#### Sparseness-controlled Adaptive Algorithms for Supervised and Unsupervised System Identification

PRADEEP LOGANATHAN

April 2011

A Thesis submitted in fulfilment of requirements for the degree of Doctor of Philosophy of Imperial College London

Communications and Signal Processing Group Department of Electrical and Electronic Engineering Imperial College London

To my parents, brothers, Andy, Emanuël and

To Patrick

### Abstract

In the single-channel hands-free telephony, the acoustic coupling between the loudspeaker and the microphone can be strong and this generates echoes that can seriously degrade user experience. Therefore, effective acoustic echo cancellation (AEC) is necessary to maintaining a stable system and hence improving the perceived voice quality of a call. Traditionally, adaptive filters have been deployed in acoustic echo cancellers to estimate the acoustic impulse responses (AIRs) using adaptive algorithms. The performances of a range of well-known algorithms are studied in the context of both AEC and network echo cancellation (NEC). It presents insights into their tracking performances under both time-invariant and time-varying system conditions.

In the context of AEC, the level of sparseness in AIRs can vary greatly in a mobile environment. When the response is strongly sparse, convergence of conventional approaches is poor. Drawing on techniques originally developed for NEC, a class of timedomain and a frequency-domain AEC algorithms are proposed that can not only work well in both sparse and dispersive circumstances, but also adapt dynamically to the level of sparseness using a new sparseness-controlled approach.

As the early part of the acoustic echo path is sparse while the late reverberant part of the acoustic path is dispersive, a novel approach to an adaptive filter structure that consists of two time-domain partition blocks is proposed such that different adaptive algorithms can be used for each part. By properly controlling the mixing parameter for the partitioned blocks separately, the proposed partitioned block algorithm works well in both sparse and dispersive time-varying circumstances.

A new insight into an analysis on the tracking performance of IPNLMS is presented by deriving the expression for the mean-square error. By employing the framework for both sparse and dispersive time-varying echo paths, this work validates the analytic results in practical simulations for AEC.

The time-domain second-order statistic based blind SIMO identification algorithms, which exploit the cross relation method, are investigated and then a technique with proportionate step-size control for both sparse and dispersive system identification is also developed.

### Acknowledgment

I would like to express my sincere gratitude to my supervisor Dr. Patrick Naylor for his valuable guidance and support throughout the entire project. I appreciate his tolerance and thanks to him for his time spent on advising me and evaluating my work. His helps can never be forgotten.

I am also indebted to Dr. Andy Khong, Dr. Emanuël A. P. Habets, Dr. Mark Thomas, Dr. Nikolay Gaubitch and Dr. Xiang (Shawn) Lin for being my well-wishers and thank them for their constructive remarks and unflinching support throughout this course.

I wish to thank all my friends in our CSP lab for their encouragement, good wishes and invaluable company until late night.

Finally, to make this acknowledgement complete, a special thanks to my parents and brothers for their unconditional love and support, this added an essence to my work.

## Contents

Abstra	act	5
Ackno	wledgment	7
Conter	nts	9
List of	Figures	13
List of	Tables	19
Abbre	viations	<b>2</b> 1
Mathe	ematical Symbols	25
Chapt	er 1. Introduction	29
1.1	Overview	29
1.2	Research objectives	32
1.3	Thesis structure	34
1.4	Statement of originality	36
1.5	Publications	37
Chapt	er 2. Literature Review	<b>39</b>
2.1	Overview of adaptive algorithms for echo cancellation $\ldots \ldots \ldots \ldots$	39
2.2	Acoustic echo cancellation	42
	2.2.1 Notations and definitions	42
	2.2.2 Assumptions	43
	2.2.3 Adaptive echo cancellation process	44
2.3	Performance measures	45
	2.3.1 Mean square error	45
	2.3.2 Normalized misalignment	45
2.4	Time domain adaptive algorithms for echo cancellation	46
	2.4.1 The LMS and NLMS algorithms	46

	2.4.2	The PNLMS and MPNLMS algorithm
	2.4.3	The IPNLMS algorithm
	2.4.4	Computational complexity
2.5	Frequ	ency domain adaptive algorithms for echo cancellation $\ldots \ldots \ldots \ldots 53$
	2.5.1	Notations and definitions 54
	2.5.2	The FLMS algorithm 55
	2.5.3	The MDF algorithm
	2.5.4	The IPMDF algorithm
	2.5.5	Computational complexity
2.6	Summ	nary
	2.6.1	Time domain algorithms
	2.6.2	Frequency domain algorithms
Chapte	er 3.	A Class of Sparseness-controlled Algorithms 65
3.1	Introd	luction $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $65$
3.2	Sparse	eness measure
3.3	Chara	cterization of framework for robust convergence in the time domain . 68
	3.3.1	Variation of sparseness in AIRs
	3.3.2	Effect of $\rho$ on step-size control matrix $\mathbf{Q}(n)$ for PNLMS 69
3.4	Time	domain sparseness-controlled algorithms
	3.4.1	The SC-PNLMS and SC-MPNLMS algorithm
	3.4.2	The SC-IPNLMS algorithm
	3.4.3	Performance evaluation
	3.4.4	Computational complexity
3.5	Frequ	ency domain sparseness-controlled algorithms
	3.5.1	The SC-IPMDF algorithm
	3.5.2	Performance evaluation
	3.5.3	Computational complexity
3.6	Summ	$ary \dots \dots$
	3.6.1	Sparseness-controlled time domain algorithms
	3.6.2	Sparseness-controlled frequency domain algorithms
Chapte	er 4.	A Partitioned Block Proportionate Adaptive Algorithm 99
4.1	Introd	luction
4.2	Motiv	ation $\ldots \ldots 100$
4.3	Fixed	length partitioned block IPNLMS
	4.3.1	Non-proportionate weighting
	4.3.2	Proportionate weighting
	4.3.3	Performance evaluation

4.4	4 Varial	ble length partitioned block IPNLMS
	4.4.1	Automatic control of the block length $L_1$
	4.4.2	The proposed VLPB-IPNLMS algorithm
	4.4.3	Performance evaluation
4.5	5 Summ	110 ary $110$ ary
	4.5.1	The partitioned block IPNLMS algorithms
Chap	oter 5.	Performance Analysis for Time-Varying System Identifications11
5.1	l Introd	luction
5.2	2 Time-	varying system model $\ldots \ldots 12$
5.3	B Recur	sive mean-square error analysis $\ldots \ldots 12$
	5.3.1	General formulation
	5.3.2	Assumptions
	5.3.3	Recursive mean-square error analysis for IPNLMS
	5.3.4	Steady-state analysis for IPNLMS
5.4	4 Perfor	mance evaluation $\ldots \ldots 12$
	5.4.1	Experimental setup
	5.4.2	Performances prediction under different rates of time-varying systems 12
	5.4.3	Performances prediction using different step-sizes
	5.4.4	Performances prediction using different input signal variances $13$
	5.4.5	Performances prediction using different additive noise variances $13$
5.5	5 Summ	$ary \ldots 13$
Chap	oter 6.	A Class of Sparseness-controlled Affine Projection Algorithms
fo	r Blind	SIMO System Identification 13
6.1	l Introd	luction
6.2	2 Signal	model
6.3	B Cross	relation method
6.4	4 Gener	al cost function for SIMO BSI using the CR error $\ldots \ldots \ldots \ldots 14$
	6.4.1	Dual channel $(M = 2)$
	6.4.2	Multichannel $(M \ge 2)$
6.5	5 Multi	channel sparseness-controlled PAPA
	6.5.1	General formulation
	6.5.2	The MC-APA algorithm
	6.5.3	The MC-PAPA formulation
	6.5.4	The MC-SCP-APA and MC-SCMP-APA algorithms
6.6	6 Perfor	mance evaluation $\ldots \ldots 15$
6.7	7 Summ	$ary \ldots \ldots$

Chapte	er 7. (	Conclus	sion	s ar	nd	Fu	tur	e	Wo	ork														153
7.1	Summa	ary			• •							•		•		•	•	•	•		•		•	. 153
7.2	Conclu	sion .			•					•						•			•				•	. 156
7.3	Future	Work			•			•		•	• •	•	•••	•		•	• •	•	•	•	• •		•	. 157
Bibliog	graphy																							159

# List of Figures

1.1	Illustration of acoustic echo in a loudspeaker-room-microphone sys-	
1.2	tem (LRMS)	30
	responses, generated using the method of images, for the cases when the	
1.3	separation is 0.9 m and 7.7 m	33
	$\xi(\mathbf{h}), \xi(\mathbf{h}_1)$ and $\xi(\mathbf{h}_2)$ respectively denote the sparseness measures [2, 3]	
	of the full impulse response, the first block with size of 256 and the second	
	block	34
2.1	Adaptive system for acoustic echo cancellation in a loudspeaker-room-	
	microphone system (LRMS).	43
3.1 3.2	Sparseness measure of different impulse responses	67
	phone, $a$ . The impulse responses are obtained from the image model using	
3.3	a fixed room dimensions of $8 \times 10 \times 3$ m	69
	signal. Impulse responses in Fig. $1.2~{\rm are}$ used as sparse and dispersive AIRs	
3.4	respectively. $[\mu_{\text{PNLMS}} = 0.3, \text{ SNR} = 20 \text{ dB}]$	70
	$\hat{h}_l(n)$ in PNLMS	71
$3.5 \\ 3.6$	$\hat{h}_l(n)$ in PNLMS	71 72
3.5 3.6	$\hat{h}_l(n)$ in PNLMS	<ul><li>71</li><li>72</li><li>74</li></ul>

3.7	Time to reach -20 dB normalized misalignment level for different values of	
	$\lambda$ in SC-PNLMS using WGN input signal. Impulse response in Fig. 1.2	
	used as sparse AIR and dispersive AIR respectively. [ $\mu_{\rm SC-PNLMS} = 0.3$ ,	
3.8	$SNR = 20 \text{ dB}] \dots \dots$	76
	signal with an echo path change at $3.5$ s. Impulse response is changed from	
3.9	Fig. 1.2 (a) to (b) and $\mu_{\text{SC-PNLMS}} = 0.3$ , $\text{SNR} = 20$ dB Relative convergence of NLMS, PNLMS and SC-PNLMS using WGN input	77
	signal with an echo path change at $3.5$ s. Impulse response is changed from	
	that shown from Fig. 1.2 (a) to (b) and $\mu_{\text{NLMS}} = \mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}} =$	
	0.3, SNR = 20  dB.	78
3.10	Relative convergence of NLMS, PNLMS and SC-PNLMS using speech input	
	signal with echo path changes at 58 s. Impulse response is changed from	
	that shown in Fig. 1.2 (a) to (b) and $\mu_{\text{NLMS}} = 0.3$ , $\mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}}$	
9.11	= 0.1,  SNR = 20  dB.	79
3.11	Relative convergence of NLMIS, MPNLMIS and SC-MPNLMIS using WGN	
	input signal with an echo path change at 3.5 s. Impulse response is changed	
	from that shown from Fig. 1.2 (a) to (b) and $\mu_{\text{NLMS}} = 0.3$ , $\mu_{\text{MPNLMS}} =$	~~~
3.12	$\mu_{\text{SC}-\text{MPNLMS}} = 0.25$ , SNR = 20 dB	80
	input signal with echo path changes at 58 s. Impulse response is changed	
	from that shown in Fig. 1.2 (a) to (b) and $\mu_{\rm NLMS}$ = 0.3, $\mu_{\rm MPNLMS}$ =	
3.13	$\mu_{\rm SC-MPNLMS} = 0.25$ , SNR = 20 dB	81
	put signal with an echo path change at $3.5$ s. Impulse response is changed	
	from that shown from Fig. 1.2 (a) to (b) and $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ ,	
3.15	$\mu_{\rm SC-IPNLMS} = 0.7$ , SNR = 20 dB	82
	ness measures of 8 systems for NLMS, PNLMS, SC-PNLMS, IPNLMS and	
	SC-IPNLMS.	82

3.14	Relative convergence of NLMS, IPNLMS and SC-IPNLMS using speech in-	
	put signal with echo path changes at $58 \text{ s.}$ Impulse response is changed from	
	that shown in Fig. 1.2 (a) to (b) and $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ , $\mu_{\text{SC-IPNLMS}}$	
3.16	= 0.8, $SNR = 20 \text{ dB}$ Time to reach the -20dB normalized misalignment against different sparse-	83
3.17	ness measures of 8 systems for NLMS, MPNLMS and SC-MPNLMS Sparse impulse responses, sampled at 8 kHz, giving (a) $\xi(n) = 0.88$ and	83
3.18	(b) $\xi(n) = 0.85$ respectively	84
	SC-IPNLMS using WGN input signal with an echo path change at $3.5$ s.	
	Impulse response is changed from that shown in Fig. $3.17$ (a) to (b) and	
3.19	$\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3,  \mu_{\text{SC-IPNLMS}} = 0.7,  \text{SNR} = 20  \text{dB.} \dots$ Sparseness measure of the generated impulse responses using the method	85
	of image with $L = 1024$ against iteration number (t) and the generated	
3.20	impulse responses at $t = 200,4000$ and 8000, respectively Relative tracking performances of NLMS, PNLMS and SC-PNLMS, using	86
3.21	WGN input signal, under a time-varying unknown system	87
3.22	using WGN input signal, under a time-varying unknown system Relative tracking performances of NLMS, IPNLMS and SC-IPNLMS, using	88
3.23	WGN input signal, under a time-varying unknown system Convergence of IPMDF for different values of $\alpha_{\text{IPMDF}}$ using WGN input	88
	signal. Impulse responses in Fig. $1.2$ (a) and (b) are used as sparse and	
3.24 3.25	dispersive AIRs respectively. $[\tau = 0.2, \mathcal{K} = 8, \text{SNR} = 20 \text{ dB}] \dots$ Variation of $\alpha_{\text{IPMDF}}$ for minimum $T_{20}$ against AIRs with different sparseness. Schematic of the Sparseness-controlled improved proportionate Multidelay	90 92
3.26	filtering (SC-IPMDF) algorithm	93
	signal with an echo path changes at 8 s and 16 s with $\tau = 0.2$ , $\mathcal{K} = 8$ ,	
	$\mathrm{SNR}=20~\mathrm{dB}.$ The dispersive and sparse AIRs are as shown in Fig. 1.2 (b)	
	and Fig. 1.2 (a) respectively	94

3.27	Relative convergence of MDF, IPMDF and SC-IPMDF using speech input
	signal with an echo path changes at 9.5 s and 19 s with $\tau = 0.2$ , $\mathcal{K} = 8$ ,
	SNR = 20  dB. The dispersive and sparse AIRs are as shown in Fig. 1.2 (b)
	and Fig. 1.2 (a) respectively. $\dots \dots 95$
4.1	Normalized misalignments (NM) of IPNLMS with different mixing param-
4.2	eters, $\alpha$ , for identification of a sparse impulse response
4.3	eters, $\alpha$ , for identification of a dispersive impulse response
	non-proportionate and proportionate weight allocation approaches, using
	WGN input signal with an echo path change at 4 s.Impulse response is
4.4	changed from that shown in Fig. 1.3 (a) to (b) and $\mu = 0.3$ , SNR = 20 dB. 107 The normalized misalignments (NM) for the overall convergence perfor-
4.5	mance illustrated in Fig. 4.3
	algorithm with $\alpha = -1$ and $\alpha = 0.9$ and 0.5 for the non-proportionate
4.6	PB-IPNLMS algorithm
	with non-proportionate and proportionate weight allocation approaches,
	using the input signal generated by $(4.11)$ with an echo path change at
	4 s. Impulse response is changed from that shown in Fig. 1.3 (a) to (b) and
4.7	$\mu = 0.3$ , SNR = 20 dB
	$\chi$ in (4.12) for PB-IPNLMS using the input signal generated by (4.13). Im-
	pulse response in Fig. $1.3$ (b) and (c) were used as sparse AIR and dispersive
4.8	AIR respectively. $[\mu = 0.3, \text{SNR} = 20 \text{ dB}] \dots \dots$
	$\xi(\mathbf{h}), \xi(\mathbf{h}_1)$ and $\xi(\mathbf{h}_2)$ respectively denote the sparseness measures [3] of
	the full impulse response, the first block with size $L_1$ and the second block. 112

4.9	a) Relative convergence of IPNLMS for $\alpha = -1$ and 0.9, PB-IPNLMS with
	proportionate weight allocation technique and VLPB-IPNLMS; <b>b</b> ) evolu-
	tion of $\frac{\ \hat{\mathbf{h}}_1(n)\ _1}{\ \hat{\mathbf{h}}(n)\ _1}$ ; <b>c)</b> evolution of $L_1$ ; The input signal was generated us-
	ing $(4.13)$ with an echo path change at 10 s and 20 s, from that shown in
	Fig. 4.8 (a) to (b) and then to (c), $\mu = 0.3$ and SNR = 20 dB
5.1	Sparseness measure of the generated impulse responses using the modified
	Markov model with $L = 1024$ , $\sigma_s^2 = 1$ and $\beta = 0.9999$ , against iteration
	number $(n)$ and generated impulse responses at $n = 0,1000$ and 8000,
5.2	respectively
5.3	$\sigma_w^2 = 10^{-6}, \sigma_s^2 = 1, \delta = \delta_{ip} = 10^{-4}.$
	ing step-sizes $\mu$ with $\varepsilon = 1 - 10^{-9}$ , $\alpha = -0.75$ , $\sigma_x^2 = 10^{-3}$ , $\sigma_w^2 = 10^{-6}$ ,
5.4	$\sigma_s^2 = 1,  \delta = \delta_{ip} = 10^{-4}$ 131 MSE of IPNLMS in dispersive and time-variant system identification for
	varying $\mu$ with $\varepsilon = 1 - 10^{-7}$ , $\alpha = -0.75$ , $\sigma_x^2 = 10^{-3}$ , $\sigma_w^2 = 10^{-6}$ , $\sigma_s^2 = 1$ ,
5.5	$\delta = \delta_{ip} = 10^{-4}$
	ing $\sigma_x^2$ with $\varepsilon = 1 - 10^{-8}$ , $\alpha = -0.75$ , $\mu = 0.7$ , $\sigma_w^2 = 10^{-6}$ , $\sigma_s^2 = 1$ ,
5.6	$\delta = \delta_{ip} = 10^{-4}$
	varying $\sigma_x^2$ with $\varepsilon = 1 - 10^{-7}$ , $\alpha = -0.75$ , $\mu = 0.7$ , $\sigma_w^2 = 10^{-6}$ , $\sigma_s^2 = 1$ ,
5.7	$\delta = \delta_{ip} = 10^{-4}$
	ing $\sigma_w^2$ with $\varepsilon = 1 - 10^{-9}$ , $\alpha = -0.75$ , $\mu = 0.7$ , $\sigma_x^2 = 10^{-3}$ , $\sigma_s^2 = 1$ ,
5.8	$\delta = \delta_{ip} = 10^{-4}$
	varying $\sigma_w^2$ with $\varepsilon = 1 - 10^{-7}$ , $\alpha = -0.75$ , $\mu = 0.7$ , $\sigma_x^2 = 10^{-3}$ , $\sigma_s^2 = 1$ ,
	$\delta = \delta_{ip} = 10^{-4}.$ 134
6.1	Illustration of an acoustic FIR SIMO system

- 6.2 Sparse (left) and Dispersive (right) acoustic impulse responses of a singleinput three-output system used in the simulation for blind identification. . . 150
- 6.3 Relative convergence of MC-APA, MC-SCP-APA and MC-SCMP-APA using WGN input signal with an echo path change at 4 s. Impulse response is changed from that shown from Fig. 6.2 (left) to (right) and  $\mu_{\text{MC}-\text{APA}} =$  $\mu_{\text{MC}-\text{SCP}-\text{APA}} = \mu_{\text{MC}-\text{SCMP}-\text{APA}} = 0.2$ , P = 2, SNR = 50 dB. . . . . . . 151
- 7.1 Acoustic impulse responses obtained using the method of images [1]. . . . . 157

# List of Tables

2.1	Complexity of algorithms' coefficients update - Addition, Multiplication,
	Division, Logarithm (Log) and Comparison
2.2	Computational complexity of FLMS, MDF and IPMDF for $\mathcal{K}=1$ 61
2.3	The NLMS, PNLMS, MPNLMS and IPNLMS Algorithms 62
2.4	The FLMS Algorithm
2.5	The MDF and IPMDF Algorithm
3.1	Complexity of algorithms' coefficients update - Addition, Multiplication,
	Division, Logarithm (Log) and Comparison
3.2	Computational complexity of MDF, IPMDF and SC-IPMDF 96
3.3	The Sparseness-controlled time domain Algorithms
3.4	The SC-IPMDF Algorithm
4.1	The Partitioned Block IPNLMS algorithms

# Abbreviations

AEC:	Acoustic echo cancellation
AIR:	Acoustic impulse response
APA:	Affine projection algorithms
AR:	Autoregressive
BSI:	Blind system identification
CPNLMS:	Composite proportionate NLMS
CR:	Cross relation
DFT:	Discrete Fourier transform
DTD:	Double talk detector
ERLE:	Echo return loss enhancement
FFT:	Fast Fourier transform
FIR:	Finite impulse response
FLC:	Filter length control
FLMS:	Frequency-domain fast-LMS
GNGD:	Generalized normalized gradient descent
HOS:	Higher-order statistics
IDFT:	Inverse discrete Fourier transform
IFT:	Inverse Fourier transform
<b>IPMCMDF:</b>	Improved proportionate multichannel MDF algorithm
IPMDF:	Improved proportionate MDF algorithm
IPNLMS:	Improved proportionate normalized least-mean-square
<b>IPNMCFLMS:</b>	Improved proportionate normalized multichannel FLMS algorithm

LMS:	Least-mean-square
LRMS:	Loudspeaker-room-microphone system
MCLMS:	Multichannel LMS algorithm
MDF:	Multidelay filtering
MIMO:	Multiple-input multiple-output
MPNLMS:	$\mu\text{-}\text{law}$ proportionate normalized least-mean-square
MSE:	Mean square error
MC-APA:	Multichannel affine projection algorithm
MC-PAPA:	Multichannel proportionate affine projection algorithm
MC-SCP-APA:	Multichannel sparseness-controlled proportionate APA
MC-SCMP-APA:	Multichannel sparseness-controlled $\mu\text{-law}$ proportionate APA
NEC:	Network echo cancellation
NLMS:	Normalized least-mean-square
NM:	Normalized misalignment
NMCFLMS:	Normalized multichannel freq. domain LMS algorithm
NPM:	Normalized projection misalignment
PAPA:	Proportionate affine projection algorithm
<b>PB-IPNLMS</b> :	Partitioned block improved proportionate NLMS
PNLMS:	Proportionate normalized least-mean-square
PNLMS++:	Proportionate normalized least-mean-square++
PSTN:	Public switched telephone network
RLS:	Recursive least-square algorithm
SAF:	Sub-band adaptive filtering
SC-IPMDF:	Sparseness-controlled improved proportionate MDF
SC-IPNLMS:	Sparseness-controlled improved proportionate NLMS
SC-MPNLMS:	Sparseness-controlled $\mu\text{-law}$ proportionate NLMS
SC-PNLMS:	Sparseness-controlled proportionate NLMS
SIMO:	Single-input multiple-output

SPNLMS:	Segment proportionate NLMS
SNR:	Signal-to-noise ratio
SOS:	Second-order statistics
VLPB-IPNLMS:	Variable length partitioned block IPNLMS
VoIP:	Voice over internet protocol
VSS:	Variable step-size
WGN:	White Gaussian noise

## Mathematical Symbols

#### Operators

- $[.]^T$ : Non-conjugate matrix transpose
- $[.]^{-1}$ : Matrix inverse
- $[.]^*$ : Complex conjugate operation
- |.| : Absolute operator
- [.]: Ceiling operator
  - \*: Convolution operator
- $\odot$ : Hadamard product operator
- $diag\{.\}$ : Diagonal operator
  - $E\{.\}$ : Expectation operator
  - $e^{(.)}$ : Exponential function
  - $erf\{.\}$ : Error function
  - $\log(.)$ : Logarithmic function
- $\max\{.\}$ : Maximum function
  - $tr\{.\}$ : Trace operator
  - $var\{.\}$ : Variance operator
    - $\|.\|_p: \ell_p$ -norm
      - $\nabla: \quad \text{Gradient vector} \quad$
      - $\widehat{\nabla}$ : Estimation of a gradient vector
    - $\frac{\partial(.)}{\partial(.)}$ : Partial derivative

#### General notations

- x: Scalar quantity
- $\mathbf{x}$ : Vector quantity
- **X** : Matrix quantity
- x(n): Function of a discrete variable at time index n
- $\mathbb{X}(m)$ : Time-domain vector at  $m^{th}$  frame index
- $\underline{\mathbb{X}}(m)$ : Frequency-domain vector at  $m^{th}$  frame index

#### Symbols and Variables

- $[.]_{m \times n}$ : Matrix of dimension m rows and n columns
- $\mathbf{0}_{m \times n}$ : Null matrix of dimension m rows and n columns
- $\mathbf{F}_{2L \times 2L}$ : Fourier matrix of dimension 2L rows and 2L columns
  - $\mathbf{I}_{m \times m}$ : Identity matrix of dimension *m* rows and *m* columns
    - *a* : Distance between a loudspeaker and a microphone
  - $b_m(n)$ : Additive noise at the  $m^{th}$  channel in the SIMO system
  - e(n): a priori error
  - $e_{ij}(n)$ : a priori cross relation error between channel i and j
  - $e_p(n)$ : a posteriori error
    - $f_s$ : Sampling frequency
  - $\mathbf{h}(n)$ : True room impulse response
  - $\widehat{\mathbf{h}}(n)$ : Estimated room impulse response
  - $\widehat{\mathbf{h}}_{j}(n): j^{th}$  partitioned block of the estimated impulse response in Chap.4 and
    - : Estimated impulse response of the  $j^{th}$  channel in Chap.6
  - $\mathcal{J}(n)$ : Cost function
    - $\mathcal{K}$ : Number of blocks in MDF structure
    - k: Block index for the MDF structure
    - L: Length of adaptive filter
    - $L_1$ : Block length of the partitioned block IPNLMS

- M: Number of channels in the SIMO system
- m: Frequency-domain frame index
- N: Block length of multi-delay filtering (MDF) structure
- n: Time-domain sample index
- P: Projection order
- $\mathbf{p}: \quad \mathrm{Cross-correlation} \ \mathrm{vector}$
- $\mathbf{Q}(n)$ : Diagonal step-size control matrix
- $\mathbf{Q}_{j}(n)$ : Diagonal step-size control matrix of  $j^{th}$  partitioned block
- $q_l(n)$ : Diagonal elements of  $\mathbf{Q}(n)$ 
  - **R** : Auto-correlation matrix
- $\mathbf{s}(n)$ : Additive noise in the AR and Markov models
- w(n): Additive noise in LRMS
- $\mathbb{X}(m)$ : Tap input of  $m^{th}$  frame in time-domain
- $\underline{\mathbb{X}}(m)$ : Tap input of  $m^{th}$  frame in frequency-domain
- $\mathbf{x}(n)$ : Tap input vector
- $x_m(n)$ :  $m^{th}$  sensor signal in the SIMO system
  - y(n): Receiving room microphone signal
  - $\widehat{y}(n)$ : Adaptive filter output
  - $\mathbf{z}(n)$ : Weight deviation vector
    - $\alpha$ : Control parameter for the IPNLMS based algorithms
    - $\alpha_i$ : Control parameter of the  $j^{th}$  block for PB-IPNLMS
  - $\beta(n)$ : Proportionality control factor for PB-IPNLMS
    - $\chi$ : Proportionality constant in  $\beta(n)$
    - $\delta$ : Regularization parameter
    - $\epsilon$  : Vicinity value for MPNLMS and SC-MPNLMS
    - $\gamma$ : Parameter to prevent coefficients from stalling when they're smaller
    - $\kappa$  : Threshold value for proportionate weighting PB-IPNLMS
  - $\kappa_{\min}:~$  Minimum threshold value for VLPB-IPNLMS

- $\kappa_{\rm max}$  : Maximum threshold value for VLPB-IPNLMS
  - $\lambda$ : Exponential decaying factor for SC-PNLMS and SC-MPNLMS
  - $\mu$ : Step-size parameter
  - $\nu$ : Compensation factor for MPNLMS and SC-MPNLMS
  - $\rho$ : Parameter to prevent coefficients from stalling at initialization
- $\rho(n):~$  Time-dependent of  $\rho$  for SC-PNLMS and SC-MPNLMS
  - $\sigma_s^2$  : Variance of the added noise in AR and Markov models
  - $\sigma_w^2$ : Variance of the additive noise
  - $\sigma_x^2$ : Variance of input signal
  - $\varepsilon$ : Control parameter for Markov model
- $\vartheta_{\max}$ : The largest eigen value of  ${f R}$
- $\xi(\mathbf{h})$ : Sparseness measure of the vector  $\mathbf{h}$
- $\widehat{\xi}(n)$ : Sparseness measure of the vector  $\widehat{\mathbf{h}}(n)$ 
  - $\zeta$  : Forgetting factor in the normalization of freq-domain tap-update
  - $\Delta \ell$ : Number of coefficients by which  $L_1$  gets enlarged/reduced

### Chapter 1

### Introduction

#### 1.1 Overview

WIRELESS phones are increasingly being regarded as essential communications tools due to their flexibility. As the use for the in-car hands free telephony has gained much popularity in recent years due to the rise in safety concerns, and also the need for an automated service delivery system, digital wireless subscribers are becoming ever more critical of the voice quality they receive from network providers. One factor that affects the voice quality is echo.

An echo is said to occur when delayed and possibly distorted versions of a signal are reflected back to the source of that signal. This delayed replica is only noticeable if the amplitude of the echo is significantly high or the time delay between the speech and the echo exceeds 16 ms (32 ms round trip) [4]. The study carried out at Bell laboratories found that echoes above 250 ms can make it impossible to have natural conversation.

Hybrid echo [5] is a type of echo generated in telephone networks comprising mixed packet-switched and circuit-switched components. Echo cancellation in telephone networks requires the identification and compensation of echo systems with various levels of sparseness. The network echo response in such systems is typically of length 64-128 ms, characterized by an unknown bulk delay dependant on network loading, encoding and jitter buffer delays [6]. This results in an 'active' region in the range of 8-12 ms duration



Figure 1.1: Illustration of acoustic echo in a loudspeaker-room-microphone system (LRMS).

and consequently, the impulse response is dominated by 'inactive' regions where coefficient magnitudes are close to zero, making the impulse response sparse.

Acoustic echo is the other type of echo which is produced by strong voice coupling between the earpiece and microphone in handsets and hands-free devices [7, 8]. The length of the acoustic echo response in a typical teleconferencing room is in the region of 100 to 400 ms and hence adaptive filters employing 1024 taps or more (at 8 kHz of sampling frequency) are typically required in order to achieve adequate levels of echo cancellation [9]. As shown in Fig. 1.1, a signal,  $\mathbf{x}(n)$ , from a loudspeaker is heard by a listener, as intended. However, this same sound also is picked up by the microphone, both directly and indirectly, after bouncing off the wall. The result of this reflection is the creation of echo which is transmitted back to the far end and is heard by the talker as echo. A typical office or living room exhibits reverberation time in the order of 50 to 300 ms [10].

The time variation of the near-end acoustic impulse response (AIR), in the acoustic echo cancellation (AEC) system may arise due to, for example, a change in temperature [11], pressure and changes in the acoustic environment [12]. Therefore, adaptation of the filter coefficients once, at the beginning of communication and a subsequent freezing of the filter coefficients is not sufficient for a permanent echo cancellation. Since the filter should be able to track the variations of the echo system, it should be adaptive.

In the earlier days of telecommunications, echo suppression was used for echo cancellation in satellite communication [5]. In essence these devices rely upon the fact that most telephone conversations are half-duplex. That is one person speaks while the other listens. Nowadays, adaptive filters are used in echo cancellers which model and subtract the echo from the return path and therefore, outperformed the suppression-based devices by aiding full-duplex capabilities (parties at both ends can speak simultaneously). The adaptive filtering technique is crucial for many other applications in the field of telecommunication, such as noise cancellation and channel equalization [13]. Although adaptive filters can be used when the echo path is initially unknown, their application is unavoidable when facing time-varying environments.

For the adaptation of the filter coefficients several families of adaptive algorithms such as the recursive least-square algorithm (RLS) [13, 14, 15], the affine projection algorithm (AP) [16, 17, 18], or the normalized least-mean-square (NLMS)-based algorithms [13] can be applied. They differ in their iterative updating scheme, their computational complexity and their convergence speed [19]. All these algorithms compute the new filter coefficients (at time sample n, for example) by correcting the old estimation (at time sample n-1) with an innovation vector weighted by a step-size. Details of the algorithmdependent innovation vector will be addressed in the next chapter for the NLMS-based algorithms and the former two families of the adaptive algorithms will not be addressed in this thesis.

