i,j]}(m)$, are calculated adaptively and individually. The reflection coefficient vector in j -th MALP stage, i.e. $\underline{K}_j(m)$ shown in Figure 3-2, is defined by

$$\underline{K}_j(m) = \left[k_{[1,j]}(m), k_{[2,j]}(m), \dots, k_{[M,j]}(m), k_{[M+1,j]}(m) \right]^T \quad (3.2.8)$$

The forward prediction error vector on j -th MALP stage, i.e. $\underline{F}_j(m)$ shown in Figure 3-2, is defined by

$$\underline{F}_j(m) = \left[f_{[1,j]}(m), f_{[2,j]}(m), \dots, f_{[M,j]}(m), f_{[M+1,j]}(m) \right]^T \quad (3.2.9)$$

The backward prediction error vector on j -th MALP stage, i.e. $\underline{B}_j(m)$ in shown in Figure 3-2, is defined by

$$\underline{B}_j(m) = \left[b_{[1,j]}(m), b_{[2,j]}(m), \dots, b_{[M,j]}(m), b_{[M+1,j]}(m) \right]^T \quad (3.2.10)$$

According to the definition of the lattice stage, each element within the forward prediction error vector $\underline{F}_j(m)$ is calculated by

$$f_{[i,j]}(m) = f_{[i,j-1]}(m) - k_{[i,j]}(m) b_{[i,j-1]}(m-1) \quad (3.2.11)$$

and each element within the backward prediction error vector $\underline{B}_j(m)$ is calculate by

$$b_{[i,j]}(m) = b_{[i,j-1]}(m-1) - k_{[i,j]}(m) f_{[i,j-1]}(m) \quad (3.2.12)$$

where $i = 1, 2, \dots, M+1$ and $j = 1, 2, \dots, L-1$.

From the previous description of the NLMS algorithm in (2.2.6), (2.2.7) and (2.2.8), the time-varying step size $\mu_{[i,j]}(m)$ is defined by

$$\mu_{[i,j]}(m) = \frac{\mu}{\hat{P}_{[i,j]}(m)} \quad (3.2.13)$$

where μ is a constant, and $\hat{P}_{[i,j]}(m)$ is the power estimate of the sum of the forward and backward prediction error squares at the input of lattice module in i -th channel and j -th MALP stage, defined by

$$\hat{P}_{[i,j]}(m) = (1-\alpha)\hat{P}_{[i,j]}(m-1) + \alpha [f_{[i,j-1]}^2(m) + b_{[i,j-1]}^2(m-1)] \quad (3.2.14)$$

where α is a smoothing factor and $0 < \alpha \ll 1$.

The reflection coefficient of the lattice module in i -th channel and j -th MALP stage, $k_{[i,j]}(m)$, can be recursively updated by

$$k_{[i,j]}(m+1) = k_{[i,j]}(m) + \mu_{[i,j]}(m) [f_{[i,j]}(m)b_{[i,j-1]}(m-1) + b_{[i,j]}(m)f_{[i,j-1]}(m)] \quad (3.2.15)$$

The uncorrelated backward prediction error vectors $\underline{B}_0(m)$, $\underline{B}_1(m)$, ..., and $\underline{B}_{L-1}(m)$ are generated stage by stage accordingly.

3.2.3 MMRTF Module

The uncorrelated backward predictor error vectors, generated by the MALP modules, are fed into the MMRTF (multichannel-multiple-regression-transversal-filter) modules. The major function of MMRTF module is to produce the filtered output from the generated backward prediction error vectors $\underline{B}(m)$ and the noisy primary signal $d(m)$.

The structure of j -th MMRTF stage is illustrated in Figure 3-5. In (3.2.10), the uncorrelated backward prediction error vector $\underline{B}_j(m)$, which is generated by the j -th

MALP stage, contains $M+1$ elements. The coefficients within j -th MMRTF stage construct the coefficient vector $\underline{W}_j(m)$, as defined by

$$\underline{W}_j(m) = [w_{[1,j]}(m), w_{[2,j]}(m), \dots, w_{[M+1,j]}(m)]^T \quad (3.2.16)$$

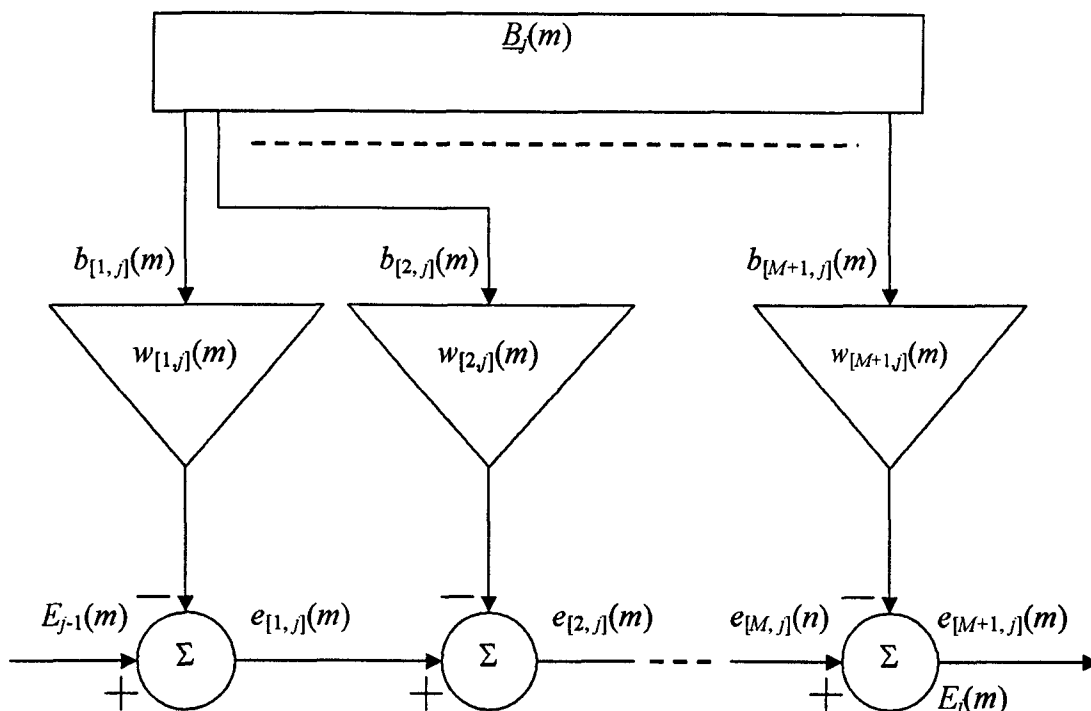


Figure 3-5: Structure of j -th MMRTF stage

In the first channel of j -th MMRTF stage, the error output $e_{[1,j]}(m)$ is calculated by

$$e_{[1,j]}(m) = E_{j-1}(m) - w_{[1,j]}(m)b_{[1,j]}(m) \quad (3.2.17)$$

then the rest of $e_{[i,j]}(m)$, where $1 < i \leq M+1$, is calculated recursively by

$$e_{[i,j]}(m) = e_{[i-1,j]}(m) - w_{[i,j]}(m)b_{[i,j]}(m) \quad (3.2.18)$$

where $b_{[i,j]}(m)$ is the i -th element in $\underline{B}_j(m)$, which is the backward predictor error vector from j -th MALP stage.

The error output $e_{[M+1,j]}(m)$ becomes the filtered output of j -th MMRTF stage, as defined by

$$e_{[M+1,j]}(m) = E_j(m) \quad (3.2.19)$$

By combining (3.2.17), (3.2.18) and (3.2.19), the error output of j -th MMRTF stage can be calculated by

$$E_j(m) = E_{j-1}(m) - \sum_{i=1}^{M+1} w_{[i,j]}(m) b_{[i,j]}(m) = E_{j-1}(m) - \underline{W}_j(m)^T \underline{B}_j(m) \quad (3.2.20)$$

where $\underline{W}_j(m)$ and $\underline{B}_j(m)$ are defined in (3.2.16) and (3.2.10).

At the initial MMRTF stage, where $j = 0$, (3.2.20) is replaced by

$$E_0(m) = d(m) - \underline{W}_0(m)^T \underline{B}_0(m) \quad (3.2.21)$$

where $d(m)$ is the noisy primary signal.

Using NLMS algorithm, the filter coefficient in i -th channel and j -th MMRTF stage, $w_{[i,j]}(m)$, is updated by

$$w_{[i,j]}(m+1) = w_{[i,j]}(m) + \frac{\mu}{\hat{Q}_{[i,j]}(m)} b_{[i,j]}(m) e_{[i,j]}(m) \quad (3.2.22)$$

where μ is a constant and $\hat{Q}_{[i,j]}(m)$ is the power estimate of the backward predictor error at the i -th channel within $\underline{B}_j(m)$, i.e. the backward prediction error vector at the j -th MALP stage. $\hat{Q}_{[i,j]}(m)$ can be calculated recursively by

$$\hat{Q}_{[i,j]}(m) = (1 - \alpha) \hat{Q}_{[i,j]}(m-1) + \alpha b_{[i,j]}^2(m) \quad (3.2.23)$$

and the filtered output of the last MMRTF stage, i.e. $(L-1)$ -th stage, becomes the estimate of the noise-free primary signal $s(m)$, i.e. $\hat{s}(m)$, as defined by

$$E_{L-1}(m) = \hat{s}(m) \quad (3.2.24)$$

3.2.4 Summary of the Algorithm

After all the detailed descriptions in the previous sections, the architecture of MCLS is illustrated. SIMO module, MALP module and MMRTF module are described individually in details. The definition and calculation of the inputs and outputs inside and between these modules are illustrated.

In this section, the overall algorithm of MCLS will be summarized, from where the simulations can be implemented.

The overall algorithm is shown as the follows.

- Set initial conditions:
 - As the system is causal, the value of all variables is zero when $m \leq 0$.
 - Set the constants, μ and α , let $0 < \mu, 0 < \alpha \ll 1$.
- For the sample iterations (m): $m = 1, 2, 3, \dots$
 - (a) SIMO module:
 - Generate the multichannel reference signal vector $\underline{X}(m)$ from the single-channel reference signal $x(m)$ using (3.2.4).
 - (b) $e_{[0,0]}(m) = d(m)$.
 - (c) For the stage iterations (j): $j=0, 1, \dots, L-1$

i. MALP stage:

- If $j = 0$, $\underline{F}_0(m) = \underline{B}_0(m) = \underline{X}(m)$;
- If $j \neq 0$,
 - Calculate $\underline{K}_j(m)$, i.e. the reflection coefficient vector in j -th MALP stage, using (3.2.8), (3.2.13), (3.2.14), and (3.2.15).
 - Calculate $\underline{F}_j(m)$ and $\underline{B}_j(m)$, i.e. the forward prediction error vector and backward prediction error vector on j -th MALP stage, using (3.2.9), (3.2.10), (3.2.11), and (3.2.12),.

ii. MMRTF stage:

- Calculate the filter coefficient vector on j -th MMRTF stage, i.e. $\underline{W}_j(m)$, using (3.2.16), (3.2.22), and (3.2.23).
- Calculate the error output at j -th MMRTF stage, i.e. $E_j(m)$, using (3.2.20) or (3.2.21).

(d) The filtered output of last MMRTF stage, $E_{L-1}(m)$, becomes the estimate of noise-free primary signal $s(m)$, i.e. $E_{L-1}(m) = \hat{s}(m)$.

From the above summary of the algorithm on MCLS, the simulation and comparison will be made to verify whether MCLS has the performance improvement over SCLS when the transmission channel has nonlinear properties.

3.3 Relationship between MCLS and SCLS

The detailed structure of MCLS is illustrated in Figure 3-2, Figure 3-3, Figure 3-4 and Figure 3-5. The detailed structure of the existing SCLS is illustrated in Figure 2-3 respectively. When the total number of channels becomes one, i.e. only linear part is going through the structure, MCLS will be the same structure as SCLS, which means that

the existing SCLS is a special case of MCLS. MCLS can be considered as an extension of SCLS to provide ANC under nonlinear cases.

4 SIMULATION AND COMPARISON

This chapter evaluates the performance of MCLS, which is discussed extensively in Chapter 3, and compares the performance between MCLS and SCLS. Both MCLS and SCLS are implemented using MATLAB and the simulations are carried out using a wide variety of simulated inputs and simulated transmission channels. Figure 4-1 shows the test model for the simulation and comparison in this thesis.

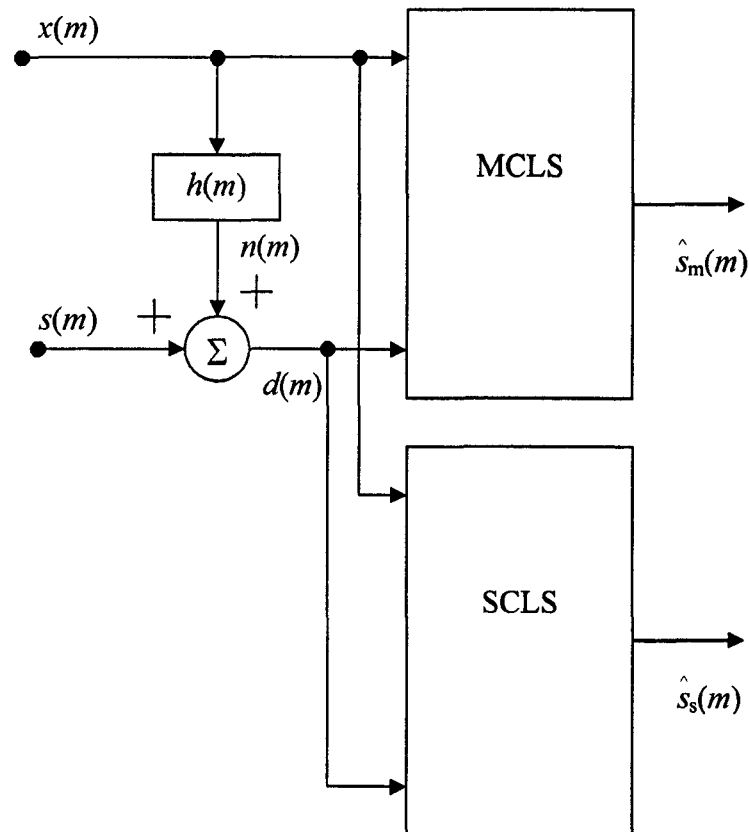


Figure 4-1: Test model for the simulation and comparison

The definitions of the variables in Figure 4-1 are shown as follows:

- $x(m)$ is the reference signal.
- $s(m)$ is the noise-free primary signal.
- $n(m)$ is the additive noise signal, where $n(m) = h[x(m)]$. It is the distorted version of the reference signal $x(m)$, as shown in (3.1.1).

- $h(m)$ is the impulse response of the simulated transmission channel.
- $d(m)$ is the noisy primary signal, i.e. the noise-free primary signal $s(m)$ with the additive noise signal $n(m)$.
- $\hat{s}_m(m)$ is the filtered output signal of MCLS, as illustrated in Figure 3-2. It is one estimate of the noise-free primary signal $s(m)$.
- $\hat{s}_s(m)$ is the filtered output of SCLS, as illustrated in Figure 2-3. It is another estimate of the noise-free primary signal $s(m)$.

The simulations were implemented using a number of different noise-free primary signals, different types of simulated transmission channels, and different type of reference signals. Both structures, MCLS and SCLS, are evaluated to compare the stability and the performance upon different simulation scenarios.

4.1 Definitions

Before starting the simulation analysis, the definition of measurements, the definition of different simulated transmission channels and the definition of input signals are described in this section.

4.1.1 Measurements

To implement the simulation and comparison, the definitions of some frequently-used measurements need to be emphasized.

4.1.1.1 RE

The RE (residue-error) is the difference between the noise-free primary signal and its estimate, i.e. the filtered output either from MCLS or from SCLS. $RE_m(m)$ and $RE_s(m)$ are used to represent the residue error from MCLS and SCLS respectively. They are defined by

$$\text{RE}_m(m) = \hat{s}_m(m) - s(m) \quad (4.1.1)$$

$$\text{RE}_s(m) = \hat{s}_s(m) - s(m) \quad (4.1.2)$$

where $s(m)$ is the noise-free primary signal, $\hat{s}_m(m)$ is the filtered output signal from MCLS, $\hat{s}_s(m)$ the filtered output signal from SCLS, as shown in Figure 4-1.

RE is an instant measurement, which shows the instant result from both structures to evaluate and compare their performance.

4.1.1.2 MSE

Sometimes, the residue error can not show the performance difference clearly. Another measure, MSE (mean-square-error), is introduced. The MSE measurement is a weighted average of the squared distance between the noise-free primary signal and its estimate with the number of samples as the weight factors. It is used to measure the mean square-error results of the simulations under certain input SNR level, defined by

$$\text{MSE} = \frac{1}{M} \sum_{m=1}^M e^2(m) = \frac{1}{M} \sum_{m=1}^M [\hat{s}(m) - s(m)]^2 \quad (4.1.3)$$

where $e(m)$ is the residue error. $s(m)$ is the noise-free primary signal. $\hat{s}(m)$, i.e. the estimate of $s(m)$, is the filtered output signal from either MCLS or SCLS. M is the number of samples within the measurement period.

MSE is an average measurement.

4.1.1.3 SNR

The SNR (signal-to-noise ratio) is a measurement of the signal strength relative to the additive noise. The ratio is usually measured in decibels (dB). In this thesis, the input and output SNR are defined by

$$\text{SNR}_{\text{input}} = 10 \log_{10} \left(\frac{\sum_{m=1}^M s^2(m)}{\sum_{m=1}^M n^2(m)} \right) (\text{dB}) \quad (4.1.4)$$

$$\text{SNR}_{\text{output}} = 10 \log_{10} \left(\frac{\sum_{i=1}^M s^2(m)}{\sum_{i=1}^M e^2(m)} \right) (\text{dB}) \quad (4.1.5)$$

where $s(m)$ is the noise-free primary signal. $n(m)$ is the additive noise signal, as shown in (3.1.1). $e(m)$ is the residue error signal, as shown in (4.1.3). M is the total number of samples in $s(m)$, $y(m)$, and $e(m)$ within the given measurement period.

SNR defined in (4.1.4) and (4.1.5) are the average measurement. They are used to provide the general evaluation of the performance from both MCLS and SCLS. The SNR levels mentioned in the following simulations and comparisons are all referred to those average SNR levels.

In the following simulations, different input SNR levels are generated to evaluate MCLS and SCLS over different noise environments. Different output SNR levels are measured to compare the performance between MCLS and SCLS regarding the ability of noise cancellation.

4.1.2 Transmission Channels

There are three types of transmission channels being used in this thesis:

- Second-order nonlinear transmission channel, the transmission channel with second-order nonlinearity, named 'Second-Non' in the simulation titles.
- Linear transmission channel, the transmission channel without nonlinearity, named 'Linear' in the simulation titles.

- Third-order nonlinear transmission channel, the transmission channel with third-order nonlinearity, named 'Third-Non' in the simulation titles.

A certain number of channels are utilized in each transmission channel type. The description of those channels is illustrated in Table 4-1. Various transmission channels are simulated to provide the solid evidence of the evaluation and comparison upon MCLS and SCLS.

