University of Windsor Scholarship at UWindsor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2011

Efficient channel equalization algorithms for multicarrier communication systems

Ishaq Gul Muhammad University of Windsor

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

Recommended Citation

Gul Muhammad, Ishaq, "Efficient channel equalization algorithms for multicarrier communication systems" (2011). *Electronic Theses and Dissertations*. 7939. https://scholar.uwindsor.ca/etd/7939

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

EFFICIENT CHANNEL EQUALIZATION ALGORITHMS FOR MULTICARRIER COMMUNICATION SYSTEMS

by

Ishaq Gul Muhammad

A Dissertation Submitted to the Faculty of Graduate Studies through the Department of Electrical and Computer Engineering in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy at the University of Windsor

Windsor, Ontario, Canada

2011

© 2011 Ishaq Gul Muhammad



Library and Archives Canada

Published Heritage Branch

395 Wellington Street Ottawa ON K1A 0N4 Canada Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington Ottawa ON K1A 0N4 Canada

> Your file Votre référence ISBN: 978-0-494-80241-0 Our file Notre référence ISBN: 978-0-494-80241-0

NOTICE:

The author has granted a nonexclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or noncommercial purposes, in microform, paper, electronic and/or any other formats.

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

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

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

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

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



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

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

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

Blind adaptive algorithm that updates time-domain equalizer (TEQ) coefficients by Adjacent Lag Auto-correlation Minimization (ALAM) is proposed to shorten the channel for multicarrier modulation (MCM) systems. ALAM is an addition to the family of several existing correlation based algorithms that can achieve similar or better performance to existing algorithms with lower complexity. This is achieved by designing a cost function without the sum-square and utilizing symmetrical-TEQ property to reduce the complexity of adaptation of TEQ to half of the existing one. Furthermore, to avoid the limitations of lower unstable bit rate and high complexity, an adaptive TEQ using equal-taps constraints (ETC) is introduced to maximize the bit rate with the lowest complexity. An IP core is developed for the low-complexity ALAM (LALAM) algorithm to be implemented on an FPGA. This implementation is extended to include the implementation of the moving average (MA) estimate for the ALAM algorithm referred as ALAM-MA. Unit-tap constraint (UTC) is used instead of unit-norm constraint (UNC) while updating the adaptive algorithm to avoid all zero solution for the TEQ taps. The IP core is implemented on Xilinx Vertix II Pro XC2VP7-FF672-5 for ADSL receivers and the gate level simulation guaranteed successful operation at a maximum frequency of 27 MHz and 38 MHz for ALAM-MA and LALAM algorithm, respectively. FEQ equalizer is used, after channel shortening using TEQ, to recover distorted QAM signals due to channel effects. A new analytical learning based framework is proposed to jointly solve equalization and symbol detection problems in orthogonal frequency division multiplexing (OFDM) systems with

QAM signals. The framework utilizes extreme learning machine (ELM) to achieve fast training, high performance, and low error rates. The proposed framework performs in real-domain by transforming a complex signal into a single 2--tuple real-valued vector. Such transformation offers equalization in real domain with minimum computational load and high accuracy. Simulation results show that the proposed framework outperforms other learning based equalizers in terms of symbol error rates and training speeds.

to my

MOTHER and FATHER

with love

Acknowledgements

First of all, thanks to almighty ALLAH for his unending blessings and bounties on me. I would like to express my deep-felt appreciation to my advisors, Dr. Kemal E. Tepe and Dr. Esam Abdel-Raheem for their continuous support and guidance thorough out my research at the University of Windsor. I wish to extend my thanks to PhD committee members Dr. Murat Uysal, Dr. Walid Abdul-Kader, Dr. Jonathan Wu, and Dr. Huapeng Wu for their valuable suggestions and comments on my thesis. Also, my sincere gratitude to former teachers who restlessly contributed towards my education since the day I started learning how to read and write. Especially, my eldest sister, Ameera who was my earliest teacher, my brother Abdul-Rahman as my mentor for mathematics in high school, and all teachers from grade one till today.

It is worth acknowledging my friends for their perpetual support during moiling hours of research to help me concentrate far beyond current impediments at different times. I would like to thank staff and my group fellows at department of ECE for all their hard work and dedication, providing the means to achieve my long-dreamt objective.

My special thanks go to my wife: Sajida, Sons: Muhammad and Ahmad, and Daughter: Sara, for their patience and support while me being away for four long years to finish my graduate studies. I would also like to extend the appreciations to my entire family including my brothers Hussain, Abdul-Rahman, Ibrahim, and Ismail who have supported me in numerous ways, including financial assistance, and took special care of my lovely family. My veracious thanks for sisters Ameera and Aziza who always offered strong affection and support for their youngest brother. Finally, it is an honor for me to dedicate this degree to my mother: Aisha, and father: Gul Muhammad, who are the reason behind each and every accomplishment I ever made, of course, after ALLAH the almighty.

Table of Contents

			I	Page
Aut	hor	's Dec	claration of Originality	iii
Abs	tra	ct		iv
Ded	ica	tion .		vi
Ack	nov	vledge	ments	vii
List	of	Tables	5	xii
\mathbf{List}	of	Figure	e s	xiii
\mathbf{List}	of	Abbre	eviations	xvi
1 I:	ntr	oducti	ion	1
1.	1	Overvi	ew of Multicarrier Modulation Systems	1
1.	2	Chann	el Equalization in MCM	4
		1.2.1	Time-Domain Equalizer (TEQ)	4
		1.2.2	Frequency-Domain Equalizer (FEQ)	5
1.	3	Neural	Networks	6
		1.3.1	Architectures of Neural Networks	6
		1.3.2	Learning Processes	7
		1.3.3	Applications of ANNs	8
1.	4	Resear	ch Objectives	9
1.	5	Resear	ch Contributions	9
1.	6	Organi	ization of Dissertation	11