In hands-free systems, reverberation also affects the quality and intelligibility of speech and is a significant problem for speech recognition applications [20]. Dereverberation is therefore another important speech enhancement process for hands-free terminals, other than the echo cancellation process. Although many approaches [21] have been developed for speech dereverberation, blind system identification (BSI), based on the cross relation (CR), will be addressed in this thesis, as BSI is believed to be the key to thoroughly solving the dereverberation problem [22].

#### **1.2** Research objectives

Variation in the sparseness of AIRs can also occur in AEC within an enclosed space. The problem can be formulated by considering an example case illustrated in Fig. 1.2, where the distance, a, between a loudspeaker and the user using, for example, a wireless microphone is varying. It shows two AIRs, generated using the method of images [1, 23] using room dimensions of  $8 \times 10 \times 3$  m and 0.57 as the reflection coefficient. The loudspeaker is fixed at  $4 \times 9.1 \times 1.6$  m in the loudspeaker-room-microphone system (LRMS) while the microphone is positioned at  $4 \times 8.2 \times 1.6$  m and  $4 \times 1.4 \times 1.6$  m giving impulse responses as shown in Fig. 1.2 (a) and (b) for a = 0.9 m and a = 7.7 m respectively. As can be seen, the sparseness of these AIRs varies significantly with the loudspeaker-microphone distance. Hence, algorithms developed for mobile hands-free terminals are required to be robust to the variations in the sparseness of the acoustic path.

The partitioned block technique is another useful approach to consider in order to improve the convergence and tracking performances of the existing AEC algorithms. For sparse and dispersive AIRs, the partitioned block of the echo path that consists of the direct path and a few early reflections is almost always sparse while the other partitioned block is always dispersive. To validate this, consider an example case where two AIRs of length L = 1024 were simulated using the method of images [1], under the same experimental set-up as before. Figure 1.3 (a) shows the AIR obtained when the loudspeaker-microphone distance is 0.85 m in the LRMS with 0.3 reflection coefficient. Figure 1.3 (b) illustrates the AIR attained when the loudspeaker-microphone distance is 5 m in the LRMS with 0.53 reflection coefficient. As can be seen from the figure and the sparseness measure [2, 3], the first block is always more sparse than the second block. Hence, a sparse algorithm is desired for the first block, whereas a non-sparse algorithm is desired for the second block. Moreover, due to the nature of the time-varying environments, which is commonly encountered in the LRMS, the length of the partitioned blocks should be made adaptive to exploit the full use of this technique.

It is pragmatically useful to study an insight of how the performances of adaptive algorithms, in particular the gradient based algorithms, are affected by conditions such as



Figure 1.2: Loudspeaker-room-microphone system (LRMS) and two acoustic impulse responses, generated using the method of images, for the cases when the separation is 0.9 m and 7.7 m.

the degree of variation of the unknown system and the step-size used for adaptation in single channel AEC. Developing a framework under time-varying unknown system conditions is therefore reputable, as it can be used to predict sensible values for the designer adjustable parameters depending on the specific needs of the application.

Motivated by the time-varying nature of the AIR within the enclosed environment, adaptive algorithms developed for blind single-input multiple-output (SIMO) system identification problems should also necessitated to be robust to the variations in the sparseness of the multiple acoustic paths. The forthcoming chapters of this thesis will concentrate on:

• sparseness-controlled techniques to develop both the time and frequency domains



Figure 1.3: Acoustic impulse responses obtained using the method of images [1].  $\xi(\mathbf{h}), \xi(\mathbf{h}_1)$  and  $\xi(\mathbf{h}_2)$  respectively denote the sparseness measures [2, 3] of the full impulse response, the first block with size of 256 and the second block.

AEC algorithms to improve the convergence limitations on the well-known existing adaptive algorithms.

- partitioned block technique, with adaptive control for the lengths of the partitionedblock AIR, to improve the performance of any AEC algorithms compared to that without the partitioned block technique.
- a framework to predict the tracking performance for IPNLMS for a time-varying echo system, so that, for example, one can choose an appropriate value for the step-size depending on their satisfactory level for the (misadjustment) error.
- a class of time domain sparseness-controlled affine projection algorithms for blind SIMO system identification based on second-order statistics which exploit the cross relation method.

#### 1.3 Thesis structure

The chapters of this thesis are organized as follows. Chapter 2 starts with formulating the AEC process using an adaptive filter with finite impulse response (FIR) structure and the necessary assumptions made for simplicity and mathematical tractability. The main time

domain adaptive algorithms for AEC and NEC are reviewed. This includes the NLMS algorithm, the proportionate normalized least-mean-square (PNLMS) algorithm [24], the  $\mu$ -law proportionate NLMS (MPNLMS) algorithm [25] and the improved proportionate NLMS (IPNLMS) algorithm [26]. Frequency domain adaptive algorithms, in particular the fast-LMS (FLMS) algorithm [27], the multidelay filtering (MDF) structure [28] and the improved proportionate MDF (IPMDF) algorithm [29] are reviewed.

Chapter 3 presents a novel class of time domain and a new frequency domain algorithms that are robust to the sparseness variation of AIRs. These algorithms compute a sparseness measure of the estimated impulse response at each iteration of the adaptive process and incorporate it into their conventional methods. As will be shown, the proposed sparseness-controlled algorithms achieve fast convergence for both sparse and dispersive AIRs and are effective for time-varying AEC.

In Chapter 4, block partitioning is proposed as a novel technique. A partitioned block IPNLMS algorithm with a control mechanism for the dynamic adjustment of the block size is developed and the convergence performance for identification of a time-varying echo system is compared with the classical IPNLMS without block partitioning.

In chapter 5, following the approach as presented in [30], the tracking performance of adaptive algorithms under time-varying unknown system conditions is analyzed. A general framework is developed such that the analysis can be applied to the NLMS-based algorithms. The aim of this analysis is to provide an insight of how the performances of such adaptive algorithms, in particular IPNLMS, are affected by conditions such as the degree of variation of the unknown system in single channel AEC. The proposed framework is evaluated in detail by comparing the theoretical and experimental performances of IPNLMS.

The focus of this thesis then moves to the problem of blind identification of singleinput multiple-output (SIMO) acoustic systems. Chapter 6 begins with an introduction to the blind SIMO identification and the existing main algorithms in the literature. Later, it presents a more rigorous way to derive proportionate affine projection algorithms (PAPA). The sparseness-controlled techniques developed in Chapter 3 is then exploited into the novel framework for the individual step-size control, in order to improve the robustness in the sparseness variation of AIRs. The simulated results, with a short channel length, for the proposed sparseness-controlled algorithms show fast convergence in both sparse and dispersive AIRs.

The thesis is concluded and further work is discussed in Chapter 7.

#### **1.4** Statement of originality

As far as the author is aware, the following aspects of the thesis are believed to be original contributions:

- Investigation into the variation in the sparseness of AIRs in AEC within an enclosed space, by varying the distance between a fixed loudspeaker and a moving wireless microphone.
- 2. Development of a class of sparseness-controlled time domain adaptive algorithms for AEC application which is robust to the level of sparseness encountered in the impulse response of the echo path.
- 3. Development of a frequency domain algorithm for AEC which dynamically adjusts its step-size according to the sparseness variation in AIR that arises in a mobile environment.
- 4. Investigation into the variation in sparseness measure of the early part (i.e., direct path and early reflections) of the acoustic echo path and the late reverberant part of the acoustic path, regardless of the overall sparseness measure.
- 5. Implementation of a partitioned block IPNLMS algorithm with a self-configuration method based on the ratio between the  $\ell_1$ -norms of the two partitioned blocks.
- 6. Implementing a generalized framework for the analysis on tracking performance of NLMS based algorithms under both non-stationary and stationary unknown system conditions in single channel AEC.
7. Development of a class of sparseness-controlled proportionate affine projection algorithms for blind SIMO system identification.

## 1.5 Publications

The following publications were produced during the course of this work:

- Journal paper
  - P. Loganathan, A. W. H. Khong and P. A. Naylor, "A Class of Sparsenesscontrolled Algorithms for Echo Cancellation", *IEEE Transactions on Speech*, *Audio and Language Processing*, vol. 17, no. 8, pp. 1591-1601, Nov 2009.
- Conference proceedings
  - P. Loganathan, E. A. P. Habets and P. A. Naylor, "A Proportionate adaptive algorithm with variable partitioned block length for acoustic echo cancellation", *The 36th International Conference on Acoustics, Speech, and Signal Processing* (ICASSP), Prague, Czech Republic, May 2011.
  - P. Loganathan, E. A. P. Habets and P. A. Naylor, "A Partitioned Block Proportionate Adaptive Algorithm for Acoustic Echo Cancellation", *Proceedings of Asia-Pacific signal and Information Processing Association (APSIPA) Annual Summit and Conference*, Biopolis, Singapore, Dec 2010.
  - P. Loganathan, E. A. P. Habets and P. A. Naylor, "Performance Analysis of IPNLMS for Identification of Time-Varying Systems", *The 35th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Dallas, USA, March 2010.
  - P. Loganathan, X. S. Lin, A. W. H. Khong and P. A. Naylor, "Frequency domain Adaptive Multi-delay Algorithm with Sparseness-control for Acoustic Echo Cancellation", Proc. European Signal Processing Conf. (EUSIPCO), Glasgow, Scotland, August 2009.

 P. Loganathan, A. W. H. Khong and P. A. Naylor, "A sparseness-controlled proportionate algorithm for acoustic echo cancellation", *Proc. European Signal Processing Conf. (EUSIPCO)*, Lausanne, Switzerland, August 2008.

# Chapter 2

# Literature Review

## 2.1 Overview of adaptive algorithms for echo cancellation

**T**RADITIONALLY, echo cancellers are realized by a FIR structure to achieve echo cancellation using an adaptive algorithm. The NLMS algorithm is a popular choice due to its simplicity, both in terms of computational load and easiness of implementation. It has, therefore, been successfully applied to a wide variety of adaptive filtering problems, including plant identification [31] and noise cancellation applications [32]. A generalized normalized gradient descent (GNGD) algorithm [33] was proposed as an extension of the NLMS, where an additional stabilization and faster convergence were introduced by making the compensation term in the normalization of the NLMS step-size gradient adaptive. However, for sparse systems such as encountered in NEC, the NLMS algorithm suffers from slow convergence and therefore new algorithms have been proposed in the literature for sparse adaptive filtering.

Several approaches have been proposed over recent years to improve the performance of the standard NLMS algorithm in various ways for NEC. These include the variable step-size (VSS) algorithms [34, 35, 36], data reusing technique [37, 38, 39], partial update adaptive filtering techniques [40, 41, 42] and sub-band adaptive filtering (SAF) schemes [43, 44, 45]. The VSS algorithm [34] improves the performance of the adaptive algorithm by employing larger step-size at the beginning of the adaptation, for fast initial convergence, and a smaller step-size during later stage of adaptation, in order to reduce the tradeoff between misadjustment and tracking ability of the fixed step-size LMS algorithm. Data reusing is another technique which was introduced to achieve improvement in convergence rate. This approach reuses the current desired response and data vector repeatedly to update the adaptive tap-weight vector several times during each iteration. Partial update algorithms are proposed to reduce the computational complexity of an adaptive filter by updating only a subset of filter coefficients for each iteration based on a selection criteria. SAF has also been introduced in AEC to achieve complexity reduction whilst achieving an improved rate of convergence compared to the conventional full-band structure. In contrast to these approaches, sparse adaptive algorithms have been developed specifically to address the performance of adaptive filters in sparse system identification.

In this thesis, attention is devoted to sparse adaptive algorithms in time and frequency domains, because of their ease of implementation and moreover the framework can be applied to most of the aforementioned approaches.

The idea of exploiting the sparse character of echo paths has appeared in [46, 47, 48]. However, one of the first sparse adaptive filtering algorithms considered as a milestone for NEC is PNLMS [24] in which each filter coefficient is updated with an independent step-size that is linearly proportional to the magnitude of that estimated filter coefficient. It is well known that PNLMS has very fast initial convergence for sparse impulse responses after which its convergence rate reduces significantly, sometimes resulting in a slower overall convergence than NLMS. In addition, PNLMS suffers from slower convergence compared to NLMS when estimating dispersive impulse responses [49, 50]. To address the latter problem, subsequent improved versions, such as PNLMS++ [49], were proposed. The PNLMS++ algorithm achieves improved convergence by alternating between NLMS and PNLMS for each sample period. However, as shown in [26], the PNLMS++ algorithm only performs best in the cases when the impulse response is sparse or highly dispersive.

An IPNLMS [26] algorithm was proposed to exploit the 'proportionate' idea by introducing a controlled mixture of proportionate (PNLMS) and non-proportionate (NLMS) adaptation. A sparseness-controlled IPNLMS (SC-IPNLMS) algorithm was proposed in [2] to improve the robustness of IPNLMS to the sparseness variation in impulse responses. Composite proportionate NLMS (CPNLMS) [51] adaptation was proposed to control the switching of PNLMS++ between the NLMS and PNLMS algorithms. For sparse impulse responses, CPNLMS performs the PNLMS adaptation to update the large coefficients and subsequently switches to NLMS, which has better performance for the adaptation of the remaining small taps. The MPNLMS [25, 52] algorithm was proposed to address the uneven convergence rate of PNLMS during the estimation process. As proposed in [25], MPNLMS uses optimal step-size control factors to achieve faster overall convergence until the adaptive filter reaches its steady state.

The main limitation of all these adaptive algorithms is that their performances are subject to a tradeoff between the speed of convergence and high precision. Algorithms with higher step-size achieve faster convergence, but the mismatch between the true system and the predicted system is worse compared to that with smaller step-size. To overcome this tradeoff, a combination framework was proposed in [53, 54], which adaptively combines two independent least-mean-square (LMS) filters with large and small step-sizes to obtain fast convergence with low mis-adjustment.

These time domain algorithms have also been proposed in the frequency [55, 27, 56] and wavelet [57, 58] domains. These frequency domain adaptive algorithms have become popular because of their efficient implementation, compared to the above time domain algorithms. They perform computations by incorporating block updating strategies, rather than performing sample-by-sample computations. In addition, exploiting the computational efficiency of the fast Fourier transform (FFT) for computing the discrete Fourier transform (DFT), so as to perform linear convolution and gradient estimation, further increases the efficiency of such algorithms. They also use the pseudo-orthogonality property of the DFT [59] to speed up the convergence rate.

The concept of frequency domain adaptive filtering was first introduced in [55]. The fast-LMS (FLMS) algorithm [27] was proposed, where the overlap-save method [60] for implementing linear convolution using FFT [61] blocks is employed to avoid the effects of circular convolution encountered in the direct implementation of the frequency domain LMS algorithm [62, 63].

Although substantial computational savings can be achieved, one of the drawbacks, however, is the delay introduced between the input and output, which is equivalent to the length of the adaptive filter L. For AIRs with several hundreds of coefficients, this delay can be significant. To overcome this, the MDF algorithm was proposed in [28] to partition the adaptive filter into blocks each of length N such that the delay is reduced by a factor of  $\mathcal{K} = L/N$ , although  $\mathcal{K} = 1$  is the optimum choice in terms of computational complexity. Combining proportionate updating of filter coefficients, the improved proportionate MDF (IPMDF) algorithm [29] achieves a fast convergence with a low delay for  $\mathcal{K} > 1$ in NEC, and a similar improvement has also been shown in [64] for blind estimation of multichannel AIRs.

Although sparse adaptive filtering algorithms, such as those described above, have originally been developed for NEC, it has been shown in [64] that such algorithms give good convergence performance in the AEC system. The tracking capabilities of these sparse NLMS-based algorithms can also be exploited to cope with the time-varying nature of AIRs.

## 2.2 Acoustic echo cancellation

The source of acoustic echo originates from the acoustic coupling between the microphone and loudspeaker. As its name suggests, an acoustic echo canceller attempts to cancel, rather than suppresses, the acoustic echo. Figure 2.1 shows a LRMS describing a typical AEC system, with an echo canceller employing an adaptive filter.

#### 2.2.1 Notations and definitions

An adaptive FIR filter with coefficients

$$\widehat{\mathbf{h}}(n) = [\widehat{h}_0(n)\widehat{h}_1(n)\dots\widehat{h}_{L-1}(n)]^T, \qquad (2.1)$$



Figure 2.1: Adaptive system for acoustic echo cancellation in a loudspeaker-roommicrophone system (LRMS).

is deployed to cancel acoustic echo, where L is the length of the adaptive filter assumed to be equal to the unknown room impulse response  $\mathbf{h}(n)$ , defined by

$$\mathbf{h}(n) = [h_0(n) \ h_1(n) \ \dots \ h_{L-1}(n)]^T, \tag{2.2}$$

and  $[\cdot]^T$  is the transposition operator. The time-varying far-end signal x(n) is transmitted to the near-end loudspeaker in the LRMS. The microphone in the near-end room receives the desired signal (the output of the LRMS), which is given by

$$y(n) = \mathbf{h}^T(n)\mathbf{x}(n) + w(n), \qquad (2.3)$$

where  $\mathbf{x}(n) = [x(n) \ x(n-1) \ \dots \ x(n-L+1)]$  and w(n) is additive noise. If no echo canceller is presented, the echo y(n) is transmitted back to the far-end with a delay.

#### 2.2.2 Assumptions

In order to simplify the mathematical derivations of algorithms without loss of generality the following assumptions are made throughout this project:

• The length of  $\mathbf{h}(n)$  is same as the length of  $\widehat{\mathbf{h}}(n)$ , which is L. In reality, the length

of the adaptive filter is often less than the receiving room impulse responses. This is due to the fact that the computational complexity of an adaptive algorithm increases monotonically with the length of the adaptive filter. Therefore, L must be long enough to achieve a low system mismatch and computational complexity.

- The noise signal in the LRMS, w(n), is additive.
- There is no near-end signal in the LRMS.
- A transversal FIR filter configuration is used due to its stability characteristics.

#### 2.2.3 Adaptive echo cancellation process

An echo canceller's objective is to estimate  $\mathbf{h}(n)$  as closely as possible at each iteration. An *a posteriori* error signal  $e_p(n)$ , can be computed by subtracting the output of the echo canceller  $\hat{y}(n)$  from the desired signal y(n), given by

$$e_p(n) = y(n) - \widehat{y}(n)$$
  
=  $y(n) - \widehat{\mathbf{h}}^T(n)\mathbf{x}(n)$   
=  $\left[\mathbf{h}^T(n) - \widehat{\mathbf{h}}^T(n)\right]\mathbf{x}(n) + w(n).$  (2.4)

Note that the *a posteriori* error  $e_p(n)$  in (2.4) is computed after the adaptive filter coefficients have been updated. In contrast, by using the previous estimation of the impulse response  $\hat{\mathbf{h}}^T(n-1)$ , an *a priori* error signal e(n) at each iteration is computed as

$$e(n) = y(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}(n).$$
(2.5)

For effective echo cancellation, e(n) must come significantly smaller at each iteration, as the filter coefficients converge to the unknown true impulse response  $\mathbf{h}(n)$ . The system identification performance of the echo canceller can be quantified by the misalignment which will be discussed in Section 2.3.

A common feature of the different structures proposed is that the canceller should be adaptive. This is necessary in order to track the time-varying nature of the echo path and for initial convergence since the echo path is initially unknown. The time variation of the AIR in the AEC system may arise due to a change in temperature [11] and pressure and to changes in the acoustic environment [12], for example movements of people, doors, windows or furniture. For this reason, adaptive filters are utilized to track and compensate any changes in the LRMS.

## 2.3 Performance measures

Evaluation of performance measures influences the choice of one algorithm over the wide variety of others. The commonly adopted measures will next be reviewed.

#### 2.3.1 Mean square error

The mean square error (MSE) is one of the ways to define an objective function. It is defined as the expected value of the square of the error and, as can be seen from (2.6), a lower MSE value is favorable.

MSE = 
$$E\{e^2(n)\}$$
. (2.6)

#### 2.3.2 Normalized misalignment

Normalized misalignment is one of the most commonly used performance used in the literature of system identification [13], defined by

$$\eta[\mathbf{h}(n), \widehat{\mathbf{h}}(n)] = \frac{\|\mathbf{h}(n) - \widehat{\mathbf{h}}(n)\|_2^2}{\|\mathbf{h}(n)\|_2^2},$$
(2.7)

where  $\|\cdot\|_2$  is the  $\ell_2$ -norm. It measures the closeness of an estimated system to that of the true system and is particularly useful to study the tracking capability of a time-varying system. It should be noted that this measure is applicable only for 'oracle' simulations in which the true 'unknown' system is known.

#### 2.4 Time domain adaptive algorithms for echo cancellation

As the acoustic echo path has a relatively long duration (hundreds of milliseconds for a typical office environment), a longer adaptive filter (800-1600 filter coefficients with 8 kHz sampling frequency) is required to model the unknown impulse response more closely. This results in a need for a large amount of computations and memory for the employed adaptive algorithm. Also, the echo path may vary with time due to changes in room characteristics (e.g. temperature, pressure and movement of talker). This makes tracking ability a necessary condition for the adaptive algorithm. Moreover, the non-stationary statistical nature of the speech signal ( $\mathbf{x}(n)$ ), i.e.: the eigenvalue spread of the speech signals autocorrelation matrix, causes a slow rate of convergence as compared to white Gaussian noise (WGN) with zero mean. All these combined factors demand for a robust and effective algorithm for AEC.

Several adaptive algorithms and their extensions have been proposed over the past decades for either AEC or NEC in the time domain, with the aim of increasing the rate of convergence. However, the discussion in this thesis is limited to NLMS, PNLMS, MPNLMS and IPNLMS. These algorithms will form the basis for the proposed algorithms in the next chapters.

#### 2.4.1 The LMS and NLMS algorithms

The LMS algorithm [13, 65] is an iterative formulation which solves, in the limit, the Wiener-Hopf equations recursively using a stochastic approximation to the method of steepest descent. The LMS algorithm computes the optimum coefficients of a linear filter by minimizing a statistical cost function defined as the MSE in (2.6). The following derivation shows the source of Wiener-Hopf equation and discusses how the method of steepest descent can be applied to form LMS. As this chapter deals with system identification in the application of echo cancellation, the input signal is always a real-valued speech signal. Hence, throughout the following time domain derivations, the use of conjugate operator is omitted and the non-conjugate matrix transpose  $([.]^T)$  is used instead of the Hermitian

transposition ([.]<sup>H</sup>).

By using the *a priori* error signal e(n) in (2.5), the cost function  $\mathcal{J}(\hat{\mathbf{h}}(n-1))$  yields an expression as

$$\mathcal{J}\left(\widehat{\mathbf{h}}(n-1)\right) = E\left\{e^{2}(n)\right\}$$
$$= E\left\{\left[y(n) - \widehat{\mathbf{h}}^{T}(n-1)\mathbf{x}(n)\right]\left[y(n) - \mathbf{x}^{T}(n)\widehat{\mathbf{h}}(n-1)\right]\right\}$$
$$= E\left\{y^{2}(n)\right\} - 2\mathbf{p}^{T}\widehat{\mathbf{h}}(n-1) + \widehat{\mathbf{h}}^{T}(n-1)\mathbf{R}\widehat{\mathbf{h}}(n-1), \qquad (2.8)$$

where **p** is the *L*-by-1 cross-correlation vector between y(n) and  $\mathbf{x}(n)$  defined as follows:

$$\mathbf{p} = E\left\{y(n)\mathbf{x}(n)\right\}$$
$$= E\left\{\begin{array}{c}y(n)\mathbf{x}(n)\\y(n)x(n-1)\\\vdots\\y(n)x(n-L+1)\end{array}\right\},\qquad(2.9)$$

and  $\mathbf{R}$  is the *L*-by-*L* auto-correlation matrix of the tap inputs in the transversal filter and can be defined as

$$\mathbf{R} = E\left\{\mathbf{x}(n)\mathbf{x}^{T}(n)\right\}$$

$$= E\left\{\begin{array}{cccc} x(n)x(n) & x(n)x(n-1) & \cdots & x(n)x(n-L+1) \\ x(n-1)x(n) & x(n-1)x(n-1) & \cdots & x(n-1)x(n-L+1) \\ \vdots & \vdots & \ddots & \vdots \\ x(n-L+1)x(n) & x(n-L+1)x(n-1) & \cdots & x(n-L+1)x(n-L+1) \end{array}\right\},$$
(2.10)

Thus, the MSE cost function has a quadratic form in the impulse response vector  $\hat{\mathbf{h}}$  and the minimum of the error surface can be obtained by setting the partial derivatives of  $\mathcal{J}$ , with respect to each filter coefficient, to zero. Therefore, the unique optimum impulse response is given by

$$\widehat{\mathbf{h}}_{opt} = \mathbf{R}^{-1} \mathbf{p}, \qquad (2.11)$$

which is known as Wiener-Hopf solution. This method provides minimum MSE and therefore can be used to estimate the unknown room impulse response. However, this approach is not appropriate in dealing with non-stationarity signals like speech signals and furthermore, the autocorrelation and cross-correlations are unknown.

The method of steepest decent is a gradient type iterative technique that has been employed to optimize cost functions [13]. The basic concept of the method of steepest descent is such that from an arbitrary starting point on the error performance surface (defined by (2.8)), a small step is taken in the direction where the cost function decreases fastest. The filter coefficients thus progress towards the minimum point on the error performance surface as the number of iterations increases. Thus, the definition of the filter coefficient update equation be of the form

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \frac{\mu}{2} \nabla \mathcal{J}\left(\widehat{\mathbf{h}}(n-1)\right), \qquad (2.12)$$

where  $\mu$  is a step-size and  $\nabla \mathcal{J}(\widehat{\mathbf{h}}(n-1))$  is the gradient vector of the cost function (i.e. the direction where the cost function changes fastest), which is computed from (2.8) as

$$\nabla \mathcal{J}\left(\widehat{\mathbf{h}}(n-1)\right) = \frac{\partial E\left\{e^{2}(n)\right\}}{\partial\widehat{\mathbf{h}}(n-1)}$$
$$= -2E\left\{\left[\mathbf{x}(n)\right]\left[y(n) - \mathbf{x}^{T}(n)\widehat{\mathbf{h}}(n-1)\right]\right\}$$
$$= -2\mathbf{p} + 2\mathbf{R}\widehat{\mathbf{h}}(n-1).$$
(2.13)

Thus, the filter coefficient update equation for the method of steepest descent is given by

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mu \left[ \mathbf{p} - \mathbf{R}\widehat{\mathbf{h}}(n-1) \right].$$
(2.14)

The purpose of the adaptive step-size  $\mu$  is to allow control over the rate of convergence such that a higher  $\mu$  gives a faster rate of convergence. For stability it must be lie with the range

$$0 < \mu < \frac{2}{\vartheta_{max}},\tag{2.15}$$

where  $\vartheta_{max}$  is the largest eigen-value of the auto-correlation matrix **R**.

The method of steepest descent still requires the explicit knowledge of the statistics of the input signal, according to (2.14). The filter coefficients update equation for LMS replaces the gradient vector in (2.13) with an instantaneous estimate of the gradient vector, as

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \frac{\mu}{2}\widehat{\nabla}\mathcal{J}\left(\widehat{\mathbf{h}}(n-1)\right), \qquad (2.16)$$

where  $\widehat{\nabla} \mathcal{J}(\widehat{\mathbf{h}}(n-1))$  is defined by using the instantaneous estimates for **R** and **p** that are based on sample values of  $\mathbf{x}(n)$  and y(n) [13]. This yields

$$\widehat{\nabla}\mathcal{J}\left(\widehat{\mathbf{h}}(n-1)\right) = -2\mathbf{x}(n)y(n) + 2\mathbf{x}(n)\mathbf{x}^{T}(n)\widehat{\mathbf{h}}(n-1)$$
$$= -2\mathbf{x}(n)e(n).$$
(2.17)

Hence, the filter coefficients update equation for LMS is expressed as

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mu \mathbf{x}(n)e(n).$$
(2.18)

The LMS algorithm can be considered as only an estimator of the Wiener filter due to the approximation of the gradient vector. In (2.18), the filter coefficient adjustment is directly proportional to the tap input vector,  $\mathbf{x}(n)$ . Therefore, when the  $\mathbf{x}(n)$  vector is large, the LMS suffers from a gradient noise amplification problem. To overcome this problem, the adjustment applied to the tap weight vector at each iteration can be normalized with respect to the squared Euclidean norm of  $\mathbf{x}(n)$ . This is then known as the NLMS algorithm and its filter coefficient update equation is given by

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \mu \frac{\mathbf{x}(n)e(n)}{\mathbf{x}^{T}(n)\mathbf{x}(n) + \delta_{\text{NLMS}}},$$
(2.19)

where the regularization parameter  $\delta_{\text{NLMS}} = \sigma_x^2$  (the variance of the input signal), which prevents division by zero, especially during initialization when  $\mathbf{x}(n) = \mathbf{0}$ .

The NLMS algorithm is one of the most popular for AEC due to its straightforward implementation and relatively low complexity. One of the main drawbacks of the NLMS algorithm is that its convergence rate reduces significantly when the impulse response is sparse, such as often occurs in NEC. The poor performance has been addressed by several sparse adaptive algorithms such as those described below that have been developed specifically to identify sparse impulse responses in NEC applications.

The filter coefficients update equation for many of the adaptive algorithms can be described by (2.5) and the following set of generalized equations [10]:

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{Q}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta},$$
(2.20)

$$\mathbf{Q}(n-1) = \operatorname{diag}\{q_0(n-1) \ \dots \ q_{L-1}(n-1)\}.$$
(2.21)

The diagonal step-size control matrix  $\mathbf{Q}(n)$  is introduced here to determine the step-size of each filter coefficient and is dependent on the specific algorithm. For NLMS, since the step-size is the same for all filter coefficients,

$$\mathbf{Q}(n) = \mathbf{I}_{L \times L},\tag{2.22}$$

with  $\mathbf{I}_{L \times L}$  being an  $L \times L$  identity matrix.

#### 2.4.2 The PNLMS and MPNLMS algorithm

It is important for an adaptive filter to identify rapidly the active coefficients in sparse impulse responses. The PNLMS and MPNLMS algorithms have been proposed for such sparse system identification. Diagonal elements  $q_l$  of the step-size control matrix  $\mathbf{Q}(n)$  in (2.21) for the PNLMS [24] and MPNLMS [25] algorithms can be expressed as

$$q_l(n) = \frac{\kappa_l(n)}{\frac{1}{L} \sum_{i=0}^{L-1} \kappa_i(n)}, \qquad 0 \le l \le L - 1,$$
(2.23)

$$\kappa_l(n) = \max \Big\{ \rho \times \max\{\gamma, F(|\hat{h}_0(n)|) \dots F(|\hat{h}_{L-1}(n)|)\}, F(|\hat{h}_l(n)|) \Big\}, \quad (2.24)$$

where  $F(|\hat{h}_l(n)|)$  is specific to the algorithm. The parameter  $\gamma = 0.01$  in (2.24) prevents the filter coefficients  $\hat{h}_l(n)$  from stalling when  $\hat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$  at initialization and  $\rho$ , with a typical value of 0.01, prevents the coefficients from stalling when they are much smaller than the largest coefficient.

The PNLMS algorithm achieves a high rate of convergence by employing step-sizes that are proportional to the magnitude of the estimated impulse response coefficients where elements  $F(|\hat{h}_l(n)|)$  are given by

$$F(|\widehat{h}_l(n)|) = |\widehat{h}_l(n)|. \tag{2.25}$$

Hence, PNLMS employs larger step-sizes for 'active' coefficients than for 'inactive' coefficients and consequently achieves faster convergence than NLMS for sparse impulse responses. However, it is found that PNLMS achieves fast initial convergence but this is followed by a slower second phase convergence [25].

The MPNLMS algorithm was proposed to improve the convergence of PNLMS. It achieves this by computing the optimal proportionate step-size during the adaptation process. The MPNLMS algorithm was derived such that all coefficients attain a converged value to within a vicinity  $\epsilon$  of their optimal value in the same number of iterations [25]. As a consequence,  $F(|\hat{h}_l(n)|)$  for MPNLMS is specified by

$$F(|\hat{h}_l(n)|) = \ln(1 + \nu|\hat{h}_l(n)|), \qquad (2.26)$$

with  $\nu = 1/\epsilon$  and  $\epsilon$  (vicinity) is a very small positive number chosen as a function of the noise level [25]. It has been shown in [25] that  $\epsilon = 0.001$  is a good choice for typical echo cancellation. The positive bias of 1 in (2.26) is introduced to avoid numerical instability during the initialization stage when  $|\hat{h}_l(0)| = 0$ ,  $\forall l$ . In order to reduce the expensive computational complexity of the MPNLMS algorithm, two straight lines are proposed [25] to approximate the logarithmic function in (2.26).