Table 4-1: Description of the transmission channels used in the simulations

Channel Type	Chan #	Impulse response of the transmission channel
Second-Non	1	$h[x(m)] = 0.7x(m) + 0.2x(m)x(m) + 0.1x(m)x(m-2)$
	2	$h[x(m)] = 0.1x(m) + 0.4x(m)x(m-1) + 0.5x(m)x(m-2)$
	3	$h[x(m)] = 0.2x(m) + 0.1x(m-1) + 0.1x(m-2) + 0.1x^2(m) + 0.1x^2(m-1) + 0.1x^2(m-2) + 0.1x(m)x(m-1) + 0.1x(m)x(m-2) + 0.1x(m-1)x(m-2)$
	4	$h[x(m)] = 0.2x(m) + 0.3x(m-5)x(m-7) + 0.5x(m-11)x(m-12)$
Linear	1	$h[x(m)] = 0.7x(m) + 0.2x(m-1) + 0.1x(m-2)$
	2	$h[x(m)] = 0.3x(m) + 0.7x(m-2)$
Third-Non	1	$h[x(m)] = 0.1x(m) + 0.3x(m-1)x(m-1) + 0.6x(m)x(m-1)x(m-2)$
	2	$h[x(m)] = 0.4x(m) + 0.3x(m)x(m) + 0.3x(m-2)x(m-2)x(m-2)$

4.1.3 Signals

The types of noise-free primary signal and reference signal are defined in this section. These signals will be fed into MCLS and SCLS to evaluate and compare the results upon different simulation scenarios.

4.1.3.1 Signal Type

There are three types of input used in this thesis, single/multiple-frequency signal, voice signal, and WGN/CGN signal.

4.1.3.1.1 Single/Multiple-Frequency Signal

The single/multiple-frequency signal is an input constructed by one or several different frequencies. It can be used either as the noise-free primary signal or as the reference signal. In this thesis, the single-frequency signal is defined by

$$s(m) = \cos\left(2\pi \frac{f_1}{f_s} m\right) \quad (4.1.6)$$

and the two-frequency signal is defined by

$$s(m) = \frac{1}{2} \cos\left(2\pi \frac{f_1}{f_s} m\right) + \frac{1}{2} \cos\left(2\pi \frac{f_2}{f_s} m\right) \quad (4.1.7)$$

where f_1 and f_2 are different frequencies, and f_s is the sampling frequency. 16 kHz is used as the sampling frequency in this thesis.

The magnitude of the frequency response of one-frequency signal, where $f_1 = 100\text{Hz}$, is shown in Figure 4-2(a). The magnitude of the frequency response of two-frequency signal, where $f_1 = 500\text{Hz}$ and $f_2 = 2\text{ kHz}$, is shown in Figure 4-2(b).

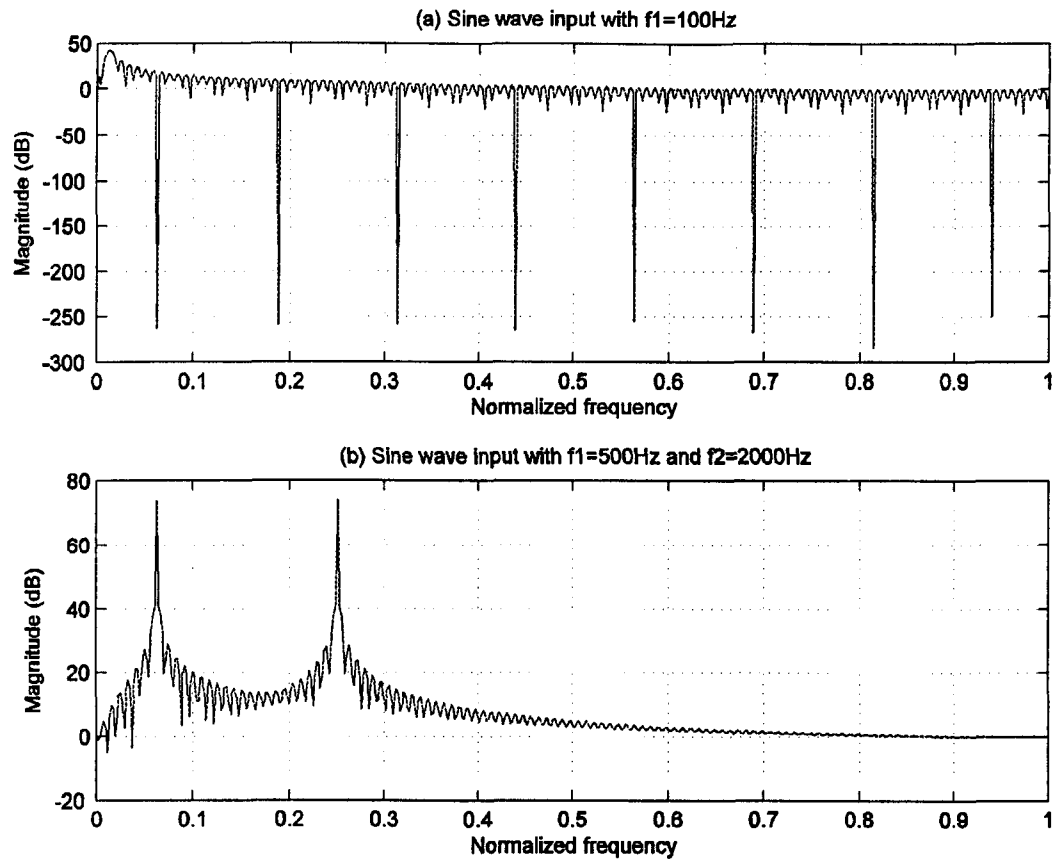


Figure 4-2: Frequency response of single/multiple-frequency signal

4.1.3.1.2 Voice Signal

The voice signals, i.e. the digitized wave files chosen from the TIMIT speech database, are used as the noise-free primary signal. The sampling frequency f_s is 16 kHz. The description of the voice signals used in this thesis is illustrated in Table 4-2.

Table 4-2: Definition of voice signals

Name	Sex	Dialect Region /File Set	Sentence	Length (samples)
si1573.wav	Female	DR1 ² /TEST ³	His captain was thin and haggard and his beautiful boots were worn and shabby.	79565

² DR1 – Speaker dialect region number 1, which is New England region.

Table 4-2 (Continued)

si919.wav	Male	DR3 ⁴ /TEST	Obviously, the bridal pair has many adjustments to make to their new situation.	79975
si2194.wav	Female	DR6 ⁵ /TEST	He had fallen into a soft job, and now the job was gone and he was stranded.	75572

Different voice signals from different speakers and over different dialect regions are selected to evaluate and compare the performance of noise cancellation ability using both MCLS and SCLS.

4.1.3.1.3 WGN/CGN Signal

White noise is simply a sequence of uncorrelated random variables. A random process consisting of a sequence of uncorrelated real-valued Gaussian random variables is a white noise process referred to as WGN (white Gaussian noise) [30]. In this thesis, the WGN signal is generated by the *wgn(.)* function in MATLAB.

According to the definition, the magnitude response of ideal WGN signal should be flattened in the frequency domain. The CGN (coloured Gaussian noise) signal is a random process where the magnitude response is not flattened in frequency domain.

The CGN signal can be generated by passing the WGN signal through a filter. In this thesis, we use the 6-order Butterworth lowpass filter with the specific cutoff-frequency f_c to generate the CGN signal. An example of the magnitude frequency responses of both WGN signal and CGN signal with $f_c = 2$ kHz are shown in Figure 4-3. The magnitude of the frequency response of WGN signal is flattening, while CGN signal is not.

WGN signal and CGN signal are used as the reference signal in this thesis.

³ TEST – Test set of wave files

⁴ DR3 – Speaker dialect region number 3, which is North Midland region.

⁵ DR6 – Speaker dialect region number 6, which is New York City region.

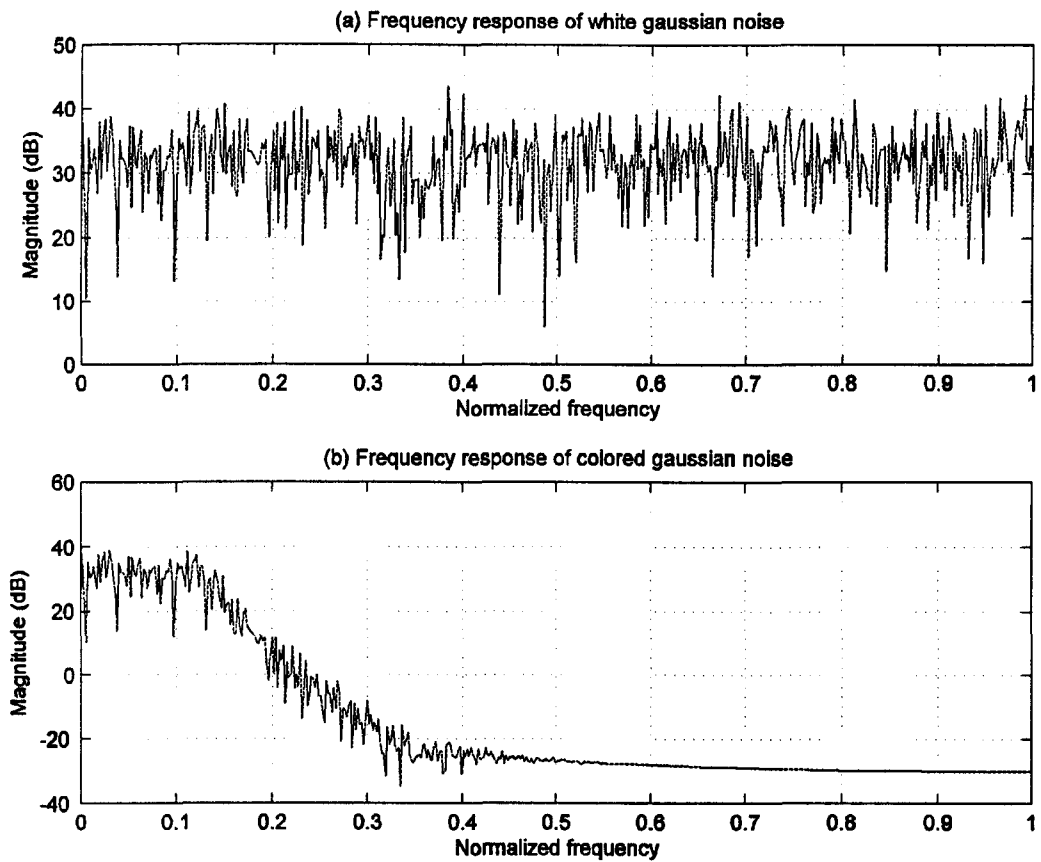


Figure 4-3: Frequency response of WGN and CGN

4.1.3.2 Noise-free Primary Signal

Four different noise-free primary signals, i.e. $s(m)$ shown in Figure 4-1, are used in the simulations. The description of those noise-free primary signals is illustrated in Table 4-3. #2, #3 and #4 are 3 different voice signals. This implies that the research in this thesis will be focused on the performance in speech applications.

4.1.3.3 Reference Signal

Three different reference signals, i.e. $x(m)$ shown in Figure 4-1, are used. The description of these reference signals is illustrated in Table 4-4. #2 is WGN signal and #3 is CGN signal, which implies that the research in this thesis will be focused on the performance evaluation under common noise environments.

Table 4-3: Description of the noise-free primary inputs used in the simulations

Noise-free primary input #	Description
1	two-frequency signal, as shown in (4.1.7), with $f_1=500\text{Hz}$, $f_2=2000\text{Hz}$
2	voice input, “si1573.wav” in Table 4-2
3	voice input, “si919.wav” in Table 4-2
4	voice input, “si2194.wav” in Table 4-2

Table 4-4: Description of the reference inputs used in the simulations

Reference input #	Description
1	one-frequency signal, as shown in (4.1.6), with $f_1=100\text{Hz}$
2	WGN signal, as illustrated in 4.1.3.1.3
3	CGN signal, as illustrated in 4.1.3.1.3, where $f_c=6000\text{Hz}$

4.1.3.4 Noisy Primary Signal

The noisy primary signal, i.e. $d(m)$ shown in Figure 4-1, is generated by passing the reference signal $x(m)$, as shown in Figure 4-1, through the simulated transmission channel, multiplying a factor that can control the input SNR level, then adding up to the noise-free primary signal $s(m)$.

The simulated noisy primary input, $d(m)$, is calculated by

$$d(m) = s(m) + \frac{h[x(m)]}{\lambda} \quad (4.1.8)$$

where $s(m)$ is the noise-free primary signal, $h[x(m)]$ is calculated using the impulse response illustrated in Table 4-1, and λ is a factor which can control the input SNR level, as defined by

$$\lambda = 10^{\frac{\text{SNR}_{\text{Input}}}{20}} \times \sqrt{\frac{\sum_{i=1}^M \{h[x(m)]\}^2}{\sum_{i=1}^M [s(m)]^2}} \quad (4.1.9)$$

where $\text{SNR}_{\text{Input}}$ is the input SNR level in decibel (dB) and M is the total number of samples in either reference signal or noise-free primary signal.

The noisy primary signal generated by (4.1.8) has the designated input SNR level.

4.1.4 Title

The title of the simulations is defined in the following format:

<Chan Type>-CH<Chan #>-O<NFP #>-N<Ref #>

where **<Chan Type>** is the type name of the transmission channel, **<Chan #>** is the channel index within the transmission channel type, the description of **<Chan Type>** and **<Chan #>** can be found in Table 4-1. **<NFP #>** is the index of the noise-free primary signal, which is illustrated in Table 4-3. **<Ref #>** is the index of the reference signal, which is illustrated in Table 4-4. For example, if the transmission channel is second-order nonlinear transmission channel, the channel index is 1, the simulation uses noise-free primary signal #2 and reference signal #3, the title of this simulation comes to **Second-Non-CH1-O2-N3**.

4.2 Method of Analysis

The simulation `Second-Non-CH1-O1-N1` is used as an example to illustrate the method of analysis on the simulation result in this thesis. This simulation uses the Second-order nonlinear transmission channel #1, as described in Table 4-1. The noise-free primary signal #1, as described in Table 4-3, is a two-frequency signal with $f_1 = 500\text{Hz}$ and $f_2 = 2\text{ kHz}$ under the sampling frequency $f_s = 16\text{ kHz}$. The reference signal #1, as described in Table 4-4, is a one-frequency signal, with $f_1 = 100\text{Hz}$ under the same sampling frequency.

Figure 4-4 shows the residue error results from both MCLS and SCLS when the input SNR level is 10dB. The upper part of Figure 4-4 shows the residue error result from MCLS, i.e. RE_m . The lower part of Figure 4-4 shows the residue error result from SCLS, i.e. RE_s . The residue error is calculated by subtracting the noise-free primary signal from the filtered output, as shown in (4.1.1) and (4.1.2).

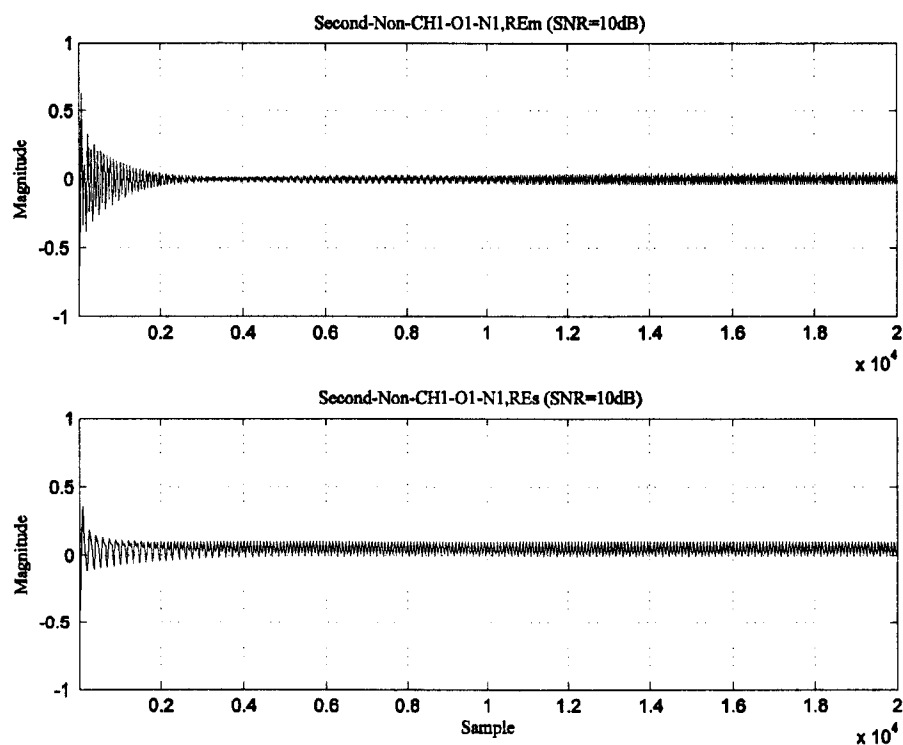


Figure 4-4: RE in `Second-Non-CH1-O1-N1` when $SNR_{input} = 10\text{dB}$

From the result in Figure 4-4, we can not distinguish the performance difference between MCLS and SCLS clearly. This is the reason on the introduction of MSE measurement, which is used to calculate the mean-square-error in a certain measurement period, as shown in (4.1.3). In the following simulations, the total number of samples in MSE measurements is selected to be all equal to 200. Figure 4-5 shows the mean-square-error (MSE) from MCLS and SCLS when the input SNR level is 10dB. The solid line shows the MSE result from MCLS and the dashed line shows the MSE result from SCLS.

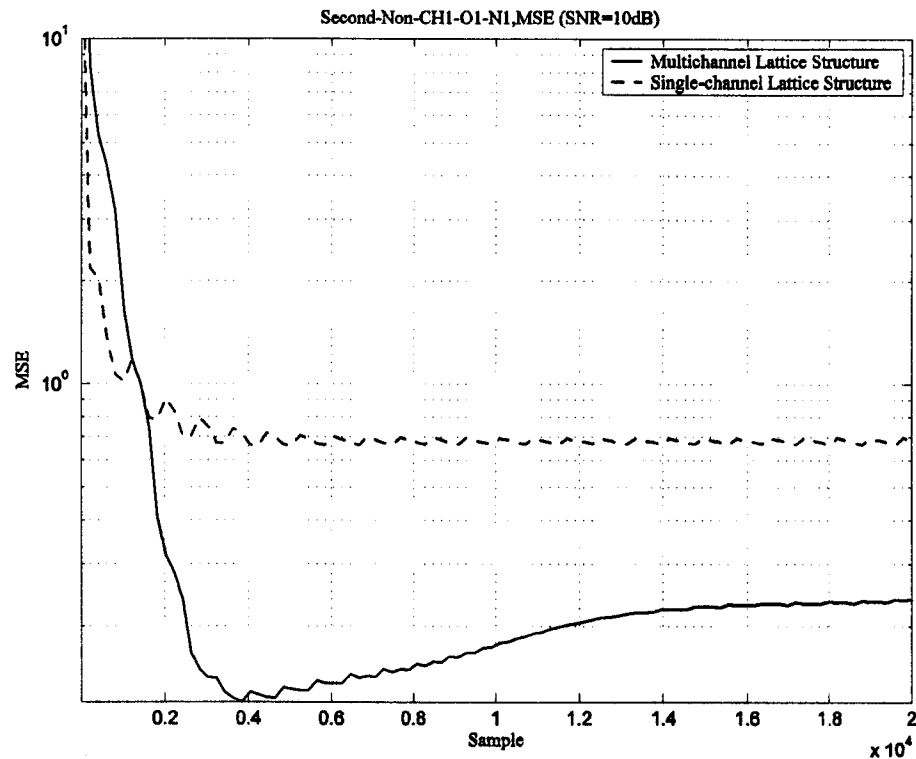


Figure 4-5: MSE in **Second-Non-CH1-O1-N1** when $\text{SNR}_{\text{input}} = 10\text{dB}$

From the result in Figure 4-5, it can be distinguished clearly that the MSE result from MCLS is lower than the MSE result from SCLS, which means that the performance of MCLS is better than SCLS when the transmission channel has the second-order nonlinearity (Second-Non-CH1) and the input SNR level is 10dB.

Using RE and MSE measurements, the performance can be evaluated on a given input SNR level. When the input SNR level changes, there will be a lot of plotting results of RE and MSE measurements.