	2	Bli	nd Adaptive Time-Domain Equalizer Algorithm for ADSL	
		\mathbf{Sys}	stems by Adjacent Lag Auto-correlation Minimization (ALAM)	13
		2.1	Introduction	13
		2.2	Preliminaries	15
			2.2.1 System Model	15
			2.2.2 Conventional Autocorrelation Based Algorithms	16
		2.3	Proposed Low Complexity Algorithm	18
`		2.4	Simulation Results	21
			2.4.1 Validation of ALAM Approach	22
			2.4.2 ALAM-DSS	24
		2.5	Conclusions	26
	3	Effi	cient Blind Adaptive TEQ Designs	27
		3.1	Introduction	27
		3.2	Proposed Low Complexity Algorithms	28
			3.2.1 Low-complexity ALAM, SAM, and SLAM	28
			3.2.2 Exploiting Symmetrical TEQ 3	30
			3.2.3 Adaptive Algorithm for TEQ Using Equal-Taps Constraint (ETC)	32
		3.3	Simulation Results	33
			3.3.1 Validation of LALAM Algorithm	33
			3.3.2 Symmetrical TEQ Taps Design	35
			3.3.3 Equal-Taps TEQ Design	37
		3.4	Conclusions	37
	4	FP	GA Implementation of a Low Complexity Time Domain Equal-	
		izer	for ADSL Systems	1
		4.1	Introduction	1
		4.2	Standard FPGA Design Flow	12
		4.3	TEQ Design and Objectives	4
		4.4	Fixed-point Analysis	5

	4.5	FPGA	A Implementation of ALAM	48
		4.5.1	Weight Update Block	48
		4.5.2	Calculation of Expectation-Term Block	50
	4.6	Imple	mentation Results	52
	4.7	Concl	usion	56
5	QA	M Eq	ualization and Symbol Detection in OFDM Systems us-	
	ing	Extre	eme Learning Machine	60
	5.1	Introd	luction	60
	5.2	Prelin	ninaries	62
		5.2.1	OFDM System Model	62
		5.2.2	Extreme Learning Machine	64
	5.3	Propo	sed Algorithm	67
	5.4	Result	ts and Discussion	70
		5.4.1	Optimal Parameter Selection	71
		5.4.2	Varying Packet Length and Doppler Frequency	77
		5.4.3	Performance Analysis of 4-, 16-, and 64-QAM	77
		5.4.4	Computational Comparison	81
	5.5	Conclu	usions	82
6	Co	nclusio	on and Future Work	84
	6.1	Conclu	uding Remarks	84
	6.2	Future	e Work	86
R	References			87
v	Vita Auctoris			98

۰.

List of Tables

3.1	Complexity comparison in terms of number of multiplications and ad-		
	dition per TEQ update	31	
4.1	Virtex implementation (post-layout) results for clock frequency and area	56	
5.1	Simulation parameters	71	
5.2	Complexity time comparison of training and testing for different algo-		
	rithms	82	

.

List of Figures

1.1	Basic multicarrier transmitter	1
1.2	MCM transceiver architecture	2
1.3	MCM symbol with cyclic prefix.	3
1.4	Learning of ANN.	6
1.5	Feedforward neural network model.	7
1.6	Feedback neural network model	8
2.1	System model for blind adaptive TEQ algorithm for MCM system.	15
2.2	Effective (shortened) channel impulse response using MA method. \ldots .	23
2.3	TEQ taps obtained using MA method	23
2.4	Achievable bitrate vs. iteration number using MA method.	25
2.5	Achievable bitrate vs. SNR using MA and instantaneous estimates	25
3.1	Effective (shortened) channel impulse response using instantaneous method.	34
3.2	TEQ taps obtained using instantaneous method. \ldots	34
3.3	Achievable bitrate vs. iteration number using instantaneous method. \ldots	36
3.4	Achievable bitrate vs. SNR using MA and instantaneous estimates	36
3.5	$ \mbox{ Effective (shortened) channel impulse response for symmetrical ALAM, SAM, } $	
	and SLAM	38
3.6	Achievable bitrate vs. iteration number using symmetrical TEQs	38
3.7	TEQ taps obtained for equal-taps constrained ALAM, SAM, and SLAM	39

3.8	Achievable bitrate vs. iteration number using Equal-taps constrained ALAM,	
	SAM, and SLAM.	39
3.9	Achievable bitrate vs. SNR using symmetrical TEQs and equal-taps con-	
	strained	40
4.1	FPGA design flow.	43
4.2	Fixed-point representation.	45
4.3	Impact of the I/O WL on the performance using 14-bit WL for internal	
	signals	47
4.4	Impact of the internal signals WL on the performance using 10-bit WL $$	
	for I/O signals.	47
4.5	Shortened channel using LALAM Fixed-point simulations	49
4.6	TEQ taps using LALAM Fixed-point simulations	49
4.7	Achievable bit rate using LALAM Fixed-point simulations	50
4.8	Realization of adaptive equalizer using ALAM algorithm	51
4.9	Realization of update equation using ALAM algorithm.	51
4.10	Realization of calculating the gradient of the cost function using (a) MA	
	estimate (b) Instantaneous estimate.	52
4.11	TEQ taps using LALAM RTL simulations	53
4.12	Achievable bit rate using LALAM RTL simulations	53
4.13	TEQ taps using ALAM-MA RTL simulations	54
4.14	Achievable bit rate using ALAM-MA RTL simulations.	54
4.15	TEQ taps using LALAM gate-level simulations.	57
4.16	Achievable bit rate using LALAM gate-level simulations	57
4.17	TEQ taps using ALAM-MA gate-level simulations.	58
4.18	Achievable bit rate using ALAM-MA gate-level simulations	58
5.1	OFDM system model.	63
5.2	ELM architecture.	64

5.3	Proposed block diagram of OFDM system for joint equalization and symbol	
	detection using ELM	68
5.4	Equalization and symbol detection using ELM for 4-QAM.	69
5.5	4 and 16 QAM quadrant distribution.	70
5.6	Performance analysis for 4-QAM for different activation functions using the	
	proposed scheme and C-ELM.	73
5.7	Performance analysis for 4-QAM for varying percentage of training data	
	using proposed framework	73
5.8	Performance analysis for varying number of hidden neurons of ELM in the	
	proposed framework using 4-QAM	74
5.9	Performance analysis for varying number of hidden neurons of ELM in the	
	proposed framework using 16-QAM.	75
5.10	SER vs. Doppler frequency using packet length of 200 symbols 4-QAM. $% \mathcal{A} = \mathcal{A} = \mathcal{A}$.	78
5.11	SER vs packet length using 500 Hz Doppler frequency for 4-QAM. \ldots	78
5.12	Error probability of various scheme for 4-QAM.	79
5.13	Error probability of various scheme for 16-QAM.	80
5.14	Error probability of various scheme for 64-QAM.	80

.