It is important to note that both PNLMS and MPNLMS suffer from slow convergence when the unknown system  $\mathbf{h}(n)$  is dispersive [49, 50]. This is because when  $\mathbf{h}(n)$ is dispersive,  $\kappa_l(n)$  in (2.24) becomes significantly large for most  $0 \le l \le L - 1$ . As a consequence, the denominator of  $q_l(n)$  in (2.23) is large, giving rise to a small step-size for each large coefficient. This causes a significant degradation in convergence performance for PNLMS and MPNLMS when the impulse response is dispersive such as can occur in AIRs.

#### 2.4.3 The IPNLMS algorithm

The IPNLMS [26] algorithm was originally developed for NEC and was further developed for the identification of acoustic room impulse responses [64]. It employs a combination of proportionate (PNLMS) and non-proportionate (NLMS) adaptation, with the relative significance of each controlled by a factor  $\alpha_{\text{IP}}$  such that the diagonal elements of  $\mathbf{Q}(n)$  are given as

$$q_l(n) = \frac{1 - \alpha_{\rm IP}}{2L} + \frac{(1 + \alpha_{\rm IP})|\hat{h}_l(n)|}{2\|\hat{\mathbf{h}}(n)\|_1 + \delta_{\rm IP}}, \qquad 0 \le l \le L - 1.$$
(2.27)

where  $\|\cdot\|_1$  is defined as the  $l_1$ -norm and the first and second terms are the NLMS and the proportionate terms respectively. It can be seen that IPNLMS is the same as NLMS when  $\alpha_{\rm IP} = -1$  and PNLMS when  $\alpha_{\rm IP} = 1$ . Use of a higher weighting for NLMS adaptation, such as  $\alpha_{\rm IP} = 0$ , -0.5 or -0.75, is a favorable choice for most AEC/NEC applications [26]. It has been shown that, although the IPNLMS algorithm has faster convergence than NLMS and PNLMS regardless of the impulse response nature [26], but it was noted from the simulations that IPNLMS does not outperform MPNLMS for highly sparse impulse responses with the above choices of  $\alpha_{\rm IP}$ .

#### 2.4.4 Computational complexity

It is also necessary to examine the computational complexity of these algorithms. Although many factors contribute to the complexity of an algorithm, the relative complexity of NLMS, PNLMS, MPNLMS and IPNLMS in terms of the total number of additions, multiplications, division, logarithm (Log) and comparisons (C) per iteration for adaptation of filter coefficients is assessed in Table 2.1.

The followings should be noted:

• The computation of the 2-norm  $\|\mathbf{x}(n)\|_2^2$  requires two multiplications and one addi-

tion using the following recursive method

$$\|\mathbf{x}(n)\|_{2}^{2} = \|\mathbf{x}(n-1)\|_{2}^{2} + (1-\upsilon)x^{2}(n), \qquad (2.28)$$

where v is the forgetting factor.

• The comparison between two numbers takes one subtraction. But, in this content, comparison is regarded as an operator.

Table 2.1: Complexity of algorithms' coefficients update - Addition, Multiplication, Division, Logarithm (Log) and Comparison.

Algorithm	Addition	Multiplication	Division	Log	Comparison
NLMS	L+3	L+3	1	0	0
PNLMS	2L + 1	5L + 2	2	0	2L
MPNLMS	3L + 1	6L + 2	2	L	2L
IPNLMS	3L + 2	5L + 2	2	0	0

It can be noticed from Table 2.1 that the overall computational complexities of PNLMS, MPNLMS and IPNLMS are increased or stayed same, compared to NLMS. To compensate this increased complexities their convergence performances must be significantly higher.

# 2.5 Frequency domain adaptive algorithms for echo cancellation

In this section, echo cancellation using adaptive algorithms in frequency domain is studied. Since its introduction in [55], adaptive filtering in the frequency domain has attracted a great deal of research interest for the following three main reasons:

• By taking advantage of the computational efficiency of the FFT for computing the DFT, a convolution of two signals can be quickly calculated in frequency domain.

- They perform computations block-by-block, by incorporating block updating strategies, rather than performing sample-by-sample computations. Since the filter output and tap updates are computed only after a block of data has been accumulated, their computational complexities reduce proportional to the block length.
- A DFT processes a time sequence like a filter bank, which orthogonalizes the data, and therefore the coefficients of a frequency domain adaptive filter can converge independently or even uniformly if the update is normalized properly [66].

Deriving a frequency domain adaptive algorithm can involve a large number of variables in the form of both vectors and matrices, in time and frequency domains. Therefore, Section 2.5.1 clarifies the notations and definitions used in the following sections, before reviewing the main adaptive algorithms in the frequency domain, such as the FLMS algorithm, the MDF algorithm and the IPMDF algorithm.

#### 2.5.1 Notations and definitions

For consistency, these notations are adopted from [10]. The  $N \times N$  identity matrix is represented as  $\mathbf{I}_{N \times N}$  and a null matrix of the same dimension is denoted as  $\mathbf{0}_{N \times N}$ . The  $2L \times 2L$  Fourier matrix is defined as

$$\mathbf{F}_{2L\times 2L} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-i2\pi/2L} & e^{-i4\pi/2L} & \cdots & e^{-i2\pi(2L-1)/2L} \\ 1 & e^{-i4\pi/2L} & e^{-i8\pi/2L} & \cdots & e^{-i4\pi(2L-1)/2L} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & e^{-i2\pi(2L-1)/2L} & e^{-i4\pi(2L-1)/2L} & \cdots & e^{-i2\pi(2L-1)^2/2L} \end{bmatrix}_{2L\times 2L}, \quad (2.29)$$

where  $i = \sqrt{-1}$  and its inverse is defined by [63]

$$\mathbf{F}_{2L\times 2L}^{-1} = \frac{1}{2L} \mathbf{F}_{2L\times 2L}^*,\tag{2.30}$$

where \* denotes complex conjugate operation. All frequency domain variables are denoted with an underscore. The windowing matrices, with L and N denote the sizes, are defined  $\mathbf{as}$ 

$$\mathbf{G}_{L\times 2L}^{10} = \mathbf{F}_{L\times L} \mathbf{W}_{L\times 2L}^{10} \mathbf{F}_{2L\times 2L}^{-1}, \qquad (2.31)$$

$$\mathbf{G}_{2L\times 2L}^{01} = \mathbf{F}_{2L\times 2L} \mathbf{W}_{2L\times 2L}^{01} \mathbf{F}_{2L\times 2L}^{-1}, \qquad (2.32)$$

where

$$\mathbf{W}_{L\times 2L}^{10} = \begin{bmatrix} \mathbf{I}_{L\times L} & \mathbf{0}_{L\times L} \end{bmatrix}_{L\times 2L}, \qquad (2.33)$$

$$\mathbf{W}_{2L\times 2L}^{01} = \begin{bmatrix} \mathbf{0}_{L\times L} & \mathbf{0}_{L\times L} \\ \mathbf{0}_{L\times L} & \mathbf{I}_{L\times L} \end{bmatrix}_{2L\times 2L} .$$
(2.34)

#### 2.5.2 The FLMS algorithm

The FLMS algorithm [27] adapts its filter coefficients by first arranging the input signal x(n) into frames and employing an arbitrary overlapping factor between successive frames. These frames are then transformed into their DFT sequences using the FFT algorithm for efficient implementation. By defining m as the frame-index, the  $m^{\text{th}}$  input frame is given by

$$\mathbb{X}(m) = \left[ x(mL-L), \ x(mL-L+1), \ \cdots, \ x(mL-1), \ x(mL), \\ x(mL+1), \ \cdots, \ x(mL+L-1) \right]_{1 \times 2L}^{T},$$
(2.35)

while the estimated impulse response is given by

$$\widehat{\mathbf{h}}(m) = \left[\widehat{h}_0(m) \ \widehat{h}_1(m) \ \cdots \ \widehat{h}_{L-1}(m)\right]^T.$$
(2.36)

The frequency domain input sequence can be expressed as

$$\underline{\mathbb{X}}(m) = \mathbf{F}_{2L \times 2L} \,\mathbb{X}(m) \\ = \left[\underline{x}_0(m) \,\underline{x}_1(m) \,\cdots \,\underline{x}_{2L-1}(m)\right]^T, \quad (2.37)$$

where  $\underline{x}_l(m)$  is the  $l^{\text{th}}$  frequency-bin of the input signal for  $l = 0, 1, \dots, 2L - 1$ . The  $L \times 1$ 

received microphone signal is given by

$$\mathbb{Y}(m) = \left[ y(mL) \ y(mL+1) \ \cdots \ y(mL+L-1) \right]^T.$$
(2.38)

The frequency domain output of the adaptive filter can be expressed as

$$\widehat{\underline{\mathbb{Y}}}(m) = \mathbf{F}_{2L \times 2L} \begin{bmatrix} \mathbf{0}_{L \times 1} \\ \widehat{\mathbb{Y}}(m) \end{bmatrix}_{2L \times 1} \\
= \mathbf{G}_{2L \times 2L}^{01} \underline{\mathbb{D}}(m) \, \widehat{\underline{\mathbf{h}}}(m-1),$$
(2.39)

where

$$\underline{\mathbb{D}}(m) = \operatorname{diag}\left\{\underline{\mathbb{X}}(m)\right\} \\ = \begin{bmatrix} \underline{x}_{0}(m) & 0 & \cdots & 0 \\ 0 & \underline{x}_{1}(m) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \underline{x}_{2L-1}(m) \end{bmatrix}_{2L \times 2L}, \qquad (2.40)$$

and

$$\widehat{\underline{\mathbf{h}}}(m) = \mathbf{F}_{2L \times 2L} \begin{bmatrix} \widehat{\mathbf{h}}(m) \\ \mathbf{0}_{L \times 1} \end{bmatrix}_{2L \times 1}.$$
(2.41)

Consequently, the time domain and frequency domain *a priori* block error can be respectively expressed as

$$\mathbb{E}(m) = \left[ e(mL) \ e(mL+1) \ \cdots \ e(mL+L-1) \right]^T$$
$$= \mathbb{Y}(m) - \widehat{\mathbb{Y}}(m), \qquad (2.42)$$

$$\underline{\mathbb{E}}(m) = \mathbf{F}_{2L \times 2L} \begin{bmatrix} \mathbf{0}_{L \times 1} \\ \mathbb{E}(m) \end{bmatrix}_{2L \times 1}$$
$$= \underline{\mathbb{Y}}(m) - \underline{\widehat{\mathbb{Y}}}(m)$$
$$= \underline{\mathbb{Y}}(m) - \mathbf{G}_{2L \times 2L}^{01} \underline{\mathbb{D}}(m) \ \widehat{\mathbf{h}}(m-1).$$
(2.43)

Similar to the time domain adaptive algorithms, the FLMS employs a gradient estimate given by the correlation between the *a priori* error and the input sequence. Therefore, the time domain gradient estimation is given by

$$\widehat{\nabla}(m) = \mathbf{W}_{L \times 2L}^{10} \mathbf{F}_{2L \times 2L}^{-1} \underline{\mathbb{D}}^*(m) \underline{\mathbb{E}}(m), \qquad (2.44)$$

where  $\mathbf{W}_{L\times 2L}^{10}$  is used to obtain the first L terms in  $\mathbf{F}_{2L\times 2L}^{-1} \underline{\mathbb{D}}^*(m) \underline{\mathbb{E}}(m)$ . Hence, the equivalent filter coefficients update equation of frequency domain LMS can be given by

$$\underline{\widehat{\mathbf{h}}}(m) = \underline{\widehat{\mathbf{h}}}(m-1) + \mu \mathbf{F}_{L \times L} \mathbf{W}_{L \times 2L}^{10} \mathbf{F}_{2L \times 2L}^{-1} \underline{\mathbb{D}}^*(m) \underline{\mathbb{E}}(m)$$
(2.45)

$$= \underline{\widehat{\mathbf{h}}}(m-1) + \mu \ \mathbf{G}_{L \times 2L}^{10} \ \underline{\mathbb{D}}^*(m) \ \underline{\mathbb{E}}(m).$$
(2.46)

The FLMS algorithm quantifies each frequency-bin such that the effective step-size for each element in the gradient vector is inversely proportional to the energy of the input signal at that frequency-bin. As a result, a more uniform convergence can be achieved across different frequency-bins. This energy can be estimated recursively using a  $2L \times 2L$ matrix [67]

$$\underline{\mathcal{S}}_{\text{FLMS}}(m) = \zeta \, \underline{\mathcal{S}}_{\text{FLMS}}(m-1) + (1-\zeta) \, \underline{\mathbb{D}}^*(m) \, \underline{\mathbb{D}}(m) \\ = \text{diag} \Big\{ \underline{\mathcal{S}}_0(m) \, \underline{\mathcal{S}}_1(m) \, \cdots \, \underline{\mathcal{S}}_{2L-1}(m) \Big\}, \tag{2.47}$$

where  $\underline{S}_{l}(m)$  is the energy of the input signal in the  $l^{\text{th}}$  frequency-bin and  $0 \ll \zeta < 1$  is the forgetting factor. Hence, by normalizing the energy, the filter coefficients update equation

of FLMS is defined by

$$\widehat{\underline{\mathbf{h}}}(m) = \widehat{\underline{\mathbf{h}}}(m-1) + 2\mu (1-\zeta) \mathbf{G}_{L\times 2L}^{10} \underline{\mathbb{D}}^*(m) [\underline{\mathcal{S}}_{\text{FLMS}}(m) + \delta_{\text{FLMS}} \mathbf{I}_{2L\times 2L}]^{-1} \underline{\mathbb{E}}(m),$$
(2.48)

where  $\delta_{\text{FLMS}}$  is the regularization parameter.

Compared to the time domain adaptive algorithms, substantial computational savings can be achieved by employing the FLMS algorithm, especially when the AIRs contain several hundreds of coefficients. However, one of the drawbacks is the delay introduced between the input and output, which is equivalent to the length of the adaptive filter L, since the FLMS algorithm computes the output  $\widehat{\mathbb{Y}}(m)$  for every L input samples.

#### 2.5.3 The MDF algorithm

The MDF structure [28] was developed to mitigate the delay problem inherent in FLMS. It partitions the adaptive filter into blocks each of length N such that the delay is reduced by a factor of  $\mathcal{K} = L/N$ . The MDF structure can be described by first defining, for the  $m^{\text{th}}$  frame, the *a priori* error, which is similar to that of FLMS in (2.43),

$$\underline{\mathbf{e}}(m) = \underline{\mathbf{y}}(m) - \mathbf{G}_{2L \times 2L}^{01} \sum_{k=0}^{\mathcal{K}-1} \underline{\mathbf{D}}(m-k) \ \widehat{\underline{\mathbf{h}}}_k(m-1), \qquad (2.49)$$

where

$$\underline{\mathbf{D}}(m-k) = \operatorname{diag}\left\{\operatorname{FFT}\left\{x(mN-kN-N) \dots x(mN-kN+N-1)\right\}\right\}, (2.50)$$

and

$$\widehat{\mathbf{h}}_{k}(m) = \mathbf{F}_{2N \times 2N} \mathbf{W}_{2N \times N}^{10} \widehat{\mathbf{h}}_{k}(m).$$
(2.51)

where k is denoted as the block-index and the realization of  $\hat{\mathbf{h}}(m)$  and the sub-filter  $\hat{\mathbf{h}}_k(m)$ can be explicitly expressed by

$$\widehat{\mathbf{h}}(m) = \left[\underbrace{\widehat{h}_0(m) \cdots \widehat{h}_{N-1}(m)}_{\widehat{\mathbf{h}}_0(m)} \cdots \cdots \underbrace{\widehat{h}_{L-N}(m) \cdots \widehat{h}_{L-1}(m)}_{\widehat{\mathbf{h}}_{\mathcal{K}-1}(m)}\right]^T. \quad (2.52)$$

This recursive relation for the energy estimation, similar to that of FLMS in (2.47), can be given by

$$\underline{\mathcal{S}}_{\mathrm{MDF}}(m) = \zeta \, \underline{\mathcal{S}}_{\mathrm{MDF}}(m-1) + (1-\zeta) \, \underline{\mathbf{D}}^*(m) \, \underline{\mathbf{D}}(m).$$
(2.53)

Hence, the  $k^{\text{th}}$  sub-filter of the MDF structure is updated by

$$\widehat{\underline{\mathbf{h}}}_{k}(m) = \widehat{\underline{\mathbf{h}}}_{k}(m-1) + \mu_{\mathrm{F}} \mathbf{G}_{L\times 2L}^{10} \underline{\mathbf{D}}^{*}(m-k) \left[\underline{\mathcal{S}}_{\mathrm{MDF}}(m) + \delta_{\mathrm{MDF}} \mathbf{I}_{2L\times 2L}\right]^{-1} \underline{\mathbf{e}}(m),$$
(2.54)

where

$$\mu_{\rm F} = \tau (1 - \zeta), \tag{2.55}$$

with  $0 < \tau \leq 1$ . Letting  $\sigma_x^2$  be the input signal variance, the initial regularization parameters [10] are  $\underline{S}_{\text{MDF}}(0) = \sigma_x^2/100$  and  $\delta_{\text{MDF}} = 20\sigma_x^2 N/L$ . For N = L and  $\mathcal{K} = 1$ , MDF is equivalent to FLMS [27]. It is also interesting to note that the smaller block size (N), allows the filter coefficients to be updated more frequently, hence resulting in faster convergence. On the other hand, larger block size results a computationally efficient structure. Therefore, a good compromise should be made when choosing N, depending on the application. The convergence analysis on MDF can be found in [68].

#### 2.5.4 The IPMDF algorithm

The MDF structure has also been proposed for sparse system identification. The IP-MDF algorithm [29] was proposed to combine the fast convergence of IPNLMS and the efficient implementation brought about by the MDF structure. To achieve this, the stepsize control matrix with diagonal elements given by (2.27) is employed in each subfilter  $\widehat{\mathbf{h}}_k(m)$  in the time domain such that

$$q_{kN+l}(m) = \frac{1 - \alpha_{\text{IPMDF}}}{2L} + \frac{(1 + \alpha_{\text{IPMDF}})|\hat{h}_{kN+l}(m)|}{2\|\hat{\mathbf{h}}(m)\|_1 + \epsilon}$$
(2.56)

for  $k = 0, 1, \dots, \mathcal{K} - 1, l = 0, 1, \dots, N - 1$ , and

$$\mathbf{Q}_{k}(m) = \operatorname{diag}\{q_{kN}(m) \ q_{kN+1}(m) \ \dots \ q_{kN+N-1}(m)\}.$$
(2.57)

Accordingly, the filter coefficients adaptation is performed in the time domain by using (2.49), the energy recursion equation (similar to (2.53)) defined by

$$\underline{\mathcal{S}}_{\text{IPMDF}}(m) = \zeta \, \underline{\mathcal{S}}_{\text{IPMDF}}(m-1) + (1-\zeta) \, \underline{\mathbf{D}}^*(m) \, \underline{\mathbf{D}}(m), \qquad (2.58)$$

and

$$\widehat{\underline{\mathbf{h}}}_{k}(m) = \widehat{\underline{\mathbf{h}}}_{k}(m-1) + L\mu_{\mathrm{F}}\mathbf{Q}_{k}(m)\widetilde{\mathbf{G}}_{N\times 2N}^{10} \ \underline{\mathbf{D}}^{*}(m-k) \ [\underline{\mathcal{S}}_{\mathrm{IPMDF}}(m) + \delta_{\mathrm{IPMDF}}]^{-1} \ \underline{\mathbf{e}}(m),$$
(2.59)

where  $\widetilde{\mathbf{G}}_{N\times 2N}^{10} = [\mathbf{I}_{N\times N} \ \mathbf{0}_{N\times N}]\mathbf{F}_{2N\times 2N}^{-1}$  and  $\mu_{\rm F}$  is defined as in (2.55). The initial regularization parameters are given by  $\underline{\mathcal{S}}_{\rm IPMDF}(0) = (1 - \alpha_{\rm IPMDF})\underline{\mathcal{S}}_{\rm MDF}(0)$  and  $\delta_{\rm IPMDF} = (1 - \alpha_{\rm IPMDF})\delta_{\rm MDF}$ .

#### 2.5.5 Computational complexity

In terms of the computational complexities, N = L is the optimal choice for MDF and IP-MDF, as they have single block ( $\mathcal{K} = 1$ ) without any overlapped input samples. Moreover, with  $\mathcal{K} = 1$ , MDF is similar to FLMS. The relative computational complexity of FLMS, MDF and IPMDF in terms of the total number of additions, multiplications and divisions per iteration for adaptation of filter coefficients is shown in Table 2.2 for  $\mathcal{K} = 1$ . Since IPMDF updates the filter coefficients in time domain, it requires an additional  $L \log_2(L)$ real multiplications and  $L \log_2(L)$  additions to compute the radix-2 FFT.

Algorithm	Addition	Multiplication	Division
FLMS & MDF	$4L \log_2(L) + 4L$	$4L\log_2(L) + 6L$	L
IPMDF	$5L \log_2(L) + 6L + 2$	$5L \log_2(L) + 8L + 2$	L+2

Table 2.2: Computation	al complexity of FL	MS. MDF an	d IPMDF for	$\mathcal{K} = 1$
rasic mpacation				

## 2.6 Summary

Echo cancellers can be potentially employed in telecommunication systems so that the undesired echoes, both acoustic and hybrid, can be controlled. The formation of single channel acoustic echo canceller has been looked at in detail, where the functioning of the adaptive filter has been studied.

By minimizing the mean square value of the error signal for linear filtering problems, Wiener filters can be employed to find the optimal filter which, when applied to the input signal, produces a signal that is close to the desired signal. This approach requires knowledge of certain statistical information of the input signal and needs to perform heavy computation each time the statistics changes. The well-known recursive optimization technique, steepest descent, can be applied and eventually converges to the Wiener solution from an arbitrary starting point on the error performance surface. It iteratively improves the solution by progressing towards the minimum point on the error performance surface as the number of iterations increases. As the gradient vector of the steepest decent method still requires the explicit knowledge of the statistics of the input signal, stochastic gradient based algorithms are next considered as they avoid this need for statistical knowledge by estimating the gradient vector at each iteration. To conclude the three methods, the Wiener filter is a closed-form solution, whereas the steepest descent and the stochastic gradient methods approach the Wiener solution by taking calculated and estimated steps, respectively.

Several time and frequency domains algorithms were reviewed and the main adaptive algorithms, including NLMS, PNLMS, MPNLMS, IPNLMS, FLMS, MDF and IP-MDF, were studied in detail by also studying their computational complexities. Their tracking performances will be compared in the next chapter, along with the proposed algorithms extended using these conventional methods.

## 2.6.1 Time domain algorithms

#### Table 2.3: The NLMS, PNLMS, MPNLMS and IPNLMS Algorithms

Initializations:  $\widehat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$  $0 < \mu \leq 1$ 

#### General Computations:

$$e(n) = y(n) - \widehat{\mathbf{h}}^{T}(n-1)\mathbf{x}(n)$$
$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{Q}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^{T}(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta}$$
$$\mathbf{Q}(n-1) = \operatorname{diag}\left\{q_{0}(n-1), \dots, q_{L-1}(n-1)\right\}$$

#### NLMS

 $q_l(n) = 1, \qquad 0 \le l \le L - 1$ 

#### PNLMS

$$q_l(n) = \frac{\kappa_l(n)}{\frac{1}{L} \sum_{i=0}^{L-1} \kappa_i(n)}, \quad 0 \le l \le L-1$$
  
$$\kappa_l(n) = \max\left\{\rho \times \max\left\{\gamma, \ |\hat{h}_0(n)|, \dots, |\hat{h}_{L-1}(n)|\right\}, |\hat{h}_l(n)|\right\}$$

#### MPNLMS

$$q_{l}(n) = \frac{\kappa_{l}(n)}{\frac{1}{L}\sum_{i=0}^{L-1}\kappa_{i}(n)}, \quad 0 \le l \le L-1$$
  

$$\kappa_{l}(n) = \max\left\{\rho \times \max\left\{\gamma, \ F(|\hat{h}_{0}(n)|), \dots, F(|\hat{h}_{L-1}(n)|)\right\}, F(|\hat{h}_{l}(n)|)\right\}$$
  

$$F(|\hat{h}_{l}(n)|) = \ln(1+\nu|\hat{h}_{l}(n)|)$$

#### **IPNLMS**

$$q_l(n) = \frac{(1 - \alpha_{\rm IP})}{2L} + \frac{(1 + \alpha_{\rm IP})|\hat{h}_l(n)|}{2\|\hat{\mathbf{h}}(n)\|_1 + \delta_{\rm IP}}, \qquad 0 \le l \le L - 1$$

## 2.6.2 Frequency domain algorithms

## Table 2.4: The FLMS Algorithm

ζ	=	$\left[1-\frac{1}{3L}\right]^L$
$\mathbb{X}(m)$	=	$[x(mL-L) \ x(mL-L+1) \ \cdots \ x(mL+L-1)]^T$
$\underline{\mathbb{X}}(m)$	=	$\mathbf{F}_{2L  imes 2L} \ \mathbb{X}(m)$
$\underline{\mathbb{D}}(m)$	=	diag $\{\underline{\mathbb{X}}(m)\}$
$\underline{\widehat{\mathbb{Y}}}(m)$	=	$\mathbf{G}_{2L\times 2L}^{01} \ \underline{\mathbb{D}}(m) \ \widehat{\mathbf{h}}(m-1)$
$\underline{\mathbb{E}}(m)$	=	$\underline{\mathbb{Y}}(m) - \underline{\widehat{\mathbb{Y}}}(m)$
$\underline{\mathcal{S}}_{\mathrm{FLMS}}(m)$	=	$\zeta \ \underline{S}_{\text{FLMS}}(m-1) + (1-\zeta) \ \underline{\mathbb{D}}^*(m) \ \underline{\mathbb{D}}(m)$
$\widehat{\underline{\mathbf{h}}}(m)$	=	$\widehat{\underline{\mathbf{h}}}(m-1) + 2\mu \ (1-\zeta) \ \mathbf{G}_{L\times 2L}^{10} \ \underline{\mathbb{D}}^*(m) \ \left[\underline{\mathcal{S}}_{\mathrm{FLMS}}(m) + \delta_{\mathrm{FLMS}} \ \mathbf{I}_{2L\times 2L}\right]^{-1} \ \underline{\mathbb{E}}(m)$

#### Table 2.5: The MDF and IPMDF Algorithm

ζ	$=\left[1-\frac{1}{3L}\right]^{L}$
$\mu_{ m F}$	$= \tau(1-\zeta),  0 < \tau \le 1$
$\underline{\mathbf{D}}(m-k)$	= diag {FFT { $x(mN - kN - N) \dots x(mN - kN + N - 1)$ }}
$\underline{\mathbf{e}}(m)$	$= \underline{\mathbf{y}}(m) - \mathbf{G}_{2L \times 2L}^{01} \sum_{k=0}^{\mathcal{K}-1} \underline{\mathbf{D}}(m-k) \ \widehat{\underline{\mathbf{h}}}_{k}(m-1)$

#### MDF

$$\underline{S}_{\text{MDF}}(m) = \zeta \, \underline{S}_{\text{MDF}}(m-1) + (1-\zeta) \, \underline{\mathbf{D}}^{*}(m) \, \underline{\mathbf{D}}(m) 
\underline{\widehat{\mathbf{h}}}_{k}(m) = \underline{\widehat{\mathbf{h}}}_{k}(m-1) + \mu_{\text{F}} \mathbf{G}_{L\times 2L}^{10} \underline{\mathbf{D}}^{*}(m-k) \left[\underline{S}_{\text{MDF}}(m) + \delta_{\text{MDF}} \mathbf{I}_{2L\times 2L}\right]^{-1} \underline{\mathbf{e}}(m)$$

#### IPMDF

$$\underline{S}_{\text{IPMDF}}(m) = \zeta \, \underline{S}_{\text{IPMDF}}(m-1) + (1-\zeta) \, \underline{\mathbf{D}}^*(m) \, \underline{\mathbf{D}}(m)$$

$$q_{kN+l}(m) = \frac{1-\alpha_{\text{IPMDF}}}{2L} + \frac{(1+\alpha_{\text{IPMDF}})|\hat{h}_{kN+l}(m)|}{2\|\hat{\mathbf{h}}(m)\|_1 + \epsilon} \quad k = 0, \dots \mathcal{K} - 1, \ l = 0, \dots N - 1$$

$$\mathbf{Q}_k(m) = \text{diag}\{q_{kN}(m) \ q_{kN+1}(m) \ \dots \ q_{kN+N-1}(m)\}$$

$$\underline{\hat{\mathbf{h}}}_k(m) = \underline{\hat{\mathbf{h}}}_k(m-1) + L\mu_{\text{F}}\mathbf{Q}_k(m)\widetilde{\mathbf{G}}_{N\times 2N}^{10} \, \underline{\mathbf{D}}^*(m-k) \ [\underline{S}_{\text{IPMDF}}(m) + \delta_{\text{IPMDF}}]^{-1} \underline{\mathbf{e}}(m)$$

# Chapter 3

# A Class of Sparseness-controlled Algorithms

THE aim of the previous chapter was to introduce the first and most fundamental adaptive algorithms that identify unknown acoustic impulse responses in the time and frequency domains. In this chapter, more sophisticated time and frequency domain single-input single-output adaptive algorithms that better suit the acoustic environment are studied. The earlier version of these works were published in [69, 70, 71].

## 3.1 Introduction

The use of adaptive filters for system identification has found applications in both network and acoustic echo cancellation. Such adaptive filters are employed to estimate the unknown impulse response of the system and algorithms developed for such applications require fast convergence as well as good tracking performance.

In reality, the acoustic echo in the receiving room does not always follow a same response. The path may vary with time influenced by the distance between the loudspeaker and the microphone and due to change in room characteristics, including temperature [11], pressure and movement of the talker. The acoustic characteristics of environment can be evaluated by the reverberation time, which is proportional to the volume of the enclosed space and inversely proportional to the absorption area [12]. For an outdoor environment, the reverberation time is reduced significantly due to the lack of reflections from any enclosure. The outdoor environment refers here to a typical urban area or a rural area with sparsely placed acoustically reflecting objects such as display boards. The sparseness of the AIR of an outdoor environment is significantly greater than typical indoor environments and equally, if not more, variable.

In such a time-varying environment, the underlying impulse response may vary over a sufficiently large range that its sparsity could change from sparse to dispersive. Therefore, there is a need for an algorithm which can work effectively and be robust to the variations in the sparseness of the acoustic path.

In this chapter, a new approach is proposed to improve convergence of proportionate adaptive algorithms for dispersive impulse responses estimation, in the time (Section 3.4) and frequency (Section 3.5) domains. The proposed algorithms compute a sparseness measure of the estimated impulse response at each iteration of the adaptive process and incorporate it into their methods. As will be shown, the proposed sparseness-controlled algorithms achieve fast convergence for both sparse and dispersive AIRs and are robust to the sparseness variation of AIRs, hence they are effective for AEC.