To evaluate and compare the general performance between MCLS and SCLS, input/output SNR measurement, as defined in (4.1.4) and (4.1.5), is used to evaluate the average output SNR level upon different input SNR level. To get the output SNR in more stable states, the output SNR level is calculated from the later 80% of the samples. For example, if there are 10000 samples in total, the output SNR level will be calculated from the samples in the range of 2001-10000. This will avoid the unstable initial state of both MCLS and SCLS, and will get a more accurate result.

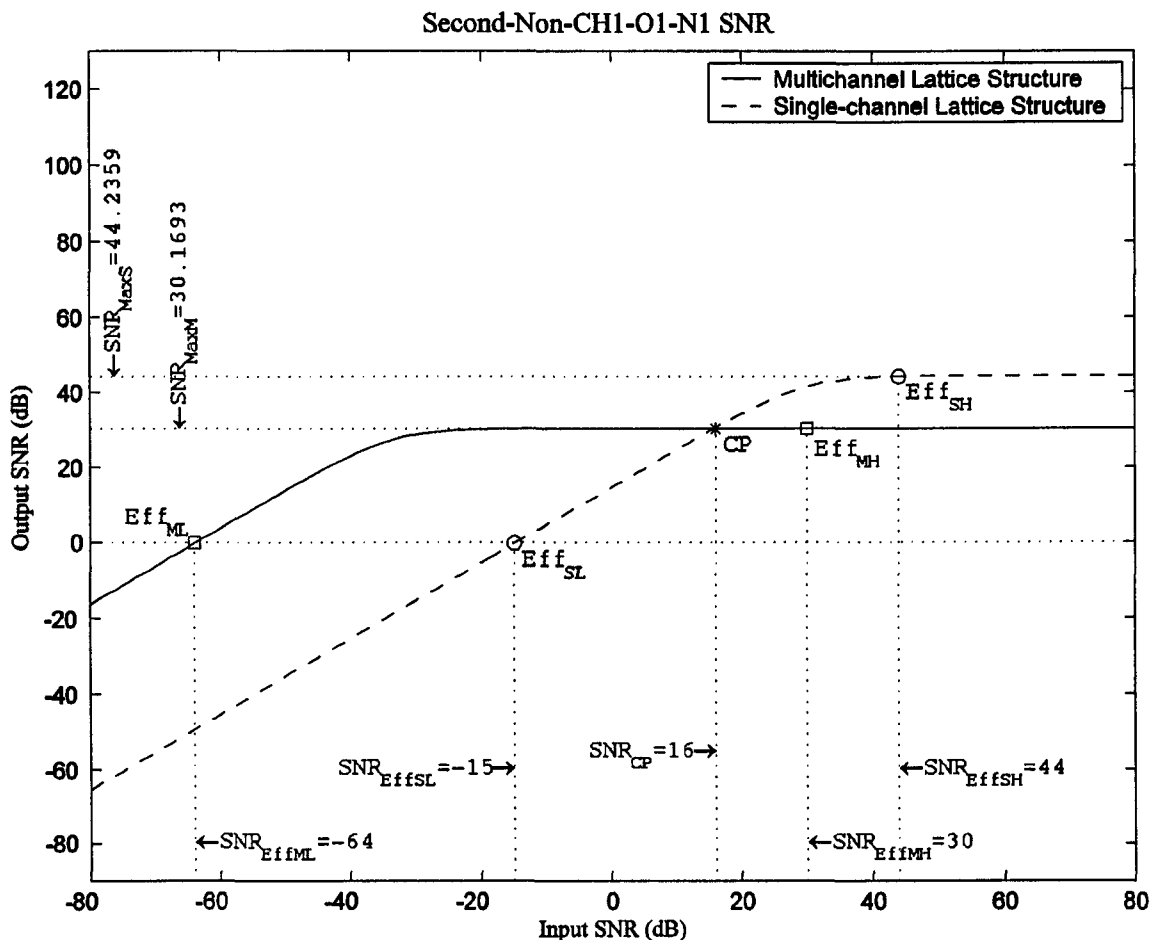


Figure 4-6: Input/Output SNR in Second-Non-CH1-O1-N1

Figure 4-6 shows the average output SNR performance of MCLS and SCLS when the input SNR changes from -80dB to 80dB. The solid line shows the result from MCLS and the dashed line shows the result from SCLS.

The effective range of input SNR level is defined for applicability reason, where $SNR_{output} > SNR_{input}$. In the input/output SNR plotting, as shown in Figure 4-6, the low-end input SNR level of effective range is where the output SNR level is equal to 0dB⁶. The high-end input SNR level of effective range is the highest input SNR level within the effective range. The concept of effective range is very useful to evaluate and compare the performance between MCLS and SCLS.

In Figure 4-6, two squared points, labelled Eff_{ML} and Eff_{MH} , are the low-end and high-end of the effective range on MCLS. Two circled points, labelled Eff_{SL} and Eff_{SH} , are the low-end and high-end of the effective range on SCLS.

One asterisked point, labelled CP , is the crossover point where the input SNR levels of both MCLS and SCLS are the same. Meanwhile, the output SNR level of both MCLS and SCLS are the same as well.

The dotted lines point to the different input or output SNR levels with the specific corresponding names. The description of the points and their corresponding input/output SNR levels in Figure 4-6 is illustrated in Table 4-5.

From the description in Table 4-5, if the effective range does not exist, the given structure can not provide the functionality of noise cancellation. If CP does not exist, SCLS is always better than MCLS. If CP exists, when the $SNR_{input} > SNR_{CP}$, the performance of SCLS is better than MCLS. When the $SNR_{input} < SNR_{CP}$, the performance of MCLS is better than SCLS.

⁶ In speech applications, when the output SNR is equal to 0dB, the noise-free primary signal can not be detected by the human ear. It means that the result is useless.

If $SNR_{CP} > SNR_{EffML}$, MCLS has the performance improvement on noise cancellation comparing with SCLS. Otherwise, MCLS has no improvement over SCLS.

Table 4-5: Description of the points and input/output SNR levels in Figure 4-6

Point Label	Input SNR level	Output SNR level	Description
CP	SNR_{CP}	Varied	Crossover Point. SNR_{CP} is the input SNR level at the point CP, where the outputs SNR levels from both MCLS and SCLS are the same.
Eff _{ML}	SNR_{EffML}	0dB	Low-end of effective range using MCLS, where the output SNR level is 0dB and the input SNR level is SNR_{EffML} .
Eff _{MH}	SNR_{EffMH}	SNR_{MaxM}	High-end of effective range using MCLS, where the input SNR level is SNR_{EffMH} . The output SNR level at this point is SNR_{MaxM} , called the maximum effective output SNR level using MCLS.
Eff _{SL}	SNR_{EffSL}	0dB	Low-end of effective range using SCLS, where the output SNR level is 0dB and the input SNR level is SNR_{EffSL} .
Eff _{SH}	SNR_{EffSH}	SNR_{MaxS}	High-end of effective range using SCLS, where the input SNR level is SNR_{EffSH} . The output SNR level at this point is SNR_{MaxM} , called the maximum effective output SNR level using SCLS.

If $SNR_{EffML} < SNR_{CP} < SNR_{EffMH}$, the performance of MCLS is better than the SCLS when the input SNR in the range from SNR_{EffML} to SNR_{CP} . If $SNR_{CP} > SNR_{EffMH}$, the range will be from SNR_{EffML} to SNR_{EffMH} . If $SNR_{CP} < SNR_{EffML}$, the range is not existed. This range is also called effective SNR improvement range, which shows the effective improvement of MCLS comparing with SCLS.

The weighted average of the output SNR difference within the effective improvement range between MCLS and SCLS is called average SNR improvement (SNR_{Imp}), as defined by

$$\text{SNR}_{\text{Imp}} = \begin{cases} 0 & , \text{ if } \text{SNR}_{\text{CP}} < \text{SNR}_{\text{EffML}} \\ \frac{1}{M} \sum_{\text{SNR}_{\text{input}} = \text{SNR}_{\text{EffML}}}^{\text{SNR}_{\text{input}} = \text{SNR}_{\text{CP}}} (\text{SNR}_{\text{Moutput}} - \text{SNR}_{\text{Soutput}}) & , \text{ if } \text{SNR}_{\text{EffML}} \leq \text{SNR}_{\text{CP}} \leq \text{SNR}_{\text{EffMH}} \\ \frac{1}{M} \sum_{\text{SNR}_{\text{input}} = \text{SNR}_{\text{EffML}}}^{\text{SNR}_{\text{input}} = \text{SNR}_{\text{EffMH}}} (\text{SNR}_{\text{Moutput}} - \text{SNR}_{\text{Soutput}}) & , \text{ if } \text{SNR}_{\text{CP}} > \text{SNR}_{\text{EffMH}} \end{cases} \quad (4.2.1)$$

where M is the number of points within the effective improvement range, $\text{SNR}_{\text{Moutput}}$ is the output SNR level from MCLS at the given input SNR level, $\text{SNR}_{\text{Soutput}}$ is the output SNR level from SCLS at the given input SNR level. If the effective improvement range exists, the input SNR level will stay in the effective range.

SNR_{MaxS} , defined in Table 4-5, shows the maximum output SNR level that SCLS can provide in the corresponding simulation. SNR_{MaxM} , defined in Table 4-5, shows the maximum output SNR level that MCLS can provide respectively.

From the result in Figure 4-6, CP is at the point where the input SNR level of either MCLS or SCLS is 16dB. The effective range of MCLS is where the input SNR level stays between -64dB and 30dB. The effective range of SCLS is where the input SNR level stays between -15dB and 44dB.

The maximum output SNR level using SCLS is about 44dB, which is about 14dB higher than the one from MCLS. This is because MCLS includes the additional adaptation steps for the second-order nonlinear elements, which introduces higher output level of base noise. On the contrary, the existing SCLS only includes the linear elements with less adaptation steps, resulting lower output level of base noise.

The effective improvement range of MCLS over SCLS is where the input SNR level stays between -64dB and 16dB. When the input SNR level stays in this range, MCLS has performance improvement comparing with SCLS.

In this example, two-frequency signal is used as the noise-free primary signal and one-frequency signal is used as the reference signal. The transmission channel uses second-order nonlinear channel #1. The result is analyzed step-by-step. In the remaining simulations, and the typical simulations⁷ are discussed in detail to analyze and compare the performance between MCLS and SCLS over different scenarios.

4.3 Simulations and Analysis

In this thesis, the simulations are grouped upon different type of transmission channels to verify whether MCLS has better performance in ANC comparing with SCLS over different scenarios.

4.3.1 Second-order Nonlinear Transmission Channel

The main focus of this thesis is to verify whether MCLS, which is based on second-order Volterra series expansion, can provide better performance than SCLS when the transmission channel has second-order nonlinearity. Extensive simulations will be carried out to provide the evidences with positive results.

4.3.1.1 Simulation Scenarios

There are totally 60 simulations carried out over 4 different second-order nonlinear transmission channels, 4 different noise-free primary signals and 3 different reference signals as described above.

Channel #1 and #2, as illustrated in Table 4-1, are used to provide the evidences that MCLS has the performance improvement over SCLS when the transmission channel only contains the basic second-order nonlinear elements⁸.

⁷ Typical simulation: In this thesis, the simulations with the voice signal as noise-free primary signal and WGN signal as reference signal are selected as the typical scenario of the simulations. Detailed analysis will be given in those simulations.

⁸ Basic second-order nonlinear elements are the elements contained in the output of SIMO module (3.2.4).

There are 30% output of the impulse response with second-order nonlinearity in channel #1, and 90% in channel #2 respectively. Simulation results show whether the different proportion of nonlinear elements will affect the performance of both MCLS and SLCS. 12 simulations (No. 1-12), as illustrated in Table 4-6, are carried out over channel #1. Another 12 simulations (No. 13-24), as illustrated in Table 4-7, are carried out over channel #2. L is the maximum number of stages of both MCLS and SCLS.

Table 4-6: Simulation titles using second-order nonlinear channel #1

No.	Simulation Title	L	No.	Simulation Title	L
1	Second-Non-CH1-O1-N1	4	7	Second-Non-CH1-O3-N1	4
2	Second-Non-CH1-O1-N2	4	8	Second-Non-CH1-O3-N2	4
3	Second-Non-CH1-O1-N3	4	9	Second-Non-CH1-O3-N3	4
4	Second-Non-CH1-O2-N1	4	10	Second-Non-CH1-O4-N1	4
5	Second-Non-CH1-O2-N2	4	11	Second-Non-CH1-O4-N2	4
6	Second-Non-CH1-O2-N3	4	12	Second-Non-CH1-O4-N3	4

Table 4-7: Simulation titles using Second-order nonlinear channel #2

No.	Index Text	L	No.	Index Text	L
13	Second-Non-CH2-O1-N1	4	19	Second-Non-CH2-O3-N1	4
14	Second-Non-CH2-O1-N2	4	20	Second-Non-CH2-O3-N2	4
15	Second-Non-CH2-O1-N3	4	21	Second-Non-CH2-O3-N3	4
16	Second-Non-CH2-O2-N1	4	22	Second-Non-CH2-O4-N1	4
17	Second-Non-CH2-O2-N2	4	23	Second-Non-CH2-O4-N2	4
18	Second-Non-CH2-O2-N3	4	24	Second-Non-CH2-O4-N3	4

Channel #3, as illustrated in Table 4-1, is used to provide the evidence that MCLS has the performance improvement over SCLS when the transmission channel not only includes the basic second-order nonlinear elements, but also includes the non-basic second-order nonlinear elements⁹. 12 simulations (No. 25-36), as illustrated in Table 4-8, are carried out over channel #3. L is the maximum number of stages of both MCLS and SCLS.

Table 4-8: Simulation titles using Second-order nonlinear channel #3

No.	Index Text	L	No.	Index Text	L
25	Second-Non-CH3-O1-N1	4	31	Second-Non-CH3-O3-N1	4
26	Second-Non-CH3-O1-N2	4	32	Second-Non-CH3-O3-N2	4
27	Second-Non-CH3-O1-N3	4	33	Second-Non-CH3-O3-N3	4
28	Second-Non-CH3-O2-N1	4	34	Second-Non-CH3-O4-N1	4
29	Second-Non-CH3-O2-N2	4	35	Second-Non-CH3-O4-N2	4
30	Second-Non-CH3-O2-N3	4	36	Second-Non-CH3-O4-N3	4

Channel #4, as illustrated in Table 4-1, is used to provide the evidence that even the absolute maximum delay¹⁰ exceeds the second-order Volterra series expansion, but the relative maximum delay¹¹ stays inside the limit of SIMO module output. In this case, by increasing the number of MALP and MMRTF stages, the performance improvement can be achieved between MCLS and SCLS.

12 simulations (No. 37-48), as illustrated in Table 4-9, are carried out over Channel #4 using $L = 4$. Another 12 simulations (No. 49-60), as illustrated in Table 4-9, are carried out over Channel #4 using $L = 13$.

⁹ Non-basic second-order nonlinear elements are the elements contained in the second-order Volterra series expansion (3.2.2), but not included in output of SIMO module (3.2.4).

¹⁰ The maximum delay within the impulse response of the transfer function, for example, $x(n-11)x(n-12)$ comes to 12 at Second-Non channel #4 in Table 4-1.

¹¹ The maximum difference of the delay within each nonlinear element, for example, $x(n-11)x(n-12)$ comes to 1 at Second-Non channel #4 in Table 4-1.

Table 4-9: Simulation titles using Second-order nonlinear channel #4

No.	Index Text	L	No.	Index Text	L
37	Second-Non-CH4-O1-N1-L4	4	49	Second-Non-CH4-O1-N1-L13	13
38	Second-Non-CH4-O1-N2-L4	4	50	Second-Non-CH4-O1-N2-L13	13
39	Second-Non-CH4-O1-N3-L4	4	51	Second-Non-CH4-O1-N3-L13	13
40	Second-Non-CH4-O2-N1-L4	4	52	Second-Non-CH4-O2-N1-L13	13
41	Second-Non-CH4-O2-N2-L4	4	53	Second-Non-CH4-O2-N2-L13	13
42	Second-Non-CH4-O2-N3-L4	4	54	Second-Non-CH4-O2-N3-L13	13
43	Second-Non-CH4-O3-N1-L4	4	55	Second-Non-CH4-O3-N1-L13	13
44	Second-Non-CH4-O3-N2-L4	4	56	Second-Non-CH4-O3-N2-L13	13
45	Second-Non-CH4-O3-N3-L4	4	57	Second-Non-CH4-O3-N3-L13	13
46	Second-Non-CH4-O4-N1-L4	4	58	Second-Non-CH4-O4-N1-L13	13
47	Second-Non-CH4-O4-N2-L4	4	59	Second-Non-CH4-O4-N2-L13	13
48	Second-Non-CH4-O4-N3-L4	4	60	Second-Non-CH4-O4-N3-L13	13

In all these 60 simulations, as described in Table 4-6, Table 4-7, Table 4-8 and Table 4-9, some other parameters in both MCLS and SCLS are set as the follows:

- Maximum delay of the SIMO module: $M - 1 = 2$.
- Smoothing factor: α , as defined in (3.2.14) and (3.2.23), is set to be 0.00075.
- Step size factor: μ , as defined (3.2.13) and (3.2.22), is set to be 0.00075.

4.3.1.2 Analysis over Channel #1

The result of simulation No. 5, i.e. **Second-Non-CH1-O2-N2** in Table 4-6, is analyzed here. It uses the second-order nonlinear transmission channel #1 in Table 4-1, the noise-free primary signal #2 in Table 4-3 and the reference signal #2 in Table 4-4.