List of Abbreviations

ADSL	Asymmetric Digital Subscriber Line
ALAM	Adjacent Lag Autocorrelation Minimization
ALAM-DSS	ALAM- Decaying Step Size
ANN	Artificial Neural Network
AR	Auto Regressive
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
СВР	Complex Back-Propagation
CELM	Complex Extreme Learning Machine
CLB	Configurable Logic Block
CMA	Constant Modulus Algorithm
CMRAN	Complex Minimal Resource Allocation Network
СР	Cyclic Prefix
CSA	Carrier Serving Area
DAB	Digital Audio Broadcasting
DFT	Discrete Fourier Transform
DMT	Discrete Multi-Tone
DVB	Digital Video Broadcasting

DD-LMS	Decision Directed-LMS
DSL	Digital Subscriber Line
DUT	Design Under Test
ELM	Extreme Learning Machine
ETC	Equal Taps Constraint
FDM	Frequency Division Multiplexing
FIR	Finite Impulse Response
FFT	Fast Fourier Transform
FNN	Feedforward Neural Network
FPGA	Field Programmable Gate Array
FWL	Fractional Word Length
ICI	Inter-Carrier Interference
iid	Independent and Identically Distributed
I/O	Input-Output
IP	Intellectual Property
ISI	Inter-Symbol Interference
IWL	Integer Word Length
k-NN	k-Nearest Neighbor

LALAM	Low-complexity ALAM
LE	Logic Element
LMS	Least Mean Square
LS	Least Square
LSAM	Low-complexity SAM
LSLAM	Low-complexity SLAM
MA	Moving Average
MAC	Multiply ACcumulate
MCM	Multi-Carrier Modulation
MERRY	Multicarrier Equalization by Restoration of Redundancy
MFB	Matched Filter Bound
MLP	MultiLayer Perceptron
MMSE	Minimum Mean Square Error
MSB	Most Significant Bit
MSSNR	Maximum Shortening Signal-to-Noise Ratio
OFDM	Orthogonal Frequency Division Multiplexing
PAR	Place-And-Route
PLC	Power Line Communications

\mathbf{QAM}	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RBF	Radial Basis Function
RLS	Recursive Least Square
RLS	Recursive Least Square
RNN	Recurrent Neural Network
SAAM	Sum-Absolute Autocorrelation Minimization
SAM	Sum-squared Autocorrelation Minimization
SG-Boosting	Stochastic Gradient Boosting
SLAM	Single Lag Autocorrelation Minimization
SLFNN	Single-hidden Layer Feedforward Neural Network
SNR	Signal-to-Noise Ratio
SOM	Self-Organizing Map
TEQ	Time-domain EQualizer
UTC	Unit Tap Constraint
WLAN	Wireless Local Area Network
WL	Word Length
WSS	Wide Sense Stationary
ZF	Zero Forcing

Chapter 1

Introduction

1.1 Overview of Multicarrier Modulation Systems

Multicarrier modulation (MCM) technique was first introduced in late 1950s [1], [2]; it is a novel approach to design high bandwidth communication system in the presence of channel distortion. The basic idea of MCM is to split the transmitting data into several parallel data to be transmitted on subcarriers, instead of single carriers [3]. The basic multicarrier based transmitter is shown in Fig. 1.1. For example, given



Figure 1.1: Basic multicarrier transmitter.

bandwidth of W to transmit M data in parallel over subcarriers of Δf bandwidth for each subchannel results in N number of subchannels, (i.e., $N = \frac{W}{\Delta f}$). For each subchannel, m amount of bits are modulated with subcarrier f_i , for i = 1, 2, ..., N. These subchannels of Δf bandwidth are nearly flat to avoid intersymbol interference

(ISI) caused by frequency selective channel. Although MCM is a form of frequency division multiplexing (FDM), it has the additional feature of overlapping subchannels to maximize the spectral efficiency. In order to avoid interchannel interference (ICI) between adjacent overlapping subcarriers, precise orthogonality between subcarriers is required. The spacing between each subcarriers has to be equal to Δf , otherwise ICI is introduced which causes a severe degradation in performance. This property can be obtained using discrete Fourier transform (DFT) [4]. The n_{th} MCM input symbol x_n is modulated so that there are N subcarriers which are orthogonal to each other. Furthermore, with the existence of modern digital signal processors to implement efficient fast Fourier transform, DFT became a standard used in MCM techniques [5]. Orthogonal frequency division multiplexing (OFDM) and discrete multitone (DMT) are two types of MCM techniques that are used in several wireless systems and Asymmetric Digital Subscriber Line (ADSL) systems, respectively. Examples include digital video broadcast (DVB), digital audio broadcast (DAB), wireless local area networks (WLAN), digital subscriber line (DSL), and power line communications (PLC). Fig. 1.2 shows a standard MCM transceiver architecture. In



Figure 1.2: MCM transceiver architecture.

order to transmit data, regrouping of bit sequence is performed to form blocks of size N by serial-to-parallel converter. Each block forms in fact an OFDM or DMT symbol that consists of, for example, quadrature phase shift keying (QPSK) signal modulated using QPSK modulator. Higher quadrature amplitude modulations (QAM) are used for higher data rate applications. The MCM symbols are then processed using inverse fast Fourier transform (IFFT) to transmit each modulated signal over an orthogonal subcarrier. Cyclic prefix (CP) is added as a guard interval between MCM symbols to avoid ISI and ICI [6]. As shown in Fig. 1.3, CP is a copy of the last portion



Figure 1.3: MCM symbol with cyclic prefix.