#### 3.2 Sparseness measure

Impulse responses are very different from one to another in networks or under different room conditions, so it is important to quantify how sparse or dispersive they are. The degree of sparseness for an impulse response can be qualitatively measured ranging from strongly dispersive to strongly sparse. The sparsity  $\xi(\mathbf{h})$  of a vector  $\mathbf{h}$  can be quantitatively measured by [2, 3]

$$\xi(\mathbf{h}) = \frac{L}{L - \sqrt{L}} \left\{ 1 - \frac{\|\mathbf{h}\|_1}{\sqrt{L} \|\mathbf{h}\|_2} \right\},\tag{3.1}$$



Figure 3.1: Sparseness measure of different impulse responses.

where L is the length of the vector  $\mathbf{h}$  and  $\|\mathbf{h}\|_1$  and  $\|\mathbf{h}\|_2$  represent  $\ell_1$  and  $\ell_2$ -norms of  $\mathbf{h}$ , respectively defined as:

$$\|\mathbf{h}\|_{1} = \sum_{l=0}^{L-1} |h_{l}|, \qquad (3.2)$$
$$\|\mathbf{h}\|_{2} = \sqrt{\sum_{l=0}^{L-1} |h_{l}^{2}|}$$
$$= \sqrt{\mathbf{h}^{T} \mathbf{h}}. \qquad (3.3)$$

As  $1 \leq \frac{\|\mathbf{h}\|_1}{\|\mathbf{h}\|_2} \leq \sqrt{L}$ , it can be shown [2, 3] that  $0 \leq \xi(\mathbf{h}) \leq 1$ . In the extreme but unlikely case when

$$h_{l} = \begin{cases} \pm k, & l = l_{1}, \\ 0, & 0 \le l \le L - 1, \ l \ne l_{1}, \end{cases}$$
(3.4)

where  $l_1 \in \{[0, L-1]\}$  and  $k \in \Re$  as shown in the rightmost plot of Fig. 3.1, then  $\xi(\mathbf{h}) = 1$ . On the other hand, when  $h_l = \pm k \forall l$  as shown in the leftmost plot of Fig. 3.1, then  $\xi(\mathbf{h}) = 0$ . It is also interesting to note that the measure is independent of the sorting order of the impulse response coefficients and not affected by a non-zero scaling factor, i.e.:

$$\xi(\mathcal{C}\mathbf{h}) = \xi(\mathbf{h}) \quad \forall \mathcal{C} \neq 0. \tag{3.5}$$

## 3.3 Characterization of framework for robust convergence in the time domain

In this Section, an illustrative example is provided to show how the sparseness of AIRs varies with the loudspeaker-microphone distance in an enclosed space such as when the user is using a wireless microphone for tele/video conferencing. This serves as a motivation to develop new algorithms which are robust to the sparseness variation of AIRs in the next Section. In addition, it also demonstrates how the choice of  $\rho$  in (2.24) affects the step-size of each filter coefficient for PNLMS.

#### 3.3.1 Variation of sparseness in AIRs

In reality  $\mathbf{h}(n)$  and hence  $\xi(\mathbf{h}(n))$  is time-varying and depends on factors such as temperature, pressure and reflectivity [11]. As explained in Section 3.1, the sparseness of AIRs  $\xi(\mathbf{h}(n))$  varies with the location of the receiving device in an open or enclosed environment, because of the open environment has fewer acoustically reflecting objects than the enclosed environment. The sparseness measure  $\xi(\mathbf{h}(n))$  can also vary with the loudspeaker-microphone distance in an enclosed space.

Consider an example case where the distance, a, between a fixed position loudspeaker and the talker using a microphone is varying. Figure 1.2 shows two AIRs, generated using the method of images [1, 23] with 1024 coefficients using room dimensions of  $8 \times 10 \times 3$  m and 0.57 as the reflection coefficient. The loudspeaker was fixed at  $4 \times 9.1 \times 1.6$  m in the LRMS while the microphone was positioned at  $4 \times 8.2 \times 1.6$  m and  $4 \times 1.4 \times 1.6$  m giving impulse responses as shown in Fig. 1.2 (a) and (b) for a = 0.9 m and a = 7.7 m respectively. Figure 3.2 illustrates how  $\xi(\mathbf{h}(n))$  of such AIRs varies with a. For each loudspeaker-microphone distance a, the microphone was directly in front of the loudspeaker. As can be seen,  $\xi(\mathbf{h}(n))$  reduces with increasing a, since for increasing a, the sound field becomes more diffuse. Since  $\xi(\mathbf{h}(n))$  varies with a, it is proposed to incorporate  $\xi(\mathbf{h}(n))$  into PNLMS, MPNLMS and IPNLMS in order to improve their robustness to the sparseness of AIRs in AEC. Since  $\mathbf{h}(n)$  is unknown during adaptation,  $\hat{\xi}(n)$  is employed



Figure 3.2: Sparseness measure against the distance between loudspeaker and microphone, *a*. The impulse responses are obtained from the image model using a fixed room dimensions of  $8 \times 10 \times 3$  m.

to estimate the sparseness of an impulse response, where at each sample iteration,

$$\widehat{\xi}(n) = \frac{L}{L - \sqrt{L}} \left\{ 1 - \frac{\|\widehat{\mathbf{h}}(n)\|_1}{\sqrt{L} \|\widehat{\mathbf{h}}(n)\|_2} \right\},\tag{3.6}$$

which uses the estimation of the impulse response at the iteration  $(\widehat{\mathbf{h}}(n))$ , instead of the unknown impulse response  $\mathbf{h}(n)$ .

#### **3.3.2** Effect of $\rho$ on step-size control matrix $\mathbf{Q}(n)$ for PNLMS

As explained in Section 2.4.2, the parameter  $\rho$  in (2.24) was originally introduced to prevent freezing of the filter coefficients when they are much smaller than the largest coefficient. Figure 3.3 shows the effect of  $\rho$  for both sparse and dispersive AIRs on the convergence performance of PNLMS measured using the normalized misalignment defined in (2.7).

A zero mean white Gaussian noise (WGN) sequence was used as the input signal while another WGN sequence w(n) was added to give an SNR of 20 dB. Impulse responses



Figure 3.3: Convergence of the PNLMS for different values of  $\rho$  using WGN input signal. Impulse responses in Fig. 1.2 are used as sparse and dispersive AIRs respectively.  $[\mu_{\text{PNLMS}} = 0.3, \text{ SNR} = 20 \text{ dB}]$ 

as shown in Fig. 1.2 were used as sparse and dispersive AIRs, and  $\mu_{\text{PNLMS}} = 0.3$ . It can be seen from this illustration that, for a sparse  $\mathbf{h}(n)$ , a low value of  $\rho$  is desired while, for a dispersive unknown system  $\mathbf{h}(n)$ , a high value of  $\rho$  is desired. This is due to the resulting effect of how different values of  $\rho$  affect the step-size control element  $q_l(n)$  as illustrated in Fig. 3.4. It can be observed that a higher value of  $\rho$  will reduce the influence of the proportional update term meaning that all filter coefficients are updated at a more uniform rate. This provides a good convergence performance for PNLMS for a dispersive AIR. On the other hand, a lower  $\rho$  will increase the degree of proportionality hence giving good convergence performance when the AIR is sparse. As a consequence of this important observation, it is proposed to incorporate  $\hat{\xi}(n)$  into  $\rho$  for both PNLMS and MPNLMS as described in the next section.

## 3.4 Time domain sparseness-controlled algorithms

In this section, an improvement in the robustness of PNLMS, MPNLMS and IPNLMS to varying levels of sparseness of impulse response such as encountered in, for example AEC, is proposed. As will be shown in the following, this is achieved by incorporating the sparseness measure of the estimated AIRs into the adaptation process. These approaches will be discussed conceptually and with simulation results on both WGN and speech input



Figure 3.4: Magnitude of  $q_l(n)$  for  $0 \le l \le L-1$  against the magnitude of coefficients  $\hat{h}_l(n)$  in PNLMS.

signals.

#### 3.4.1 The SC-PNLMS and SC-MPNLMS algorithm

In order to address the problem of slow convergence in PNLMS and MPNLMS for dispersive AIR, the step-size control elements  $q_l(n)$  require to be robust to variation in the sparseness of the impulse response. Several choices can be employed to obtain the desired effect of achieving a high  $\rho$  when  $\hat{\xi}(n)$  is small when estimating dispersive AIRs. An example exponential function is considered as

$$\rho(n) = e^{-\lambda\xi(n)}, \qquad \lambda \in \mathbb{R}^+. \tag{3.7}$$

The variation of  $\rho(n)$  in PNLMS for the exponential function is plotted in Fig. 3.5 for the cases where  $\lambda = 4, 6$  and 8. It can be noted that a linear function  $\rho(n) = 1 - \hat{\xi}(n)$  also achieves the desired condition. This case was first tested and found it to be performing worse than the more general form of (3.7), as the value of  $\rho(n)$  is not small enough to



Figure 3.5: Variation of  $\rho$  against sparseness measure  $\hat{\xi}(n)$  of impulse response.

achieve the desired proportionality control determined by  $q_l(n)$  when the AIR is strongly sparse such as for  $0.8 \leq \hat{\xi}(n) \leq 1$ . So, the choice of the linear function will not be considered further.

It can be seen that low values of  $\rho(n)$  are allocated for a large range of sparse impulse responses such as when  $\hat{\xi}(n) > 0.4$ . As a result of small values in  $\rho(n)$  using (3.7), the proposed sparseness-controlled PNLMS algorithm (SC-PNLMS) inherits the proportionality step-size control over a large range of sparse impulse response. When the impulse response is dispersive, such as when  $\hat{\xi}(n) < 0.4$ , the proposed SC-PNLMS algorithm inherits the NLMS adaptation control with larger values of  $\rho(n)$ . As explained in Section 3.3.2 and Fig. 3.4, this gives a more uniform step-size across  $h_l(n)$ . Hence, the exponential function described by (3.7) will achieve the overall desired effect of the robustness to sparse and dispersive AIRs.

The choice of  $\lambda$  is important. As can be seen from Fig. 3.5, a larger choice of  $\lambda$  will cause the proposed SC-PNLMS to inherit more of PNLMS properties compared to NLMS
giving good convergence performance when AIR is sparse. On the other hand, when the AIR is dispersive,  $\lambda$  must be small for good convergence performance. Hence, it will be shown in Section 3.5.2 that a good compromise is given by  $\lambda = 6$ , though the algorithm is not very sensitive to this choice in the range of  $4 \le \lambda \le 6$ .

Incorporating  $\rho(n)$  in a similar manner for the MPNLMS algorithm, the resulting sparseness-controlled MPNLMS algorithm (SC-MPNLMS) inherits more of the MPNLMS properties when the estimated AIR is sparse and distributes uniform step-size across  $h_l(n)$ , as in NLMS, when the estimated AIR is dispersive. In addition, it can be noted that when n = 0,  $\|\widehat{\mathbf{h}}(0)\|_2 = 0$  and hence, to prevent division by a small number or zero,  $\widehat{\xi}(n)$  can be computed for  $n \ge L$  in both SC-PNLMS and SC-MPNLMS. When n < L, a value of  $\rho(n) = 5/L$  can be set as described in [26].

The SC-PNLMS algorithm is thus described by (2.5)-(2.25), (3.6) and (3.7), whereas SC-MPNLMS is described by (2.5)-(2.24), (2.26), (3.6) and (3.7) with  $\lambda = 6$ , as summarized in Table 3.3.

#### 3.4.2 The SC-IPNLMS algorithm

A different approach, compared to SC-PNLMS and SC-MPNLMS, is chosen to incorporate sparseness-control into the IPNLMS algorithm (SC-IPNLMS) [2] because, as can be seen from (2.27), two terms are employed in IPNLMS for control of the mixture between proportionate and NLMS updates. The proposed SC-IPNLMS improves the performance of the IPNLMS by expressing  $q_l(n)$  for  $n \ge L$  as

$$q_{l}(n) = \left[\frac{1 - 0.5\hat{\xi}(n)}{L}\right] \frac{(1 - \alpha_{\rm SC-IP})}{2L} + \left[\frac{1 + 0.5\hat{\xi}(n)}{L}\right] \frac{(1 + \alpha_{\rm SC-IP})|\hat{h}_{l}(n)|}{2\|\hat{\mathbf{h}}(n)\|_{1} + \delta_{\rm IP}}.$$
 (3.8)

As can be seen, for large  $\hat{\xi}(n)$  when the impulse response is sparse, the algorithm allocates more weight to the proportionate term of (2.27). For comparatively less sparse impulse responses, the algorithm aims to achieve the convergence of NLMS by applying a higher weighting to the NLMS term. An empirically chosen weighting of 0.5 in (3.8) is



Figure 3.6: Magnitude of  $q_l(n)$  for  $0 \le l \le L-1$  against the magnitude of coefficients  $\hat{h}_l(n)$  in SC-IPNLMS and different sparseness measures of 8 systems.

included to balance the performance between sparse and dispersive cases, which could be further optimized for a specific application. In addition, normalization by L is introduced to reduce significant coefficient noise when the effective step-size is large for sparse AIRs with high  $\hat{\xi}(n)$ .

Figure 3.6 illustrates the step-size control elements  $q_l(n)$  for SC-IPNLMS in estimating different unknown AIRs. As can be seen, for dispersive AIRs, SC-IPNLMS allocates a uniform step-size across  $h_l(n)$  while, for sparse AIRs, the algorithm distributes  $q_l(n)$ proportionally to the magnitude of the coefficients. As a result of this distribution, the SC-IPNLMS algorithm varies the degree of NLMS and proportionate adaptations according to the nature of the AIRs. In contrast, in standard IPNLMS the mixing coefficient  $\alpha_{\rm IP}$ in (2.27) is fixed *a priori*. The SC-IPNLMS algorithm is described by (2.5)-(2.21), (3.6) and (3.8), as specified in Table 3.3.

#### 3.4.3 Performance evaluation

Simulation results are presented next, to evaluate the performance of the proposed SC-PNLMS, SC-MPNLMS and SC-IPNLMS algorithms in the context of AEC. In addition, an example case of how SC-IPNLMS can be employed in NEC and also the tracking performances of the sparseness-controlled algorithms under a time-varying unknown echo system are shown at the end of this section.

## Experimental setup

Throughout the simulations, algorithms were tested using a zero mean WGN and a male speech signal as inputs while another WGN sequence w(n) was added to give an SNR of 20 dB. The length of the adaptive filter L = 1024 was assumed to be equivalent to that of the unknown system. Two receiving room impulse responses  $\mathbf{h}(n)$  for AEC simulations have been used as described in Fig. 1.2. The sparseness measure of these AIRs are computed using (3.1) giving  $\xi(n) = 0.83$  and  $\xi(n) = 0.59$  respectively.

#### Effect of $\lambda$ on the performance of SC-PNLMS for AEC

SC-PNLMS was tested as shown in Fig. 3.7 for different  $\lambda$  values in (3.7) to illustrate the time taken to reach -20 dB normalized misalignment using a WGN sequence as the input signal. A step-size of  $\mu = 0.3$  was used in this experiment. It can be seen from the result that, for each case of  $\lambda$ , the SC-PNLMS has a higher rate of convergence for a sparse system compared to a dispersive system. This is due to the initialization choice of  $\hat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$ , where most filter coefficients are initialized close to their optimal values. In addition, a smaller value of  $\lambda$  is favorable for the dispersive AIR, since SC-PNLMS performs similarly to NLMS for small  $\lambda$  values. On the contrary, a higher value for  $\lambda$  is desirable for the sparse case. It can be noted that SC-PNLMS is exactly NLMS for  $\lambda = 0$ . It can also be seen that a range of good value for  $\lambda$  is  $4 \leq \lambda \leq 6$ .

Figure 3.8 shows the performance of SC-PNLMS with an echo path change introduced from Fig. 1.2 (a) to (b) at 3.5 s, for  $\lambda = 0, 4, 6$  and 8. It can be observed from this



Figure 3.7: Time to reach -20 dB normalized misalignment level for different values of  $\lambda$  in SC-PNLMS using WGN input signal. Impulse response in Fig. 1.2 used as sparse AIR and dispersive AIR respectively. [ $\mu_{\text{SC-PNLMS}} = 0.3$ , SNR = 20 dB]

result that the convergence rate of SC-PNLMS is high when  $\lambda$  is small for a dispersive channel. This is because, as explained in Section 3.4.1, the proposed algorithm inherits properties of the NLMS for a small  $\lambda$  value. For a high  $\lambda$ , the SC-PNLMS algorithm inherits properties of PNLMS giving good performance for sparse AIR before the echo path change. As can be seen, a good compromise of  $\lambda$  is given by  $\lambda = 6$ .

## Convergence performance of SC-PNLMS for AEC

Figure 3.9 compares the performance of NLMS, PNLMS and SC-PNLMS using WGN as the input signal. The step-size parameter for each algorithm was chosen such that all algorithms achieve the same steady-state performance asymptotically. This was achieved by setting  $\mu_{\text{NLMS}} = \mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}} = 0.3$ . An echo path change was introduced at 3.5 s from Fig. 1.2 (a) to 1.2 (b) while  $\lambda$  for the SC-PNLMS algorithm was set to 6. It can be seen from Fig. 3.9 that the convergence rate of SC-PNLMS is as fast as PNLMS



Figure 3.8: Convergence of the SC-PNLMS for different values of  $\lambda$  using WGN input signal with an echo path change at 3.5 s. Impulse response is changed from Fig. 1.2 (a) to (b) and  $\mu_{\text{SC-PNLMS}} = 0.3$ , SNR = 20 dB.

for sparse and much better than PNLMS for dispersive, therefore achieving our objective of improving robustness to varying sparseness. This is because SC-PNLMS inherits the beneficial properties of both PNLMS and NLMS. It can be seen from the result that SC-PNLMS achieves high rate of convergence similar to PNLMS giving approximately 5 dB improvement in normalized misalignment during initial convergence compared to NLMS for a sparse AIR. After the echo path change, for a dispersive AIR, the SC-PNLMS maintains its high convergence rate over NLMS and PNLMS giving approximately 4 dB improvement in normalized misalignment compared to PNLMS.

Figure 3.10 shows simulation results for a male speech input signal where the same parameters as in the case of WGN input signal were used. As can be seen, the proposed SC-PNLMS algorithm achieves the highest rate of convergence, giving convergence as fast as PNLMS and approximately 7 dB improvement during initial convergence compared to NLMS for the sparse AIR. For dispersive AIR, SC-PNLMS performs almost the same as NLMS with approximately 4 dB improvement compared to PNLMS.



Figure 3.9: Relative convergence of NLMS, PNLMS and SC-PNLMS using WGN input signal with an echo path change at 3.5 s. Impulse response is changed from that shown from Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = \mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}} = 0.3$ , SNR = 20 dB.

## Convergence performance of SC-MPNLMS for AEC

Figure 3.11 illustrates the performance of NLMS, MPNLMS and SC-MPNLMS using WGN as the input signal. As before, the step-sizes were adjusted to achieve the same steady-state misalignment for all algorithms. This corresponds to  $\mu_{\text{NLMS}} = 0.3$ ,  $\mu_{\text{MPNLMS}} = \mu_{\text{SC-MPNLMS}} = 0.25$ . The value of  $\lambda = 6$  was also used for SC-MPNLMS. As can be seen from this result, the SC-MPNLMS algorithm attains approximately 8 dB improvement in normalized misalignment during initial convergence compared to NLMS and same initial performance followed by approximately 2 dB improvement over MPNLMS for the sparse AIR. After the echo path change, SC-MPNLMS achieves approximately 3 dB improvement compared to MPNLMS and about 8 dB better performance than NLMS for dispersive AIR.

As shown in Fig. 3.12, with the speech signal as the input, the proposed SC-MPNLMS algorithm achieves approximately 10 dB improvement during initial conver-



Figure 3.10: Relative convergence of NLMS, PNLMS and SC-PNLMS using speech input signal with echo path changes at 58 s. Impulse response is changed from that shown in Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = 0.3$ ,  $\mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}} = 0.1$ , SNR = 20 dB.

gence compared to NLMS and 2 dB compared to MPNLMS for the sparse AIR. For dispersive AIR, the SC-MPNLMS algorithm achieves an improvement of approximately 4 dB compared to both NLMS and MPNLMS. It is also noted that NLMS achieves approximately 7 dB better steady-state performance than the MPNLMS-based approaches for this example with speech input. This is attributed in [72] to sensitivity to eigenvalue spread of the speech signal's autocorrelation matrix.

#### Convergence performance of SC-IPNLMS for AEC

For SC-IPNLMS performance comparison, the step-sizes were chosen as  $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ ,  $\mu_{\text{SC-IPNLMS}} = 0.7$  in order to attain same steady state performance. Proportionality control factors  $\alpha_{\text{IP}} = \alpha_{\text{SC-IP}} = -0.75$  have been used for both IPNLMS and SC-IPNLMS. It can be seen from Fig. 3.13 and 3.14 that by using both WGN and speech input signals, SC-IPNLMS achieves approximately 10 dB improvement in normal-



Figure 3.11: Relative convergence of NLMS, MPNLMS and SC-MPNLMS using WGN input signal with an echo path change at 3.5 s. Impulse response is changed from that shown from Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = 0.3$ ,  $\mu_{\text{MPNLMS}} = \mu_{\text{SC-MPNLMS}} = 0.25$ , SNR = 20 dB.

ized misalignment during initial convergence compared to NLMS for the sparse AIR. For a dispersive AIR, the SC-IPNLMS achieves a 5 dB improvement compared to NLMS. For a speech input, the improvement of SC-IPNLMS over IPNLMS is 3 dB for both sparse and dispersive AIRs. On the other hand, the improvement of SC-IPNLMS compared to NLMS are 10 dB and 6 dB for sparse and dispersive AIRs, respectively.

## Convergence performance for AIRs with different sparseness in AEC

Eight different impulse responses were extracted from a set of AIRs with sparseness measure  $0.58 \leq \xi \leq 0.93$  as shown in Fig. 3.2. The time taken to reach -20 dB normalized misalignment is plotted against  $\xi(n)$  for NLMS, PNLMS, SC-PNLMS, IPNLMS and SC-IPNLMS in Fig. 3.15, and for NLMS, MPNLMS and SC-MPNLMS in Fig. 3.16. As before, all step-sizes have been adjusted so that the algorithms achieve the same steady-state normalized misalignment. These correspond to  $\mu_{\text{NLMS}} = \mu_{\text{PNLMS}} = \mu_{\text{SC-PNLMS}} = \mu_{\text{IPNLMS}} =$ 



Figure 3.12: Relative convergence of NLMS, MPNLMS and SC-MPNLMS using speech input signal with echo path changes at 58 s. Impulse response is changed from that shown in Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = 0.3$ ,  $\mu_{\text{MPNLMS}} = \mu_{\text{SC-MPNLMS}} = 0.25$ , SNR = 20 dB.

0.3,  $\mu_{\text{MPNLMS}} = \mu_{\text{SC-MPNLMS}} = 0.25$  and  $\mu_{\text{SC-IPNLMS}} = 0.7$ . A zero mean WGN was used as an input signal while another WGN sequence w(n) was added to achieve an SNR of 20 dB. It can be seen that when the AIRs are sparse, the speed of initial convergence increases significantly for each algorithm. This is because many of the filter coefficients are initialized close to their optimum values since during initialization,  $\hat{\mathbf{h}}(0) = \mathbf{0}_{L \times 1}$ .

The sparseness-controlled algorithms (SC-PNLMS, SC-MPNLMS and SC-IPNLMS) give the overall best performance compare to their conventional methods across the range of sparseness measure. This is because the proposed algorithms take into account the sparseness measure of the estimated impulse response at each iteration.



Figure 3.13: Relative convergence of NLMS, IPNLMS and SC-IPNLMS using WGN input signal with an echo path change at 3.5 s. Impulse response is changed from that shown from Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ ,  $\mu_{\text{SC-IPNLMS}} = 0.7$ , SNR = 20 dB.



Figure 3.15: Time to reach the -20dB normalized misalignment against different sparseness measures of 8 systems for NLMS, PNLMS, SC-PNLMS, IPNLMS and SC-IPNLMS.



Figure 3.14: Relative convergence of NLMS, IPNLMS and SC-IPNLMS using speech input signal with echo path changes at 58 s. Impulse response is changed from that shown in Fig. 1.2 (a) to (b) and  $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ ,  $\mu_{\text{SC-IPNLMS}} = 0.8$ , SNR = 20 dB.



Figure 3.16: Time to reach the -20dB normalized misalignment against different sparseness measures of 8 systems for NLMS, MPNLMS and SC-MPNLMS.

#### Convergence performance of SC-IPNLMS for NEC

Additional simulations are provided to illustrate the performance of SC-IPNLMS in the context of sparse adaptive NEC, such as may occur in network gateways for mixed packetswitched and circuit-switched networks. Figure 3.17 shows two impulse responses, sampled at 8 kHz comprising a 12 ms active region located within a total duration of 128 ms. The sparseness of these impulse responses computed using (3.1) are (a)  $\xi(n) = 0.88$  and (b)  $\xi(n) = 0.85$  respectively. As before, a WGN input signal was used while another WGN sequence is added to give an SNR of 20 dB.



Figure 3.17: Sparse impulse responses, sampled at 8 kHz, giving (a)  $\xi(n) = 0.88$  and (b)  $\xi(n) = 0.85$  respectively.

Figure 3.18 shows the performances of NLMS, IPNLMS, for  $\alpha_{IP} = -0.5$  and -0.75, and the proposed SC-IPNLMS algorithm with  $\alpha_{SC-IP} = -0.75$ . An echo path change was introduced using impulse responses as shown from Fig. 3.17 (a) to (b) at 3.5 s. It can be seen from the result that the performance of IPNLMS is dependent on  $\alpha_{IP}$ . More importantly, a faster rate of convergence can be seen for SC-IPNLMS compared to NLMS and IPNLMS both at initial convergence and also after the echo path change.



Figure 3.18: Relative convergence of NLMS, IPNLMS for  $\alpha = -0.5$  and -0.75 and SC-IPNLMS using WGN input signal with an echo path change at 3.5 s. Impulse response is changed from that shown in Fig. 3.17 (a) to (b) and  $\mu_{\text{NLMS}} = \mu_{\text{IPNLMS}} = 0.3$ ,  $\mu_{\text{SC-IPNLMS}} = 0.7$ , SNR = 20 dB.

## Tracking performance under a time-varying unknown echo system

As mentioned before, acoustic channels are inherently time-varying systems. It is therefore necessary to consider the performances of the adaptive algorithms under a time-varying system model. If the channel changes slowly in time, it can be obtained using the method of image proposed in [1, 23].

Figure 3.19 illustrates the sparseness measure of the generated impulse responses, computed using (3.1), with L = 1024 against iteration number (t). For this illustration, it was assumed that a loudspeaker was fixed at  $4 \times 9.1 \times 1.6$  m inside a room with a dimension of  $8 \times 10 \times 3$  m. In order to introduce a time-varying  $\mathbf{h}(n)$ , a user using a wireless microphone was placed at  $4 \times 9.1 \times 1.6$  m initially and moved away from the loudspeaker across the room at a constant velocity of 0.2 ms<sup>-1</sup>. This example case was



Figure 3.19: Sparseness measure of the generated impulse responses using the method of image with L = 1024 against iteration number (t) and the generated impulse responses at t = 200,4000 and 8000, respectively.

simulated, with 8 kHz sampling frequency, by generating a new receiving room impulse response at every  $40^{\text{th}}$  sample iteration n by varying the microphone position by 1 mm (i.e,: n = 40t). It can be noted from Fig. 3.19 that the bulk delay represented by the leading zeros in the impulse responses is proportional to the separation distance between the loudspeaker and the microphone at that particular time instance.

To evaluate the performances of the conventional, i.e.: NLMS, PNLMS, MPNLMS and IPNLMS, and the proposed sparseness-controlled algorithms for time-varying system identification, a zero mean WGN with  $\sigma_x^2 = 1$  was used as the input signal and the rest of the parameters were carried from the previous simulation setup. Figure 3.20 - 3.22 illustrate the tracking performance of the algorithms under the time-varying echo system, with  $\mathbf{h}(0)$  initialized to the impulse response generated at t = 200 in Fig. 3.19 which describes that the microphone is 20 cm away from the loudspeaker. As it can be seen from the figures that the sparseness-controlled algorithms give better tracking performances, compared to the conventional methods. These results reinforce the suitability of SC-



Figure 3.20: Relative tracking performances of NLMS, PNLMS and SC-PNLMS, using WGN input signal, under a time-varying unknown system.

PNLMS, SC-MPNLMS and SC-IPNLMS to build echo cancelers with improved robustness to echo system sparsity.

## 3.4.4 Computational complexity

The relative complexity of NLMS, PNLMS, SC-PNLMS, IPNLMS, SC-IPNLMS, MPNLMS and SC-MPNLMS in terms of the total number of additions, multiplications, division, logarithm (Log) and comparisons per iteration for adaptation of filter coefficients is assessed in Table 3.1 (for convenient, the computational costs of the classical algorithms are also included here, that are same as in Table 2.1). The additional complexity of the proposed sparseness-controlled algorithms, on top of their conventional method, arises from the computation of the sparseness measure  $\hat{\xi}(n)$ . Given that  $L/(L - \sqrt{L})$  in (3.6) can be computed off-line, the remaining *l*-norms require an additional 2*L* additions and *L* multiplications.

The SC-PNLMS and SC-MPNLMS algorithms additionally require computations



Figure 3.21: Relative tracking performances of NLMS, MPNLMS and SC-MPNLMS, using WGN input signal, under a time-varying unknown system.



Figure 3.22: Relative tracking performances of NLMS, IPNLMS and SC-IPNLMS, using WGN input signal, under a time-varying unknown system.

Algorithm	Addition	Multiplication	Division	Log	Comparison
NLMS	L+3	L+3	1	0	0
PNLMS	2L + 1	5L + 2	2	0	2L
SC-PNLMS	4L + 2	6L + 4	3	0	2L
IPNLMS	3L + 2	5L + 2	2	0	0
SC-IPNLMS	4L + 5	6L + 8	3	0	0
MPNLMS	3L + 1	6L + 2	2	L	2L
SC-MPNLMS	5L + 2	7L + 4	3	L	2L

Table 3.1: Complexity of algorithms' coefficients update - Addition, Multiplication, Division, Logarithm (Log) and Comparison.

for (3.7). Alternatively, a look-up table with values of  $\rho(n)$  defined in (3.7) can be computed for  $0 \leq \hat{\xi}(n) \leq 1$ . Segment PNLMS (SPNLMS) is proposed in [52], to approximate the  $\mu$ -law function in MPNLMS using line segments. Since  $\|\hat{\mathbf{h}}(n)\|_1$  computation is already available from IPNLMS in (2.27), SC-IPNLMS only requires an additional L+3 additions, L+6 multiplications and 1 division. As it can be seen, the increase in the complexity is compromised by the algorithm's performance. Consequently, the trade-off between complexity and performance depend on the design choice for a particular application.

## 3.5 Frequency domain sparseness-controlled algorithms

In this section, a frequency domain adaptive algorithm for AEC is proposed. This new algorithm dynamically adjusts its step-size according to the sparseness variation in acoustic impulse responses that might arise in a mobile environment. Inheriting the beneficial properties of both the fast convergence of IPNLMS [26] and the efficient implementation of the MDF algorithm [28], the proposed sparseness-controlled improved proportionate MDF (SC-IPMDF) algorithm is evaluated using WGN and speech input signals with AIRs of various degrees of sparseness.



Figure 3.23: Convergence of IPMDF for different values of  $\alpha_{\text{IPMDF}}$  using WGN input signal. Impulse responses in Fig. 1.2 (a) and (b) are used as sparse and dispersive AIRs respectively. [ $\tau = 0.2, \mathcal{K} = 8, \text{SNR} = 20 \text{ dB}$ ]

## 3.5.1 The SC-IPMDF algorithm

Considering the highly time-varying nature of the AIR in hands-free telephony devices, the sparseness measure defined in (3.1) can be invoked at each iteration allowing an automatic adjustment of the weighting  $\alpha_{\text{IPMDF}}$ , in (2.56), between proportionate and nonproportionate updating of the filter coefficients of the IPMDF algorithm. The resultant variable weighting factor, as a function of the sparseness of the *k*th subfilter, gives rise to the proposed SC-IPMDF algorithm.

As mentioned respectively in Sections 2.4.3 and 2.5.4, the weighting factor  $\alpha_{\rm IP}$ and  $\alpha_{\rm IPMDF}$  were originally introduced to determine the significance of proportionate and non-proportionate step-size controls. To show the importance of  $\alpha_{\rm IPMDF}$  for IPMDF in terms of the convergence performance, the sparse and dispersive AIRs as shown in Fig. 1.2 were used. Employing the normalized misalignment defined in (2.7) as the performance measure, the IPMDF algorithm was then tested using a zero mean WGN as input while another WGN sequence w(n) was added to give an SNR of 20 dB. As before, it was assumed that the length of  $\hat{\mathbf{h}}(n)$  is equivalent to that of the unknown  $\mathbf{h}(n)$ .

Figure 3.23 shows the effect of  $\alpha_{\text{IPMDF}}$  to the performance of IPMDF for  $\tau = 0.2$ ,  $\mathcal{K} = 8$  in estimating the AIRs as shown in Fig. 1.2 (a) and Fig. 1.2 (b) respectively. As can be seen, a smaller value of  $\alpha_{\text{IPMDF}}$  is desirable for sparse AIR while  $\alpha_{\text{IPMDF}}$  with larger value is favorable for dispersive AIR. It can be observed that  $\alpha_{\text{IPMDF}} = -0.3$  for sparse identification gives worse initial convergence performance for IPMDF. This is because, when  $\alpha_{\text{IPMDF}} = -0.3$ , it emphasizes more the proportionate term than that for the case when  $\alpha_{\text{IPMDF}} = -0.75$ . Since the  $\|\hat{\mathbf{h}}(m)\|_1$  in (2.56) is very small during the initial convergence for a sparse impulse response,  $\alpha_{\text{IPMDF}} = -0.3$  results in more undesirable noisy step-size and therefore, giving worse initial convergence.