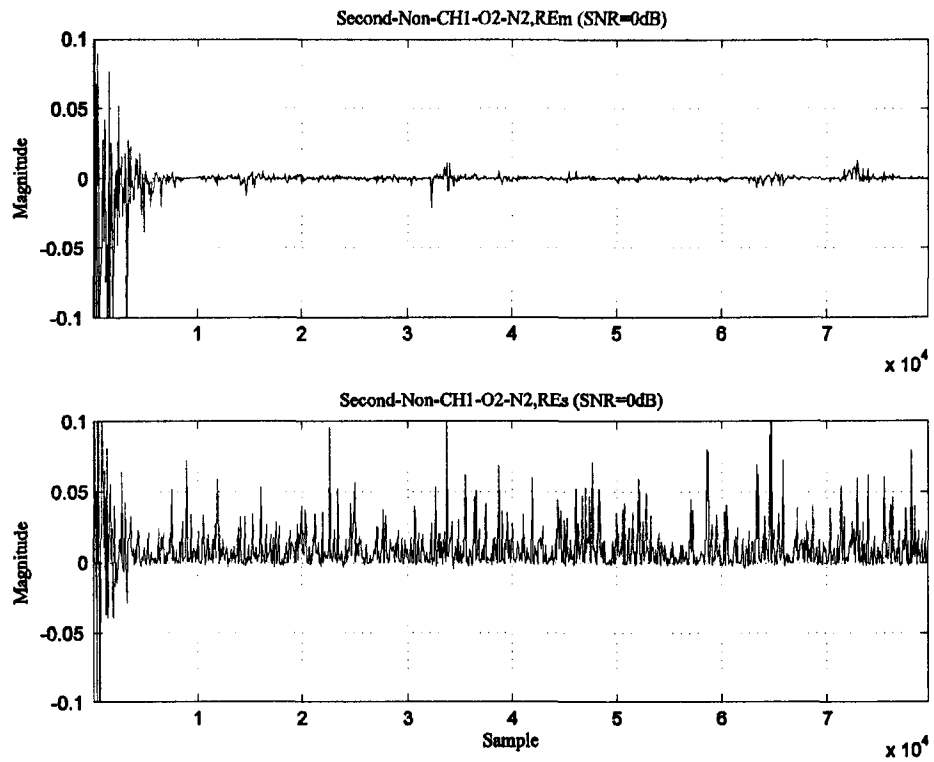


Figure 4-7: RE in Second-Non-CH1-O2-N2 when input $\text{SNR}_{\text{input}} = 0\text{dB}$

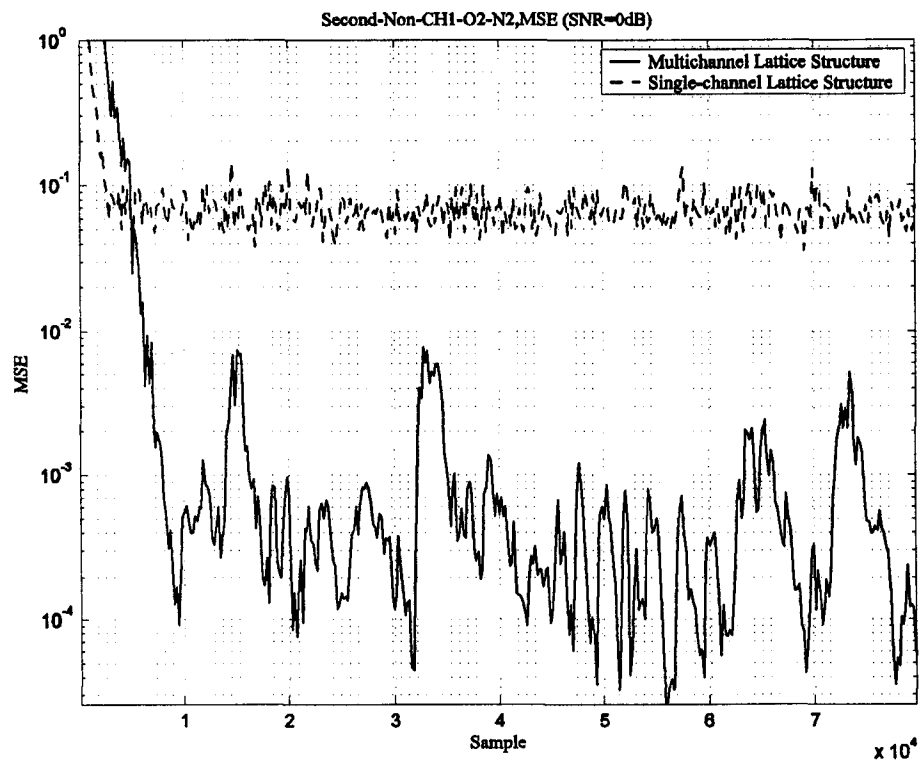


Figure 4-8: MSE in Second-Non-CH1-O2-N2 when $\text{SNR}_{\text{input}} = 0\text{dB}$

Figure 4-7 shows the RE results from both MCLS and SCLS when the input SNR level is 0dB. The RE from MCLS (RE_m) is smaller than the RE from SCLS (RE_s).

Figure 4-8 shows the MSE results from both MCLS and SCLS under 0dB input SNR level. The MSE from MCLS is lower than the MSE from the SCLS.

The result of RE/MSE shows that MCLS has improved ANC ability comparing with SCLS when the input SNR level is 0dB.

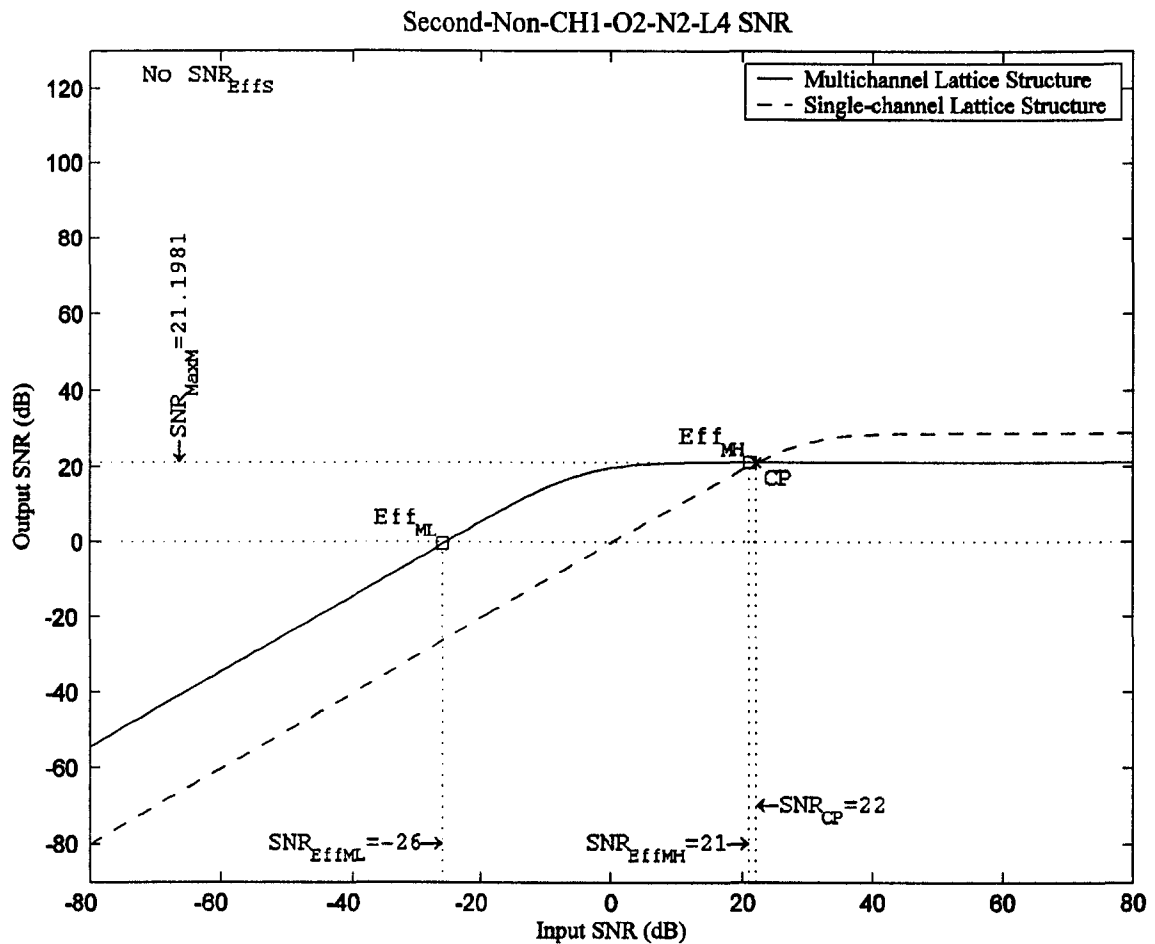


Figure 4-9: Input/Output SNR in **Second-Non-CH1-O2-N2**

Figure 4-9 shows the output SNR levels from both MCLS and SCLS when the input SNR levels ranges from -80dB to 80dB. There is no effective range when using SCLS. This means that SCLS can not provide ANC functionality in this simulation. On the

contrary, when the input SNR level is from -26dB to 21dB, MCLS has the ability to cancel the noise adaptively.

The detailed results of all 12 simulations using second-order nonlinear channel #1 can be found in Table 4-10.

Table 4-10: Simulation results using Second-Non channel #1

No	SNR _{EffS} (dB)			SNR _{EffM} (dB)			SNR _{CP} (dB)	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
1	-15	44	44.236	-64	30	30.169	16	-64	16	33.234
2	-33	27	27.223	-58	23	22.693	-9	-58	-9	17.200
3	-41	25	25.129	-55	21	21.162	-17	-55	-17	9.943
4	N/A	N/A	N/A	-35	20	20.296	41	-35	20	43.466
5	N/A	N/A	N/A	-26	21	21.198	22	-26	21	18.146
6	-7	25	25.293	-23	21	20.774	15	-23	15	11.294
7	N/A	N/A	N/A	-39	20	19.565	37	-39	20	41.440
8	-5	23	23.031	-31	21	21.232	19	-31	19	17.199
9	-13	25	24.678	-28	21	20.814	10	-28	10	11.177
10	N/A	N/A	N/A	-37	23	23.449	42	-37	23	42.851
11	-3	24	24.152	-29	22	21.870	20	-29	20	17.619
12	-11	23	22.857	-27	20	20.086	12	-27	12	10.833

The first column is the simulation index, which is defined in Table 4-6. The SNR_{EffS} represents the effective range using SCLS, where **L** shows the low-end input SNR level, **H** shows the high-end input SNR level, and **Max** shows the maximum output SNR level. The SNR_{EffM} represents the effective range using MCLS, where **L** shows the low-end input SNR level, **H** shows the high-end input SNR level, and **Max** shows the maximum output SNR level. The SNR_{CP} column shows the input SNR level on the crossover point. The SNR_{Imp} represents the effective improvement range when MCLS has performance

improvement over SCLS, where **L** shows the low-end input SNR level, **H** shows the high-end input SNR level, and IMP_{Avg} shows the average SNR improvement over the effective improvement range. IMP_{Avg} can be calculated using (4.2.1). N/A represents not available.

From the results in Table 4-10, the average SNR improvement of the simulation No. 5 is about 18.146dB when the input SNR level ranges from -26dB to 21dB. All the simulations using Second-order nonlinear channel #1 has the effective improvement range. SCLS can not provide ANC functionality in some simulations, such as No. 4, 5, 7 and 10 in Table 4-5, where MCLS can provide.

The comparison of SNR improvements over second-order nonlinear channel #1 using different noise-free primary signals and reference signals is shown in Figure 4-10.

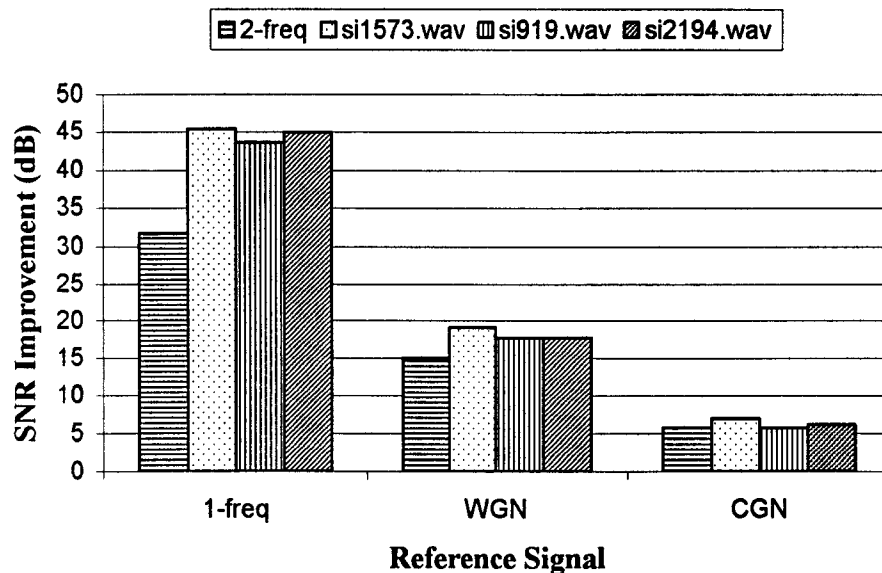


Figure 4-10: SNR Improvement over Second-Non channel #1

When the reference signal is the one-frequency signal, MCLS gets the best average SNR improvement, which is around 37dB. When the reference signal is WGN signal, the average SNR improvement is around 17dB. When the reference signal is CGN signal, the average SNR improvement is around 6dB.

From the results in Figure 4-10, the values of average SNR improvements are grouped according to different reference signals. The values of average SNR improvements are close under the same reference signal.

The comparison of maximum effective output SNR between MCLS and SCLS over second-order nonlinear channel #1 is shown in Figure 4-11. The value in the table is $SNR_{MaxDesc}$, which is defined by

$$SNR_{MaxDesc} = \begin{cases} 0 & , \text{ if } SNR_{MaxS} \text{ not available} \\ 0 & , \text{ if } SNR_{MaxS} < SNR_{MaxM} \\ SNR_{MaxS} - SNR_{MaxM} & , \text{ otherwise} \end{cases} \quad (4.3.1)$$

where SNR_{MaxS} and SNR_{MaxM} are defined in Table 4-5. $SNR_{MaxDesc}$ shows the maximum effective output SNR decrease by comparing MCLS with SCLS.

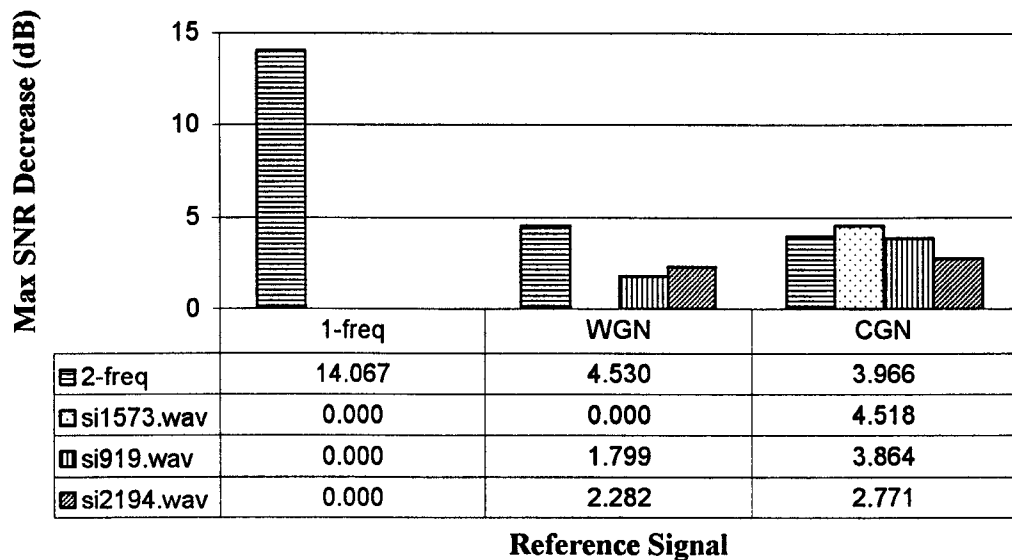


Figure 4-11: Max output SNR Decrease over Second-Non channel #1

When the noise-free primary signal is two-frequency signal and the reference signal is one-frequency signal, the $SNR_{MaxDesc}$ can be about 14dB. In other noise-free primary signal and reference signal combinations, the $SNR_{MaxDesc}$ is below 5dB.

4.3.1.3 Analysis over Channel #2

The result of simulation No. 17, i.e. **Second-Non-CH2-O2-N2** in Table 4-7, is analyzed here. It uses the second-order nonlinear transmission channel #2 in Table 4-1, the noise-free primary signal #2 in Table 4-3 and the reference signal #2 in Table 4-4.

Figure 4-12 shows the RE results from both MCLS and SCLS when the input SNR level is 10dB. Apparently, the RE from MCLS (RE_m) is smaller than the RE from SCLS (RE_s).

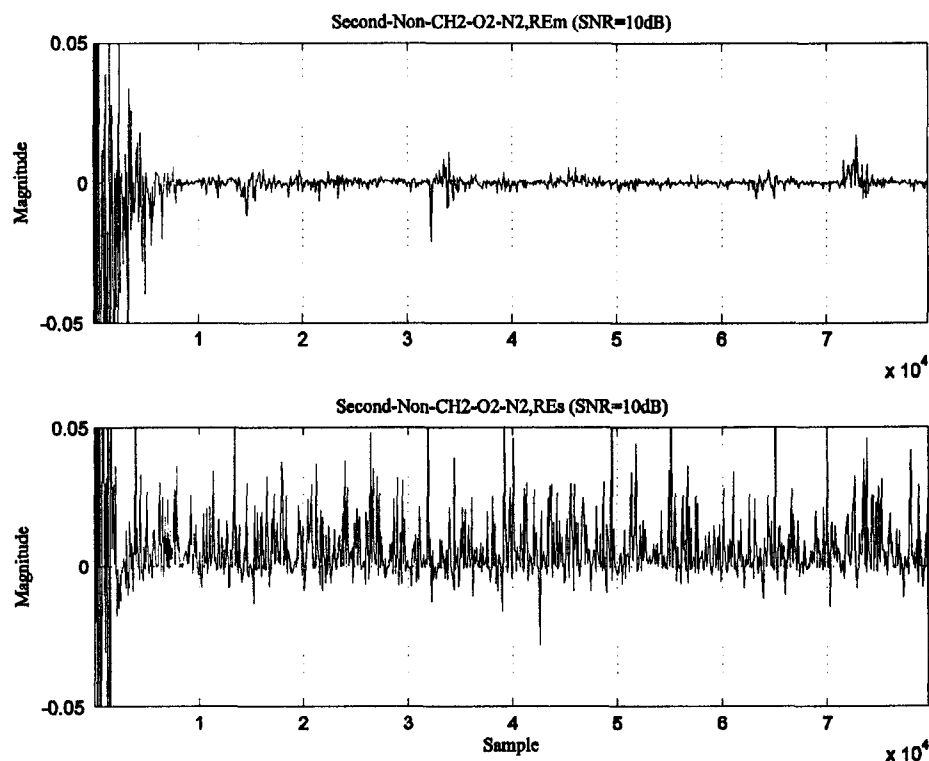


Figure 4-12: RE in **Second-Non-CH2-O2-N2** when $SNR_{input} = 10dB$

Figure 4-13 shows the MSE results from both MCLS and SCLS under the same input SNR level. The MSE from MCLS is lower than the MSE from SCLS.

The result of RE/MSE shows that MCLS has improved ANC ability comparing with SCLS when the input SNR level is 10dB.

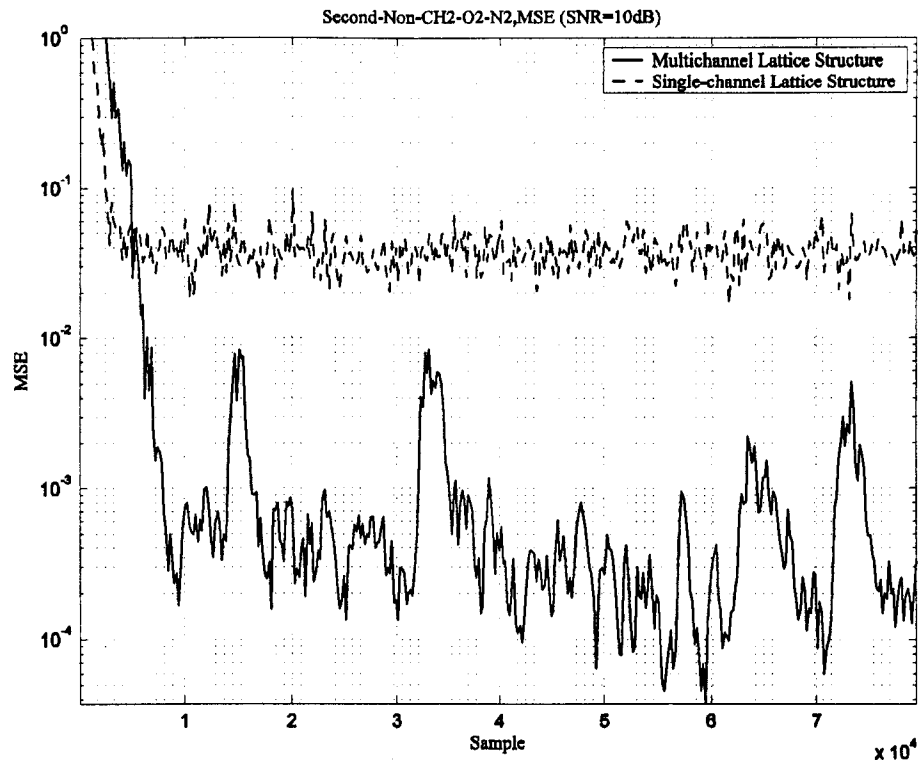


Figure 4-13: MSE in Second-Non-CH2-O2-N2 when $SNR_{input} = 10dB$

Figure 4-14 shows the output SNR levels from both MCLS and SCLS when the input SNR levels ranging from -80dB to 80dB. There is no effective range using SCLS. This means that SCLS can not provide ANC functionality in this simulation. On the contrary, when the input SNR level is from -16dB to 21dB, MCLS has the ability to cancel the noise adaptively.