of the MCM symbol. The length of the CP is directly dependent on the length of the channel impulse response. Therefore, in order to combat interference caused by multipath fading, a sufficient CP length is required, (i.e., CP length has to be larger than the delay spread of the channel to achieve no ISI). The MCM symbol with CP is a time-domain signal created using the IFFT block. This signal is converted to serial data using parallel-to-serial converter to be transmitted over the wireless or wireline channel. The received signal, at the receiver, is passed through a time-domain equalizer (TEQ) to reduce the length of the effective channel so that its length is less than the CP data. This filter was first introduced in 1990's for multicarrier systems [7]. In traditional MCM transceivers, TEQ is not used. However, when larger delay spread of the channel is present such as in ADSL channel, TEQ is used at the front end of the receivers to shorten the effective channel so that the length of CP can be reduced. Hence, reduction in the CP maximizes the throughput using the relation $\frac{N}{N+CP}$ [8]. After the TEQ operations, CP is removed and serial-to-parallel converter is used to convert the sequence data to parallel. The parallel data are then fed into the FFT block that reconstructs the QAM symbols. Following the demodulation using FFT, the effects of the channels on the modulated signals are equalized using 1-tap frequency domain equalizers (FEQ). Finally, using a demodulator the modulated signals are mapped to bits and are converted from parallel to serial. Since the objective of this thesis is channel equalization in MCM systems we provide an overview of channel equalization in the next Section.

1.2 Channel Equalization in MCM

Equalization in MCM system is divided in two parts: (i) TEQ is used to shorten the effective channel to a length less than the CP length, and (ii) single tap FEQ is used to compensate for the magnitude and phase distortion in each subchannel. TEQ is often used when there is a large delay spread, such as in ADSL lines, to shorten the channel impulse response while maximizing the transmitting bit rate at a fixed bit error rate. Once the shortening is fullfilled, then point wise equalization can be achieved using a bank of complex scalars, (i.e., single tap-FEQs).

1.2.1 Time-Domain Equalizer (TEQ)

Several algorithms have been proposed in literature using trained data to shorten the channel such as minimum mean square error (MMSE) design [9], [10], [11], maximum shortening signal-to-noise ratio (MSSNR) design [12], and others evaluated in [13]. These TEQ designs are non-blind algorithms where channel need to be estimated or it is known in advance. In non-blind equalization techniques, training data reduces the throughput of the system. For that reason, blind, adaptive channel shortening algorithms have been proposed in the literature to increase the throughput of the

system. These algorithms exploit the existing property of the multicarrier modulation systems to build the equalizer taps. Some of the properties includes: presence of the cyclic prefix, presence of null tones, constant modulus signals, etc. Some of the algorithms that exploit such properties are: Multicarrier equalization by restoration of redundancy (MERRY) [14], [15] algorithm which exploits the presence of cyclic prefix to blindly update the TEQ. MERRY algorithm updates the TEQ every MCM symbol which is slow in convergence that makes it not suitable for fast time varying channel [14], [15]. A faster convergence algorithm called sum squared autocorrelation minimization (SAM) algorithm [16], which exploits autocorrelation shortening property of the channel was proposed. Further detailed discussions of SAM and several other algorithms are reviewed in Chapter 2.

1.2.2 Frequency-Domain Equalizer (FEQ)

Single-tap equalization is performed once the shortening of the channel is achieved when the delay spread of the channel is larger than the CP. Like TEQ algorithms, there are training based and blind based algorithms to update the FEQ tap. If training is available then least mean square (LMS) or recursive least Square (RLS) algorithm is used to adapt the FEQ [17]. On the other hand, in applications where training data are not available then blind algorithms are used such as constant modulus algorithm (CMA) [18] or multimodulus algorithm (MMA) [19], [20] to adapt the FEQ by exploiting the desired finite-alphabet output of the FFT. In addition to these algorithms, nonlinear classifications, such as Bayesian decision theory, offers promising solution for equalization and symbol detection problems [3], [21]. Artificial neural networks (ANN), such as ANNs based on multilayer feed-forward neural networks and radial basis function (RBF), have been used for system identification and noise cancelation problems to recover transmitted data [22]. In order to get some insight of neural networks, a brief review of neural networks and their types, adavantages/disadvantages, and applications is introduced in the following Section

1.3 Neural Networks

Neural networks consisting of interconnecting artificial neurons is called artificial neural network (ANN). ANN is basically a generalized mathematical model of human neural system. It processes information using appropriate connection between neurons over connected links. The signal transmitted through a link is multiplied by an associated weight to this link. Activation function is applied to the sum of weighted inputs to calculate the output signals. ANN has an ability to learn the relationship between input and output in linear and non-linear system which makes it a very powerful tool. The training of ANN weights is performed adaptively, as shown in Fig. 1.4, until a criterion is met, (i.e., when output signal is close to a desired output). Once the learning phase is completed, testing data can be processed using the learned model [23].



Figure 1.4: Learning of ANN.

1.3.1 Architectures of Neural Networks

Feedforward Networks

A feedforward neural network, as shown in Fig. 1.5, allows the signal to propagate only in one direction from input-to-output passing through the hidden layer neurons.

[23].

This type of neural network is straight forward and is used normally in applications of pattern recognition.



Figure 1.5: Feedforward neural network model.

Feedback Networks

A feedback neural network shown in Fig. 1.6 allows the signal to propagate in both directions. This type of network are very powerful and complicated as it is dynamic until reaching an equilibrium state. It stays in the equilibrium state until the input changes and the network becomes dynamic again to find a new equilibrium. This type of ANN is also called recurrent neural network.

1.3.2 Learning Processes

There are two types of learning processes for adaptive neural networks which are: supervised learning, and unsupervised learning. In supervised learning, a training set is provided where the input and output are known. Using training data, the supervised learning algorithm builds a model that is used in the testing phase for any unseen data. Multilayer perceptron (MLP) network using backpropagation algorithm [24], [25] is an example of supervised learning that uses the desired output to train



Figure 1.6: Feedback neural network model.

the neural network parameters. One of the methods used to minimize the error in supervised learning to obtain convergence is the least mean square (LMS) algorithm [26]. On the other hand, unsupervised learning are not based on training data but instead, it uses local information to train the neural networks. Self-organizing map (SOM) [27], is an example of unsupervised neural networks that uses its properties to map input and output.