The desired effect can be further verified by plotting  $T_{20}$ , which denotes the minimum time for IPMDF to reach the -20 dB normalized misalignment given a specific  $\alpha_{\text{IPMDF}}$  value, against various sparseness associated with 8 simulated AIRs generated using the aforementioned setup in Section 3.4.3. As it can be seen from Fig. 3.24, an approximately monotonic relationship can be observed. By performing a least-squares curve fitting to such relationship using the Matlab's 'polyfit' function with degree 1, a variable weighting factor can be formed as a function of  $\hat{\xi}(m)$  such that the diagonal elements of the step-size control matrix is defined by

$$q_{kN+l}(m) = \frac{1 - \alpha_{\text{SC-IPMDF}}}{2L} + \frac{(1 + \alpha_{\text{SC-IPMDF}})|\widehat{h}_{kN+l}(m)|}{2\|\widehat{\mathbf{h}}(m)\|_1 + \epsilon}$$
(3.9)

where

$$\alpha_{\text{SC-IPMDF}}(m) = 1 - 2\widehat{\xi}(m). \tag{3.10}$$

The proposed SC-IPMDF algorithm, as depicted in Fig. 3.25, can be described by (2.49), (2.57) - (2.59) and (3.9) - (3.10).

## 3.5.2 Performance evaluation

Simulation results are presented next, to evaluate the performance of the proposed SC-IPMDF algorithm compared to the MDF and IPMDF algorithms, in the context of AEC.

#### Experimental setup

A time-varying AIR was obtained by switching between two simulated AIRs shown in Fig. 1.2 with two echo path changes being introduced, i.e., from Fig. 1.2 (b) to 1.2 (a) and



Figure 3.24: Variation of  $\alpha_{\text{IPMDF}}$  for minimum  $T_{20}$  against AIRs with different sparseness.

then back to Fig. 1.2 (b). The convergence performance was measured using the normalized misalignment, defined in (2.7). The length of  $\hat{\mathbf{h}}(n)$  was assumed to be equivalent to that of the unknown  $\mathbf{h}(n)$ . Proportionality control factor  $\alpha_{\text{IPMDF}} = -0.75$  was used for the standard IPMDF algorithm and the other simulation parameters were same as in the case described in Section 3.5.1.

## Convergence performance of SC-IPMDF for AEC

Figure 3.26 first compares the convergence performance of MDF, IPMDF and SC-IPMDF using WGN as the input signal. The step-size parameter for each algorithm was set to  $\tau = 0.2$ . It can be seen from Fig. 3.26 that the convergence rate of SC-IPMDF is as fast as IPMDF for the dispersive case and achieves a faster convergence performance over MDF by up to 7 dB in terms of normalized misalignment. After the echo path change, the SC-IPMDF exhibits a faster tracking performance over both MDF and IPMDF giving approximately 11 dB and 5 dB gain in normalized misalignment, respectively. After the







Figure 3.26: Relative convergence of MDF, IPMDF and SC-IPMDF using WGN input signal with an echo path changes at 8 s and 16 s with  $\tau = 0.2$ ,  $\mathcal{K} = 8$ , SNR = 20 dB. The dispersive and sparse AIRs are as shown in Fig. 1.2 (b) and Fig. 1.2 (a) respectively.

final echo path change, SC-IPMDF maintains its high initial convergence rate over MDF and IPMDF giving respectively 9 dB and 2 dB improvements.

Figure 3.27 shows the results using a male speech input signal. As can be seen, the proposed SC-IPMDF algorithm achieves the highest rate of convergence, giving approximately 1 dB and 4 dB improvements during the initial convergence compared to IPMDF and MDF, respectively, for the dispersive AIR. For sparse AIR, improvements of up to 3 dB and 7 dB normalized misalignment for SC-IPMDF can be seen in comparison with IPMDF and MDF, respectively. It is also noted that SC-IPMDF achieves better steady-state performance than IPMDF and MDF after the final echo path change.

## 3.5.3 Computational complexity

The relative complexity of MDF, IPMDF and SC-IPMDF in terms of the total number of additions, multiplications and divisions per iteration for adaptation of filter coefficients is shown in Table 3.2 for  $\mathcal{K} = 1$ . The additional complexity of the proposed SC-IPMDF arises



Figure 3.27: Relative convergence of MDF, IPMDF and SC-IPMDF using speech input signal with an echo path changes at 9.5 s and 19 s with  $\tau = 0.2$ ,  $\mathcal{K} = 8$ , SNR = 20 dB. The dispersive and sparse AIRs are as shown in Fig. 1.2 (b) and Fig. 1.2 (a) respectively.

from the computation of the sparseness measure  $\hat{\xi}(m)$ . Given that  $L/(L - \sqrt{L})$  in (3.6) can be computed off-line and that  $l_1$ -norm is available from IPMDF weight updation, the proposed SC-IPMDF only requires additional L + 2 additions, L + 3 multiplications and 1 division, compared to that of IPMDF.

## 3.6 Summary

The NLMS algorithm achieves good convergence in dispersive AIRs, whereas the proportionate algorithms, including PNLMS and MPNLMS, perform well in sparse impulse response. The IPNLMS algorithm combines the NLMS update and the proportionate term.

A class of sparseness-controlled algorithms have been proposed. They achieve improved convergence compared to classical NLMS and typical sparse adaptive filtering algorithms. The sparseness measure has been incorporated into PNLMS, MPNLMS and

Algorithm	Addition	Multiplication	Division
MDF	$4L \log_2(L) + 4L$	$4L \log_2(L) + 6L$	L
IPMDF	$5L \log_2(L) + 6L + 2$	$5L \log_2(L) + 8L + 2$	L+2
SC-IPMDF	$5L \log_2(L) + 7L + 4$	$5L \log_2(L) + 9L + 5$	L+3

Table 3.2: Computational complexity of MDF, IPMDF and SC-IPMDF.

IPNLMS for AEC to achieve fast convergence that is robust to the level of sparseness encountered in the impulse response of the echo path. The resulting SC-PNLMS, SC-MPNLMS and SC-IPNLMS algorithms take into account the sparseness measure via a modified coefficient update function. Through a series of simulations, it has been shown that the proposed sparseness-controlled algorithms are robust to variations in the level of sparseness in AIR, with only a modest increase in computational complexity.

In the frequency domain, the SC-IPMDF algorithm has been proposed for AEC, which integrates the sparseness control mechanism into the MDF structure. This has been achieved by forming a variable weighting factor for combining proportionate and non-proportionate tap updating schemes according to the sparseness of the adaptive filter, which allows the proposed SC-IPMDF algorithm to be robust to the sparseness variation of AIRs due to its time-varying nature. The incorporation of the MDF structure ensures a reduced delay for the filter output. Simulation results have shown an improved convergence and tracking performance in terms of normalized misalignment over MDF and IPMDF algorithms.

## 3.6.1 Sparseness-controlled time domain algorithms

 Table 3.3: The Sparseness-controlled time domain Algorithms

$$\begin{split} \widehat{\mathbf{h}}(0) &= \mathbf{0}_{L \times 1} \quad \text{and} \quad 0 < \mu \leq 1 \\ \alpha_{\text{SC-IP}} &= -0.75 \quad (\text{SC-IPNLMS}) \\ \lambda &= 6 \quad \text{and} \quad \rho(n) = 5/L \text{ for } n < L \quad (\text{SC-PNLMS, SC-MPNLMS}) \\ \nu &= 1000 \quad (\text{SC-MPNLMS}) \end{split}$$

$$e(n) = y(n) - \widehat{\mathbf{h}}^T(n-1)\mathbf{x}(n)$$
  

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{Q}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta}$$
  

$$\mathbf{Q}(n-1) = \operatorname{diag}\{q_0(n-1), \dots, q_{L-1}(n-1)\}$$

$$\widehat{\xi}(n) \qquad = \frac{L}{L - \sqrt{L}} \left\{ 1 - \frac{\|\widehat{\mathbf{h}}(n)\|_1}{\sqrt{L} \|\widehat{\mathbf{h}}(n)\|_2} \right\}, \qquad n \ge L$$

## SC-PNLMS

$$q_{l}(n) = \frac{\kappa_{l}(n)}{\frac{1}{L}\sum_{i=0}^{L-1}\kappa_{i}(n)}, \quad 0 \leq l \leq L-1$$
  

$$\kappa_{l}(n) = \max\left\{\rho(n) \times \max\{\gamma, |\hat{h}_{0}(n)|, \dots, |\hat{h}_{L-1}(n)|\}, |\hat{h}_{l}(n)|\right\}$$
  

$$\rho(n) = e^{-\lambda\hat{\xi}(n)}, \quad n \geq L$$

## SC-MPNLMS

$$q_{l}(n) = \frac{\kappa_{l}(n)}{\frac{1}{L} \sum_{i=0}^{L-1} \kappa_{i}(n)}, \quad 0 \le l \le L-1$$
  

$$\kappa_{l}(n) = \max \left\{ \rho(n) \times \max\{\gamma, F(|\hat{h}_{0}(n)|), \dots, F(|\hat{h}_{L-1}(n)|)\}, F(|\hat{h}_{l}(n)|) \right\}$$
  

$$F(|\hat{h}_{l}(n)|) = \ln(1+\nu|\hat{h}_{l}(n)|)$$
  

$$\rho(n) = e^{-\lambda \hat{\xi}(n)}, \quad n \ge L$$

## SC-IPNLMS

$$q_{l}(n) = \left[\frac{1 - 0.5\hat{\xi}(n)}{L}\right] \frac{(1 - \alpha_{\rm SC-IP})}{2L} + \left[\frac{1 + 0.5\hat{\xi}(n)}{L}\right] \frac{(1 + \alpha_{\rm SC-IP})|\hat{h}_{l}(n)|}{2\|\hat{\mathbf{h}}(n)\|_{1} + \delta_{\rm IP}}$$

## 3.6.2 Sparseness-controlled frequency domain algorithms

 $\begin{aligned} 0 < \mu \leq 1 \\ \zeta &= \left[1 - \frac{1}{3L}\right]^{L} \\ \mathbf{\underline{D}}(m-k) &= \text{diag} \{ \text{FFT} \{x(mN-kN-N) \ \dots \ x(mN-kN+N-1) \} \} \\ \mathbf{\underline{e}}(m) &= \mathbf{\underline{y}}(m) - \mathbf{G}_{2L\times 2L}^{0} \sum_{k=0}^{\mathcal{K}-1} \mathbf{\underline{D}}(m-k) \ \mathbf{\underline{\hat{h}}}_{k}(m-1) \\ \\ \mathbf{\underline{S}}_{\text{IPMDF}}(m) &= \zeta \ \mathbf{\underline{S}}_{\text{IPMDF}}(m-1) + (1-\zeta) \ \mathbf{\underline{D}}^{*}(m) \ \mathbf{\underline{D}}(m) \\ \mathbf{\underline{\hat{\xi}}}(m) &= \frac{L}{L - \sqrt{L}} \left\{ 1 - \frac{\|\mathbf{\widehat{h}}(m)\|_{1}}{\sqrt{L} \|\mathbf{\widehat{h}}(m)\|_{2}} \right\} \\ \alpha_{\text{SC}-\text{IPMDF}}(m) &= \left\{ \begin{array}{c} -0.75, \ m=1, \\ 1 - 2\hat{\xi}(m), \ \text{else.} \\ q_{kN+l}(m) &= \frac{1 - \alpha_{\text{SC}-\text{IPMDF}}}{2L} + \frac{(1 + \alpha_{\text{SC}-\text{IPMDF}})|\widehat{h}_{kN+l}(m)|}{2\|\mathbf{\widehat{h}}(m)\|_{1} + \epsilon} \\ &\quad k = 0, 1, \dots \mathcal{K} - 1, \ l = 0, 1, \dots N - 1 \\ \\ \mathbf{Q}_{k}(m) &= \text{diag}\{q_{kN}(m) \ q_{kN+1}(m) \ \dots \ q_{kN+N-1}(m)\} \\ \\ \mathbf{\underline{\hat{h}}}_{k}(m) &= \mathbf{\underline{\hat{h}}}_{k}(m-1) + L\mu\mathbf{Q}_{k}(m)\mathbf{\underline{G}}_{N\times 2N}^{10} \ \mathbf{\underline{D}}^{*}(m-k) \ \left[\mathbf{\underline{S}}_{\text{IPMDF}}(m) + \delta_{\text{IPMDF}}\right]^{-1} \mathbf{\underline{e}}(m) \end{aligned}$ 

## Chapter 4

# A Partitioned Block Proportionate Adaptive Algorithm

In this chapter, an adaptive filter structure that consists of two time domain partition blocks is proposed such that different adaptive algorithms can be used for each part. The earlier version of this work was published in [73, 74].

## 4.1 Introduction

Initially, research on sparse adaptive filters [75] was aimed at NEC while more recently [76] these filters have been investigated in the context of AEC. As explained in Chapter 3, the AIRs, and hence the sparseness of AIRs, are time-varying and depend on factors such as air temperature and pressure and reflectivity of the acoustic environment [11]. The level of sparseness in AIR also varies with the location of the receiving device in an open or enclosed environment. Hence, algorithms developed for AEC are required to be robust to the variations in the sparseness of the acoustic path.

For both sparse and dispersive AIR, the early part of the echo path that consists of the direct path and a few early reflections is almost always sparse while the remaining late part is normally dispersive. To validate this, consider an example case where two AIRs of length L = 1024 were simulated using the method of images [1] in a room of dimension  $8 \times 10 \times 3$  m at a sampling frequency of 8 kHz. Figure 1.3 (a) shows the AIR obtained when the loudspeaker-microphone distance is 0.85 m in the LRMS with 0.3 reflection coefficient. Figure 1.3 (b) illustrates the AIR attained when the loudspeaker-microphone distance is 5 m in the LRMS with 0.53 reflection coefficient. As can be seen from the figure and the sparseness measure [2, 3], the first block is always sparser than the second block. Hence, it would be advantageous to use a sparse adaptive algorithm for the early part of the AIR and a non-sparse adaptive algorithm for the late part.

In this chapter, it is proposed to use two IPNLMS algorithms each with a different proportionate/non-proportionate factor  $\alpha$  for the two corresponding time domain partitioned blocks and develop a fast tracking time domain adaptive algorithm for AEC. The review on the classic IPNLMS can be found in Section 2.4.3 or in Table 2.3. This chapter is organized as follows: Section 4.2 shows how the sparseness of AIRs varies when the echo path is partitioned into two blocks. Incorporating the findings, the proposed partitioned block IPNLMS (PB-IPNLMS) algorithm is developed in Section 4.3 with fixed block sizes, using two different ways to compose the step-size control matrix of each block. Simulation results shown in Section 4.3.3, in the context of AEC, demonstrate a faster tracking performance for both sparse and dispersive AIRs compared to the IPNLMS algorithm with single mixing factor  $\alpha$ . Section 4.4 demonstrates a technique proposed for adaptive block partitioning which improves the robustness of convergence of PB-IPNLMS in a situation requiring identification of time-varying echo path.

## 4.2 Motivation

Let's first express the echo path as

$$\mathbf{h}(n) = [\mathbf{h}_1^T(n) \quad \mathbf{h}_2^T(n)]^T, \tag{4.1}$$

with

$$\mathbf{h}_1(n) = [h_0(n) \dots h_{L_1-1}(n)]^T, \qquad (4.2)$$

$$\mathbf{h}_{2}(n) = [h_{L_{1}}(n) \dots h_{L-1}(n)]^{T}.$$
(4.3)

Here,  $\mathbf{h}_1(n)$  with length  $L_1$  includes the direct path and a few early reflections, which is sparser than  $\mathbf{h}_2(n)$  that includes all other reflections.

Figure 1.3 (a) and (b) show illustrative AIRs for substantially sparse and dispersive cases respectively. With L = 1024, the sparseness measure,  $\xi(\mathbf{h})$  (defined in (3.1)), of the AIR shown in Fig. 1.3 (a) equals 0.76. The measures of the first and second blocks with  $L_1 = \lceil \frac{L}{4} \rceil$  are  $\xi(\mathbf{h}_1) = 0.71$  and  $\xi(\mathbf{h}_2) = 0.37$ . The AIR shown in Fig. 1.3 (b) gives  $\xi(\mathbf{h}) = 0.40, \ \xi(\mathbf{h}_1) = 0.60$  and  $\xi(\mathbf{h}_2) = 0.28$ . As can be seen, the first block is in both cases substantially sparser than the second block.

Figure 4.1 shows the convergence performance of IPNLMS measured using the normalized misalignment (defined in (2.7)), for the sparse AIR shown in Fig. 1.3 (a), with  $\alpha = -1$  and 0.9. A zero mean WGN sequence is used as the input signal while another WGN sequence w(n) is added to give an SNR of 20 dB and  $\mu = 0.3$ . It can be seen from Fig. 4.1 (a)-(c) that, IPNLMS with  $\alpha = 0.9$  is better for the first block during the initial phase and therefore giving an overall faster initial convergence, while  $\alpha = -1$  (NLMS) is better for the second block and thus giving an improved overall steady-state performance.

The same observation is seen from Fig. 4.2 for the dispersive AIR shown in Fig. 1.3 (b), under the same experimental setup as before. As the first block of the AIR contains the dominant parts of the echo path, allocating larger individual step-sizes for the coefficients in the block gives faster initial convergence performance. Moreover, through the simulations, it was found that distributing almost equal step-sizes for the second block gives better steady-state performance. As a consequence of this important observation, a new adaptation approach is proposed for IPNLMS as described below.

## 4.3 Fixed length partitioned block IPNLMS

In this Section, an illustrative example is provided to show how the sparseness of AIRs varies when the echo path is partitioned into two blocks with different sizes. This serves as a motivation to develop a new algorithm which improves the the tracking performance of IPNLMS. In addition, it is also demonstrated how the sums of the composite diagonal elements of  $\mathbf{Q}(n)$  for the two blocks affect the overall performance of the proposed algorithm.



Figure 4.1: Normalized misalignments (NM) of IPNLMS with different mixing parameters,  $\alpha$ , for identification of a sparse impulse response.

To achieve the desired effect explained in Section 4.2, it is proposed to use IPNLMS with the mixing parameter  $\alpha_1$  close to 1 as the sparse algorithm for the first block of length  $L_1$ , where the diagonal elements  $q_l$  of the step-size control matrix of the first block  $\mathbf{Q}_1(n)$ for the proposed partitioned block IPNLMS (PB-IPNLMS) algorithm can be expressed as

$$q_l(n) = \frac{(1-\alpha_1)}{2L_1} + \frac{(1+\alpha_1)|\hat{h}_l(n)|}{2\|\widehat{\mathbf{h}}_1(n)\|_1 + \delta_{\mathrm{IP}}}, \quad 0 \le l \le L_1 - 1,$$
(4.4)

$$\mathbf{Q}_1(n-1) = \operatorname{diag} \{ q_0(n-1), \dots, q_{L_1-1}(n-1) \},$$
(4.5)



Figure 4.2: Normalized misalignments (NM) of IPNLMS with different mixing parameters,  $\alpha$ , for identification of a dispersive impulse response.

where diag{} is the diagonal operator. For the second block, as it is more dispersive compared to the first block, it is proposed to employ IPNLMS with the mixing parameter  $\alpha_2$  ( $\alpha_2 < \alpha_1$ ) close to -1, where  $q_l$  of the second block  $\mathbf{Q}_2(n)$  for PB-IPNLMS can be formulated as

$$q_l(n) = \frac{(1-\alpha_2)}{2(L-L_1)} + \frac{(1+\alpha_2)|h_l(n)|}{2\|\widehat{\mathbf{h}_2}(n)\|_1 + \delta_{\mathrm{IP}}}, \quad L_1 \le l \le L-1,$$
(4.6)

$$\mathbf{Q}_{2}(n-1) = \operatorname{diag} \{ q_{L_{1}}(n-1), \dots, q_{L-1}(n-1) \}.$$
(4.7)

When using different update rules, the constraint on  $tr{\mathbf{Q}(n)}$  of PB-IPNLMS, which is composed of  $\mathbf{Q}_1(n)$  and  $\mathbf{Q}_2(n)$ , still needs to be  $tr{\mathbf{Q}(n)} = 1$  for very small values of  $\delta_{\text{IP}}$ . Although this constraint can be satisfied in many ways, the following two different approaches are considered in this work.

## 4.3.1 Non-proportionate weighting

In the first approach, equal weights are allocated as

$$\mathbf{Q}(n-1) = \begin{bmatrix} 0.5 \ \mathbf{Q}_1(n-1) & \mathbf{0}_{L_1 \times (L-L_1)} \\ & & \\ \mathbf{0}_{(L-L_1) \times L_1} & 0.5 \ \mathbf{Q}_2(n-1) \end{bmatrix},$$
(4.8)

to satisfy the constraint on  $\mathbf{Q}(n)$  of PB-IPNLMS. This approach has been dubbed the 'non-proportionate PB-IPNLMS'. Thus, the non-proportionate PB-IPNLMS algorithm is described by (2.5), (2.20) and (4.4)-(4.8), as also specified in Table 4.1. It was noted through simulations that the non-proportionate approach works well only if  $\|\mathbf{h}_1(n)\|_1 \approx$  $\|\mathbf{h}_2(n)\|_1$ , but in practice this condition is seldom met.

## 4.3.2 Proportionate weighting

As  $\mathbf{h}_1(n)$  and  $\mathbf{h}_2(n)$  are unobservable, it is proposed to allocate weights proportional to the ratio between  $\|\widehat{\mathbf{h}}_1(n)\|_1$  and  $\|\widehat{\mathbf{h}}(n)\|_1$ , while satisfying  $\operatorname{tr}\{\mathbf{Q}(n)\} \sim 1$  for very small values of  $\delta_{\mathrm{IP}}$ . This approach is referred as the 'proportionate PB-IPNLMS'.

It is noted that, for a sparse system identification with  $\widehat{\mathbf{h}}(0) = \mathbf{0}$ , the ratio between  $\|\widehat{\mathbf{h}}_1(n)\|_1$  and  $\|\widehat{\mathbf{h}}(n)\|_1$  is close to 1 during the initial stage and decays to a value  $\kappa$  ( $0 \leq \kappa \leq 1$ ), which on average is greater than 0.5, due to the fact that the first block contains almost all the dominant echo. However, for a dispersive AIR, the ratio quickly decays to a value less than  $\kappa$ , as the second block also has many weaker acoustic reflections. In this approach, the proportionality is controlled by  $\beta(n)$  which is defined as follows, for n > 1,

in order to calculate the composed step-size control matrix  $\mathbf{Q}(n-1)$ :

$$\beta(n) = \begin{cases} \chi \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}}, & \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}} > \kappa, \\ \chi^{-1} \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}}, & \text{otherwise} \end{cases}$$
(4.9)

$$\mathbf{Q}(n-1) = \begin{bmatrix} \beta(n) \ \mathbf{Q}_1(n-1) & \mathbf{0}_{L_1 \times (L-L_1)} \\ & & \\ \mathbf{0}_{(L-L_1) \times L_1} & [1-\beta(n)] \ \mathbf{Q}_2(n-1) \end{bmatrix}.$$
(4.10)

With the formulation of  $\beta(n)$  in (4.9) for the first block, a weight that is directly proportional to the ratio between  $\|\widehat{\mathbf{h}}_1(n)\|_1$  and  $\|\widehat{\mathbf{h}}(n)\|_1$  is allocated, when the ratio is above a threshold value  $\kappa$ , where  $\chi$  (0 <  $\chi$  < 1) is introduced to allocate almost equal weights for the two blocks after the initial convergence. The factor  $\chi$  also ensures that  $1 - \beta(n)$  for the second block is always greater than zero, and therefore avoids stalling the adaptation of  $\widehat{\mathbf{h}}_2$ . Likewise,  $\chi^{-1}$  (which is  $\geq 1$ ) ensures that  $\beta(n)$  is never very small, thereby avoiding stalling the adaptation of  $\widehat{\mathbf{h}}_1$ .

When the ratio is below or equal to  $\kappa$ , the first block gets higher weight during the initial stage of a dispersive system identification and gradually reduces such that the second block gets more weight. With the experimentally determined values of  $\chi = 0.8$  and  $\kappa = 0.5$ , proportionate PB-IPNLMS not only works well in both sparse and dispersive circumstances, but also performs well when the scenario involves a time-varying system. The proposed proportionate PB-IPNLMS algorithm is thus described by (2.5), (2.20), (4.4)-(4.7), (4.9) and (4.10), as specified in Table 4.1.

## 4.3.3 Performance evaluation

Simulation results are presented to evaluate the performance of the proposed PB-IPNLMS algorithm. Throughout the simulations, algorithms were tested using a zero mean WGN signal as input while another WGN sequence w(n) was added to give an SNR of 20 dB. It was assumed that the length of the adaptive filter L = 1024 was equivalent to that of

the unknown system. Two receiving room impulse responses  $\mathbf{h}(n)$  for AEC simulations have been used, with an echo path change at 4 s. The AIR was changed from that shown in Fig. 1.3 (a) to (b) and  $\mu = 0.3$ . For PB-IPNLMS,  $L_1$  was fixed to 256 such that the first partitioned block contained the direct path and early reflections and,  $\alpha_1 = 0.9$ and  $\alpha_2 = -1$  were used, while  $\chi = 0.8$  and  $\kappa = 0.5$  were employed specifically for the proportionate PB-IPNLMS algorithm.

Figure 4.3 compares the overall performance of IPNLMS, in terms of normalized misalignment, with  $\alpha = -1$  and 0.9 and PB-IPNLMS using the non-proportionate and proportionate weight allocation approaches, while Fig. 4.4 shows the normalized misalignments of the first and second blocks. As it can be seen that the proposed non-proportionate PB-IPNLMS achieves approximately 3 dB improvements over the IPNLMS with  $\alpha = -1$ , and performs similar to the IPNLMS with  $\alpha = 0.9$  during the initial stage of the sparse system identification. After the echo path change, a similar performance pattern was observed between 4-5 s. However, below the -10 dB normalized misalignment level, the non-proportionate PB-IPNLMS algorithm performs similar to the IPNLMS with  $\alpha = -1$ , and achieves approximately 3 dB better convergence performance over the IPNLMS with  $\alpha = 0.9$ . Moreover, the proportionate PB-IPNLMS gives better performance compared to all the algorithms, notably a 2 dB improvement over the non-proportionate PB-IPNLMS after the echo path changes to a dispersive AIR. PB-IPNLMS achieves this better initial performance by exploiting the beneficial properties of the IPNLMS with  $\alpha = 0.9$  for the first block and allocates step-sizes similar to the IPNLMS with  $\alpha = -1$  for the second block, as illustrated in Fig. 4.4.

Figure 4.5 shows a detailed study on the evolution of  $\beta$  in (4.9), which is equivalent to  $\|\hat{\mathbf{h}}_1(n)\|_1$  for the IPNLMS algorithm with  $\alpha = -1$  and  $\alpha = 0.9$  and 0.5 for the non-proportionate PB-IPNLMS algorithm, throughout the simulation time for the overall performance illustrated in Fig. 4.3. As can be seen, the IPNLMS with  $\alpha = -1$  gives a small weight,  $\beta$ , for the first block at all times, therefore gives higher weight,  $(1 - \beta)$ , for the second block to achieve a better steady-state performance. While, the IPNLMS with  $\alpha = 0.9$  allocates higher weight during the early stages of before and after the echo path change, giving faster convergence performance. The proportionate PB-IPNLMS exploits the beneficial properties of IPNLMS with  $\alpha = -1$  and 0.9 and hence achieves the better overall performance.



Figure 4.3: Relative convergence of IPNLMS for  $\alpha = -1$  and 0.9 and PB-IPNLMS with non-proportionate and proportionate weight allocation approaches, using WGN input signal with an echo path change at 4 s.Impulse response is changed from that shown in Fig. 1.3 (a) to (b) and  $\mu = 0.3$ , SNR = 20 dB.

The same experiment was repeated with the exact parameter settings using a correlated unity-variance AR(2) process given by [75]

$$x(n) = 0.73 \ x(n-1) - 0.8 \ x(n-2) + s(n), \tag{4.11}$$

where s(n) is a white Gaussian noise with  $\sigma_s^2 = 0.3$ , and the relative performances are shown in Fig. 4.6. As observed in the WGN input signal case, the proportionate PB-IPNLMS outperforms all the aforementioned algorithms before and after the echo path change.



Figure 4.4: The normalized misalignments (NM) for the overall convergence performance illustrated in Fig. 4.3

## 4.4 Variable length partitioned block IPNLMS

The study in Section 4.3 on the block partitioned approach proposed to keep  $L_1$  constant. As the echo path is time-varying in practical cases, such as in an application scenario involving hands-free telephony in which the user moves through significantly different acoustic environments during the call. For example, starting in an office, then moving into an elevator, then a lobby, out into the open air and finally into a car. During this call the level and nature of sparseness in the AIR to be identified both change significantly. Hence, the block size  $L_1$  should be made time dependent, to adaptively control the structure of the adaptive filter to deal effectively with such changes, for a robust convergence performance


Figure 4.5: Evolution of  $\beta$  in (4.9), which is equivalent to  $\|\hat{\mathbf{h}}_1(n)\|_1$  for the IPNLMS algorithm with  $\alpha = -1$  and  $\alpha = 0.9$  and 0.5 for the non-proportionate PB-IPNLMS algorithm.

of PB-IPNLMS.

A filter length control (FLC) algorithm was also addressed in [77]. However, the objective in [77] was to find the length of an adaptive filter that identifies a system with unknown and time-varying system memory.

In this section, it is shown how the AIRs can be partitioned into two blocks adaptively, so that the first block, with the dominant direct path and early reflections, is more sparse than that of the second block. This serves as a motivation to develop a new algorithm which improves the robustness and the tracking performance compared to PB-IPNLMS.

#### 4.4.1 Automatic control of the block length $L_1$

As shown in Section 4.3, the composed step-size control matrix in (4.8) and (4.10) satisfies the constraint on tr{Q(n)}, even though this constraint can be satisfied in many other



Figure 4.6: Relative convergence of IPNLMS for  $\alpha = -1$  and 0.9 and PB-IPNLMS with non-proportionate and proportionate weight allocation approaches, using the input signal generated by (4.11) with an echo path change at 4 s.Impulse response is changed from that shown in Fig. 1.3 (a) to (b) and  $\mu = 0.3$ , SNR = 20 dB.

ways. To motivate this work, we begin by defining

$$\chi(n|L_1) = \frac{\|\mathbf{h}_1(n)\|_1}{\|\mathbf{h}(n)\|_1}.$$
(4.12)

Figure 4.7 shows the performance of PB-IPNLMS with the non-proportionate weighting technique, for different  $\chi$  values, which is controlled by varying  $L_1$ , to illustrate the time taken to reach -20 dB normalized misalignment defined in (2.7). The input signal was generated using an AR(2) process given by [75]

$$x(n) = 0.4 x(n-1) - 0.4 x(n-2) + s(n),$$
(4.13)

where s(n) is a WGN with  $\sigma_s^2 = 0.77$ . Another WGN sequence was chosen for w(n) such that an SNR of 20 dB was obtained. A step-size of  $\mu = 0.3$  was used in this experiment. As it can be seen from the result, the non-proportionate composition of **Q** in (4.8) works



Figure 4.7: Time to reach -20 dB normalized misalignment level for different values of  $\chi$  in (4.12) for PB-IPNLMS using the input signal generated by (4.13). Impulse response in Fig. 1.3 (b) and (c) were used as sparse AIR and dispersive AIR respectively.  $[\mu = 0.3, \text{ SNR} = 20 \text{ dB}]$ 

well in this case only if

$$\chi(n) \approx 0.5,\tag{4.14}$$

for both sparse and dispersive AIRs. It is also interesting to note that, when the ratio is 0, PB-IPNLMS employs the non-sparse algorithm. If the ratio is 1, it employs the sparse algorithm. In addition, as it can be noted from the result that, for each case of  $\chi$ , PB-IPNLMS has a higher rate of convergence for a sparse system compared to a dispersive system. This is due to the initialization choice of  $\hat{\mathbf{h}}(0) = \mathbf{0}_{L\times 1}$ , where most filter coefficients are initialized close to their optimal values.