The detailed results of all 12 simulations using Second-order nonlinear channel #2 can be found in Table 4-11, where the simulation indices are defined in Table 4-7.

The average SNR improvement of the simulation No. 17, as shown in Table 4-11, is about 19dB when the input SNR level ranges from -16dB to 21dB. All the simulations using Second-order nonlinear channel #2 has the effective improvement range. SCLS can not provide ANC functionality in some simulations, such as No. 16, 17, 18, 19, 20, 22, and 23 shown in Table 4-11, where MCLS can provide.

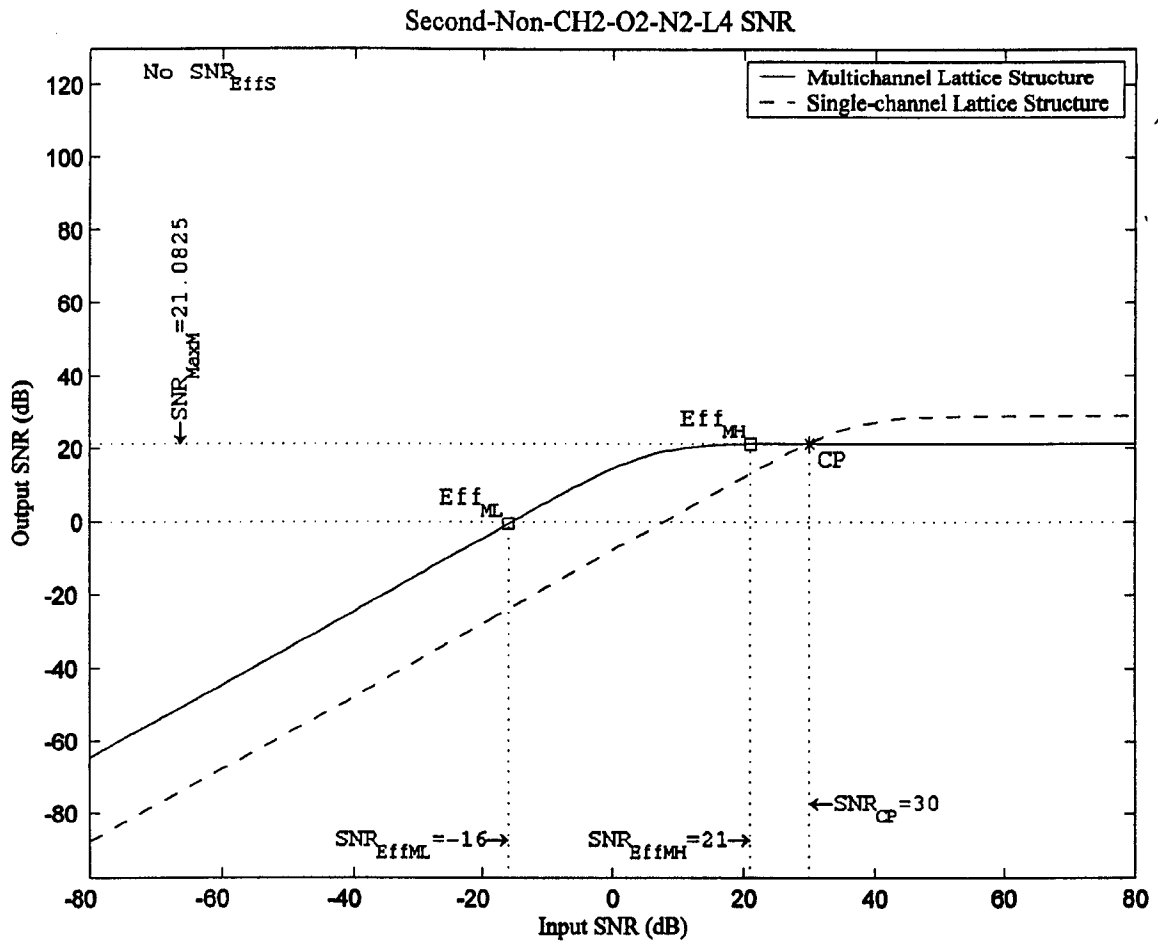


Figure 4-14: Input/Output SNR in Second-Non-CH2-O2-N2

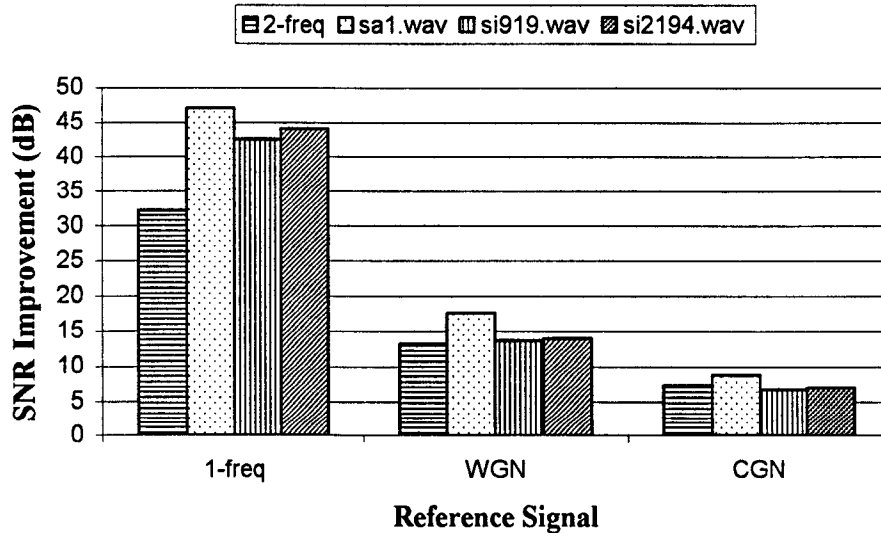


Figure 4-15: SNR Improvement over Second-Non channel #2

The comparison of SNR improvements over second-order nonlinear transmission channel #2 is shown in Figure 4-15.

Table 4-11: Simulation results using Second-Non channel #2

No	SNR _{EffS} (dB)			SNR _{EffM} (dB)			SNR _{CP} (dB)	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
13	-5	43	42.879	-51	30	30.170	25	-51	30	31.685
14	-26	27	27.220	-47	23	22.695	-1	-47	23	15.087
15	-30	25	25.119	-38	21	21.164	-7	-38	21	5.770
16	N/A	N/A	N/A	-22	20	20.273	50	-22	20	45.455
17	N/A	N/A	N/A	-16	21	21.083	30	-16	21	19.139
18	N/A	N/A	N/A	-4	20	19.670	25	-4	20	6.909
19	N/A	N/A	N/A	-26	20	19.557	47	-26	20	43.630
20	N/A	N/A	N/A	-21	21	21.231	26	-21	21	17.825
21	-2	20	20.130	-10	20	20.277	20	-10	20	5.768
22	N/A	N/A	N/A	-25	23	23.441	51	-25	23	45.014
23	N/A	N/A	N/A	-19	22	21.823	28	-19	22	17.801
24	0	10	9.995	-8	19	19.322	22	-8	19	6.135

When the reference signal is the one-frequency signal, MCLS gets the best average SNR improvement, which is around 40dB. When the reference signal is WGN signal, the SNR improvement is around 14dB. When the reference signal is CGN signal, the SNR improvement is around 7dB. From the results in Table 4-11, the values of average SNR improvement are grouped according to different reference signals. The values of average SNR improvement are close under the same reference signal.

The comparison of SNR_{MaxDesc} over second-order nonlinear channel #2 is shown in Figure 4-16. Only when the noise-free primary signal is two-frequency signal, the

$SNR_{MaxDesc}$ ranges from 4dB to 13dB. Using other noise-free primary signals, i.e. voice inputs, the $SNR_{MaxDesc}$ does not exist.

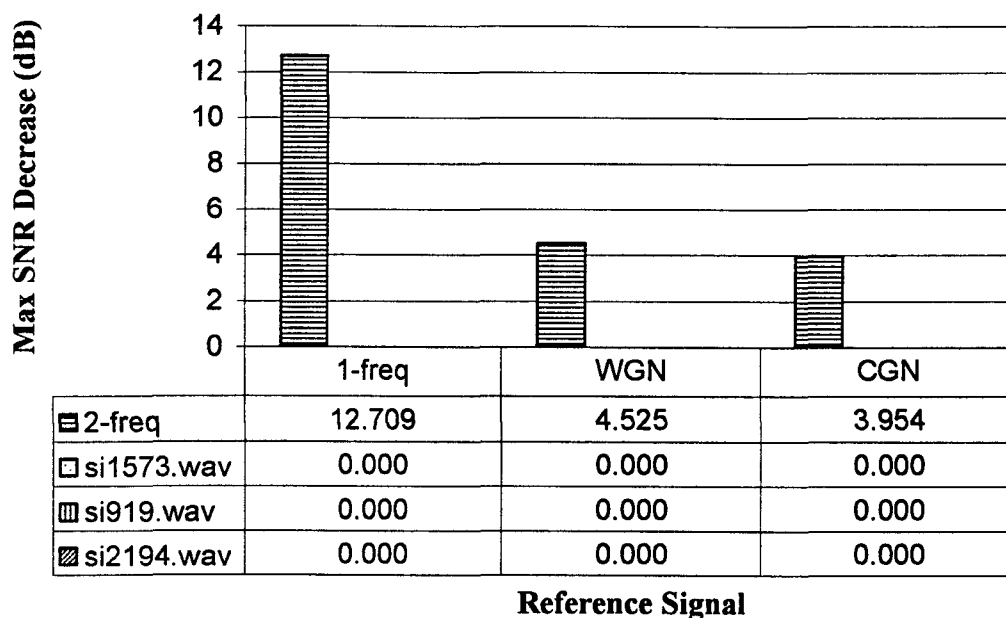


Figure 4-16: Max output SNR decrease over Second-Non channel #2

4.3.1.4 Analysis over Channel #3

The result of simulation No. 29, i.e. **Second-Non-CH3-O2-N2** in Figure 4-17 shows the output SNR levels from both MCLS and SCLS when the input SNR levels ranging from -80dB to 80dB. There is no effective range using SCLS. This means that the existing SCLS can not provide ANC functionality in this simulation. On the contrary, when the input SNR level ranging from -18dB to 21dB, MCLS has the ability to cancel the noise adaptively.

The detailed result of all 12 simulations using second-order nonlinear channel #3 is shown in Table 4-12, where the simulation indices are defined in Table 4-9.

The SNR improvement of the simulation No. 29, as shown in Table 4-12, is about 14.7dB when the input SNR level ranges from -18dB to 21dB.

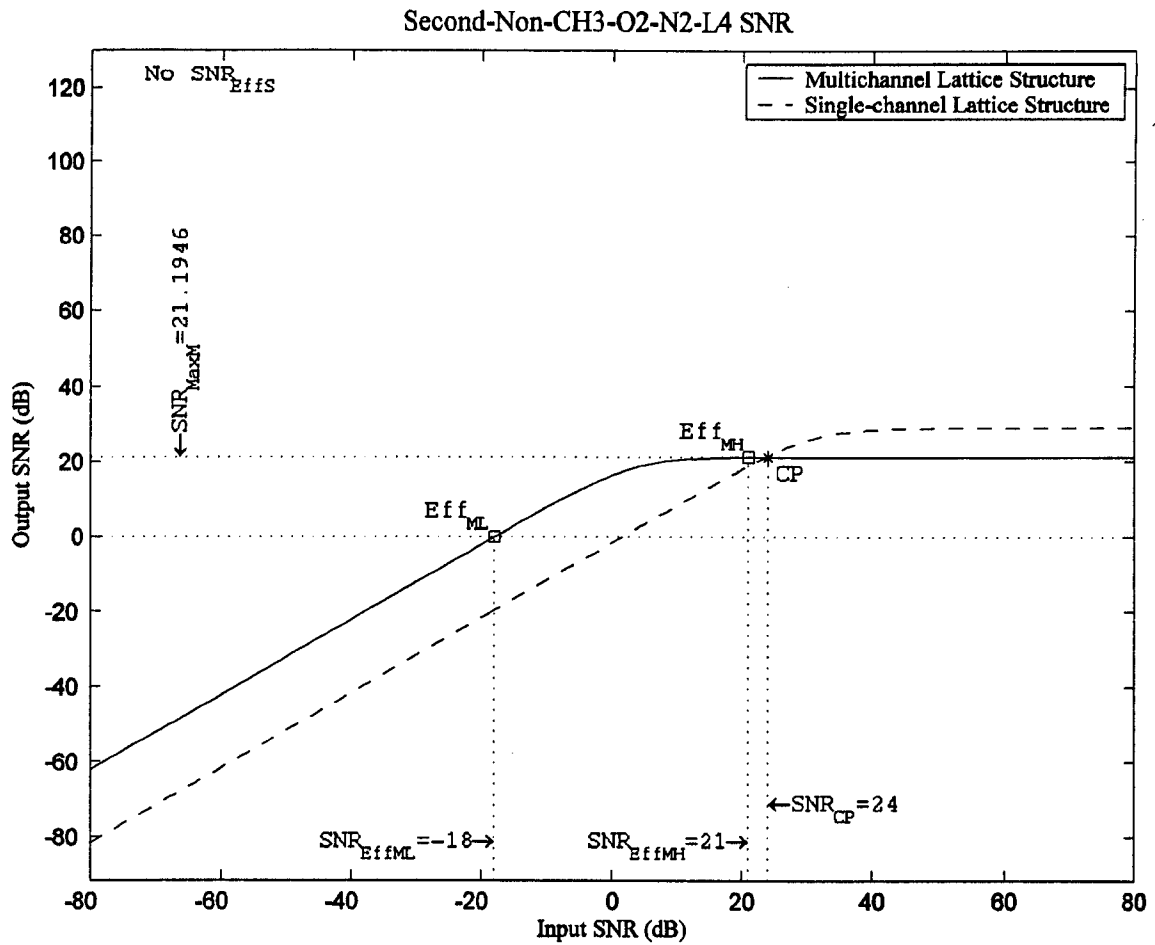


Figure 4-17: Input/output SNR in Second-Non-CH3-O2-N2

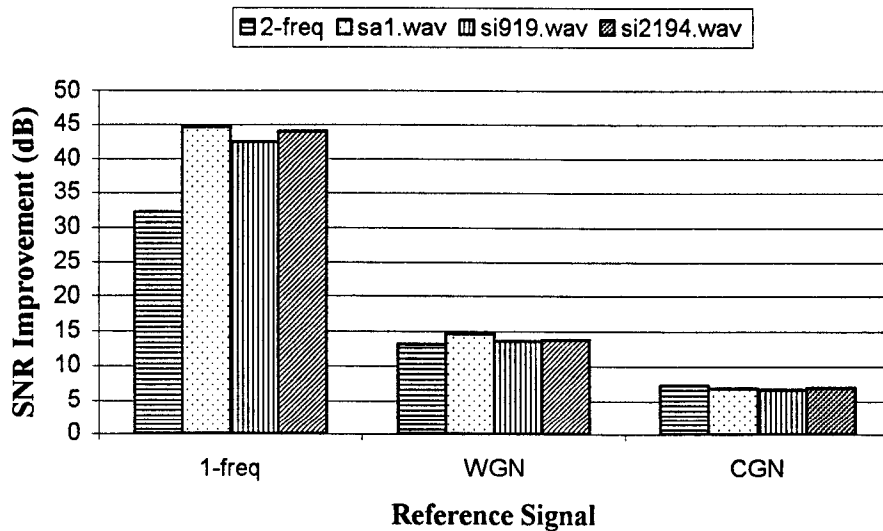


Figure 4-18: SNR improvement over Second-Non channel #3

Table 4-12: Simulation results using Second-Non channel #3

No	SNR _{EFS} (dB)			SNR _{EMM} (dB)			SNR _{CP} (dB)	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
25	-9	44	43.799	-56	30	30.170	22	-56	30	32.246
26	-32	27	27.226	-50	23	22.694	-8	-50	23	13.159
27	-35	25	25.125	-45	21	21.165	-12	-45	21	7.274
28	N/A	N/A	N/A	-27	20	20.290	47	-27	20	44.712
29	N/A	N/A	N/A	-18	21	21.195	24	-18	21	14.736
30	-1	21	20.998	-11	21	20.685	21	-11	21	6.818
31	N/A	N/A	N/A	-30	20	19.563	43	-30	20	42.614
32	-4	22	22.169	-23	21	21.245	20	-23	21	13.632
33	-7	24	23.930	-16	21	20.732	16	-16	21	6.650
34	N/A	N/A	N/A	-29	23	23.446	48	-29	23	44.097
35	-2	23	22.863	-22	22	21.897	22	-22	22	13.847
36	-5	22	21.725	-15	20	19.966	17	-15	20	6.944

All the simulations using Second-order nonlinear channel #3 has the effective improvement range. SCLS can not provide ANC functionality in some simulations, such as No 28, 29, 31 and 34 in Table 4-12, where MCLS can provide.

The comparison of average SNR improvements is shown in Figure 4-18.

When the reference input is one-frequency signal, MCLS gets the best average SNR improvement, which is around 40dB. When the reference signal is WGN signal, the average SNR improvement is around 14dB. When the reference signal is CGN signal, the SNR improvement is around 6dB. From the results in Figure 4-18, the average SNR improvements are grouped according to different reference signals. The values of average SNR improvements are close under the same reference signal.

The comparison of $SNR_{MaxDesc}$ over second-order nonlinear channel #3 is shown in Figure 4-19. When the noise-free primary signal is one-frequency signal and the reference input is two-frequency signal, the $SNR_{MaxDesc}$ is about 14dB. In other noise-free primary signal and reference signal combinations, the $SNR_{MaxDesc}$ is below 5dB.

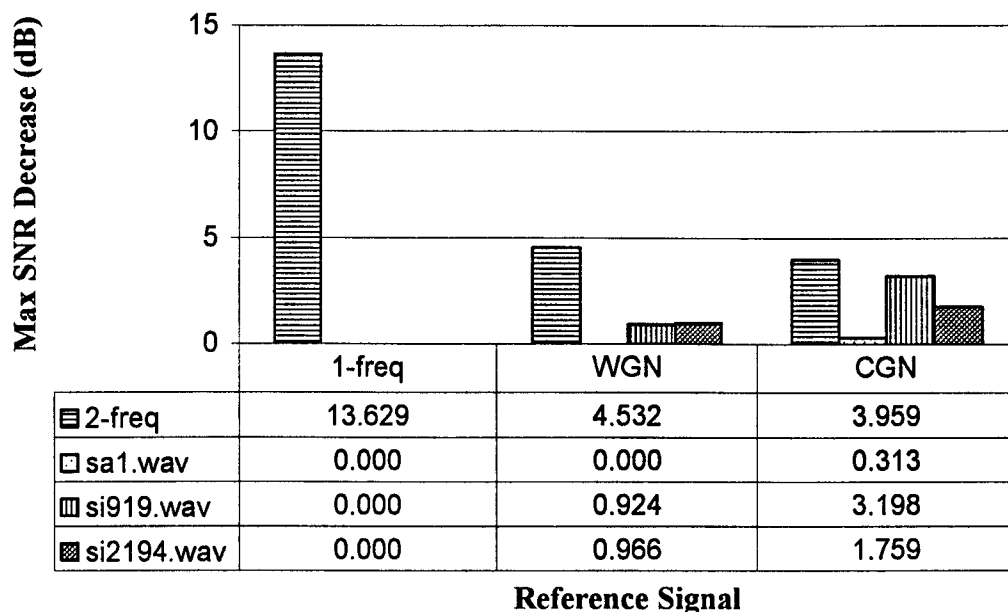


Figure 4-19: Max output SNR decrease over Second-Non channel #3

The simulations over second-order nonlinear channel #3 shows that even if the SIMO module only includes the basic elements within the second-order Volterra series expansion, as shown in (3.2.4), MCLS will provide the performance improvement over SCLS when the transmission channel includes the non-basic second-order nonlinear elements, as shown in Table 4-1.