1.3.3 Applications of ANNs

Since ANNs have the ability to solve linear and non-linear problem based on the training data provided, therefore, it is used in various object recognition applications such as recognition of signatures, shapes, words, face, color, etc [23]. ANNs has capability of forming nonlinear decision boundaries that helps in solving complex classification tasks [28] - [31]. Recently, ANN have been also used in modeling nonlinear phenomenon of channel equalizations in wireless communication systems. In the literature, different type of feedforward neural network based equalizers are used like backpropagation [32], radial basis functions [33] - [35], complex minimal resource allocation network (CMRAN) [36], [37], and complex extreme learning machine (CELM) [38] - [40].

1.4 Research Objectives

- The main objective of this thesis is to design new algorithms for channel equalization in multicarrier modulation system. New algorithms are designed to obtain optimum performance with lower complexities than the existing algorithms. Blind adaptive TEQ algorithms designed for ADSL systems are used to achieve the shortening of the effective channel to a length shorter than the cyclic prefix. With this main objective, several other sub-objectives are required to be optimized such as: maximizing the bit rate, maintain low complexity in designing the TEQ, suitability for fast time varying channel, and fast convergence rate.
- Investigate, develop and implement an IP core for the ALAM algorithm on Xilinx Verix II Pro FPGA.
- Use of analytical schemes that jointly solve the problem of equalization and symbol detection in orthogonal frequency division multiplexing (OFDM) systems with QAM signals to achieve fast training, high performance, and low error rates.

1.5 Research Contributions

This section briefly describes the contributions developed to design new algorithms for channel equalization that are efficient in terms of performance and complexity.

- New approach to adapt the TEQ blindly by minimizing the adjacent-lag autocorrelation, without sum-squared as in SAM, to lower the complexity of the adaptive algorithm.
- A decaying step size approach is used, in the proposed ALAM and in the existing SAM and SLAM algorithms, to attain stable bit rate and avoid the drop in bit

rates after reaching the maximum bit rates.

- Low complexity instantaneous estimated approach to estimate the correlation estimated in the adaptive algorithm. With this approach fast convergence is obtained and the TEQ updates every sample which makes it suitable for fast time varying channels.
- Symmetrical property is exploited to reduce the complexity of the proposed ALAM and the existing SAM and SLAM algorithms to half of the original algorithms.
- Equal taps constraint (ETC) is enforced while adapting the algorithms to achieve higher bit rates without the drop down in bit rate once reached its maximum.
- An IP core is developed to implement the low-complexity ALAM algorithm on Xilinx Verix II Pro FPGA for ADSL receivers.
- Unit tap constraint (UTC) approach is implemented in the adaptive algorithm instead of UNC constraint that is very expensive in terms of hardware implementation.
- A new learning based framework to jointly solve equalization and symbol detection problems in orthogonal frequency division multiplexing (OFDM) systems with QAM signals.
- Fully real-valued processing scheme is used to obtain fast convergence and low computational complexity. This is achieved by transforming QAM constellation points to a real-valued vector of 2-tuple, which are labeled with integer values (i.e., class number) as an output. In addition, higher accuracy is attained by associating a 2-tuple pattern to one of the *M* possible classes, which corresponds to an individual QAM constellation point.

1.6 Organization of Dissertation

The remainder of this dissertation consists of 5 chapters which are organized as follows:

- Chapter 2. Covers an introduction of channel shortening algorithms and the background of the existing algorithms for the TEQ. A new low complexity cost function is derived based on the new approach of minimizing adjacent lag autocorrelation minimization to achieve the optimum TEQ taps. In order to avoid the drop in bit rate, a decaying step size is used in ALAM algorithm to achieve stable high bit rates. Both algorithm are validated and results are compared with the existing algorithms.
- Chapter 3. An improvement to the proposed ALAM and the existing SAM and SLAM algorithms in terms of low complexity and suitability to fast time varying channel is achieved. Furthermore, symmetrical property of the TEQ is exploited to reduce the complexity of the original algorithms to half of the non-symmetrical based algorithms. ETC is used while adapting the algorithms to improve the bit rate performance and avoid drop in bit rate with minimum computational complexity. The validation of the proposed approaches used in this chapter are validated and comparative results are obtained and shown in the simulation section of this chapter.
- Chapter 4. Details the design and implementation of the IP core for ALAM algorithm on a Xilinx Vertix II Pro FPGA. For comparison, ALAM algorithm is implemented on FPGA using MA method [16] and the low-complexity instantaneous estimate. Physical synthesis report is generated that confirms the complexity reduction of the proposed method. Simulation results are shown at all stages of the design flow from floating-point simulation to Gate-level simulation to show the functionality of the FPGA implementation.

- Chapter 5. In this chapter, a review of existing schemes for equalization and symbol detection using neural networks are discussed. An analytical learning based framework using extreme learning machine is proposed that solves jointly the problem of equalization and symbol detection in OFDM receivers. Experiments are performed to show the significant improvement of the proposed framework over the exiting approaches in terms of SER and computational complexity.
- **Chapter** 6. In this chapter, conclusions of the proposed research in this thesis is summarized and some recommendation of future work is presented.

Chapter 2

Blind Adaptive Time-Domain Equalizer Algorithm for ADSL Systems by Adjacent Lag Auto-correlation Minimization (ALAM)

2.1 Introduction

In the literature, there are several TEQ designs that are either based on a training data or a blind approach [13] with high complexity. Trained methods require additional overhead that reduces the throughput of the system whereas blind methods do not require additional data; moreover it uses the property of MCM such as the presence of CP, null tones, constant modulus of signals, etc [41]. Several researchers have developed non-blind/non-adaptive TEQ design methods with high computational complexity such as maximum shortening signal-to-noise ratio (MSSNR) [12], minimum mean-square error (MMSE) [7], and several other methods which are summarized and evaluated in [13]. In comparison with non-blind methods, there are limited numbers of blind channel shortening methods for TEQ design in the literature and it has not been explored extensively as blind channel equalization [41].

Among the existing blind TEQ algorithms in the literature, there are: Sum-squared autocorrelation minimization (SAM) [16], sum-absolute autocorrelation minimization (SAAM) [42], single lag autocorrelation minimization (SLAM) [43], [44], multicarrier equalization by restoration of redundancy (MERRY) [14], the algorithms in [45], and the recent algorithm in [46].