The condition in (4.14) can be adopted as a technique to partition the block, so that the first block contains the direct path and few early reflections. The AIRs in Fig. 4.8 were partitioned into two blocks using different sizes for the first block with length  $L_1$ , giving (a)  $L_1 = 316$ , (b)  $L_1 = 196$  and (c)  $L_1 = 356$  respectively. The three AIRs of length L = 1024 were simulated under different conditions using the method of images [1] in a LRMS at a sampling frequency of 8 kHz.



Figure 4.8: Acoustic impulse responses obtained using the method of images [1].  $\xi(\mathbf{h}), \xi(\mathbf{h}_1)$  and  $\xi(\mathbf{h}_2)$  respectively denote the sparseness measures [3] of the full impulse response, the first block with size  $L_1$  and the second block.

Figure 4.8 (a) and (c) show the AIRs obtained when the loudspeaker-microphone distances, a, are 11.4 m and 5 m in a room of dimension  $8 \times 10 \times 3$  m, with 0.2 and 0.53 as the reflection coefficients, respectively. Figure 4.8(b) illustrates the AIR attained when a = 4.2 m in a room of dimension  $10 \times 15 \times 3$  m, with 0.2 as the reflection coefficient. The sparseness measures of these AIRs are computed using (3.1). As can be seen from the sparseness measures of the first blocks ( $\xi(\mathbf{h}_1)$ ) and second blocks ( $\xi(\mathbf{h}_2)$ ), the first blocks in all cases are more sparse than the second blocks. As a consequence of this important finding, a new adaptation technique for PB-IPNLMS is proposed with an efficient mechanism to make  $L_1$  time-dependent, as described below.

#### 4.4.2 The proposed VLPB-IPNLMS algorithm

An example set of AIRs illustrated in Fig. 4.8 (a)-(c) show that in order to satisfy the condition in (4.14),  $L_1$  varies depending on the initial bulk delays and the overall sparseness levels of the AIRs. An efficient mechanism for the automatic control of  $L_1(n)$  can be derived according to the ratio between  $\|\widehat{\mathbf{h}}_1(n)\|_1$  and  $\|\widehat{\mathbf{h}}(n)\|_1$  as follows

$$L_{1}(n) = \begin{cases} L/4, & n < L \\ L_{1}(n-1) + \triangle \ell, & \frac{\|\widehat{\mathbf{h}}_{1}(n)\|_{1}}{\|\widehat{\mathbf{h}}(n)\|_{1}} < \kappa_{\min} \\ L_{1}(n-1) - \triangle \ell, & \frac{\|\widehat{\mathbf{h}}_{1}(n)\|_{1}}{\|\widehat{\mathbf{h}}(n)\|_{1}} > \kappa_{\max} \\ L_{1}(n-1), & \text{otherwise} \end{cases}$$
(4.15)

where  $\Delta \ell \ll L$  denotes the number of coefficients by which  $L_1(n)$  can be enlarged/reduced. The minimum and maximum threshold values  $\kappa_{\min}$  ( $0 \ll \kappa_{\min} < 0.5$ ) and  $\kappa_{\max}$  ( $0.5 < \kappa_{\max} \ll 1$ ) can be specified in order to define the region to satisfy (4.14). With the formulation in (4.15),  $L_1$  initializes to L/4 which has been chosen empirically and kept as constant for n < L. The first block size  $L_1$  gets enlarged, when the ratio between  $\|\hat{\mathbf{h}}_1(n)\|_1$  and  $\|\hat{\mathbf{h}}(n)\|_1$  is less than a threshold value  $\kappa_{\min}$ , resulting adding some more early reflections into the first block. On the other hand,  $L_1$  reduces to exclude some of the weaker reflections when the ratio is greater than  $\kappa_{\max}$ . Otherwise,  $L_1$  stays the same. Thus, the proposed variable length partitioned block IPNLMS (VLPB-IPNLMS) algorithm is described by (2.5), (2.20), (4.4)-(4.8) and (4.15).

#### 4.4.3 Performance evaluation

Simulation results are presented to evaluate the performance of the proposed VLPB-IPNLMS algorithm. Throughout the simulations, algorithms were tested using the input signal generated by (4.13) while w(n) was a WGN sequence chosen such that an SNR of 20 dB was obtained. It was assumed that the length of the adaptive filter L = 1024was equivalent to that of the unknown system. Three receiving room impulse responses, simulated using the method of images [1], for AEC simulations have been used, with an echo path changed from that shown in Fig. 4.8 (a) to (b) and then to (c) and  $\mu = 0.3$ . For PB-IPNLMS and VLPB-IPNLMS,  $\alpha_1 = 0.9$  and  $\alpha_2 = -1$  were used respectively as sparse and dispersive adaptive algorithms, while  $\Delta \ell = 10$ ,  $\kappa_{\min} = 0.45$  and  $\kappa_{\max} = 0.65$ were employed specifically for the VLPB-IPNLMS algorithm. The  $\Delta \ell$  value was chosen empirically and the lower and upper threshold values were chosen from the results shown in Fig. 4.7.

Figure 4.9 (a) compares the overall performance of IPNLMS with  $\alpha = -1$  and 0.9, PB-IPNLMS using the proportionate weight allocation technique and VLPB-IPNLMS, in terms of the normalized misalignment. As it can be seen that the proposed VLPB-IPNLMS achieves both a high rate of initial convergence similar to IPNLMS with  $\alpha = 0.9$  for the identification of the first two AIRs and also approximately 3 dB better convergence performance for the identification of the third (dispersive) AIR. Moreover, VLPB-IPNLMS gives more than 7 dB better initial convergence performance and similar steady-state performance compared to IPNLMS with  $\alpha = -1$  for the first two sparse AIRs and almost equivalent convergence performance for the dispersive AIR. Notably, VLPB-IPNLMS achieves more than 5 dB faster initial convergence compared to PB-IPNLMS for the sparse AIRs and similar performance for the dispersive AIR.

The VLPB-IPNLMS algorithm achieves this better performance by varying the block size  $L_1$ , as illustrated in Fig. 4.9 (c), such that the first block of the AIR with the dominant parts of the echo path gets larger individual step-sizes and therefore achieves faster initial convergence performance. Moreover, the second partitioned block is set to have almost equal step-sizes and as a result attains better steady-state performance. Here, the adaptation of the  $L_1$  was automatically controlled by the evolution of the ratio shown in Fig. 4.9 (b).



Figure 4.9: a) Relative convergence of IPNLMS for  $\alpha = -1$  and 0.9, PB-IPNLMS with proportionate weight allocation technique and VLPB-IPNLMS; b) evolution of  $\frac{\|\hat{\mathbf{h}}_1(n)\|_1}{\|\hat{\mathbf{h}}(n)\|_1}$ ; c) evolution of  $L_1$ ; The input signal was generated using (4.13) with an echo path change at 10 s and 20 s, from that shown in Fig. 4.8 (a) to (b) and then to (c),  $\mu = 0.3$  and SNR = 20 dB.

#### 4.5 Summary

A partitioned block IPNLMS algorithm has been presented, with two different approaches to allocate weights for the composition of the step-size control matrix of the two blocks. The proposed algorithm achieves improved convergence compared to classical IPNLMS with fixed single proportional/non-proportionate factor  $\alpha$ . For the proposed PB-IPNLMS algorithm with proportionate weighting, the ratio between the  $\ell_1$ -norm of the first block's estimated filter coefficients and that of the overall filter coefficient has been incorporated into IPNLMS for AEC to achieve fast convergence for both sparse and dispersive acoustic echo paths.

In practical scenarios of AEC for hands-free mobile telephony devices, the level of sparseness in the AIR can be highly variable. To deal with this issue, a partitioned block IPNLMS algorithm has been developed with a control mechanism for the dynamic adjustment of the block size. The proposed algorithm achieves improved convergence. For the proposed VLPB-IPNLMS, a self-configuration method has been incorporated based on the ratio between the  $\ell_1$ -norm of the first block's estimated filter coefficients and that of the overall filter coefficient into PB-IPNLMS to achieve faster convergence for both sparse and dispersive acoustic echo paths with variable bulk delay.

#### 4.5.1 The partitioned block IPNLMS algorithms

 Table 4.1: The Partitioned Block IPNLMS algorithms

$$\begin{split} e(n) &= y(n) - \widehat{\mathbf{h}}^{T}(n-1)\mathbf{x}(n) \\ \widehat{\mathbf{h}}(n) &= \widehat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{Q}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^{T}(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta} \\ \widehat{\mathbf{h}}_{1}(n) &= [\widehat{h}_{0}(n) \dots \widehat{h}_{L_{1}-1}(n)]^{T} \\ \widehat{\mathbf{h}}_{2}(n) &= [\widehat{h}_{L_{1}}(n) \dots \widehat{h}_{L-1}(n)]^{T} \\ \widehat{\mathbf{h}}(n) &= [\widehat{\mathbf{h}}_{1}(n)^{T} \quad \widehat{\mathbf{h}}_{2}(n)^{T}]^{T} \\ q_{l}(n) &= \frac{(1-\alpha_{1})}{2L_{1}} + \frac{(1+\alpha_{1})|\widehat{h}_{l}(n)|}{2\|\widehat{\mathbf{h}}_{1}(n)\|_{1} + \delta_{\mathrm{IP}}}, \quad 0 \leq l \leq L_{1} - 1 \\ \mathbf{Q}_{1}(n-1) &= \operatorname{diag}\{q_{0}(n-1), \dots, q_{L_{1}-1}(n-1)\} \\ q_{l}(n) &= \frac{(1-\alpha_{2})}{2(L-L_{1})} + \frac{(1+\alpha_{2})|\widehat{h}_{l}(n)|}{2\|\widehat{\mathbf{h}}_{2}(n)\|_{1} + \delta_{\mathrm{IP}}}, \quad L_{1} \leq l \leq L - 1 \\ \mathbf{Q}_{2}(n-1) &= \operatorname{diag}\{q_{L_{1}}(n-1), \dots, q_{L-1}(n-1)\} \end{split}$$

Non-proportionate PB-IPNLMS

$$L_{1} = L/4$$
  

$$\mathbf{Q}_{(n-1)} = \begin{bmatrix} 0.5 \ \mathbf{Q}_{1}(n-1) & \mathbf{0}_{L_{1} \times (L-L_{1})} \\ \mathbf{0}_{(L-L_{1}) \times L_{1}} & 0.5 \ \mathbf{Q}_{2}(n-1) \end{bmatrix}_{L \times L}$$

Proportionate PB-IPNLMS

$$L_{1} = L/4$$

$$\beta(n) = \begin{cases} \chi \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}}, & \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}} > \kappa, \\ \chi^{-1} \frac{\|\hat{\mathbf{h}}_{1}(n)\|_{1}}{\|\hat{\mathbf{h}}(n)\|_{1}}, & \text{otherwise} \end{cases}$$

$$\mathbf{Q}(n-1) = \begin{bmatrix} \beta(n) \mathbf{Q}_{1}(n-1) & \mathbf{0}_{L_{1} \times (L-L_{1})} \\ \mathbf{0}_{(L-L_{1}) \times L_{1}} & [1-\beta(n)] \mathbf{Q}_{2}(n-1) \end{bmatrix}_{L \times L}$$

Variable length PB-IPNLMS

$$L_{1}(n) = \begin{cases} L/4, & n < L \\ L_{1}(n-1) + \triangle \ell, & \frac{\|\widehat{\mathbf{h}}_{1}(n)\|_{1}}{\|\widehat{\mathbf{h}}(n)\|_{1}} < \kappa_{\min} \\ L_{1}(n-1) - \triangle \ell, & \frac{\|\widehat{\mathbf{h}}_{1}(n)\|_{1}}{\|\widehat{\mathbf{h}}(n)\|_{1}} > \kappa_{\max} \\ L_{1}(n-1), & \text{otherwise} \end{cases}$$
$$\mathbf{Q}(n-1) = \begin{bmatrix} 0.5 \ \mathbf{Q}_{1}(n-1) & \mathbf{0}_{L_{1} \times (L-L_{1})} \\ \mathbf{0}_{(L-L_{1}) \times L_{1}} & 0.5 \ \mathbf{Q}_{2}(n-1) \end{bmatrix}_{L \times L}$$

## Chapter 5

# Performance Analysis for Time-Varying System Identifications

In this chapter, a theoretical model is developed to predict the tracking performance of the IPNLMS algorithm, measured using the mean-squared error, for sparse and dispersive time-varying system. The earlier version of this work was published in [78].

#### 5.1 Introduction

Adaptive filters have been used to achieve AEC in time-varying environments by tracking the acoustic echo path and thereby continuously predicting the acoustic echo that is received by the microphone. By introducing a controlled mixture of proportionate (PNLMS) and non-proportionate (NLMS) adaptation, the IPNLMS algorithm is currently a favorable choice employed in real-time applications, as it performs better than NLMS and PNLMS regardless of the level of sparseness in the AIR with a modest increased in the computational complexity [26].

As explained in Chapter 3, the sparseness of the AIR is dependent on (a) the distance from the loudspeaker to the microphone and (b) the nature and number of sound reflecting surfaces in the vicinity of the microphone. Both (a) and (b) may vary with

time when a wireless microphone is used and when the terminal is mobile, respectively, and therefore the sparseness will also be time-varying. An everyday example of timevarying sparseness in the AIR is during a hands-free phone call when the caller starts in an elevator, then moves through the lobby of a building and finally moves outside onto the street. An illustration is shown in Fig. 1.2 using the image model [1], with different distances between loudspeaker and microphone in a room.

In order to explore the algorithms' performances under such continuously timevarying condition, a first-order Markov process [15, 79] is commonly used to model the unknown AIR. Tracking capability of time-variations by the LMS algorithm has already been the subject of several studies including [79, 80]. Recently, the transient behavior of a modified PNLMS algorithm was studied in [30].

In this chapter, the analysis of the tracking performance of IPNLMS, for sparse and dispersive time-varying systems, is presented. For the analysis, the methodology proposed in [30] is adopted. A review in IPNLMS can be found in Section 2.4.3 or in Table 2.3. This chapter is organized as follows: the modified first-order Markov model [15, 81] used for the time-varying unknown system is reviewed in Section 5.2 while Section 5.3 develops expressions to predict both the transient and steady-state performances, measured in terms of mean-square error, of IPNLMS algorithm for time-varying systems. Simulation results shown in Section 5.4 demonstrate that the predicted performance and the actual performance (i.e., ensemble average of simulations) are very similar when the unknown system changes in the context of AEC.

#### 5.2 Time-varying system model

If the channel changes slowly in time, it can be adequately represented by a first order Markov model [15]. The modified first-order Markov model [15, 81] is employed to represent a time-varying unknown system

$$\mathbf{h}(n) = \varepsilon \mathbf{h}(n-1) + \sqrt{1 - \varepsilon^2} \, \mathbf{s}(n), \tag{5.1}$$

where

$$\mathbf{h}(n) = [h_0(n) \ h_1(n) \ \dots \ h_{L-1}(n)]^T, \tag{5.2}$$

is the unknown room impulse response with length L and

$$\mathbf{s}(n) = [s_0(n) \ s_1(n) \ \dots \ s_{L-1}(n)]^T, \tag{5.3}$$

is a random sequence with elements drawn from a normal (Gaussian) distribution with zero mean and variance  $\sigma_s^2$ . The parameter  $\varepsilon$  ( $0 \ll \varepsilon < 1$ ) controls the relative contributions to the instantaneous values of the "system memory" and "innovations" [15]. It can be noted that  $\varepsilon = 1$  represents a time-invariant environment.

Figure 5.1 shows the sparseness measure (defined in (3.1)) of the generated impulse responses with L = 1024, using the modified Markov model in (5.1) with parameters set to  $\sigma_s^2 = 1$  and  $\varepsilon = 0.9999$ , against iteration number (n). As time progresses, this dynamic model keeps  $E\{\mathbf{h}(n)\}$  constant and the covariance matrix of  $\mathbf{h}(n)$  tends to a finite steady-state value that is equal to the covariance matrix of the sequence  $\{\mathbf{s}(n)\}$ . Hence, the model always gives a dispersive system as  $n \to \infty$ . However, by initializing  $\mathbf{h}(0)$  to a sparse system and choosing a value close to 1 for  $\varepsilon$ , this model can be employed to simulate a slowly time-varying sparse system.

#### 5.3 Recursive mean-square error analysis

The main purpose of this study is to explore the sensitivity of the transient and stead-state performances to the input signal x(n), additive noise w(n) and step-size  $\mu$ . To motivate for the convergence analysis in terms of the mean-square output error (MSE) in (2.6), the following generalized equation (also defined in (2.20)) of the adaptive algorithms is considered

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{Q}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta} , \qquad (5.4)$$

with the notations as defined in Section 2.4. However, the generalized formulation that is developed in Section 5.3.1, with the set of assumptions in Section 5.3.2, is proceeded by considering the MSE specifically for the IPNLMS algorithm (described in Section 2.4.3).



Figure 5.1: Sparseness measure of the generated impulse responses using the modified Markov model with L = 1024,  $\sigma_s^2 = 1$  and  $\beta = 0.9999$ , against iteration number (n) and generated impulse responses at n = 0,1000 and 8000, respectively.

#### 5.3.1 General formulation

With the weight deviation vector defined as

$$\mathbf{z}(n) = \mathbf{h}(n) - \widehat{\mathbf{h}}(n), \tag{5.5}$$

using (2.3) and (2.5), e(n) can be rearranged as

$$e(n) = w(n) + \sum_{j=0}^{L-1} x_j(n) z_j(n), \qquad (5.6)$$

where  $x_l(n) \triangleq x(n-l+1)$ . Hence, the MSE in (2.6) can be reformulated as

$$MSE(n) = E \{e^{2}(n)\} = \sigma_{w}^{2} + \sigma_{x}^{2} \sum_{l=0}^{L-1} E \{z_{l}^{2}(n)\},$$
(5.7)

where  $\sigma_w^2$  and  $\sigma_x^2$  are the variances of the additive noise and the input signal, respectively. In order to calculate (5.7), the expected values of the square weight deviations,  $E\left\{z_l^2(n)\right\}$ , can be calculated as follows.

By substituting (5.4) and (5.1) into (5.5), with e(n) defined as in (5.6), the component-wise weight deviation is given by

$$z_{l}(n) = z_{l}(n-1) + (\varepsilon - 1)h_{l}(n-1) + \sqrt{1 - \varepsilon^{2}}s_{l}(n) - \left[\frac{\mu q_{l}(n-1)x_{l}(n)}{\mathbf{x}^{T}(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta}\right] \left[w(n) + \sum_{j=0}^{L-1} x_{j}(n)z_{j}(n)\right].$$
(5.8)

In order to calculate the MSE in (2.6), the component-wise form of the recursion for the square of the wight deviation in (5.8) is given by

$$\begin{aligned} z_l^2(n) &= z_l^2(n-1) + 2z_l(n-1)(\varepsilon - 1)h_l(n-1) + \\ &(\varepsilon - 1)^2 h_l^2(n-1) + 2z_l(n-1)\sqrt{1 - \varepsilon^2} s_l(n) + \\ &(\varepsilon - 1)h_l(n-1)\sqrt{1 - \varepsilon^2} s_l(n) + (1 - \varepsilon^2) s_l^2(n) - \\ &\left[ \frac{2\mu(\varepsilon - 1)h_l(n-1)q_l(n-1)x_l(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta} \right] \left[ w(n) + \sum_{j=0}^{L-1} x_j(n)z_j(n) \right] - \\ &\left[ \frac{2\mu z_l(n-1)q_l(n-1)x_l(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta} \right] \left[ w(n) + \sum_{j=0}^{L-1} x_j(n)z_j(n) \right] + \\ &\left[ \frac{\mu^2 q_l^2(n-1)x_l^2(n)}{\left(\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta\right)^2} \right] \left[ w(n) + \sum_{j=0}^{L-1} x_j(n)z_j(n) \right]^2 - \\ &\left[ \frac{2\mu\sqrt{1 - \varepsilon^2} s_l(n)q_l(n-1)x_l(n)}{\mathbf{x}^T(n)\mathbf{Q}(n-1)\mathbf{x}(n) + \delta} \right] \left[ w(n) + \sum_{j=0}^{L-1} x_j(n)z_j(n) \right]^2. \end{aligned}$$

#### 5.3.2 Assumptions

The subsequent theoretical analysis will rely on the following assumptions [13, 30, 82] that have been extensively used in the adaptive filtering literature to match reasonably well with their actual performance:

I) The step-size  $\mu$  is chosen sufficiently small such that  $z_l(n)$  changes slowly relative to

 $x_l(n)$ .

- II) The length of the adaptive filter L is equivalent to that of the unknown system.
- III) The expected value of the normalization term in (5.9) and the expected valued of its square can be assumed to be [30]

$$E\left\{\mathbf{x}^{T}(n)\mathbf{Q}(n-1)\mathbf{x}(n)+\delta\right\} = \sigma_{x}^{2} + \delta$$
(5.10)

$$E\left\{\left(\mathbf{x}^{T}(n)\mathbf{Q}(n-1)\mathbf{x}(n)+\delta\right)^{2}\right\}=\left(\sigma_{x}^{2}+\delta\right)^{2}.$$
(5.11)

IV) Using the 'separable approach' theory developed in [30], for  $a, b \in \{1, 2\}$ ,

$$E\{q_l^a(n-1)\} = E\{q_l(n-1)\}^a$$
(5.12)

$$E\left\{q_{l}^{a}(n-1)z_{l}^{b}(n)\right\} = E\left\{q_{l}(n-1)\right\}^{a} E\left\{z_{l}^{b}(n)\right\}.$$
(5.13)

V) The  $l^{th}$  component of the weight deviation at each iteration,  $z_l(n)$ , follows a normal distribution with  $\bar{z}_l(n) \triangleq E\{z_l(n)\}$  and variance  $\sigma_l^2(n)$  [30]. This implies that the each adaptive filter coefficient  $\hat{h}_l(n)$  is also distributed as

$$\widehat{h}_l(n) \sim \mathcal{N}\left(m_l(n), \sigma_l^2(n)\right), \tag{5.14}$$

with p.d.f

$$f\left(|\hat{h}_{l}(n)|\right) = \frac{1}{\sqrt{2\pi\sigma_{l}^{2}(n)}} \left[e^{-\frac{\left(|\hat{h}_{l}(n)|-m_{l}(n)\right)^{2}}{2\sigma_{l}^{2}(n)}} + e^{-\frac{\left(|\hat{h}_{l}(n)|+m_{l}(n)\right)^{2}}{2\sigma_{l}^{2}(n)}}\right] U\left(\hat{h}_{l}(n)\right),$$
(5.15)

where

$$m_l(n) = h_l(n) - \bar{z}_l(n),$$
 (5.16)

$$\sigma_l^2(n) \triangleq E\{z_l^2(n)\} - E^2\{z_l(n)\}, \qquad (5.17)$$

 $\quad \text{and} \quad$ 

$$U\left(\widehat{h}_l(n)\right) = \begin{cases} 0, & \widehat{h}_l(n) < 0\\ 1, & \widehat{h}_l(n) \ge 0; \end{cases}$$

$$(5.18)$$

It follows from (5.15) that the mean of this distribution is given by

$$E\left\{\left|\widehat{h}_{l}(n)\right|\right\}$$

$$=\int_{-\infty}^{\infty}\left|\widehat{h}_{l}(n)\right| f\left(\left|\widehat{h}_{l}(n)\right|\right) d\left|\widehat{h}_{l}(n)\right|$$

$$=m_{l}(n) \operatorname{erf}\left(\frac{m_{l}(n)}{\sqrt{2\sigma_{l}^{2}(n)}}\right) + \sqrt{\frac{2}{\pi}}\sigma_{l}(n)e^{-\frac{m_{l}^{2}(n)}{2\sigma_{l}^{2}(n)}},$$
(5.19)

with

$$\operatorname{erf}(x) \triangleq \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt.$$
 (5.20)

#### 5.3.3 Recursive mean-square error analysis for IPNLMS

By employing these assumptions, the expectations  $E\{.\}$  of the weight deviation in (5.8) and the square weight deviation in (5.9) are respectively given by the following recursive forms:

$$E\{z_{l}(n)\} = E\{z_{l}(n-1)\} - \frac{\mu\sigma_{x}^{2}}{\sigma_{x}^{2}+\delta}E\{q_{l}(n-1)\} E\{z_{l}(n)\},$$
(5.21)  

$$E\{z_{l}^{2}(n)\} = E\{z_{l}^{2}(n-1)\} + 2(1-\varepsilon)\sigma_{s}^{2} - \frac{2\mu\sigma_{x}^{2}}{\sigma_{x}^{2}+\delta}E\{q_{l}(n-1)\} E\{z_{l}^{2}(n)\} + \frac{\mu^{2}\sigma_{x}^{2}\sigma_{w}^{2}}{(\sigma_{x}^{2}+\delta)^{2}}E\{q_{l}(n-1)\}^{2} + \frac{\mu^{2}\sigma_{x}^{4}}{(\sigma_{x}^{2}+\delta)^{2}}E\{q_{l}(n-1)\}^{2}\sum_{j=0}^{L-1}E\{z_{j}^{2}(n)\},$$
(5.22)

with their initial values assigned to

$$E\{z_l(0)\} = h_l(0), (5.23)$$

$$E\{z_l^2(0)\} = h_l^2(0), (5.24)$$

and

$$E\{q_l(n-1)\} = \frac{1-\alpha}{2L} + \frac{(1+\alpha)E\{|\hat{h}_l(n)|\}}{2\sum_{j=0}^{L-1}E\{|\hat{h}_j(n)|\} + \delta_{\rm ip}}.$$
(5.25)

Given (5.19)-(5.25), the MSE in (5.7) can be now recursively computed which, as will be shown in Section 5.4, allows to investigate the sensitivity of the IPNLMS algorithm's tracking performance to the parameters such as variances of the input signal ( $\sigma_x^2$ ) and the additive noise ( $\sigma_w^2$ ), step-size ( $\mu$ ) and rate of the time-varying system ( $\varepsilon$ ). The formulation can also be used to compute the steady-state MSE value, as described in the next section.

#### 5.3.4 Steady-state analysis for IPNLMS

The steady-state MSE value,  $MSE(\infty)$ , can be estimated by evaluating (5.7) at  $n = \infty$ , as

MSE(
$$\infty$$
) =  $\sigma_w^2 + \sigma_x^2 \sum_{l=0}^{L-1} E\{z_l^2(\infty)\}$ . (5.26)

The sum of all the weight deviation components in (5.26) can be computed, by evaluating (5.22) at  $n = \infty$ , as

$$0 = \sum_{l=0}^{L-1} 2(1-\varepsilon)\sigma_s^2 - \sum_{l=0}^{L-1} \frac{2\mu\sigma_x^2}{\sigma_x^2 + \delta} E\{q_l(\infty)\} E\{z_l^2(\infty)\} + \sum_{l=0}^{L-1} \frac{\mu^2 \sigma_x^2 \sigma_w^2}{(\sigma_x^2 + \delta)^2} E\{q_l(\infty)\}^2 + \sum_{l=0}^{L-1} \frac{\mu^2 \sigma_x^4}{(\sigma_x^2 + \delta)^2} E\{q_l(\infty)\}^2 \sum_{j=0}^{L-1} E\{z_j^2(\infty)\},$$
(5.27)

which assumes convergent behavior such that when  $n = \infty$ 

$$\sum_{l=0}^{L-1} E\left\{z_l^2(n)\right\} = \sum_{l=0}^{L-1} E\left\{z_l^2(n-1)\right\}.$$
(5.28)

The  $\sum_{l=0}^{L-1} E\{q_l(\infty)\} E\{z_l^2(\infty)\}$  term in (5.27) can be computed as

$$\sum_{l=0}^{L-1} E\{q_l(\infty)\} E\{z_l^2(\infty)\} = \sum_{l=0}^{L-1} \left[ \left( \frac{1-\alpha}{2L} + \frac{(1+\alpha)|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} \right) E\{z_l^2(\infty)\} \right]$$
$$= \frac{1-\alpha}{2L} \sum_{l=0}^{L-1} E\{z_l^2(\infty)\} + (1+\alpha) \sum_{l=0}^{L-1} \left[ \frac{|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} E\{z_l^2(\infty)\} \right].$$
(5.29)

where the second term can be estimated by considering the extreme dispersive and sparse cases as follows:

• For the extreme sparse case, the estimated impulse response  $\hat{h}(\infty)$  with length L is assumed to have only one coefficient with a non-zero magnitude and rest of the coefficients are zero, so the second term in (5.29) becomes

$$\sum_{l=0}^{L-1} \left[ \frac{|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} E\left\{ z_l^2(\infty) \right\} \right] = \frac{\sum_{l=0}^{L-1} E\left\{ z_l^2(\infty) \right\}}{2 + \delta_{\rm ip}}.$$
 (5.30)

• For the extreme dispersive case, the estimated impulse response  $\hat{h}(\infty)$  with length L is assumed to have equal magnitude for all coefficients, so the second term in (5.29) becomes

$$\sum_{l=0}^{L-1} \left[ \frac{|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} E\left\{ z_l^2(\infty) \right\} \right] = \frac{\sum_{l=0}^{L-1} E\left\{ z_l^2(\infty) \right\}}{2L + \delta_{\rm ip}}.$$
 (5.31)

Therefore, by using the linear interpolation, the second term in (5.29) can be estimated in terms of  $\hat{\xi}(\infty)$  as

$$\sum_{l=0}^{L-1} \left[ \frac{|\widehat{h}_l(\infty)|}{2\|\widehat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} E\left\{z_l^2(\infty)\right\} \right] \simeq \frac{\sum_{l=0}^{L-1} E\left\{z_l^2(\infty)\right\}}{\left[2L(1-\widehat{\xi}(\infty)) + 2\widehat{\xi}(\infty)\right] + \delta_{\rm ip}}$$
(5.32)

where  $\hat{\xi}(\infty)$  is the sparseness measure of the estimated impulse response at  $n = \infty$ , defined in (3.6). Finally, the  $\sum_{l=0}^{L-1} E\{q_l(\infty)\}^2$  term in (5.27) can be expressed as

$$\sum_{l=0}^{L-1} E\{q_l(\infty)\}^2 = \sum_{l=0}^{L-1} \left[ \frac{1-\alpha}{2L} + \frac{(1+\alpha)|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} \right]^2$$
$$= \sum_{l=0}^{L-1} \left[ \frac{1-\alpha}{2L} \right]^2 + \sum_{l=0}^{L-1} \left[ \frac{(1+\alpha)|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}} \right]^2 + \sum_{l=0}^{L-1} \frac{(1-\alpha)}{L} (1+\alpha) \frac{|\hat{h}_l(\infty)|}{2\|\hat{\mathbf{h}}(\infty)\|_1 + \delta_{\rm ip}}.$$
(5.33)

By ignoring the  $\delta_{ip}$  in the denominator terms, (5.33) can be approximated as

$$\sum_{l=0}^{L-1} E\left\{q_{l}(\infty)\right\}^{2} \simeq \frac{(1-\alpha)^{2}}{4L} + \sum_{l=0}^{L-1} \left[\frac{(1+\alpha)\left|\hat{h}_{l}(\infty)\right|}{2\|\hat{\mathbf{h}}(\infty)\|_{1}}\right]^{2} + \frac{(1-\alpha^{2})}{2L\|\hat{\mathbf{h}}(\infty)\|_{1}} \sum_{l=0}^{L-1}\left|\hat{h}_{l}(\infty)\right| \\ \simeq \frac{(1-\alpha)^{2}}{4L} + \left[\frac{(1+\alpha)}{2\sqrt{L}-2\hat{\xi}(\infty)\left[\sqrt{L}-1\right]}\right]^{2} + \frac{(1-\alpha^{2})}{2L}.$$
 (5.34)

Substituting (5.32) into (5.29), (5.27) becomes

$$\sum_{l=0}^{L-1} E\left\{z_{l}^{2}(\infty)\right\} = \left\{\frac{2L(1-\varepsilon)\sigma_{s}^{2} + \frac{\mu^{2}\sigma_{x}^{2}\sigma_{w}^{2}}{(\sigma_{x}^{2}+\delta)^{2}}\sum_{l=0}^{L-1} E\left\{q_{l}(\infty)\right\}^{2}}{\frac{\mu\sigma_{x}^{2}}{\sigma_{x}^{2}+\delta}\left[\frac{1-\alpha}{L} + \frac{1+\alpha}{\left[L(1-\widehat{\xi}(\infty))+\widehat{\xi}(\infty)\right]+\delta_{\mathrm{ip}}}\right] - \frac{\mu^{2}\sigma_{x}^{4}}{(\sigma_{x}^{2}+\delta)^{2}}\sum_{l=0}^{L-1} E\left\{q_{l}(\infty)\right\}^{2}}\right\},$$
(5.35)

where  $\sum_{l=0}^{L-1} E\left\{z_l^2(\infty)\right\}$  is defined in (5.34).