4.3.1.5 Analysis over Channel #4

The result of simulation No. 41 and No. 53, i.e. **Second-Non-CH4-O2-N2-L4** and **Second-Non-CH4-O2-N2-L13** in Table 4-9, are analyzed here. They use the second-order nonlinear transmission channel #4 in Table 4-1, the noise-free primary

signal #2 in Table 4-3 and the reference signal #2 in Table 4-4. $L = 4$ in simulation No. 41 and $L = 13$ in simulation No. 53.

Figure 4-20 shows the output SNR levels from both MCLS and SCLS when the input SNR levels ranges from -80dB to 80dB when $L = 4$. Both MCLS and SCLS have no effective range, which means that they all do not have the ANC ability in this simulation.

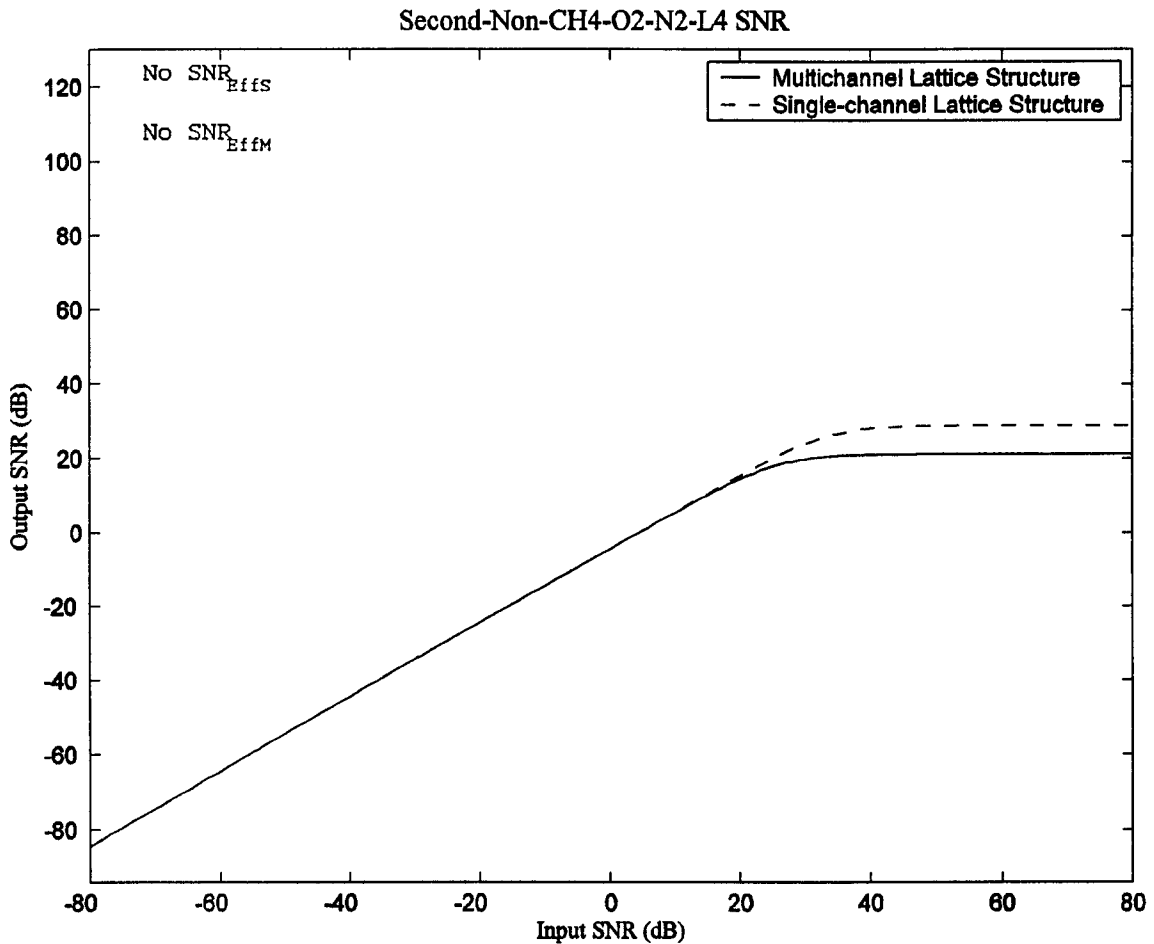


Figure 4-20: Input/output SNR in Second-Non-CH4-O2-N2-L4

The detailed result of all 12 simulations (No. 37-48), defined in Table 4-9, using Second-order nonlinear channel #4 with $L = 4$ is shown in Table 4-13. From the result in Table 4-13, most simulations either have no effective SNR improvement range or show very small average SNR improvement. This is because the absolute maximum delay within the transmission channel exceeds the limit of the SIMO module.

Table 4-13: Simulation results using Second-Non channel #4 with $L=4$

No	SNR _{EffS} (dB)			SNR _{EffM} (dB)			CP	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
37	-6	43	43.175	-30	30	30.163	24	-30	30	17.874
38	-29	27	27.215	-29	23	22.690	N/A	N/A	N/A	N/A
39	-34	25	25.128	-35	21	21.158	-20	-35	21	0.502
40	N/A	N/A	N/A	-7	19	19.318	49	-7	19	33.971
41	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
42	-1	19	19.031	-2	16	15.870	13	-2	13	0.408
43	N/A	N/A	N/A	-11	19	19.160	45	-11	19	32.815
44	-1	18	17.927	-1	14	14.047	N/A	N/A	N/A	N/A
45	-6	24	23.764	-7	20	19.867	8	-7	20	0.459
46	N/A	N/A	N/A	-10	23	23.053	50	-10	23	33.438
47	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
48	-5	21	21.247	-5	19	18.725	10	-5	19	0.461

As discussed in 4.3.1.1, the absolute maximum delay of channel #4 exceeds the limit of SIMO module. But the relative maximum delay does not. By increasing the number of MALP and MMRTF stages, the desired result can be achieved.

The simulation No. 53, **Second-Non-CH4-O2-N2-L13** in Table 4-9, is used to verify the result using $L = 13$.

Figure 4-21 shows the output SNR levels from both structures when the input SNR levels ranges from -80dB to 80dB when $L = 13$. There is no effective point from SCLS, which means that the existing SCLS does not have ANC ability in this simulation. On the contrary, when input SNR is from -14dB to 16dB, MCLS has the ability to cancel the noise adaptively.

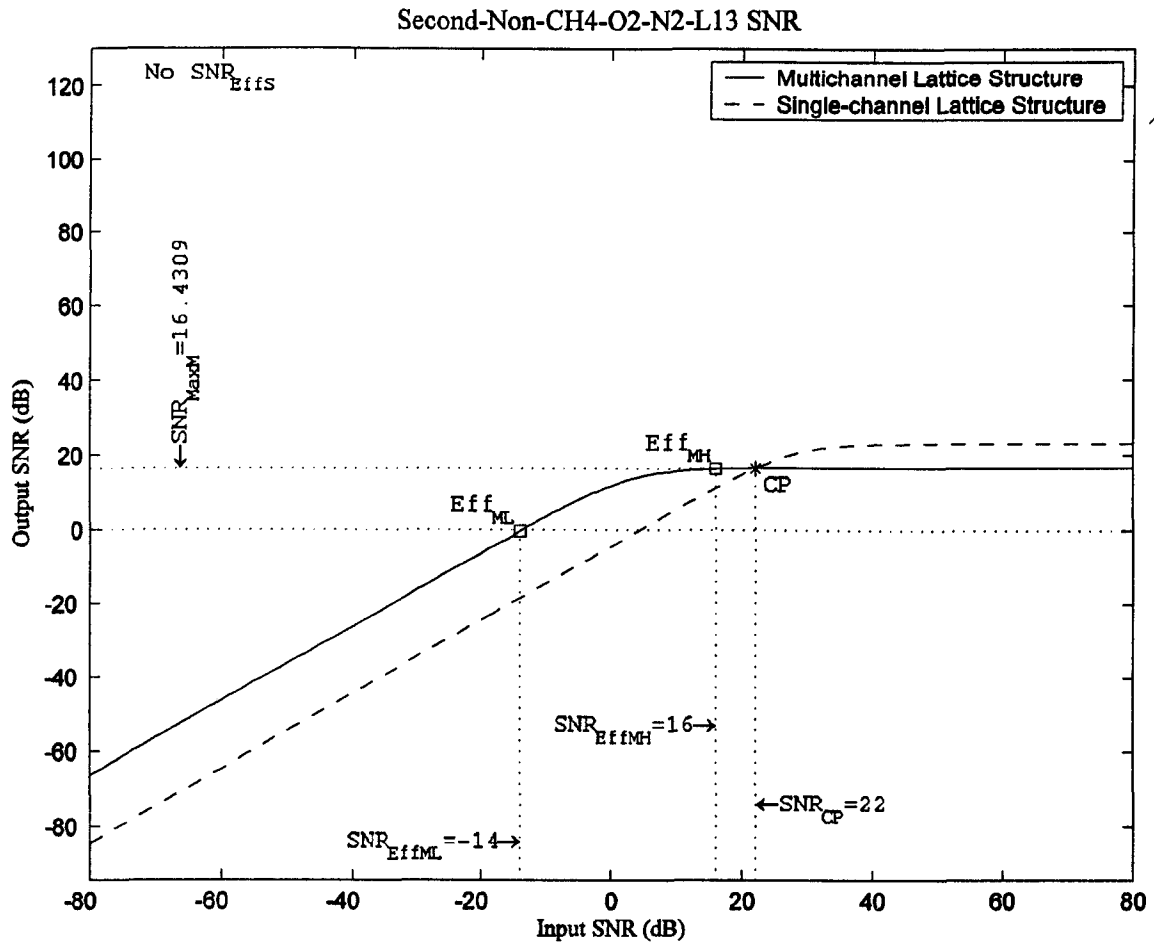


Figure 4-21: Input/output SNR in Second-Non-CH4-O2-N2-L13

This simulation shows that even if the absolute maximum delay exceeds the limit of SIMO module, but the relative maximum stays in the limit of SIMO module, ANC can be achieved by increasing the total number of stages of MALP and MMRTF modules in MCLS. The detailed result of all 12 simulations (No. 49-60), as defined in Table 4-9, using second-order nonlinear channel #4 with $L = 13$ is shown in Table 4-14.

The average SNR improvement of the simulation No. 53, as shown in Table 4-14, is about 14.2dB when the input SNR level ranges from -14dB to 16dB. All the simulations using Second-order nonlinear channel #4 with $L = 13$ have the effective improvement SNR range. SCLS can not provide ANC functionality in some simulations, such as No 52, 53, 55, 58 and 59 in Table 4-14, where MCLS can provide.

Table 4-14: Simulation results using Second-Non channel #4 with $L=13$

No	SNR _{EffS} (dB)			SNR _{EffM} (dB)			CP	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
49	-6	42	42.197	-34	29	29.383	23	-34	29	20.502
50	-29	22	22.164	-46	17	17.258	-10	-46	17	11.560
51	-34	21	21.398	-42	17	16.719	-17	-42	17	4.883
52	N/A	N/A	N/A	-8	14	14.209	44	-8	14	34.453
53	N/A	N/A	N/A	-14	16	16.431	22	-14	16	14.192
54	-1	15	15.059	-8	15	14.901	15	-8	15	4.550
55	N/A	N/A	N/A	-12	14	13.881	40	-12	14	33.019
56	-1	14	13.993	-19	16	16.057	17	-19	16	12.266
57	-6	19	19.115	-13	16	16.022	11	-13	16	4.663
58	N/A	N/A	N/A	-11	19	19.026	46	-11	19	33.921
59	N/A	N/A	N/A	-17	16	16.379	19	-17	16	12.919
60	-5	17	17.320	-11	15	15.067	12	-11	15	4.625

The comparison of SNR improvements is shown in Figure 4-22. The values of average SNR improvement are grouped according to different reference signals, i.e. the values of average SNR improvements are close under the same reference signal. When $L = 4$, under WGN signal or CGN signal, MCLS does not provide or only provides a little performance improvement. But when $L = 13$, the average SNR improvement is about 12dB under WGN and about 5dB under CGN respectively. This shows that the performance of MCLS is closely related to the maximum delay property of the transmission channel.

The comparison of SNR_{MaxDesc} over second-order nonlinear channel #4 with $L = 4$ and $L = 13$ is shown in Figure 4-23. When the noise-free primary signal is one-frequency signal and the reference input is two-frequency signal, the SNR_{MaxDesc} is about 13dB. In

other noise-free primary signal and reference signal combinations, the $SNR_{MaxDesc}$ is below 5dB.

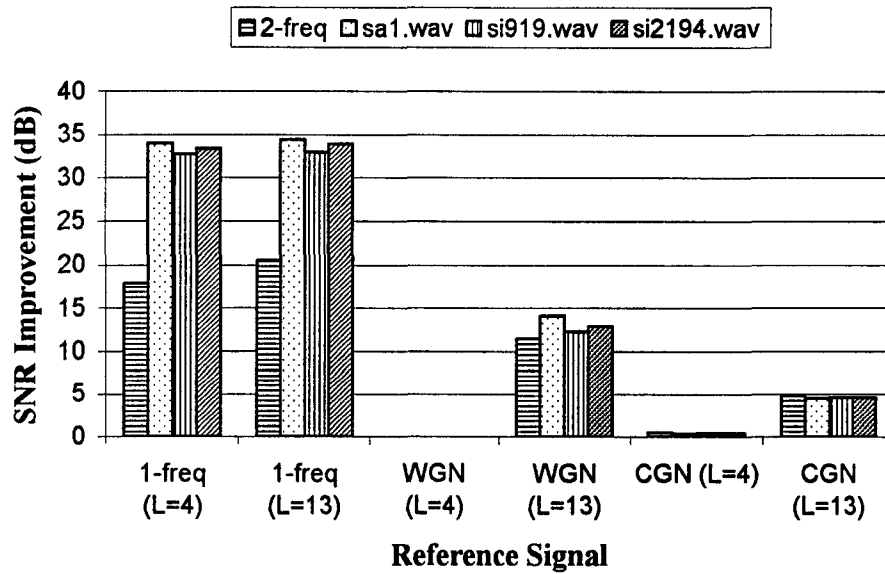


Figure 4-22: SNR improvement over Second-Non channel #4

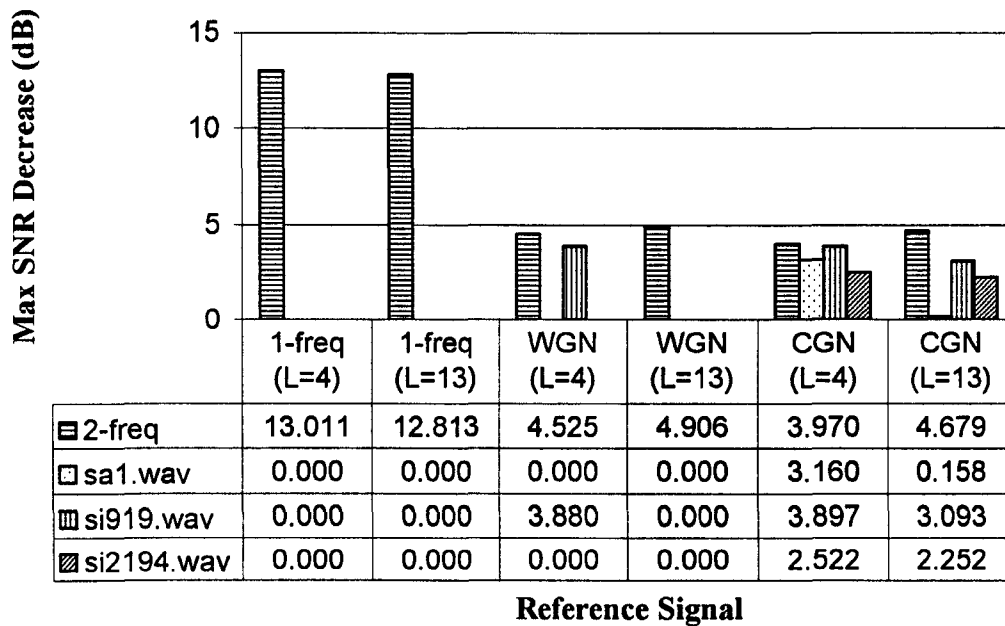


Figure 4-23: Max output SNR decrease over Second-Non channel #4

4.3.1.6 Summary on Second-order Nonlinear Transmission Channels

From the analysis of the previous 60 simulations, MCLS has the performance improvement comparing with SCLS when the transmission channel has second-order nonlinearity.

The values of average SNR improvement are grouped according to the different reference signal. When the reference signal is one-frequency signal, the average SNR improvement is about 37dB. When the reference input is WGN signal, the average SNR improvement is about 15dB. When the reference input is CGN signal, the average SNR improvement is about 6dB. The values of maximum output SNR decrease is around 13dB when the noise-free primary signal is two-frequency signal and the reference signal is one-frequency signal, and around 5dB for other scenarios respectively.

Different distribution of linear and second-order nonlinear elements within the transmission channel will not produce much difference on the performance improvement.

Even if the SIMO module only includes the basic elements within the second-order Volterra series expansion, MCLS has the performance improvement over SCLS when the transmission channel also includes the non-basic second-order nonlinear elements.

When the absolute maximum delay in transmission channel exceeds the limit of the definition of SIMO module and the relative maximum delay stays in the limit, the performance improvement can be achieved by increasing the total number of stages (L) of MCLS.

4.3.2 Linear Transmission Channel

Simulations over linear transmission channel are implemented to evaluate and compare the performance of MCLS with SCLS when the transmission channel only has the linear elements. There are totally 24 simulations carried out over 2 different linear

transmission channels, 4 different noise-free primary signals and 3 different reference signals, as described in Table 4-1, Table 4-3 and Table 4-4.

Table 4-15: Simulation titles using linear channel #1

No.	Index Text	L	No.	Index Text	L
61	Linear-CH1-O1-N1	4	67	Linear-CH1-O3-N1	4
62	Linear-CH1-O1-N2	4	68	Linear-CH1-O3-N2	4
63	Linear-CH1-O1-N3	4	69	Linear-CH1-O3-N3	4
64	Linear-CH1-O2-N1	4	70	Linear-CH1-O4-N1	4
65	Linear-CH1-O2-N2	4	71	Linear-CH1-O4-N2	4
66	Linear-CH1-O2-N3	4	72	Linear-CH1-O4-N3	4

Table 4-16: Simulation titles using linear channel #2

No.	Index Text	L	No.	Index Text	L
73	Linear-CH2-O1-N1	4	79	Linear-CH2-O3-N1	4
74	Linear-CH2-O1-N2	4	80	Linear-CH2-O3-N2	4
75	Linear-CH2-O1-N3	4	81	Linear-CH2-O3-N3	4
76	Linear-CH2-O2-N1	4	82	Linear-CH2-O4-N1	4
77	Linear-CH2-O2-N2	4	83	Linear-CH2-O4-N2	4
78	Linear-CH2-O2-N3	4	84	Linear-CH2-O4-N3	4

12 Simulations (No. 61-72), as illustrated in Table 4-15, are carried out over Linear channel #1. Another 12 simulations (No. 73-84), as illustrated in Table 4-16, are carried out over Linear channel #2.

The result of simulation No. 65, i.e. Linear-CH1-O2-N2 in Table 4-15, is analyzed here. It uses the linear transmission channel #1 in Table 4-1, the noise-free

primary signal #2 is a voice file named "si1573.wav" under the sampling frequency 16 kHz in Table 4-3 and the reference signal #2 defined in Table 4-4.