MERRY is a low computational complexity algorithm, but has a very slow convergence rate because it updates the TEQ coefficients once per symbol, therefore, it is not suitable for fast time varying environments. On the other hand, SAM, SAAM, and SLAM are high complexity algorithms due to the use of minimizing the squared-autocorrelation and moving average (MA) methods to estimate expectation. Minimizing the squared-autocorrelation produces more than one product terms that increases the complexity of the algorithms. This complexity is obvious when taking gradient of the squared cost function that results in a high complexity adaptive algorithm to update the TEQ coefficients. SAAM [42] on the other hand which is an extensions of SAM, is based on minimizing the sum of absolute values instead of sum squared values as in SAM. This algorithm uses signum function in its adaptive algorithm, that is why it has lower complexity than SAM. These algorithms share the same drawback of drop down in bitrate which can be treated by freezing the TEQ once it reaches the maximum shortening SNR as in SLAM [44]. However, freezing the TEQ makes the algorithm non-adaptive and that makes it not suitable for fast time varying environments.

The proposed algorithm in this chapter is focused on reducing the complexity while achieving similar performance as SAM and SLAM. ALAM is the proposed blind adaptive algorithm that exploits the uncorrelation of the adjacent data samples produced by IFFT in ADSL systems to shorten the channel. This property is degraded due to the large delay spread in ADSL channel. The restoration of this property is achieved while adapting the TEQ by minimizing the adjacent-lag autocorrelation of the TEQ output. In this approach, without the *square* operator as in SAM and



Figure 2.1: System model for blind adaptive TEQ algorithm for MCM system.

SLAM, the gradient function is a single product term of the correlation between the TEQ's input and output. Therefore, the complexity of ALAM adaptive algorithm is reduced without the degradation in performance. Unit-norm constraint (UNC) is implemented on the adaptive algorithm as in SAM and SLAM to avoid all zero solution on the TEQ taps [16]. In order to validate the proposed ALAM algorithm, MA method used in SAM [43] is implemented on ALAM to calculate the correlation estimates. A comparative simulations for ALAM, SAM, and SLAM are performed to show effectiveness of the low complexity proposed algorithms. The algorithms using MA estimate are referred in this thesis as ALAM-MA, SAM-MA, and SLAM-MA. All of those algorithms have the drawback of drop in bitrate once the bitrate reaches its maximum. In order to treat such a problem, a decaying step size is used to control the convergence and achieve steady bitrate performance.

2.2 Preliminaries

2.2.1 System Model

In MCM systems, each of the M subcarrier is modulated with a quadrature amplitude modulation (QAM) signal. The modulation is formed using M-point FFT to produce orthogonal data to avoid ISI and ICI. Figure 2.1 illustrates the system model used to shorten the effective channel in multicarrier systems. The source sequence x(n) contains M + v number of samples (bins) and v here is length of the cyclic prefix. These x(n) samples are uncorrelated to each other and are produced using IFFT of the QAM symbol data X_k . The signal x(n) is transmitted through linear finite-impulse-response (FIR) channel of length $L_h + 1$ taps. Those bins form a random process which is white, wide sense stationary (WSS), and real with zero-mean and unit variance [16].

The received sequence r(n) is described by

$$r(n) = \sum_{k=0}^{L_h} h(k)x(n-k) + v(n), \qquad (2.1)$$

where n is the sample index and v(n) is a zero-mean AWGN sequence, uncorrelated with the source sequence and has a variance σ_v^2 . The sequence r(n) is fed to the TEQ of length $L_w + 1$ taps. The effective channel is a discrete time convolution between the channel **h** and equalizer tap vector **w**, i.e., $\mathbf{c} = \mathbf{h} * \mathbf{w}$, where the length of the effective channel is $L_c + 1$, where $L_c = L_h + L_w$. The output sequence of the TEQ is given by

$$y(n) = \sum_{k=0}^{L_w} w(k) r(n-k) = \mathbf{w}^T \mathbf{r}_n, \qquad (2.2)$$

where $\mathbf{w} = [w_0 \ w_1 \ w_2 \ \dots \ w_{L_w}]^T$ and the TEQ filter input regressor vector $\mathbf{r}_n = [r(n) \ r(n-1) \ r(n-2) \ \dots \ r(n-L_w)]^T$.

2.2.2 Conventional Autocorrelation Based Algorithms

The conventional correlation based blind adaptive channel-shortening algorithms are SAM [16] and SLAM [43], [44]. Both use received signals to adapt the TEQ coefficient, SAM cost function is designed by minimizing the sum-squared autocorrelation of the received signal defined by

$$\boldsymbol{J}_{v+1} = \sum_{l=v+1}^{L_c-1} | R_{cc}(l) |^2, \qquad (2.3)$$
where $R_{cc}(l) = R_{yy}(l)$ under noiseless scenario [16]. The SAM adaptive algorithm over the cost function defined in Equation (2.3) is defined as

$$\mathbf{w}^{k+1} = \mathbf{w}^{k} - \mu \sum_{l=\nu+1}^{L_{c}} \nabla_{w} (E[y(n)y(n-l)])^{2}
= \mathbf{w}^{k} - 2\mu \sum_{l=\nu+1}^{L_{c}} \{E[y(n)y(n-l)]\} \{E[y(n)\mathbf{r}_{n-l} + y(n-l)\mathbf{r}_{n}]\},$$
(2.4)

and the implementation of the above adaptive algorithm is performed using MA or auto regressive (AR) estimators. The coefficient update equation of the TEQ using SAM algorithm with MA method to estimate the expectation is given by

$$\mathbf{w}^{k+1} = \mathbf{w}^{k} - \mu \sum_{l=\nu+1}^{L_{c}} \left[\left\{ \sum_{l=kN}^{(k+1)N-1} \frac{y(n)y(n-l)}{N} \right\} \left\{ \sum_{l=kN}^{(k+1)N-1} \frac{y(n)\mathbf{r}_{n-l} + y(n-l)\mathbf{r}_{n}}{N} \right\} \right],$$
(2.5)

where N is the design parameter that defines the window size of samples (averaging block) to estimate the expectation, larger N gives better estimate but higher complexity since the complexity is directly proportional to N. SAM adaptive algorithm is based on estimating the correlation between the equalizer outputs and the received samples using MA estimator. The estimator can also be implemented using AR method [16] which is faster in convergence however AR method depends depends on previous estimated values, and hence its only good for slow varying channels. A simplified version of SAM was proposed by Nawaz and Chambers called SLAM [43], [44]. SLAM also used MA and AR estimator and has similar performance as SAM at a lower complexity. The coefficient update equation of the TEQ using SLAM algorithm with MA method to estimate the expectation is given by

$$\mathbf{w}^{k+1} = \mathbf{w}^{k} - \mu \left[\left\{ \sum_{l=kN}^{(k+1)N-1} \frac{y(n)y(n-l)}{N} \right\} \left\{ \sum_{l=kN}^{(k+1)N-1} \frac{y(n)\mathbf{r}_{n-l} + y(n-l)\mathbf{r}_{n}}{N} \right\} \right].$$
(2.6)