Hence, the steady-state MSE value can be estimated, by substituting (5.35) into (5.26), as

$$MSE(\infty) = \sigma_{w}^{2} + \sigma_{x}^{2} \left\{ \frac{2L(1-\varepsilon)\sigma_{s}^{2} + \frac{\mu^{2}\sigma_{x}^{2}\sigma_{w}^{2}}{(\sigma_{x}^{2}+\delta)^{2}}\sum_{l=0}^{L-1}E\left\{q_{l}(\infty)\right\}^{2}}{\frac{\mu\sigma_{x}^{2}}{\sigma_{x}^{2}+\delta}\left[\frac{1-\alpha}{L} + \frac{1+\alpha}{\left[L(1-\widehat{\xi}(\infty))+\widehat{\xi}(\infty)\right]+\delta_{ip}}\right] - \frac{\mu^{2}\sigma_{x}^{4}}{(\sigma_{x}^{2}+\delta)^{2}}\sum_{l=0}^{L-1}E\left\{q_{l}(\infty)\right\}^{2}}\right\}}.$$
(5.36)

#### 5.4 Performance evaluation

In this section, the theoretical results of the IPNLMS algorithm derived for the transient performance in Section 5.3.3 and the steady-state performance in Section 5.3.4 are confirmed for different experimental setups, for sparse and dispersive time-varying echo systems, in the context of AEC.

#### 5.4.1 Experimental setup

In all simulations, the adaptive filter length was set to L = 1024, a zero mean white Gaussian noise (WGN) was used as the input signal x(n) while another WGN sequence w(n) was used for the additive noise in the receiving room. The proportionality control factor for IPNLMS was set to  $\alpha = -0.75$  [26] and  $\sigma_s^2 = 1$  was chosen to model a scenario described in Section 5.4.2. The regularization parameters were chosen to satisfy the assumptions in (5.10) and (5.11) and the estimation in (5.32). These were achieved by setting to  $\delta = \delta_{ip} = 10^{-4}$  [30]. In all simulation cases, the results were obtained by the Monte Carlo simulations with 100 independent trials.

### 5.4.2 Performances prediction under different rates of time-varying systems

The transient and steady-state performances of the IPNLMS algorithm is assessed under different first order Markov systems by changing the time-varying rate  $\varepsilon$  in (5.1). The step-size and the variances of the input signal and the additive noise were set to  $\mu = 0.7$ ,  $\sigma_x^2 = 10^{-3}$  and  $\sigma_w^2 = 10^{-6}$ , respectively, while the other parameters were fixed to those described in Section 5.4.1.

Figure 5.2 shows the MSEs for different  $\varepsilon$ , including  $\varepsilon = 1$ , which models a timeinvariant system, and  $\varepsilon = 1 - 10^{-8}$  which models an equivalent scenario of a source moving approximately at 0.35 ms<sup>-1</sup> in a room dimensions of  $8 \times 10 \times 3$  m. In all cases,  $\mathbf{h}(0)$  was initialized to the sparse impulse response shown in Fig. 1.3 (a) in order to model sparse time-varying systems as explained in Section 5.2. For these values of  $\varepsilon$  the predicted MSEs provide results close to the MSE obtained by the simulations. In addition, it can be noticed that the steady-state MSE( $\infty$ ) increases when  $\varepsilon$  decreases (i.e, the system becomes more



Figure 5.2: MSE of IPNLMS for varying  $\varepsilon$  with  $\mu = 0.7$ ,  $\alpha = -0.75$ ,  $\sigma_x^2 = 10^{-3}$ ,  $\sigma_w^2 = 10^{-6}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .

time-variant).

#### 5.4.3 Performances prediction using different step-sizes

Figure 5.3 shows MSEs of IPNLMS for different step-sizes. The sparse time-varying system was modeled by initializing  $\mathbf{h}(0)$  to the sparse impulse response shown in Fig. 1.2 (a). By setting the time-varying rate  $\varepsilon$  to  $1 - 10^{-9}$ , the system changes slowly over time such that the sparseness measure in the first 4 seconds of the sparse response ranges between 0.66 and 0.83. The variances of the WGN sequences for the input signal and the additive noises were chosen as  $\sigma_x^2 = 10^{-3}$  and  $\sigma_w^2 = 10^{-6}$ , respectively. It can be seen that the transient and steady-state MSEs correspond very well with the simulated MSE.

With the similar setup as before, Fig. 5.4 shows the theoretical and simulated MSEs of IPNLMS for a dispersive time-varying system. The dispersive time-varying system was simulated by initializing  $\mathbf{h}(0)$  to the dispersive impulse response shown in Fig. 1.2 (b). With  $\varepsilon$  empirically set to  $1 - 10^{-7}$ , the sparseness measure of the dispersive impulse response in the first 4 seconds ranged between 0.33 and 0.59. As shown in Fig. 5.4,



Figure 5.3: MSE of IPNLMS in sparse and time-variant system identification for varying step-sizes  $\mu$  with  $\varepsilon = 1 - 10^{-9}$ ,  $\alpha = -0.75$ ,  $\sigma_x^2 = 10^{-3}$ ,  $\sigma_w^2 = 10^{-6}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .

the transient and steady-state MSEs can be accurately predicted by (5.7) and (5.36), respectively.

This analysis enables the designers to choose a sensible step-size for the adaptive algorithm, depending on their application and the background noise level, as the designer adjustable parameter involves a trade-off between mis-adjustment and the convergence speed.

#### 5.4.4 Performances prediction using different input signal variances

Figure 5.5 and 5.6 show the results for predicted and simulated MSEs of IPNLMS using different input signal variances, to track the sparse and dispersive time-varying systems, respectively. The time-varying systems were imitated as described in Section 5.4.3. Zero mean WGN sequences with three different variances were used as the input signals, while step-size  $\mu = 0.7$  and the additive noise variance  $\sigma_w^2 = 10^{-6}$  were chosen in these examples.

The results in Fig. 5.5 and 5.6 indicate that the predictions are accurate to within



Figure 5.4: MSE of IPNLMS in dispersive and time-variant system identification for varying  $\mu$  with  $\varepsilon = 1 - 10^{-7}$ ,  $\alpha = -0.75$ ,  $\sigma_x^2 = 10^{-3}$ ,  $\sigma_w^2 = 10^{-6}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .

1 dB. It can also be observed that the predicted MSE slightly deviates (approximately 2-3 dB in terms of MSE) from the simulated MSE for the sparse time-varying system, during the initial stage, due to the assumption in (5.13).

#### 5.4.5 Performances prediction using different additive noise variances

Simulations to support the analysis of the transient and steady-state MSEs for timevarying system identifications are shown in Fig. 5.7 and 5.8, respectively, for different additive noise variances  $\sigma_w^2$ . In these simulations, the sparse and dispersive time-varying systems were modeled as described above and  $\mu = 0.7$  and  $\sigma_x^2 = 10^{-3}$ . Predictions accurate to within 2 dB, in terms of MSE, can be seen from the results.

It can also be noted from Fig. 5.7 that the developed model to predict the transient performance of IPNLMS can also forecast any misconvergence behavior.



Figure 5.5: MSE of IPNLMS in sparse and time-variant system identification for varying  $\sigma_x^2$  with  $\varepsilon = 1 - 10^{-8}$ ,  $\alpha = -0.75$ ,  $\mu = 0.7$ ,  $\sigma_w^2 = 10^{-6}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .



Figure 5.6: MSE of IPNLMS in dispersive and time-variant system identification for varying  $\sigma_x^2$  with  $\varepsilon = 1 - 10^{-7}$ ,  $\alpha = -0.75$ ,  $\mu = 0.7$ ,  $\sigma_w^2 = 10^{-6}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .



Figure 5.7: MSE of IPNLMS in sparse and time-variant system identification for varying  $\sigma_w^2$  with  $\varepsilon = 1 - 10^{-9}$ ,  $\alpha = -0.75$ ,  $\mu = 0.7$ ,  $\sigma_x^2 = 10^{-3}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .



Figure 5.8: MSE of IPNLMS in dispersive and time-variant system identification for varying  $\sigma_w^2$  with  $\varepsilon = 1 - 10^{-7}$ ,  $\alpha = -0.75$ ,  $\mu = 0.7$ ,  $\sigma_x^2 = 10^{-3}$ ,  $\sigma_s^2 = 1$ ,  $\delta = \delta_{ip} = 10^{-4}$ .

#### 5.5 Summary

A performance analysis has been presented for IPNLMS, one of the best known sparse adaptive filtering algorithms. The analysis considers the tracking case in which the unknown system to be identified is not only sparse or dispersive but also time-varying. The cases of slowly time-varying sparse and dispersive echo paths have been included in the study with varying levels of sparseness.

Due to the specific assumption in this prediction of a stochastic process, the theoretical and experimental results are very close but not exactly the same for the sparse time-varying case. Nevertheless, simulations presented, in the context of AEC, have been shown to verify the theoretical analysis to within 3 dB in terms of MSE which accurately describes the performances of the algorithm under different experimental setups.

## Chapter 6

# A Class of Sparseness-controlled Affine Projection Algorithms for Blind SIMO System Identification

 $I_{(APA)}^{N}$  the literature of supervised sparse system identification, affine projection algorithms  $I_{(APA)}^{N}$  with proportionate step-size allocation techniques have been proposed based on extensions of PNLMS-type algorithms, especially to achieve a superior convergence rate for correlated input signals such as speech signals. In this chapter, a generalized framework for a class of proportionate APA (PAPA) is derived for blind system identification (BSI), similar to the supervised PAPA formulated in [83], and then the sparseness-controlled techniques developed in Chapter 3 are exploited in the novel framework for the individual step-size control.

#### 6.1 Introduction

In contrast to the supervised unknown system identification studied in the previous chapters where a reference source signal is known, a priori knowledge of the source speech signal is inaccessible in applications like speech dereverberation. This demands a method for blind identification of acoustic multichannel systems. In the recent time, there has been a significant increase in the interest of applying BSI [84] techniques on applications for both civil and military purposes. For example, integration of microphone arrays in portable devices has become popular, which has made robust multichannel BSI algorithms extremely desirable for speech dereverberation and subsequent speech enhancement applications.

Since the second-order statistics (SOS) [85] of a scalar channel output do not contain information about the system phase, early methods of blind estimation were based on higher-order statistics (HOS) [86]. However, accurate estimation of the higher-order moments usually requires long segments of data, which is problematic for time-varying channels. This can be overcome by using multiple receivers, so that the lost phase information is recovered, thereby giving a SIMO channel model. A large number of SOS-based multichannel BSI algorithms have been developed [87, 88], among which celebrated work include the cross-relation (CR) method [87], the subspace method [89], the LP-based subspace algorithm [90], and the two-step maximum likelihood algorithm [91]. They identify the unknown channels from multichannel observations, providing

- 1. the transfer functions of the unknown channels do not share any common zeros
- 2. the autocorrelation matrix of the input signal is of full rank.

For a real-time implementation, an adaptive BSI algorithm is desirable due to track changes in the acoustic environment. In [92], a systematic way to design adaptive algorithms for BSI has been introduced and also proposed the multichannel LMS (MCLMS) method which is computationally efficient. Its frequency domain version, the normalized multichannel frequency domain LMS (NMCFLMS), was proposed in [93]. However such an LMS-based method converges slowly particularly when the SIMO system to be identified is sparse , which is a common drawback of the LMS-type algorithms [83]. Motivated by the sparse nature of AIRs, proportionate blind adaptive algorithms, such as improved proportionate NMCFLMS (IPNMCFLMS) and improved proportionate multichannel multi-delay filter (IPMCMDF) [64], have been proposed so that the convergence can be accelerated in identifying sparse AIRs blindly.

The rate of convergence of the aforementioned LMS-type algorithms is highly dependent on the sample correlation coefficient of successive input signal vectors. To alleviate this dependency, a generalized framework is developed in this chapter for blind SIMO system identification, by exploiting the affine projection technique [16] into the PNLMS-type



Figure 6.1: Illustration of an acoustic FIR SIMO system.

sparseness-controlled algorithms described in Chapter 3. The sparseness measure is exploited to deal efficiently with systems having different degrees of sparsity. In addition, the new class of algorithms takes into account the proportionate history [83], yielding further improvement in performance and makes efficient use of computations due to the recursive implementation of the proportionate history.

This chapter is organized as follows: Section 6.2 and 6.3 define the multichannel system model and the cross relation method, respectively. A generalized framework for SIMO system identification using the CR error is derived in Section 6.4. A class of multichannel sparseness-controlled PAPA is then described in Section 6.5. Finally, experimental results are presented in Section 6.6 before summarize the chapter.

#### 6.2 Signal model

For an *M*-channel SIMO system as shown in Fig. 6.1, the  $m^{\text{th}}$  impulse response with *L* coefficients can be denoted as

$$\mathbf{h}_m = [h_{m,0} \ h_{m,1} \ \dots \ h_{m,L-1}]^T, \tag{6.1}$$

for m = 1, 2, ..., M, which corresponds to the  $m^{\text{th}}$  unique acoustic channel resulted from multipath reflections of the transmitted source signal s(n). The  $m^{\text{th}}$  sensor signal can be expressed as

$$x_m(n) = \sum_{l=0}^{L-1} h_{m,l} \, s(n-l) + b_m(n), \tag{6.2}$$

where  $b_m(n)$  is the additive noise. The additive noise is assumed to be zero-mean and uncorrelated with the source signal. In vector form, (6.2) can be written as

$$\mathbf{x}_m(n) = \mathbf{H}_m \,\mathbf{s}(n) + \mathbf{b}_m(n),\tag{6.3}$$

where  $\mathbf{s}(n) = [s(n) \ s(n-1) \ \dots \ s(n-2L+1)]^T$ ,  $\mathbf{x}_m(n) = [x_m(n) \ x_m(n-1) \ \dots \ x_m(n-L+1)]^T$ ,  $\mathbf{b}_m(n) = [b_m(n) \ b_m(n-1) \ \dots \ b_m(n-L+1)]^T$ , and  $\mathbf{H}_m$  is the  $L \times (2L-1)$  matrix for the  $m^{\text{th}}$  channel such that

$$\mathbf{H}_{m} = \begin{bmatrix} h_{m,0} & h_{m,1} & \cdots & h_{m,L-1} & \cdots & \cdots & 0\\ 0 & h_{m,0} & h_{m,1} & \cdots & h_{m,L-1} & \cdots & 0\\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots\\ 0 & \cdots & \cdots & h_{m,0} & h_{m,1} & \cdots & h_{m,L-1} \end{bmatrix}.$$
 (6.4)

Since the impulse responses are assumed to be slowly time-varying,  $\mathbf{H}_m$  is independent of *n*. By concatenating all *M* outputs of (6.3), a system of equations

$$\mathbf{x}(n) = \mathbf{H}\,\mathbf{s}(n) + \mathbf{b}(n) \tag{6.5}$$

can be obtained using the following quantities

$$\mathbf{x}(n) = [\mathbf{x}_1^T(n) \ \mathbf{x}_2^T(n) \ \dots \ \mathbf{x}_M^T(n)]^T,$$
(6.6)

$$\mathbf{H} = [\mathbf{H}_1^T \ \mathbf{H}_2^T \ \dots \ \mathbf{H}_M^T]^T, \tag{6.7}$$

$$\mathbf{b}(n) = [\mathbf{b}_1^T(n) \ \mathbf{b}_2^T(n) \ \dots \ \mathbf{b}_M^T(n)]^T.$$
(6.8)

The problem of blind SIMO system identification is to find  $\mathbf{h} = [\mathbf{h}_1^T \ \mathbf{h}_2^T \ \dots \ \mathbf{h}_M^T]^T$  using only the received signals  $x_m(n)$ . Hence, given the received signals, a unique solution to  $\mathbf{h}$ should be obtained up to a non-zero scale factor across all channels. This scale factor is irrelevant in most of acoustic signal processing applications.

#### 6.3 Cross relation method

The CR method [87] is based on the fact that

$$x_m(n) * h_{k,l} = s(n) * h_{m,l} * h_{k,l} = x_k(n) * h_{m,l},$$
(6.9)

a cross-relation between the  $m^{\text{th}}$  and  $k^{\text{th}}$  channel outputs, in the absence of noise, can be formulated as

$$\mathbf{x}_m^T(n)\mathbf{h}_k = \mathbf{x}_k^T(n)\mathbf{h}_m, \ m, k = 1, 2, \dots, M, \ m \neq k,$$
(6.10)

where  $\mathbf{h}_k$  and  $\mathbf{h}_m$  denote the  $k^{\text{th}}$  and  $m^{\text{th}}$  channel impulse responses, respectively. In the presences of noise the *a priori* CR error is defined as the difference between the left- and right-hand sides of (6.10)

$$e_{mk}(n) = \mathbf{x}_m^T(n)\mathbf{h}_k - \mathbf{x}_k^T(n)\mathbf{h}_m.$$
(6.11)

Channel estimation can be performed by minimizing the CR error between different sensor pairs subject to the identifiability conditions, mentioned in Section 6.1, being satisfied.

#### 6.4 General cost function for SIMO BSI using the CR error

In Section 6.4.1 the novel multichannel proportionate APA (MC-PAPA) algorithm is derived specifically for M = 2, while the framework is then generalized for  $M \ge 2$  in Section 6.4.2.

#### 6.4.1 Dual channel (M = 2)

By denoting the estimator of  $\mathbf{h}(n)$  at time n as  $\hat{\mathbf{h}}(n)$ , the *a posteriori* CR error can be defined as

$$\epsilon(n) = \mathbf{R}(n)\mathbf{\hat{h}}(n), \tag{6.12}$$

where

$$\mathbf{R}(n) = \begin{bmatrix} \mathbf{x}_{2}^{T}(n) & -\mathbf{x}_{1}^{T}(n) \\ \mathbf{x}_{2}^{T}(n-1) & -\mathbf{x}_{1}^{T}(n-1) \\ \vdots & \vdots \\ \mathbf{x}_{2}^{T}(n-P+1) & -\mathbf{x}_{1}^{T}(n-P+1) \end{bmatrix}_{P \times 2L}, \qquad (6.13)$$

with P ( $P \leq L$ ) denoting the projection order that controls the tradeoff between the computational complexity and rate of convergence. It is well-known that when the projection order increases, the convergence rate of the filter coefficient vector also increases. However, this also leads to an increased computational complexity. The normalized *a posteriori* CR error can be similarly defined as

$$\tilde{\epsilon}(n) = \left[\mathbf{R}(n)\mathbf{G}(n-1)\mathbf{R}^{T}(n)\right]^{-\frac{1}{2}}\epsilon(n), \qquad (6.14)$$

where

$$\mathbf{G}(n) = \begin{bmatrix} \mathbf{G}_1(n) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2(n) \end{bmatrix}_{2L \times 2L},$$
(6.15)

with  $\mathbf{G}_m(n) = \text{diag}\{\mathbf{g}_m(n)\} = \text{diag}\{g_{m,0}(n), g_{m,1}(n), \dots, g_{m,L-1}(n)\}\$  for  $1 \le m \le M$  is an  $L \times L$  diagonal matrix that usually depends on  $\widehat{\mathbf{h}}_m(n-1)$ .

Similar to the supervised case in [95], the criterion of proportionate adaptive algorithms for blind SIMO system identification can be formulated as

$$J(n) = d\left[\widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1)\right] + \tilde{\epsilon}^T(n)\tilde{\epsilon}(n), \qquad (6.16)$$

where d  $\left[\widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1)\right]$  denotes a distance measure between the estimated channels at time n and n-1. Differentiating J(n) with respect to  $\widehat{\mathbf{h}}(n)$  produces

$$\frac{\partial J(n)}{\partial \widehat{\mathbf{h}}(n)} = \frac{\partial d \left[\widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1)\right]}{\partial \widehat{\mathbf{h}}(n)} + 2 \mathbf{R}^{T}(n) \left[\mathbf{R}(n)\mathbf{G}(n-1)\mathbf{R}^{T}(n)\right]^{-1} \mathbf{R}(n)\widehat{\mathbf{h}}(n).$$
(6.17)

Equating the derivative in (6.17) to zero, it can be seen that any adaptive algorithms need

to satisfy

$$\mathbf{P}_{\mathbf{x}}(n) \left[ \widehat{\mathbf{h}}(n) - \widehat{\mathbf{h}}(n-1) \right] + \frac{1}{2} \frac{\partial d \left[ \widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1) \right]}{\partial \widehat{\mathbf{h}}(n)} = -\mathbf{R}^{T}(n) \left[ \mathbf{R}(n) \mathbf{G}(n-1) \mathbf{R}^{T}(n) \right]^{-1} \mathbf{e}(n),$$
(6.18)

where

$$\mathbf{P}_{x}(n) = \mathbf{R}^{T}(n) \left[ \mathbf{R}(n)\mathbf{G}(n-1)\mathbf{R}^{T}(n) \right]^{-1} \mathbf{R}(n)$$
(6.19)

and

$$\mathbf{e}(n) = \mathbf{R}(n)\widehat{\mathbf{h}}(n-1) \tag{6.20}$$

is the *a priori* CR error.

In order to increase robustness of the system performance for example, after a large noise sample perturbs it, d  $[\hat{\mathbf{h}}(n), \hat{\mathbf{h}}(n-1)]$  should be constrained efficiently at each iteration. Depending on the choice for the distance function, different algorithms can be derived.

### 6.4.2 Multichannel $(M \ge 2)$

In the case of M number of channels, there exist M(M-1)/2 number of independent CR errors defined in (6.11). To include all the CR possibilities, the full rank matrix  $\mathbf{R}(n)$  can be generalized as follows for M channels [66]:

$$\mathbf{R}(n) = \begin{bmatrix} \mathbf{C}_{1}(n) - \mathbf{D}_{1}(n) \\ \mathbf{C}_{2}(n) - \mathbf{D}_{2}(n) \\ \vdots \\ \mathbf{C}_{M-1}(n) - \mathbf{D}_{M-1}(n) \end{bmatrix}_{\frac{PM(M-1)}{2} \times ML}$$
(6.21)

where

$$\mathbf{C}_{m}(n) = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \mathbf{X}_{m+1}(n) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{X}_{m+2}(n) & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{X}_{M}(n) & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}_{P(M-m) \times ML} , \qquad (6.22)$$

г

$$\mathbf{D}_{m}(n) = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} & \mathbf{X}_{m}(n) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{X}_{m}(n) & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{X}_{m}(n) \end{bmatrix}_{P(M-m) \times ML}$$
(6.23)

with the  $P \times L$  matrix  $\mathbf{X}_m(n) = [\mathbf{x}_m(n) \mathbf{x}_m(n-1) \cdots \mathbf{x}_m(n-P+1)]^T$ . Finally, the generalized step-size control matrix is defined as

$$\mathbf{G}(n) = \begin{bmatrix} \mathbf{G}_{1}(n) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{2}(n) & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{G}_{M}(n) \end{bmatrix}_{ML \times ML}, \qquad (6.24)$$

which allows different adaptation techniques for the coefficients according to their influence in the adaptive filter. As for the sparse supervised system identification case, this could increase the convergence speed of the BSI adaptive algorithms when the system is sparse.

#### 6.5Multichannel sparseness-controlled PAPA

In this Section, a generalized framework is developed for blind SIMO system identification, based on the CR error, by exploiting the following 3 existing techniques:

- 1. affine projection [16], to deal with coloured input signal
- 2. history of proportionate step-size control [83], to improve the convergence performance in sparse system identification
- 3. sparseness-controlled [70], which makes it robust to the sparseness variation of AIRs.

#### 6.5.1General formulation

From all the possible choice of functions for d  $[\widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1)]$ , Riemannian manifolds<sup>1</sup> [96] are used as the function facilitates for proportionate coefficient update techniques. Hence,

and

<sup>&</sup>lt;sup>1</sup>a curved differentiable manifold where the distance properties are not uniform along the space
the distance measure is defined as

$$d\left[\widehat{\mathbf{h}}(n), \widehat{\mathbf{h}}(n-1)\right] = \left[\widehat{\mathbf{h}}(n) - \widehat{\mathbf{h}}(n-1)\right]^T \mathbf{Q}_x \left[\mathbf{G}(n-1)\right] \left[\widehat{\mathbf{h}}(n) - \widehat{\mathbf{h}}(n-1)\right], \quad (6.25)$$

where  $\mathbf{Q}_x [\mathbf{G}(n-1)]$  is a symmetric positive-definite matrix that depends on  $\mathbf{x}(n)$  and  $\mathbf{G}(n-1)$  and thus indirectly on  $\hat{\mathbf{h}}(n-1)$ . Let's define the  $ML \times ML$  matrix  $\mathbf{Q}_x [\mathbf{G}(n-1)]$  as

$$\mathbf{Q}_x \left[ \mathbf{G}(n-1) \right] = \mu^{-1} \mathbf{G}^{-1}(n-1) - \mathbf{P}_x(n), \tag{6.26}$$

which is know as the *Riemannian metric tensor* [97]. Using (6.18), (6.19) and (6.26), it follows for P > 1 that

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \mu \,\mathbf{G}(n-1)\mathbf{R}^{T}(n) \Big[\mathbf{R}(n)\mathbf{G}(n-1)\mathbf{R}^{T}(n) + \delta \mathbf{I}\Big]^{-1}\mathbf{e}(n), \qquad (6.27)$$

where  $\delta$  is a positive regularization parameter to improve the numerical stability. The diagonal step-size control matrix  $\mathbf{Q}_x [\mathbf{G}(n-1)]$  determines the step-size of each filter coefficient and is dependent on the specific algorithm.

#### 6.5.2 The MC-APA algorithm

For the MC-APA algorithm [94], since the step-size is the same for all filter coefficients,

$$\mathbf{Q}_x \left[ \mathbf{G}(n-1) \right] = \mathbf{I}_{ML \times ML},\tag{6.28}$$

with  $\mathbf{I}_{ML \times ML}$  being an  $ML \times ML$  identity matrix, (6.27) becomes

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \mu \mathbf{R}^T(n) \Big[ \mathbf{R}(n) \mathbf{R}^T(n) \Big]^{-1} \mathbf{e}(n).$$
(6.29)

The main advantage of MC-APA over the MCLMS algorithm is that it has a faster convergence rate for correlated inputs. But, its convergence performance degrades when the system is sparse, which is a common drawback of the LMS-type algorithm as found from Chapter 3.

### 6.5.3 The MC-PAPA formulation

Since PNLMS-type algorithms usually outperform the NLMS algorithm for sparse impulse responses, the poor performance of MC-APA in BSI has been addressed by combining the proportionate idea with the MC-APA, thus resulting MC-PAPA. To incorporate the history of the proportionate step-size control matrix ( $\mathbf{G}$ ), let us consider rewriting (6.27) as [83]

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \mu \mathbf{Y}(n) \left[ \mathbf{R}(n) \mathbf{Y}(n) + \delta \mathbf{I} \right]^{-1} \mathbf{e}(n),$$
(6.30)

where

$$\mathbf{Y}(n) = \mathbf{G}(n-1)\mathbf{R}^{T}(n).$$
(6.31)

Denoting the operator  $\odot$  as the Hadamard product, i.e.,  $\mathbf{a} \odot \mathbf{b} = [a(1)b(1) \ a(2)b(2) \ \dots \ a(L)b(L)]^T$ , where  $\mathbf{a}$  and  $\mathbf{b}$  are two vectors of length L, the components of  $\mathbf{Y}(n)$  for an example case of M = 2 is given by (6.32):

$\begin{array}{cccc} 0 & \ddots & 0 \\ 0 & 0 & g_{1,L-1}(n-1) \end{array} & \\ x_2(n-L+1) \\ \end{array}$
$\left(\begin{array}{cccc} g_{2,0}(n-1) & 0 & 0 \end{array}\right) \left(\begin{array}{cccc} -x_1(n-1) & 0 & 0 \end{array}\right)$

(6.32)

As it can be seen from (6.32), the classical PAPA algorithms take into account the history of the last P samples of the sensor signal  $\mathbf{x}_m(n)$ , but do not use the history of the step-size control matrix  $\mathbf{G}_m(n-1)$ . The modified MC-PAPA algorithm exploits the histories of both the sensor signal and the proportionate step-size control. Thus,for M = 2, (6.32) becomes

$$\tilde{\mathbf{Y}}(n) = \begin{bmatrix} \mathbf{g}_1(n-1) \odot \mathbf{x}_2(n) & \mathbf{g}_1(n-2) \odot \mathbf{x}_2(n-1) & \cdots & \mathbf{g}_1(n-P) \odot \mathbf{x}_2(n-P+1) \\ \mathbf{g}_2(n-1) \odot -\mathbf{x}_1(n) & \mathbf{g}_2(n-2) \odot -\mathbf{x}_1(n-1) & \cdots & \mathbf{g}_2(n-P) \odot -\mathbf{x}_1(n-P+1) \end{bmatrix},$$
(6.33)

and it can be generalized for M-channels as

$$\tilde{\mathbf{Y}}(n) = \tilde{\mathbf{G}}^T(n-1) \odot \mathbf{R}^T(n), \tag{6.34}$$

where

$$\tilde{\mathbf{G}}(n-1) = \begin{bmatrix} \begin{pmatrix} \mathbf{g}_{1}(n-1) & \mathbf{g}_{2}(n-1) & \cdots & \mathbf{g}_{M}(n-1) \\ \vdots & \vdots & & \vdots \\ \mathbf{g}_{1}(n-1) & \mathbf{g}_{2}(n-1) & \cdots & \mathbf{g}_{M}(n-1) \end{pmatrix}_{\frac{M(M-1)}{2} \times ML} \\ \begin{pmatrix} \mathbf{g}_{1}(n-2) & \mathbf{g}_{2}(n-2) & \cdots & \mathbf{g}_{M}(n-2) \\ \vdots & \vdots & & \vdots \\ \mathbf{g}_{1}(n-2) & \mathbf{g}_{2}(n-2) & \cdots & \mathbf{g}_{M}(n-2) \end{pmatrix}_{\frac{M(M-1)}{2} \times ML} \\ \vdots \\ \begin{pmatrix} \mathbf{g}_{1}(n-P) & \mathbf{g}_{2}(n-P) & \cdots & \mathbf{g}_{M}(n-P) \\ \vdots & \vdots & & \vdots \\ \mathbf{g}_{1}(n-P) & \mathbf{g}_{2}(n-P) & \cdots & \mathbf{g}_{M}(n-P) \end{pmatrix}_{\frac{M(M-1)}{2} \times ML} \end{bmatrix}_{\frac{PM(M-1)}{2} \times ML}$$
(6.35)

The modification entails not only an improved convergence performance, but also a reduction in the computational complexity. Since only the latest reverberant signals are multiplied by the new proportionate gains at each time instant,  $\tilde{\mathbf{Y}}(n)$  can be computed recursively. Hence, the computational complexity, in terms of number of multiplications, is decreased by a factor of P((6.32) needs  $PLM^2(M-1)/2$  multiplications while the recursive relation on (6.34) requires only  $LM^2(M-1)/2$  multiplications) at the cost of increased memory requirement, compared to that in implementing (6.32). Thus, with a proper choice of  $\mathbf{g}_m(n-1)$ , the generalized MC-PAPA algorithm, for M-channels, is described by (6.20)-(6.23), (6.34), (6.35) and

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) - \mu \, \widetilde{\mathbf{Y}}(n) \Big[ \mathbf{R}(n) \, \widetilde{\mathbf{Y}}(n) + \delta \mathbf{I} \Big]^{-1} \mathbf{e}(n).$$
(6.36)

In the next section, two particular choices of  $\mathbf{g}_m(n)$  are employed into this framework for robust performance in identifying AIRs blindly.