Figure 4-24 shows the output SNR levels from both MCLS and SCLS when the input SNR level ranges from -80dB to 80dB.

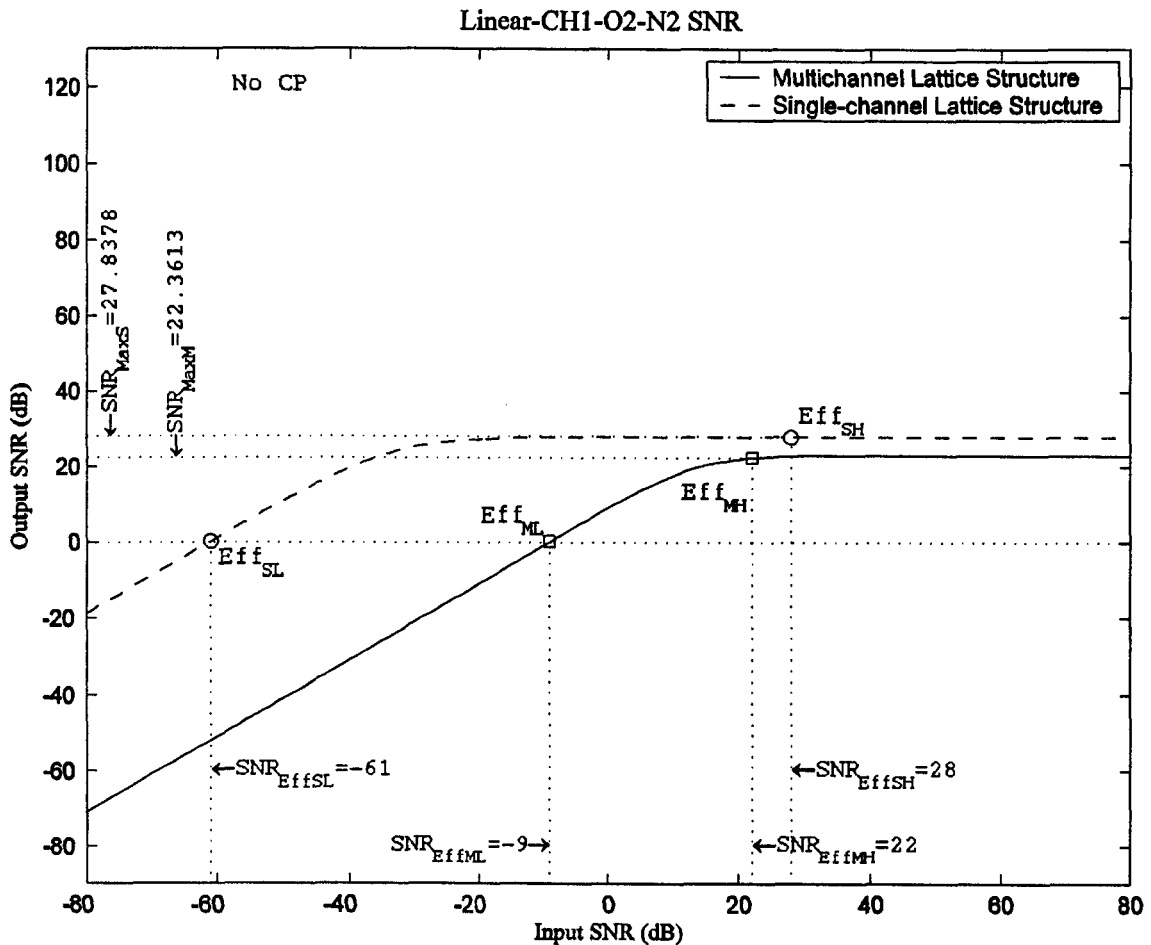


Figure 4-24: Input/Output SNR in Linear-CH1-O2-N2

Both MCLS and SCLS have the effective range. There is no CP existed, which means the ANC performance of SCLS is always better than MCLS.

The detailed result of all 24 simulations using linear transmission channel #1 and #2 is shown in Table 4-17. None of them has the effective SNR improvement range.

Table 4-17: Simulation results using Linear channels

No	SNR _{EMS} (dB)			SNR _{EMM} (dB)			CP	SNR _{Imp} (dB)		
	L	H	Max	L	H	Max		L	H	IMP _{Avg}
61	-74	44	44.392	-62	30	30.169	N/A	N/A	N/A	N/A
62	-66	27	27.223	-50	23	22.694	N/A	N/A	N/A	N/A
63	-60	25	25.130	-53	21	21.164	N/A	N/A	N/A	N/A
64	-42	35	35.293	-25	16	16.459	N/A	N/A	N/A	N/A
65	-61	28	27.838	-9	22	22.361	N/A	N/A	N/A	N/A
66	-47	24	24.159	-13	21	21.136	N/A	N/A	N/A	N/A
67	-56	26	26.206	-38	20	19.565	N/A	N/A	N/A	N/A
68	-80	25	24.583	-22	21	21.224	N/A	N/A	N/A	N/A
69	-61	25	24.871	-27	21	20.830	N/A	N/A	N/A	N/A
70	-54	32	32.075	-37	23	23.449	N/A	N/A	N/A	N/A
71	-80	27	26.899	-21	22	21.912	N/A	N/A	N/A	N/A
72	-59	23	23.207	-24	20	20.073	N/A	N/A	N/A	N/A
73	-58	44	44.395	-52	30	30.169	N/A	N/A	N/A	N/A
74	-65	27	27.223	-38	23	22.691	N/A	N/A	N/A	N/A
75	-56	25	25.129	-43	21	21.171	N/A	N/A	N/A	N/A
76	-31	35	35.301	-14	16	16.297	N/A	N/A	N/A	N/A
77	-59	28	27.838	N/A	N/A	N/A	N/A	N/A	N/A	N/A
78	-38	24	24.153	-1	16	16.103	N/A	N/A	N/A	N/A
79	-45	26	26.206	-27	20	19.559	N/A	N/A	N/A	N/A
80	-75	25	24.583	-10	21	20.881	N/A	N/A	N/A	N/A
81	-51	25	24.871	-15	21	20.766	N/A	N/A	N/A	N/A
82	-43	32	32.074	-26	23	23.435	N/A	N/A	N/A	N/A
83	-74	27	26.899	-9	22	21.560	N/A	N/A	N/A	N/A
84	-50	23	23.206	-12	20	19.838	N/A	N/A	N/A	N/A

However, almost all the cases in Table 4-17, except No. 77, have the effective SNR range using MCLS, which means that MCLS has the ANC ability over the linear transmission channel.

To summarize the simulations over linear transmission channel, MCLS shows the ANC ability. But the performance of MCLS is worse than SCLS. So under the linear transmission channel, use MCLS has no advantage at all. According to 3.3, the SCLS is a special case of the MCLS. It should be considered to switch MCLS into SCLS in case of the linear transmission channel.

4.3.3 Third-order Nonlinear Transmission Channel

Simulations over third-order nonlinear transmission channels are implemented to evaluate and compare the performance of MCLS with SCLS. There are totally 24 simulations carried out over 2 different third-order nonlinear transmission channels, 4 different noise-free primary signals and 3 different reference signals as described in Table 4-1, Table 4-3 and Table 4-4.

12 Simulations (No. 85-96), as illustrated in Table 4-18, are carried out over third-order nonlinear channel #1. Another 12 simulations (No. 97-108), as illustrated in Table 4-19, are carried out over third-order nonlinear channel #2.

The result of simulation No. 92, i.e. `Third-Non-CH1-O3-N2` in Table 4-18, is analyzed here. It uses the third-order nonlinear transmission channel #1 in Table 4-1, the noise-free primary signal #3 in Table 4-3 and the reference signal #2 in Table 4-4.

Figure 4-25 shows the output SNR levels from three structures when the input SNR level ranges from -80dB to 80dB. The solid line refers to T-structure, which is the MCLS with the third-order SIMO module, as illustrated in (3.2.5). The dashed line refers to M-structure, which is the MCLS with the second-order SIMO module, as illustrated in (3.2.4). The dash-dotted line refers to S-structure, which is SCLS.

Table 4-18: Simulation titles using Third-Non channel #1

No.	Index Text	L	No.	Index Text	L
85	Third-Non-CH1-O1-N1	4	91	Third-Non-CH1-O3-N1	4
86	Third-Non-CH1-O1-N2	4	92	Third-Non-CH1-O3-N2	4
87	Third-Non-CH1-O1-N3	4	93	Third-Non-CH1-O3-N3	4
88	Third-Non-CH1-O2-N1	4	94	Third-Non-CH1-O4-N1	4
89	Third-Non-CH1-O2-N2	4	95	Third-Non-CH1-O4-N2	4
90	Third-Non-CH1-O2-N3	4	96	Third-Non-CH1-O4-N3	4

Table 4-19: Simulation titles using Third-Non channel #2

No.	Index Text	L	No.	Index Text	L
97	Third-Non-CH2-O1-N1	4	103	Third-Non-CH2-O3-N1	4
98	Third-Non-CH2-O1-N2	4	104	Third-Non-CH2-O3-N2	4
99	Third-Non-CH2-O1-N3	4	105	Third-Non-CH2-O3-N3	4
100	Third-Non-CH2-O2-N1	4	106	Third-Non-CH2-O4-N1	4
101	Third-Non-CH2-O2-N2	4	107	Third-Non-CH2-O4-N2	4
102	Third-Non-CH2-O2-N3	4	108	Third-Non-CH2-O4-N3	4

The effective SNR range of S-structure is between Eff_{SL} and Eff_{SH} , while Eff_{ML} and Eff_{MH} for M-structure, Eff_{TL} and Eff_{TH} for T-structure respectively. There are maximum three crossover points among those three structures, from where we can find out the SNR improvement range using the similar method as we discussed in 4.2.

The detailed result of all 24 simulations using third-order nonlinear channel #1 and #2 is shown in Table 4-20.

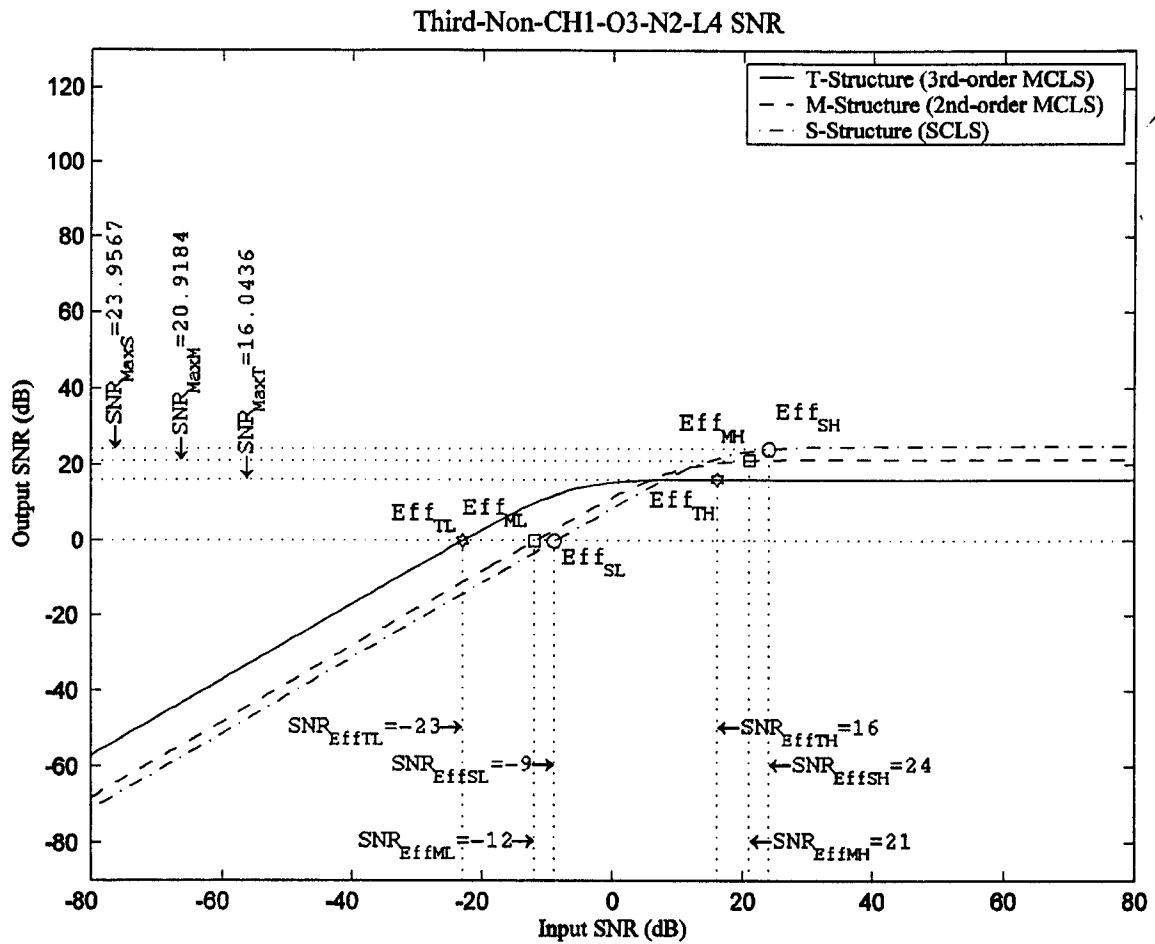


Figure 4-25: Input/Output SNR in Third-Non-CH1-O3-N2

In Table 4-20, SNR_{Eff} stands for the effective SNR range, while **S** for S-structure, **M** for M-structure and **T** for T-structure. **L** stands for the low-end input SNR level. **H** stands for the high-end input SNR level. **Max** stands for the maximum output SNR level.

SNR_{CP} stands for the input SNR on the crossover point, where **MS** for the crossover point between M-structure and S-structure, while **TS** for T-structure and S-structure, and **TM** for T-structure and M-structure respectively.

SNR_{IMP} stands for the effective SNR improvement range. IMP_{Avg} represents the average SNR improvement, where **MS** for the crossover point between M-structure and S-structure, while **TS** for T-structure and S-structure, and **TM** for T-structure and M-structure respectively.

Table 4-20: Simulation results using Third-Non channel #1 and #2

No	SNR _{Eff} (dB)				SNR _{CP}		SNR _{IMP} (dB)		
		S	M	T	(dB)		L	H	IMP _{Avg}
85	L	-16	-17	6	MS	6	-17	30	0.582
	H	44	30	-4	TS	N/A	N/A	N/A	N/A
	Max	44.282	30.102	12.885	TM	N/A	N/A	N/A	N/A
86	L	-36	-38	-48	MS	-15	-38	23	1.754
	H	27	23	18	TS	-17	-48	18	8.303
	Max	27.222	22.693	22.693	TM	-19	-48	18	6.714
87	L	-45	-47	-50	MS	-25	-47	21	1.788
	H	25	21	16	TS	-30	-50	16	3.424
	Max	25.128	21.159	21.159	TM	-33	-50	16	1.595
88	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A
89	L	N/A	N/A	-10	MS	26	N/A	N/A	N/A
	H	N/A	N/A	17	TS	23	-10	17	11.942
	Max	N/A	N/A	14.397	TM	21	-10	17	9.307
90	L	-4	-6	-9	MS	17	-6	20	1.796
	H	22	20	14	TS	10	-9	14	3.687
	Max	21.907	20.109	17.768	TM	7	-9	14	1.876
91	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A
92	L	-9	-12	-23	MS	12	-12	12	2.339
	H	24	21	16	TS	8	-23	8	9.742
	Max	23.957	20.918	16.044	TM	6	-23	6	7.454
93	L	-17	-20	-23	MS	3	-20	21	2.187
	H	25	21	15	TS	-3	-23	15	4.024
	Max	24.803	20.772	20.652	TM	-7	-23	15	1.877
94	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A

Table 4-20 (Continued)

95	L	-7	-10	-21	MS	14	-10	21	2.203
	H	26	21	18	TS	12	-21	18	9.867
	Max	25.914	21.343	20.877	TM	10	-21	18	7.784
96	L	-15	-18	-21	MS	4	-18	20	2.059
	H	23	20	15	TS	-1	-21	15	3.782
	Max	23.076	20.016	19.866	TM	-5	-21	15	1.834
97	L	-9	-29	-16	MS	22	-29	30	15.432
	H	44	30	3	TS	-7	-16	3	3.884
	Max	43.793	30.173	28.090	TM	N/A	N/A	N/A	N/A
98	L	-24	-42	-44	MS	0	-42	23	12.935
	H	27	23	18	TS	-5	-44	18	13.914
	Max	27.210	22.693	22.692	TM	-26	-44	18	1.797
99	L	-33	-51	-53	MS	-9	-51	21	12.658
	H	25	21	16	TS	-16	-53	16	13.352
	Max	25.126	21.160	21.160	TM	-39	-53	16	0.919
100	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A
101	L	N/A	-1	-4	MS	N/A	N/A	N/A	N/A
	H	N/A	17	16	TS	35	-4	16	20.174
	Max	N/A	17.112	16.347	TM	15	-4	16	1.958
102	L	N/A	-12	-14	MS	N/A	N/A	N/A	N/A
	H	N/A	21	15	TS	24	-14	15	18.423
	Max	N/A	21.059	20.306	TM	0	-14	15	1.572
103	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A
104	L	N/A	-15	-18	MS	28	N/A	N/A	N/A
	H	N/A	21	16	TS	21	-18	16	16.320
	Max	N/A	21.118	20.848	TM	-1	-18	16	1.866
105	L	-4	-26	-27	MS	19	-26	21	14.488
	H	23	21	15	TS	11	-27	15	15.015

Table 4-20 (Continued)

	Max	23.015	20.832	20.823	TM	-14	-27	15	1.155
106	L	N/A	N/A	N/A	MS	N/A	N/A	N/A	N/A
	H	N/A	N/A	N/A	TS	N/A	N/A	N/A	N/A
	Max	N/A	N/A	N/A	TM	N/A	N/A	N/A	N/A
107	L	N/A	-13	-16	MS	29	N/A	N/A	N/A
	H	N/A	22	18	TS	24	-16	18	17.098
	Max	N/A	21.684	21.402	TM	4	-16	18	2.044
108	L	-3	-23	-25	MS	20	N/A	N/A	N/A
	H	20	20	15	TS	13	-25	15	14.536
	Max	19.971	20.049	19.984	TM	-11	-25	15	1.330

In simulation No. 92, as shown in Table 4-20, when the input SNR level ranges from -12dB to 12dB, the performance of M-structure is better than S-structure, and the average SNR improvement is 2.339dB. When the input SNR level ranges from -23dB to 8dB, the performance of T-structure is better than S-structure, and the average SNR improvement is 9.742dB. When the input SNR level ranges from -23dB to 6dB, the performance of T-structure is better than M-structure, and the average SNR improvement is 7.454dB. The maximum effective output SNR of S-structure is 23.957dB, while M-structure is 20.918dB and T-structure is 16.044dB respectively.

The comparison of SNR improvement (T-structure vs. M-structure) over third-order nonlinear transmission channel #1 is shown in Figure 4-26. When the noise-free primary signal is two-frequency signal, there is no improvement. When the reference signal is WGN signal, the average SNR improvement is about 7dB. When the reference signal is CGN signal, the average SNR improvement is about 1.7dB.

The comparison of $SNR_{MaxDesc}$ (T-structure vs. M-structure) over third-order nonlinear channel #1 is shown in Figure 4-27. When the noise-free primary signal is two-frequency signal and the reference signal is one-frequency signal, the maximum output SNR decrease is about 17dB. Otherwise, it is below 2.5dB.