Another extensions of SAM reported in [42] called SAAM is based on minimizing the sum of absolute values instead of sum-squared values as in SAM. This algorithm uses *signum* function in its adaptive algorithm and that is why it has lower complexity than SAM.

2.3 Proposed Low Complexity Algorithm

Since the channel input, x(n) sequence excluding the cyclic prefix data, is white with zero-mean, therefore the covariance matrix $E[\mathbf{x}_n \mathbf{x}_n^T]$ is equal to the correlation matrix which is defined as

$$E[\mathbf{x}_{n}\mathbf{x}_{n}^{T}] = \begin{bmatrix} R(0) & R(1) & R(2)... & R(M) \\ R(1) & R(0) & R(1)... & : \\ R(2) & R(1) & R(0)... & R(1) \\ \vdots & & & \\ R(M) & .. & R(1)... & R(0) \end{bmatrix},$$
(2.7)

where R(0) is the unity variance and all other non diagonal elements, R(1), R(2), ...R(M), being zero due to the the independent and identically distributed (i.i.d.) assumption that the output of IFFT produces output samples that are uncorrelated in MCM system. The correlation with lag l is defined by

$$E[\mathbf{x}_{n}\mathbf{x}_{n-l}^{T}] = \begin{bmatrix} R(l) & \dots & R(l+M) \\ \vdots & \dots & \vdots \\ R(l-M) & \dots & R(l) \end{bmatrix}.$$
 (2.8)

This correlation model is a Toelpitz matrix, with only one diagonal of nonzero entries and depending on the lag value l, the diagonal vector shifts up or down. The shifting matrix is a consequence of the i.i.d. assumption about the channel input sequences, x(n). In an MCM system, x(n) becomes non i.i.d. due to the the existence of cyclic prefix data. However, the shifting structure can still be present if and only if the correlating samples are within the M consecutive samples which means that the lag l is within the value defined by

$$l \le M - L_h - i, \tag{2.9}$$

where i is the received sample index. If this assumption is violated then the matrix in Equation (2.8) will have another non-diagonal entries that correspond to the

correlation between transmitted data and the cyclic prefix data. Since SAM uses a range of l in estimating the autocorrelation, the violation probability of the condition in Equation (2.9) would be higher if the the length of the channel L_h is large. The proposed approach utilizes only adjacent lag to estimate the correlation, this is why the condition in Equation (2.9) can be easily met. One of the lags to optimize the TEQ is sufficient and R(1) is the adjacent lag autocorrelation intended to minimize at the receiver to optimize the TEQ coefficients, which is used to formulate the proposed cost function and is defined by

$$J = R_{yy}(1) = E[y(n)y(n-1)], n \in 1, 2, \dots, M+v,$$
(2.10)

where n is the sample number of the MCM symbol. As shown in the proposed cost function in Equation (2.10), the sum and square is removed from the cost function to reduce the complexity of the proposed algorithm without any loss in performance which is shown in our simulation studies in Section 2.4. The optimization problem for the proposed algorithm is defined as

$$\mathbf{w}^{opt} = \arg_w \min_{\|\mathbf{w}\|_2^2 = 1} \ \boldsymbol{J}$$

, where a unit norm constraint is applied to avoid non-zero solution. The cost function is derived similarly as [16] and [44] but with only adjacent lag and without sum squared autocorrelation

$$J = R_{yy}(1) = E[y(n)y(n-1)]$$

= $E[(\mathbf{c}^T \mathbf{x}_n + \mathbf{w}^T \mathbf{v}_n)(\mathbf{x}_{n-1}^T \mathbf{c} + \mathbf{v}_{n-1}^T \mathbf{w})]$ (2.11)
= $R_{cc}(1) + \sigma_v^2 R_{ww}(1),$

where the term $\sigma_v^2 R_{ww}(1)$ is eliminated as being very small compare to $R_{cc}(1)$ which is the lag-1 autocorrelation of the effective channel as in [44]. Therefore, the cost function in Equation (2.11) is the autocorrelation of adjacent lag of the TEQ output sequence which is equal to the autocorrelation of adjacent lag of the effective channel, i.e., $R_{yy}(1) \approx R_{cc}(1)$, therefore the final cost function is simplified as

$$J = E[y(n)y(n-1)].$$
 (2.12)

The effect of removing the sum and square in our proposed algorithm as compared to SAM is investigated in Section 2.4 and in the adaptive algorithm for the proposed TEQ using the steepest gradient-descend algorithm to update the equalizer coefficients given by

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \mu \nabla_w E[y(n)y(n-1)]$$

= $\mathbf{w}^k - 2\mu E[y(n)\mathbf{r}_{n-1}],$ (2.13)

where $k = 0, 1, ..., L_w - 1$, n = 1, 2, ..., M + v, μ is the step size, and for simplification the delayed TEQ input regressor vector $\mathbf{r}_{n-1} \approx \mathbf{r}_n$ (This approximation is used throughout in the thesis while implementing the proposed adaptive algorithms). As shown from the derived gradient of the cost function in Equation (2.12) which is used in the proposed adaptive algorithm in Equation (2.13), the update depends on estimating the correlations between the received samples and equalizer output. Note that, SAM algorithm in Equation (2.4) has additional terms that are formed due to the squared autocorrelation in the cost function of Equation (2.3), where as the proposed algorithm has only single correlation term. A proper step size in the ALAM adaptive algorithm can compensate for additional terms in SAM, furthermore, the performance of the simplified proposed approach is investigated and compared with SAM performance in Section 2.4.