### 6.5.4 The MC-SCP-APA and MC-SCMP-APA algorithms

Drawing on techniques originally developed for sparseness-controlled PNLMS-type algorithms in Chapter 3, a class of time domain algorithms are proposed for blind SIMO system identification that can not only work well in both sparse and dispersive circumstances, but also adapt dynamically to the level of sparseness. The elements of  $\mathbf{g}_m(n)$  can be formulated as

$$g_{m,l}(n) = \frac{\kappa_{m,l}(n)}{\frac{1}{L} \sum_{l=0}^{L-1} \kappa_{m,l}(n)} \quad 0 \le l \le L-1,$$
(6.37)

$$\kappa_{m,l}(n) = \max \Big\{ \rho_m(n) \times \max\{\gamma, \\ F(|\hat{h}_{m,0}(n)|), F(|\hat{h}_{m,1}(n)|), \dots, F(|\hat{h}_{m,L-1}(n)|) \}, F(|\hat{h}_{m,l}(n)|) \Big\},$$
(6.38)

where  $F(|\hat{h}_{m,l}(n)|)$  is specific to the algorithm,  $\gamma$  is a small positive small which prevents the filter coefficients from stalling and

$$\rho_m(n) = e^{-\lambda \hat{\xi}_m(n)}, \qquad \lambda \in \mathbb{R}^+ \tag{6.39}$$

$$\widehat{\xi}_{m}(n) = \frac{L}{L - \sqrt{L}} \left\{ 1 - \frac{\|\widehat{\mathbf{h}}_{m}(n)\|_{1}}{\sqrt{L} \|\widehat{\mathbf{h}}_{m}(n)\|_{2}} \right\}.$$
(6.40)

Similar to the SC-PNLMS algorithm, the multichannel sparseness-controlled proportionate APA (MC-SCP-APA) algorithm achieves a high rate of convergence by employing step-sizes that are proportional to the magnitude of the estimated impulse response coefficients where elements  $F(|\hat{h}_{m,l}(n)|)$  are given by

$$F(|\widehat{h}_{m,l}(n)|) = |\widehat{h}_{m,l}(n)|, \qquad (6.41)$$



Figure 6.2: Sparse (left) and Dispersive (right) acoustic impulse responses of a singleinput three-output system used in the simulation for blind identification.

whereas the multichannel sparseness-controlled  $\mu$ -law proportionate APA (MC-SCMP-APA) algorithm employs

$$F(|\hat{h}_{m,l}(n)|) = \ln(1 + \nu|\hat{h}_{m,l}(n)|).$$
(6.42)

The MC-SCP-APA algorithm is thus described by (6.20)-(6.23), and (6.34)-(6.41), whilst MC-SCMP-APA is described by (6.20)-(6.23), (6.34)-(6.40) and (6.42).

### 6.6 Performance evaluation

In this section, simulation results are presented to investigate the effectiveness of MC-APA and the proposed MC-SCP-APA and MC-SCMP-APA algorithms in a blind SIMO system identification problem. The randomly generated sparse and dispersive AIRs for



Figure 6.3: Relative convergence of MC-APA, MC-SCP-APA and MC-SCMP-APA using WGN input signal with an echo path change at 4 s. Impulse response is changed from that shown from Fig. 6.2 (left) to (right) and  $\mu_{\text{MC-APA}} = \mu_{\text{MC-SCP-APA}} = \mu_{\text{MC-SCP-APA}} = 0.2$ , P = 2, SNR = 50 dB.

M = 3, with length L = 15, are shown in Fig. 6.2. The AIRs were obtained such that their transfer functions do not share any common zeros. The sparseness measure of these AIRs are computed using (6.40) giving  $\xi(n) = 0.36, 0.55$  and 0.46 for the sparse AIRs and  $\xi(n) = 0.18, 0.16$  and 0.23 for the dispersive AIRs. Employing the modified normalized projection misalignment, given by [98]

$$NPM(n) = 20 \log_{10} \left[ \frac{1}{\|\mathbf{h}(n)\|} \left( \mathbf{h}(n) - \frac{\mathbf{h}^{T}(n)\widehat{\mathbf{h}}(n)}{\widehat{\mathbf{h}}^{T}(n)\widehat{\mathbf{h}}(n)} \widehat{\mathbf{h}}(n) \right) \right],$$
(6.43)

as the performance measure, the multichannel algorithms were then tested using a zero mean WGN as the source signal s(n) (from the 5<sup>th</sup> random 'state' in Matlab) while another three WGN sequences  $b_m(n)$  were added for the three sensor signals to give SNR of 50 dB in all channels. It was also assumed that the length of  $\hat{\mathbf{h}}(n)$  is equivalent to that of the unknown  $\mathbf{h}(n)$ .

Figure 6.3 shows the relative convergence of MC-APA, MC-SCP-APA and MC-SCMP-APA with an echo path change at 4 s. Impulse responses were changed from that

shown in Fig. 6.2 sparse to Fig. 6.2 dispersive. The step-size parameter for each algorithm was chosen such that all algorithms achieve the same steady-state. This was achieved by setting  $\mu_{\text{MC}-\text{APA}} = \mu_{\text{MC}-\text{SCP}-\text{APA}} = \mu_{\text{MC}-\text{SCMP}-\text{APA}} = 0.2$ . The projection order was set to P = 2, while the parameters  $\gamma = 0.01$ ,  $\rho(n) = 0.1$  for n < L,  $\lambda = 1$  and  $\nu = 1000$  were set experimentally for the MC-SCP-APA and MC-SCMP-APA algorithms. As can be seen, the proposed MC-SCP-APA algorithm performs better than MC-APA, giving approximately 3 dB improvement during the initial convergence for the sparse AIRs. At the same time, the MC-SCMP-APA algorithm achieves the highest rate of convergence performance giving approximately 12 dB improvement compared to MC-APA. On the other hand, they perform almost similar for the dispersive AIRs. The ability of the class of sparseness-controlled multichannel algorithms to achieve good convergence performance for both sparse and dispersive AIRs is achieved by incorporating the beneficial properties of both proportionate and non-proportionate step-size control techniques. However, their computational complexities are increased, as they estimate the sparseness measure of the AIRs at each iteration for each channel.

### 6.7 Summary

In this chapter, a class of novel adaptive schemes based on SOS for blind estimation of a SIMO channel model has been derived. The proposed framework exploits the affine projection technique with the history of proportionate step-size control mechanism, in order to enhance the convergence performance in sparse BSI by providing a control over the tradeoff between the computational complexity and rate of convergence. Moreover, the particular choices for the step-size control by employing the sparseness-controlled technique, gives overall better performance compared to the conventional MC-APA. This is because the proposed algorithms take into account the sparseness measure of the estimated impulse response of each channel at each iteration.

# Chapter 7

# **Conclusions and Future Work**

In this final chapter, the work presented in this thesis is summarized and concluded, where major challenges addressed in this thesis are reviewed and the important achievements are highlighted. Finally, some feasible future work is proposed according to the current demands in such fields.

### 7.1 Summary

In this thesis, new time and frequency domain adaptive algorithms were developed, by employing sparseness measure exploitation and partitioned block techniques, and the IPNLMS algorithm was theoretically analyzed for single channel time-varying AEC application.

In Chapter 2, the existing time- and frequency domain adaptive algorithms in the context of AEC were studied. The AEC setup with a FIR adaptive filter in a LRMS was first introduced, along with the notations and the standard assumptions. Then, the derivation of LMS and NLMS algorithms were looked at in detail, to solve the Wiener-Hopf equation recursively using the method of steepest descent. As their convergence performances degrade when the impulse response is spare, many different intuitions were introduced in the literature to overcome this problem. In this work, the PNLMS, MPNLMS and IPNLMS sparse adaptive filtering algorithms were briefly reviewed, with their relative computational complexities for filter coefficient adaptation. Mainly to reduce the computational complexity of large adaptive FIR filters, frequency domain algorithms, including

FLMS, MDF and IPMDF, were presented next with their computational costs.

In Chapter 3, it was shown how to use a priori information on sparseness of the impulse response, in the design of adaptive algorithms, in order to make them perform better (in terms of initial convergence and tracking) than the aforementioned classical adaptive algorithms. By considering a particular example case, it was first shown that the variation in the sparsity of the impulse response exists when the loudspeaker to microphone distance is varied within an enclosed room. Exploiting the sparseness measure into the PNLMS adaptation equation, it was revealed through simulations that there exists a deterministic relationship between the sparseness measure of the impulse response and the optimal step gain matrix that can guarantee fastest convergence speed. Then, a class of new sparseness-controlled algorithms were proposed to incorporate the sparseness measure of the impulse response into PNLMS, MPNLMS and IPNLMS, via adaptively modified coefficient update functions. Simulation results were shown to evaluate the convergence performances of the proposed algorithms, using WGN and speech input signals, with instantaneously and slowly time-varying echo paths in the context of the AEC. With only a modest increase in computational complexity, it has been shown that the proposed algorithms (SC-PNLMS, SC-MPNLMS and SC-IPNLMS) perform well, compared to their classical ones, by utilizing the beneficial properties of the non-proportionate algorithm when identifying a dispersive AIR and the proportionate algorithm when identifying a sparse algorithm. Hence, they are especially suitable to a time-varying AIR. The idea of exploiting the sparseness measure was also exhibited in frequency domain, by incorporating it into the proportionate/nonproportionate control parameter of IPMDF. The simulation results with WGN and speech input signals were shown, which show improved performance for SC-IPMDF.

In Chapter 4, a novel partitioned block technique was developed in time domain to improve the performance of IPNLMS for time-varying echo path. The AIR was partitioned into two blocks, by including the direct path and a few early reflections in the first block and the later reverberant parts for the second block. An adaptive filter structure that consists of two time domain partition blocks was used with different proportionate/nonproportionate control factor for each partitioned block, such that a more proportionate step-sizes were allocated to the first block compared to the second block. With fixed block sizes, two different strategies were presented to impose the step-size control matrices. The resultant non-proportionate and proportionate weighting techniques have shown to achieve good convergence performance over the IPNLMS algorithm with single proportionate/nonproportionate control factor, where the former technique works well only if the norms of the two blocks' filter coefficients are almost equal. When the nature of the AIR is not known or the AIR is highly time-varying, a partitioned block algorithm was developed with an efficient control mechanism for the dynamic adjustment of the block size. Utilizing the self-configuration method, it was further shown through simulations that the proposed VLPB-IPNLMS improves the robustness of the PB-IPNLMS to the variable bulk delay.

In Chapter 5, both the transient and steady-state misalignment analysis for the IPNLMS algorithm was presented, under both sparse and dispersive time-varying unknown system conditions modelled using the modified Markov model. Simulation results were presented and have shown to verify that the analysis accurately describes the performances of the algorithms. It was noted that, for a time-varying system, the performance of IPNLMS in terms of steady-state misalignment degrades with increasing time-variation. It was also noted that, due to the specific assumption in this prediction of a stochastic process, the theoretical and experimental results are very close but not exactly the same for the sparse time-varying cases.

In Chapter 6, the research then moved to multichannel unsupervised scenario for the development of time domain blind SIMO system identification algorithms based on SOS and CR method. The chapter began by introducing the signal model for an acoustic FIR SIMO system. Extending the conventional definition of PAPA for the multichannel BSI, a generalized framework was developed together with the history of the proportionate step-size control. The sparseness-controlled technique proposed in Chapter 3 was then employed into this framework as the particular choice for the step-size control matrix. Results demonstrate the advantage of the proposed MC-SCP-APA and MC-SCMP-APA in that they provide control on the convergence performance in terms of NPM for both sparse and dispersive BSI problems. In addition, it was noted that the computational complexity of such algorithms is lower, compare to the similar technique without incorporating the proportionate history, using a recursive implementation

### 7.2 Conclusion

This research has addressed the significant problem caused by undesirable echoes that result from coupling between the loudspeakers and microphones in the near end room and also for the blind identification of SIMO acoustic systems. This thesis has mainly focused on the development of the adaptive filtering algorithms for sparse and dispersive systems in time- and frequency domains, emphasizing on the achievement of fast convergence rate with a modest increase in the computational cost.

A class of sparseness-controlled algorithms were developed which achieves improved convergence compared to classical NLMS and typical sparse adaptive filtering algorithms. The sparseness measure was incorporated into PNLMS, MPNLMS, IPNLMS and IPMDF for AEC to achieve fast convergence that is robust to the level of sparseness encountered in the impulse response of the echo path. The resulting SC-PNLMS, SC-MPNLMS, SC-IPNLMS and SC-IPMDF algorithms take into account the sparseness measure via a modified coefficient update function. It has been shown that the proposed sparsenesscontrolled algorithms are robust to variations in the level of sparseness in AIR with only a modest increase in computational complexity.

Due to the properties of an acoustic enclosure, it was shown that the early part (i.e., direct path and early reflections) of the acoustic echo path is sparse while the late reverberant part of the acoustic path is dispersive. Hence, it was validated through simulation results that, allocating larger individual step-sizes for the filter coefficients in the first block with the early part of the AIR gives faster initial convergence performance and, distributing almost equal step-sizes for the second block with the later part of the AIR gives better steady-state performance.

The transient and steady-state performances of IPNLMS was analyzed. Analytical results were shown to be accurate to within a very small estimation error compared with simulated results hence validating the analysis presented. Specifically, the error in prediction was noted only for the sparse time-varying system identification.

A class of novel proportionate adaptive algorithms for blind SIMO system identification was derived, based on the affine projection principle and the sparseness-controlled method. These algorithms take into account the history of the proportionate step-size



Figure 7.1: Acoustic impulse responses obtained using the method of images [1].

control for each filter coefficient, with lower computational complexity (compared to the classical technique without the proportionate history) using a recursive implementation in excess of additional memory requirement. Simulation results indicate that the proposed MC-SCP-APA and MC-SCMP-APA perform well for both sparse and dispersive BSI problems, providing robustness to the sparseness variation in AIR.

## 7.3 Future Work

In this very last section, the following prospective ideas of future work are presented:

• The technique developed in Chapter 4 partitioned the AIR into two blocks such that first block contained the early dominant peaks, whereas the second block contained the later parts. However, partitioning into two blocks may not be an optimum way. For example, consider the AIR shown in Fig. 7.1, which was generated employing the method of images [1] using room dimensions of  $8 \times 10 \times 3$  m and 0.3 as the reflection coefficient. The loudspeaker was fixed at  $7.5 \times 0.5 \times 1.6$  m in the LRMS while the microphone was positioned at  $0.5 \times 9.5 \times 1.6$  m. As it can be seen that partitioned into more blocks would give better convergence performance. It would be an interesting extension to the work in this thesis. • The development of the sparseness-controlled concept in this thesis, and in the recent works published in [99, 100, 101], has also opened several directions of future research. Since it is well-known that a sparse echo path can be identified faster than a dispersive echo path with same length, when the filter coefficients are initialized to zeros. Therefore, transforming a dispersive echo path into a sparse path, using existing or possibly a new transformation, will improve the convergence speed of a dispersive system identification. Hence, it is interesting to examine the possibility of exploiting the sparseness measure of the estimated impulse response at that time instance into an automated mechanism to efficiently switch between the different domains.

# Bibliography

- J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," J. Acoust. Soc. Am., vol. 65, no. 4, pp. 943–950, Apr. 1979.
- [2] A. W. H. Khong and P. A. Naylor, "Efficient use of sparse adaptive filters," in Proc. Asilomar Conf. on Signals, Systems and Computers, Oct. 2006.
- [3] J. Benesty, Y. Huang, J. Chen, and P. A. Naylor, "Adaptive algorithms for the identification of sparse impulse responses," in *Selected Methods for Acoustic Echo* and Noise Control, E. Hansler and G. Schmidt, Eds. Elsevier, 2006, ch. 5, pp. 125–153.
- [4] K. Ho, "A study of two adaptive filters in tandem," *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1626–1636, Jun. 2000.
- [5] M. M. Sondhi, "The history of echo cancellation," *IEEE Signal Process. Mag.*, vol. 23, pp. 95 102, 2006.
- [6] J. Radecki, Z. Zilic, and K. Radecka, "Echo cancellation in IP networks," in Proc. Fourty-Fifth Midwest Symposium on Circuits and Systems, vol. 2, 2002, pp. 219–222.
- M. Kahrs, G. W. Elko, S. J. Elliot, S. Makino, J. M. Kates, M. Bosi, and J. O. Smith, "The past, present and future of audio signal processing," *IEEE Signal Process. Mag.*, vol. 14, no. 5, pp. 30–57, Sep. 1997.
- [8] C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler,
   G. Schmidt, and J. Tilp, "Acoustic echo control: An application of very-high-order adaptive filter," *IEEE Signal Process. Mag.*, vol. 16, no. 4, pp. 42–69, Jul. 1999.

- [9] A. Gilloire, "Experiments with sub-band acoustic echo cancellers for teleconferencing," in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Apr. 1987, pp. 2141 – 2144.
- [10] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, Advances in Network and Acoustic Echo Cancellation. Springer-Verlag, 2001.
- [11] G. W. Elko, E. Diethorn, and T. Gänsler, "Room impulse response variation due to thermal fluctuation and its impact on acoustic echo cancellation," in *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, 2003, pp. 67–70.
- [12] H. Sabine, "Room acoustics," Transactions of the IRE Professional Group on Room Acoustics, vol. 1, no. 4, pp. 4–12, Jul. 1953.
- [13] S. Haykin, Adaptive Filter Theory, 4th ed. Prentice-Hall, 2002.
- [14] J. Benesty and T. Gänsler, "New insights into the RLS algorithm," EURASIP Journal on Applied Signal Processing, vol. 2004, no. 3, pp. 331–339, Mar. 2004.
- [15] N. J. Bershad, S. McLaughlin, and C. F. N. Cowan, "Performance comparison of RLS and LMS algorithms for tracking a first order Markov communications channel," in *Proc. IEEE Int. Symposium on Circuits and Systems*, vol. 1, 1990, pp. 266–270.
- [16] K. Ozeki and T. Umeda, "An adaptive filtering using an orthogonal projection to an affine subspace and properties," *Electronics and Communications in Japan*, vol. 67-A, no. 5, 1984.
- [17] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), vol. 5, 1995, pp. 3023–3026.
- [18] S. L. Gay, "Affine projection algorithms," in *Least-mean-square adaptive filters*,
  S. Haykin and B. Widrow, Eds. John Wiley & Sons Inc., 2003, ch. 7, pp. 241–291.
- [19] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic echo cancellation filters - an overview," *Signal Processing*, vol. 80, pp. 1697–1719, 2000.
- [20] P. A. Naylor and N. D. Gaubitch, "Speech dereverberation," in Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC), 2005.

- [21] T. Nakatani, M. Miyoshi, and K. Kinoshita, "One microphone blind dereverberation based on quasi-periodicity of speech signals," in *Neural Information Processing* Systems (NIPS), 2003.
- [22] Y. Lin, J. Chen, Y. Kim, and D. D. Lee, "Blind channel identification for speech dereverberation using l1-norm sparse learning," in *Neural Information Processing* Systems (NIPS), 2007.
- [23] P. M. Peterson, "Simulating the response of multiple microphones to a single acoustic source in a reverberant room." *Journal of the Acoustical Society of America*, vol. 80, no. 5, pp. 1527–1529, Nov. 1986.
- [24] D. L. Duttweiler, "Proportionate normalized least mean square adaptation in echo cancellers," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508–518, Sep. 2000.
- [25] H. Deng and M. Doroslovacki, "Improving convergence of the PNLMS algorithm for sparse impulse response identification," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 181–184, Mar. 2005.
- [26] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), vol. 2, 2002, pp. 1881– 1884.
- [27] E. R. Ferrara, "Fast implementations of LMS adaptive filters," *IEEE Trans. Acoust.*, Speech, Signal Process., vol. 28, pp. 474–475, 1980.
- [28] J. S. Soo and K. K. Pang, "Multidelay block frequency domain adaptive filter," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 373–376, Feb. 1990.
- [29] A. W. H. Khong, P. A. Naylor, and J. Benesty, "A low delay and fast converging improved proportionate algorithm for sparse system identification," *EURASIP Journal* on Audio, Speech, and Music Processing, vol. Article ID 84376, p. 8 pages, 2007.
- [30] K. Wagner and M. Doroslovacki, "Towards Analytical Convergence Analtsis of Proportion-type NLMS Algorithms," in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), 2008, pp. 3825–3828.

- [31] M. H. Hayes, Statistical Digital Signal Processing and Modeling. Wiley, 1996.
- [32] B. Widrow, J. R. Glover, Jr., J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, Jr., and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications," in *Proc. IEEE*, vol. 63, 1975, pp. 1692–1716.
- [33] D. P. Mandic, "A generalized normalized gradient descent algorithm," *IEEE Signal Process. Lett.*, vol. 11, pp. 115–118, 2004.
- [34] R. H. Kwong and E. Johston, "A variable step-size LMS algorithm," *IEEE Trans. Signal Process.*, vol. 40, pp. 1633–1642, 1992.
- [35] C. Rusu and F. N. Cowan, "The convex variable step size (CVSS) algorithm," *IEEE Signal Process. Lett.*, vol. 7, pp. 256–258, 2000.
- [36] J. Sanubari, "A new variable step size method for the LMS adaptive filter," in IEEE Asia-Pacific Conference on Circuits and systems, 2004.
- [37] S. Roy and J. J. Shynk, "Analysis of the data-reusing LMS algorithm," in Proceedings of the 32nd Midwest Symposium on Circuits and Systems, vol. 2, Aug. 1989, pp. 1127 –1130.
- [38] B. A. Schnaufer and W. K. Jenkins, "New data-reusing LMS algorithms for improved convergence," in Proc. Asilomar Conf. on Signals, Systems and Computers, 1993.
- [39] J. A. Apolinario, M. L. R. de Campos, and P. S. R. Diniz, "The binormalized datareusing LMS algorithm," *IEEE Trans. Signal Process.*, vol. 48, pp. 3235 – 3242, Nov. 2000.
- [40] A. W. H. Khong and P. A. Naylor, "Selective-tap adaptive algorithms in the solution of the non-uniqueness problem for stereophonic acoustic echo cancellation," *IEEE Signal Process. Lett.*, vol. 12, no. 4, pp. 269–272, Apr. 2005.
- [41] P. A. Naylor and A. W. H. Khong, "Affine projection and recursive least squares adaptive filters employing partial updates," in *Proc. Asilomar Conf. on Signals, Systems and Computers*, vol. 1, Nov. 2004, pp. 950–954.
- [42] H. Deng and M. Doroslovacki, "New sparse adaptive algorithms using partial update," in Proc. Intl. Conf. on Signal Processing, vol. 2, May 2004, pp. 845–848.

- [43] A. Gilloire and M. Vetterli, "Adaptive filtering in subbands with critical sampling: Analysis, experiments, and application to acoustic echo cacnellation," *IEEE Trans. Signal Process.*, vol. 40, no. 8, pp. 1862–1875, Aug. 1992.
- [44] D. R. Morgan and J. C. Thi, "A delayless subband adaptive filter architecture," *IEEE Trans. Signal Process.*, vol. 43, no. 8, pp. 1819–1830, Aug. 1995.
- [45] Y. Bendel, D. Burshtein, O. Shalvi, and E. Weinstein, "Delayless frequency domain acoustic echo cacnellation," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 5, pp. 589–597, Jul. 2001.
- [46] J. Homer, I. Mareels, R. R. Bitmead, B. Wahlberg, and A. Gustafsson, "LMS estimation via structural detection," *IEEE Trans. Signal Process.*, vol. 46, p. 2651 2663, Oct. 1998.
- [47] S. Makino, Y. Kaneda, and N. Koizumi, "Exponentially weighted step-size NLMS adaptive filter based on the statistics of a room impulse response," *IEEE Trans. Speech Audio Process.*, vol. 1, pp. 101 – 108, Jan. 1993.
- [48] A. Sugiyama, H. Sato, A. Hirano, and S. Ikeda, "A fast convergence algorithm for adaptive FIR filters under computational constraint for adaptive tap-position control," *IEEE Trans. Circuits Syst. II*, vol. 43, p. 629–636, Sep. 1996.
- [49] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in Proc. Asilomar Conf. on Signals, Systems and Computers, vol. 1, Nov. 1998, pp. 394–398.
- [50] A. Deshpande and S. L. Grant, "A new multi-algorithm approach to sparse system adaptation," in Proc. European Signal Processing Conf. (EUSIPCO), 2005.
- [51] M. A. Mehran Nekuii, "A fast converging algorithm for network echo cancelation," *IEEE Signal Process. Lett.*, vol. 11, pp. 427–430, 2004.
- [52] H. Deng and M. Doroslovacki, "Proportionate adaptive algorithms for network echo cancellation," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1794–1803, May 2006.
- [53] M. Martinez-Ramon, J. Arenas-Garcia, A. Navia-Vazquez, and A. R. Figueiras-Vidal, "An adaptive combination of adaptive filters for plant identification," in 14th International Conference on DSP, Jul. 2002, p. 1195–1198.

- [54] J. Arenas-Garca, V. Gmez-Verdejo, and A. R. Figueiras-Vidal, "New algorithms for improved adaptive convex combination of LMS transversal filters," *IEEE Trans. Instrum. Meas.*, vol. 54, pp. 2239 – 2249, 2005.
- [55] M. Dentino, J. M. McCool, and B. Widrow, "Adaptive filtering in the frequency domain," in *Proc. IEEE*, Dec. 1978, pp. 1658–1659.
- [56] A. W. H. Khong, J. Benesty, and P. A. Naylor, "Stereophonic acoustic echo cancellation: Analysis of the misalignment in the frequency domain," *IEEE Signal Process. Lett.*, vol. 13, pp. 33–36, 2006.
- [57] M. Doroslovacki and H. Fan, "Wavelet based time-varying linear system modeling and adaptive filtering," in *Proc. Intl. Symp. on Circuits and Systems*, vol. 2, 1994, pp. 49–52.
- [58] —, "Wavelet-based linear system modeling and adaptive filtering," IEEE Trans. Signal Process., vol. 44, no. 5, pp. 1156–1167, May 1996.
- [59] R. Gray, "On the asymptotic eigenvalue distribution of toeplitz matrices," *IEEE Trans. Inf. Theory*, vol. 18, no. 6, pp. 725–730, Nov. 1972.
- [60] J. J. Shynk, "Frequency-domain and multirate adaptive filtering," IEEE Signal Process. Mag., vol. 9, no. 1, pp. 14–37, Jan. 1992.
- [61] J. W. Cooley and J. W. Tukey, "An algorithm for the machine computation of complex Fourier series," *Math. Comp.*, vol. 19, pp. 297–301, Apr. 1965.
- [62] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing- Principles, Algorithms and Applications*, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 1996.
- [63] S. K. Mitra, Digital Signal Processing- A Computer-Based Approach, 2nd ed. McGraw-Hill, 2001.
- [64] R. Ahmad, A. W. Khong, and P. A. Naylor, "Proportionate frequency domain adaptive algorithms for blind channel identification," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, May 2006, pp. V29–V32.
- [65] B. B. Farhang, Adaptive Filters: Theory and Applications. Chicester: Wiley, 1998.

- [66] J. Benesty, M. M. Sondhi, and Y. Huang, Speech Processing. Springer, 2008.
- [67] P. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. 15, no. 2, pp. 70–73, Jun. 1967.
- [68] J. Lee and S.-C. Chong, "On the convergence properties of multidelay frequency domain adaptive filter," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, 1999, pp. 1865 – 1868.
- [69] P. Loganathan, A. W. H. Khong, and P. A. Naylor, "A sparseness-controlled proportionate algorithm for acoustic echo cancellation," in *Proc. European Signal Pro*cessing Conf. (EUSIPCO), 2008.
- [70] —, "A class of sparseness-controlled algorithms for echo cancellation," IEEE Trans. Audio, Speech, Lang. Process., vol. 17, no. 8, pp. 1591 – 1601, Nov. 2009.
- [71] P. Loganathan, X. S. Lin, A. W. H. Khong, and P. A. Naylor, "Frequency-domain Adaptive Multidelay algorithm with sparseness-control for acoustic echo cancellation," in Proc. European Signal Processing Conf. (EUSIPCO), 2009.
- [72] H. Deng and M. Doroslovacki, "Wavelet-based MPNLMS Adaptive Algorithm for Network Echo Cancellation," EURASIP Journal on Audio, Speech, and Music Processing, 2007.
- [73] P. Loganathan, E. A. P. Habets, and P. A. Naylor, "A partitioned block proportionate adaptive algorithm for acoustic echo cancellation," in *Proceedings of APSIPA Annual Summit and Conference 2010*, Biopolis, Singapore, Dec. 2010.
- [74] —, "A proportionate adaptive algorithm with variable partitioned block length for acoustic echo cancellation," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.
- [75] F. das Chagas de Souza, O. Tobias, R. Seara, and D. Morgan, "A PNLMS algorithm with individual activation factors," *IEEE Trans. Signal Process.*, vol. 58, pp. 2036– 2047, 2010.
- [76] C. Paleologu, J. Benesty, and S. Ciochina, Sparse Adaptive Filters for Echo Cancellation. Morgan & Claypool Publishers, 2010.

- [77] M. Zeller, L. Azpicueta-Ruiz, and W. Kellermann, "Adaptive FIR filters with automatic length optimization by monitoring a normalized combination scheme," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2009, pp. 149–152.
- [78] P. Loganathan, E. A. P. Habets, and P. A. Naylor, "Performance analysis of IPNLMS for identification of time-varying systems," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2010.
- [79] A. W. H. Khong and P. A. Naylor, "Selective-tap adaptive filtering with performance analysis for identification of time-varying systems," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 15, pp. 1681–1695, Jul. 2007.
- [80] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, "Stationary and non-stationary learning characteristics of the LMS adaptive filter," in *Proc. IEEE*, vol. 64, 1976, pp. 1151–1162.
- [81] A. W. H. Khong and P. A. Naylor, "A family of selective-tap algorithms for stereo acoustic echo cancellation," in *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 3, Mar. 2005, pp. 133–136.
- [82] A. H. Sayed, Fundamentals of adaptive filtering. Wiley, 2003.
- [83] C. Paleologu, J. Benesty, and S. Ciochina, Sparse Adaptive Filters for Echo Cancellation. Morgan & Claypool publishers, 2010.
- [84] Y. Sato, "A method of self-recovering equalization for multilevel amplitudemodulation," *IEEE Trans. Commun.*, vol. COM-23, pp. 679 – 682, 1975.
- [85] L. Tong, G. Xu, and T. Kailath, "Blind identification and equalization based on second-order statistics: a time domain approach," *IEEE Trans. Inf. Theory*, vol. 40, p. 340–349, Mar. 1994.
- [86] A. Benveniste, M. Goursat, and G. Ruget, "Robust identification of a nonminimum phase system: blind adjustment of a linear equalizer in data communications," *IEEE Trans. Autom. Control*, vol. 1, pp. 385 – 399, Jun. 1980.
- [87] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Trans. Signal Process.*, vol. 43, pp. 2982–2993, 1995.

- [88] P. A. Naylor and N. D. Gaubitch, Speech Dereverberation. Springer, 2010.
- [89] E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel fir filters," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516–525, Feb. 1995.
- [90] D. Slock, "Blind fractionally-spaced equalization, perfect reconstruction filterbanks, and multilinear prediction," in Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP), vol. 4, 1994.
- [91] Y. Hua, "Fast maximum likelihood for blind identification of multiple FIR channels," *IEEE Trans. Signal Process.*, vol. 44, pp. 661 – 672, Mar. 1996.
- [92] Y. A. Huang and J. Benesty, "Adaptive multi-channel least mean square and Newton algorithms for blind channel identification," *Signal Processing*, vol. 82, p. 1127–1138, 2002.
- [93] Y. Huang and J. Benesty, "A class of frequency-domain adaptive approaches to blind multichannel identification," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 11–24, Jan. 2003.
- [94] E. A. P. Habets, J. Benesty, and P. A. Naylor, "A cross-relation based affine projection algorithm for blind single-input multiple-output system identification," submitted to IEEE Signal Process. Lett., Oct. 2010.
- [95] Y. Huang, J. Benesty, and J. Chen, Acoustic MIMO Signal Processing. Springer, 2006.
- [96] L. R. Vega, H. Rey, J. Benesty, and S. Tressens, "A family of robust algorithms exploiting sparsity in adaptive filters," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, pp. 572 – 581, 2009.
- [97] R. E. Mahony and R. C. Williamson, "Prior knowledge and preferential structures in gradient descent learning algorithms," *Journal of Machine Learning Research*, vol. 1, pp. 311 – 355, 2001.
- [98] D. R. Morgan, J. Benesty, and M. M. Sondhi, "On the evaluation of estimated impulse responses," *IEEE Signal Process. Lett.*, vol. 5, pp. 174 – 176, 1998.

- [99] L. Liao and A. W. H. Khong, "Sparseness-controlled affine projection algorithm for echo cancelation," in *Proceedings of APSIPA Annual Summit and Conference 2010*, Biopolis, Singapore, Dec. 2010.
- [100] F. das C. de Souza, O. J. Tobias, R. Seara, and D. R. Morgan, "Alternative approach for computing the activation factor of the PNLMS algorithm," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Glasgow, Scotland, Aug. 2009.
- [101] J. Yang, X. Zhu, G. E. Sobelman, and K. K. Parhi, "Sparseness-controlled adaptive tap algorithms for partial update adaptive filter," in 7th International Conference on Information, Communications and Signal Processing, Dec. 2009, pp. 1–5.