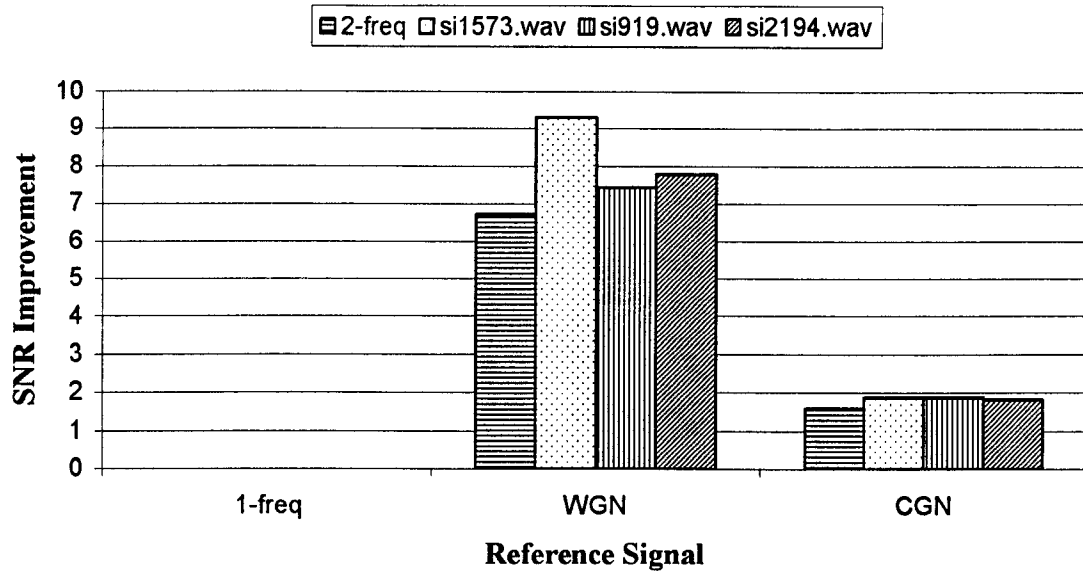


Figure 4-26: SNR improvement (TM) over Third-Non channel #1

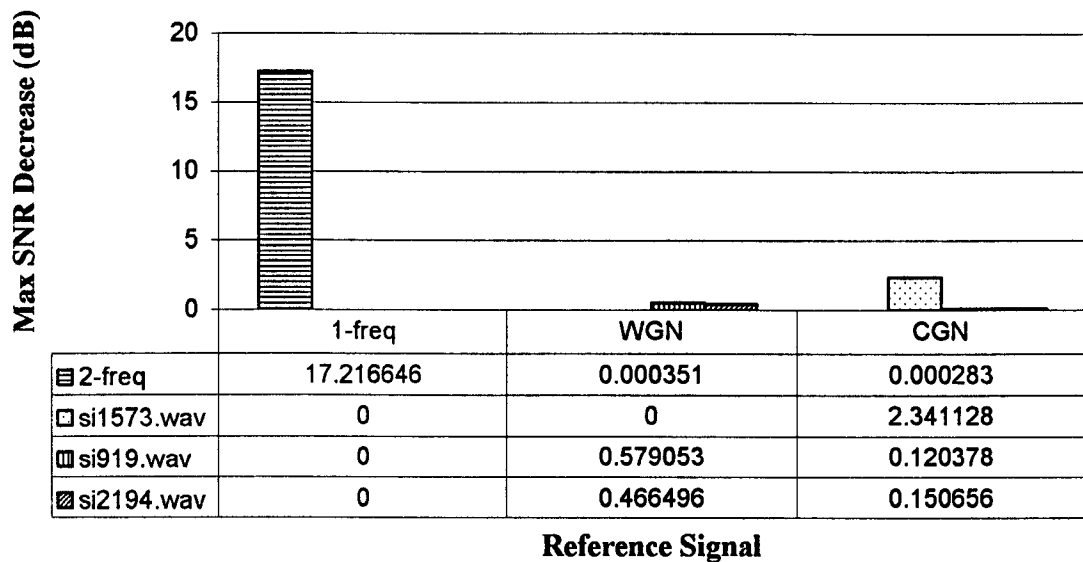


Figure 4-27: Max output SNR decrease (TM) over Third-Non channel #1

The comparison of SNR improvement (T-structure vs. S-structure) over third-order nonlinear transmission channel #2 is shown in Figure 4-28. When the reference signal is one-frequency signal, there is no improvement. When the reference signal is WGN signal,

the average SNR improvement is about 1.8dB. When the reference signal is CGN signal, the average SNR improvement is about 1dB.

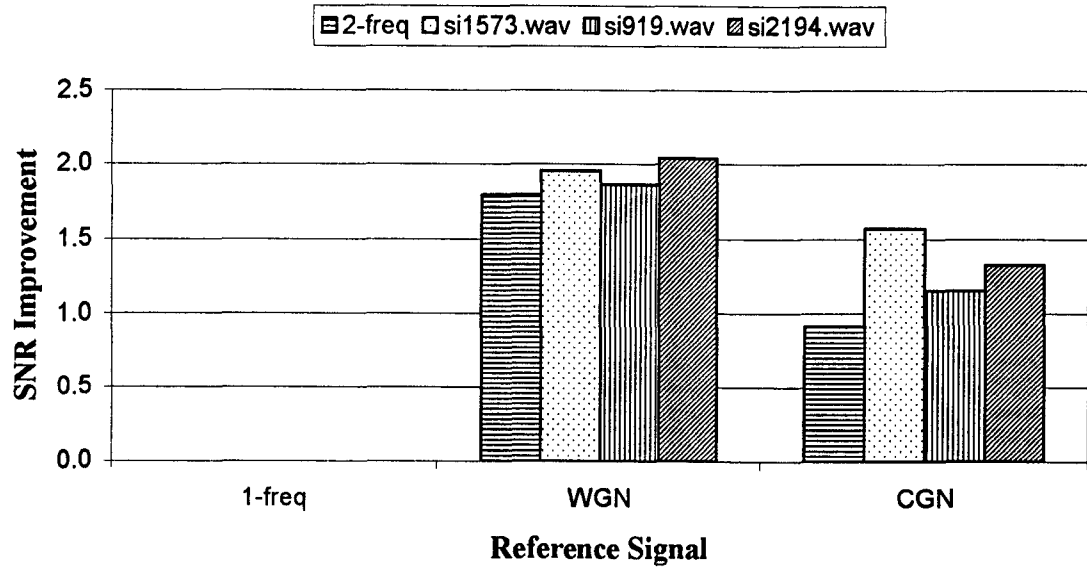


Figure 4-28: SNR improvement (TM) over Third-Non channel #2

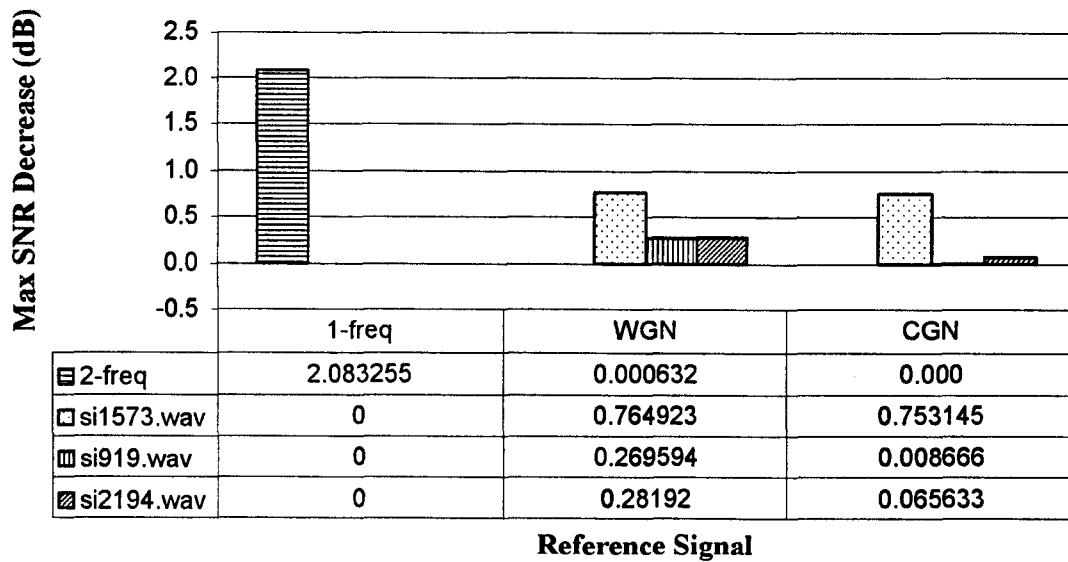


Figure 4-29: Max output SNR decrease (TM) over Third-Non channel #2

The comparison of $SNR_{MaxDesc}$ (T-structure vs. M-structure) over third-order nonlinear transmission channel #2 is shown in Figure 4-29. When the noise-free primary signal is two-frequency signal and the reference input is one-frequency signal, the maximum output SNR decrease is about 2dB. In other occasions, it is below 0.8dB.

To summarize the simulations over third-order nonlinear transmission channels, it appears that by modifying MCLS to include the third-order nonlinearity using the third-order SIMO module, it has the performance improvement when the reference signal is WGN signal or CGN signal. As more adaptation steps are added into the structure, the fixed error increased. The maximum output SNR level is decreased. The simulation result shows that the MCLS with third-order SIMO module (T-structure) can only provide the marginal performance improvement over SCLS or M-structure with the expense of lower maximum output SNR level and higher computational complexity.

5 CONCLUSIONS AND FUTURE WORK

This thesis addressed the problem of adaptive noise cancellation under nonlinear transmission channel using the typical two-microphone ANC system. A multichannel lattice structure (MCLS) is applied, where the single-channel reference signal is converted into the multichannel signal vector, where second-order or higher-order nonlinearities are included. The multichannel signal vector is processed through the multichannel adaptive lattice predictor and the multichannel multiple regression transversal filter stage-by-stage, where NLMS algorithm is used. The truncated Volterra series expansion is used to model the nonlinear transmission channel.

The results provide the evidences that the multichannel lattice structure had a performance improvement over the single-channel lattice structure, especially in low input SNR conditions. The major drawback of the single-channel lattice structure is that the performance deteriorates rapidly when the transmission channel has nonlinear properties. However, the transmission channel in real world environment tends to be nonlinear. The results shows the evidences that by correctly selecting the maximum number of delay, maximum stage number, smooth factor, and step size, significant noise attenuation can be achieved by using the multichannel lattice structure.

The results of the simulations using one-frequency signal, white Gaussian noise signal and coloured Gaussian noise signal as the reference signal, using two-frequency signal and voice signals as the noise-free primary signal, over different simulated nonlinear transmission channels, show that the multichannel lattice structure provided improvement on output SNR level and the effective SNR range over the single-channel lattice structure when the transmission channel has nonlinear properties.

When the transmission channel is purely linear, the performance of the single-channel lattice structure is better than the multichannel lattice structure. Because in the multichannel lattice structure, the adaptation steps increased, so the base noise level is increased.

The third-order nonlinear transmission channels are evaluated using the multichannel lattice structure with third-order SIMO module. The simulation results show that the multichannel lattice structure can only provide the marginal performance improvement, with the expense of high computational complexity and decreased maximum output SNR level.

Future research needs to be conducted in the evaluation of the multichannel lattice structure in real world noise environment to model its nonlinear properties. As the increased base noise level decrease the maximum output SNR level, the reason needs to be investigated to try to avoid or decrease its effect. The adaptation process needs to be monitored and optimized to speed up the convergence and increase the noise cancellation ability.

REFERENCE

- [1] D.A. George, "Continuous nonlinear systems", Technical Report 355, MIT Research Lab. Elect., 1959
- [2] A.V. Oppenheim, R.W. Schaffer, and T.G. Stockham Jr., "Nonlinear filtering of multiplied and convolved signals", Proc. IEEE, vol. 56, No. 8, pp 1264-1291, August 1968
- [3] F. Itakura and S. Saito, "Digital filtering techniques for speech analysis and synthesis", Proc. 7th Int. Conf. Acoust., vol. 3, Paper 25C-1, pp 261, 1971
- [4] A.H. Gary, Jr., and J.D. Markel, "Digital lattice and ladder filter synthesis", IEEE Trans. Audio Electroacoust., vol. AU-21, pp 491-500, Dec. 1973
- [5] B. Widrow, *et al.*, "Adaptive noise cancelling: principles and applications", Proc. IEEE, vol. 63, pp. 1692-1716, Dec. 1975
- [6] E.A. Robinson and M.T. Silvia, "Digital Signal Processing and Time Series Analysis", Holden Day, San Francisco, 1978
- [7] G. Mitzel, S. Clancy, W. Rugh, "On transfer function representations for homogeneous nonlinear systems", IEEE Trans. on Automatic Control, vol. AC-24, pp 242-249, 1979
- [8] M. Schetzen, "The Volterra and Wiener Theory of the Nonlinear Systems", Wiley and Sons, New York, 1980
- [9] M. Schetzen, "Nonlinear system modeling based on the Wiener theory", Proc. IEEE, vol. 69, No. 12, pp 1557-1573, December 1981
- [10] W.J. Rugh, "Nonlinear System Theory: The Volterra Wiener Approach", John Hopkins University Press, Baltimore, Maryland, 1981
- [11] T.A. Nodes, N.C. Gallagher, "Median filters: Some modifications and their properties", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-30, No. 5, pp 739-746, October 1982
- [12] I.W. Sandberg, "On Volterra expansions for time-varying nonlinear systems", IEEE Trans. Circuits and Systems, vol. CAS-30, No. 2, pp 61-67, February 1983
- [13] I.W. Sandberg, "Series expansions for nonlinear systems", Circuits, Systems and Signal Processing, vol. 2, No. 1, pp 77-87, 1983

- [14] S. Boyd, Y.S. Tang, O. Chua, "Measuring Volterra Kernel", IEEE Trans. On Circuit and Systems, vol. 30, no. 8, pp 571-577, 1983
- [15] A.C. Bovik, T.S. Huang, D.C. Munson, "A generalization of median filtering using linear combinations of order statistics", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-31, No. 6, pp 1342-1349, December 1983
- [16] S. Boyd, O. Chua, "Analytical Foundations of Volterra Series", IMA journal of Mathematical Control & Information, vol. 1, pp 243-282, 1984
- [17] I.W. Sandberg, "The mathematical foundations of associated expansions for mildly nonlinear systems", IEEE Trans. On Circuits and Systems, vol. 23, no. 7, pp 441-455, 1984
- [18] Y.H. Lee, S.A. Kassam, "Generalized median filtering and related nonlinear filtering systems", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-33, No. 3, pp 672-683, June 1985
- [19] Heping Ding, Chongzhi Yu, "Adaptive lattice noise canceller and optimal step size", ICASSP IEEE '86, vol. 11, pp 2939 -2942, Apr. 1986
- [20] M. Savoji, "A variable length lattice filter for adaptive noise cancellation", ICASSP IEEE '86. , vol. 11, pp 2935 -2938, Apr. 1986
- [21] P. Maragos, R.W. Schafer, "Morphological filters, part I: Their set theoretic analysis and relations to linear shift invariant filters", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-35, No. 8, pp 1153-1169, August 1987
- [22] P. Maragos, R.W. Schafer, "Morphological filters, part II: Their relation to median, order static and stack filters", IEEE Trans. Acoust., Speech, Signal Proc., vol. ASSP-35, No. 8, pp 1170-1184, August 1987
- [23] C. Davila, A. Welch, H. III Rylander, "A second-order adaptive Volterra filter with rapid convergence", IEEE Trans Acoust., Speech, and Signal Proc., vol. 35-9, pp 1259 -1263, Sep 1987
- [24] John G. Proakis, Dimitris G. Manolakis, "Introduction to Digital Signal Processing", Macmillan, 1988
- [25] A.S. Abutaleb, "An adaptive filter for noise cancelling", IEEE Trans. Circuits Syst., vol. 35-10, pp 1201 -1209, Oct. 1988

- [26] S. Ozgunel, A.N. Kayran, E. Panayirci, "Nonlinear channel equalization using multichannel adaptive lattice algorithm", IEEE Int. Symp. Circuits Syst., pp 2826-2829, 11-14 June 1991
- [27] V.J. Mathews, "Adaptive polynomial filters", IEEE Signal Processing magazine, vol. 8-3, pp 10 -26, July 1991
- [28] M.A. Syed, V.J. Mathews, "Lattice algorithm for recursive least squares adaptive second-order Volterra filtering", IEEE Trans. Circuits Syst. II, vol. 41-3, pp 202 - 214, March 1994
- [29] S. Haykin, "Neural Networks: A Comprehensive Foundation", Macmillan, New York, 1994
- [30] Monson H. Hayes, "Statistical Digital Signal Processing and Modeling", John Wiley & Sons, 1996
- [31] Sen M. Kuo and Dennis R. Morgan, "Active Noise Control Systems - Algorithms and DSP Implementations", John Wiley & Sons, 1996
- [32] Tak-Keung Yeung, Sze-Fong Yau, "A feed-back ANC system using adaptive lattice filters", ISCAS '98.vol 5, pp 186 -189, 31 May-3 June 1998
- [33] V. Madisetti, D.B. Williams, "The digital signal processing handbook", CRC, 1999
- [34] J. Minkoff, S. Novak, D. Zaff, "Dynamics of multichannel feedforward adaptive systems", Signal Processing, IEEE Transactions, vol. 47, issue 10, pp 2700-2709, October 1999
- [35] S.V. Vaseghi, "Advanced digital signal processing and noise reduction, second edition", John Wiley & Sons, New York, 2000
- [36] A.N. Venetsanopoulos, K.N. Plataniotis, "Adaptive filters and multichannel signal processing", Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC, pp 147-152, October 2000
- [37] S.J. Chen, J.S. Gibson, "Feed-forward adaptive noise control with multivariable gradient lattice filters", IEEE Trans. on Signal Proc., vol. 49 -3, pp 511 -520, March 2001
- [38] T. Li, J. Jiang, "Adaptive Volterra filters for active control of nonlinear noise processes", IEEE Trans. On Acoust., Speech, Signal Proc., vol. 49, issue 8, pp 1667-1676, August 2001

- [39] R.T. Bambang, L. Anggono, K. Uchida, "DSP based RBF neural modeling and control for active noise cancellation", *Intelligent Control, Proceedings of the 2002*, pp. 460-466, October 2002
- [40] R.A. Baki, C. Beainy, M.N. El-Gamal, "Distortion analysis of high-frequency log-domain filters using Volterra series", *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions*, vol. 50, issue 1, pp 1-11, January 2003
- [41] M.I. Yusof, "Modelling Volterra series based on input/output data" *Telecommunication Technology, NCTT 2003 Proceedings. 4th National Conference*, pp 222-225, January 2003
- [42] J.T. Cao, N. Murata, Amari, S.; A. Cichocki, T. Takeda, "A robust approach to independent component analysis of signals with high-level noise measurements", *Neural Networks, IEEE Transactions*, vol. 14, issue 3, pp 631-645, May 2003
- [43] A. Guerin, G. Faucon, R. Le Bouquin-Jeannes, "Nonlinear acoustic echo cancellation based on Volterra filters", *Speech and Audio Processing, IEEE Transactions*, vol. 11 , issue 6 , pp 672-683, November 2003
- [44] Z. Li, M.J. Er, Y. Gao, "An adaptive RBFN-based filter for adaptive noise cancellation, *Decision and Control, 2003. Proceedings. 42nd IEEE Conference*, vol. 6 , pp 6175-6180, December 2003

VITA AUCTORIS

Qing Xie was born in Shanghai, China, on September 9, 1968, the son of Yonglin Xie and Youxian Wang. He received the degree of Bachelor of Science with a major in Electronic Engineering from Shanghai Jiao Tong University in July 1991. During the following years, he was employed as a system/software engineer in Marconi and later in Motorola. In October 2000, he immigrated to Canada with his wife and his son. He was employed as a software designer in Nortel Networks after that. In May 2002, he entered the Graduate School of the University of Windsor at Windsor, Ontario.