A stochastic gradient descent update has been implemented using moving average (MA) or auto-regressive (AR) estimate in [16] and [44]. MA method is defined by

$$E[y(n)y(n-1)] = \sum_{n=1}^{N} \frac{y(n)y(n-1)}{N},$$
(2.14)

where N here is user defined window to perform averaging. On the other hand, AR estimate is defined by

$$E[y(n)y(n-1)]_{i} \approx (1-\alpha)E[y(n)y(n-1)]_{i-1} + \alpha y(n)y(n-1), \qquad (2.15)$$

where i and i - 1 are current and previous estimates, respectively.

According to [16], AR estimate is less complex, however, it depends on previous estimate which makes it unsuitable for fast time varying environments, where as MA estimate performs the estimate over a predefined window of size N samples where each update is performed every N samples. Both MA and AR implementation methods have been used in the literature to estimate the expectation. In [16] MA and AR methods have been used with a complexity of $4NL_w(L_c-v)$ and $4L_w(L_c-v)$ multiplications/additions in each update, respectively. Thus, AR method has about N times lower complexity than MA method, however, the AR method depends on the previous estimates which makes it unsuitable for fast time variation environments [16]. SLAM, on the other, has complexity equal to $3NL_w$ [43] using MA method and equal to $4L_w$ [44] using AR method. However, SLAM has the drawback of being dependable on the previous estimate which makes it unsuitable for fast time varying channel. In the proposed TEQ algorithm, the expectation is implemented by MA estimate which has a complexity of $2NL_w$ multiplications/additions per update. The implementation of our proposed ALAM using MA method validates the approximation in Equation (3.2) as shown by simulation in Section 2.4.

2.4 Simulation Results

In this Section, the low complexity ALAM algorithm using MA estimate is simulated and compared with the original SAM and SLAM algorithms. In our simulation 75 DMT symbols are used to simulate results and the ADSL downstream parameters used are: FFT size M = 512, v = 32 for CP, 16 taps for the TEQ, and 40 dB for $\frac{\sigma_x^2 ||h||^2}{\sigma_n^2}$. ADSL performance metric is the achievable bitrate, R_b , given at a fixed bit error rate (BER) which is determined by

$$R_b = \sum_{i}^{M} \ln \left(1 + \frac{SNR_i}{\Gamma}\right), \qquad (2.16)$$

where SNR_i is the SNR for each subcarrier *i*, and Γ is a parameter that depends on 9.8-dB SNR gap, 6-dB margin and 4.2-dB coding gain [13]. The simulation of the proposed algorithm and the modifications to exiting algorithms are written in Matlab using the standard carrier serving area (CSA) loop 1 available in [47] and integrated with the Matlab code in [48] and the design tool box in [47].

To test the performance of the proposed blind adaptive algorithms, TEQ is initialized with single center spike and the step sizes used for the algorithms are 0.15, 5, and 600 for ALAM-MA, SAM-MA, and SLAM-MA, respectively.

2.4.1 Validation of ALAM Approach.

The proposed ALAM algorithm is validated by comparing it with the conventional SAM in [16], SLAM in [43], MSSNR solution, and the matched filter bound (MFB). The metric used to validate our results are bitrates, TEQ taps, and shortened channel. To validate the ALAM approach of adapting TEQ using adjacent lag autocorrelation without sum-square, ALAM adaptive algorithm of Equation (2.13) is implemented using MA method used in SAM [16] to estimate the expectation. Fig. 2.2, Fig. 2.3 and Fig. 2.4 show the simulation using MA method to estimate the expectation. Simulation results show that the algorithms are slow in convergence because the updating of TEQ is performed every block. Fig. 2.2 shows that ALAM algorithm using MA methods shortens the channel to a length similar to SAM and SLAM. Furthermore, the optimized TEQ taps after convergence using the proposed ALAM-MA are demonstrated in Fig. 2.3, where the TEQ taps are almost identical to SAM and SLAM.



Figure 2.2: Effective (shortened) channel impulse response using MA method.



Figure 2.3: TEQ taps obtained using MA method.

algorithms. The achievable bitrate vs. the iteration number (average blocks) is shown in Fig. 2.4. This figure is simulated using MA estimate where each averaging block N = 32 samples which shows that the convergence for ALAM-MA, SAM-MA and SLAM-MA is achieved at approximately 320, 260, and 1000 blocks, respectively, (i.e., in terms of DMT symbols the convergence is achieved at approximately 18, 15, and 58 symbols, respectively). These figures validates the proposed approach of adjacent lag autocorrelation minimization that shows that the channel shortening is achieved with lower complexity algorithm. Fig. 2.5 shows the achievable bitrate versus SNR for MSSNR, SAM-MA, SLAM-MA and the proposed algorithm ALAM-MA. The bitrate at each SNR is obtained after transmitting 75 symbols. Observe that ALAM-MA and SAM-MA achieve almost similar results but SLAM-MA outperforms both of them. Note that SLAM in this case is outperforming because the achieved bitrate is recorded after transmitting 75 symbols, (i.e., after transmitting 1200 blocks), which is high as shown in Fig. 2.4. However, as shown in this figure that at 75 symbols it is still in the process of decaying and eventually at steady state it will be the same as the ALAM-MA and SAM-MA.

2.4.2 ALAM-DSS.

In order to avoid the drop down in bitrate once reached its maximum, a decaying step size function is used to achieve MSSNR solution. The step size controls the convergence rate. As the bitrate reaches maximum value, the step size approaches a value close to 0. This technique works similarly as the stopping criterion used in SLAM algorithm which examines the energy of the TEQ taps, and freezes the TEQ tap once reached a threshold value. Unlike the stopping criterion which requires the designer to determine the threshold, ALAM decaying step size (ALAM-DSS) does not require such condition where the problem of stopping is treated by DSS function as it approaches 0. By doing so, both techniques are in the non-adaptive mode that can not be suitable for time varying channels. The tradeoff in using ALAM-DSS is



Figure 2.4: Achievable bitrate vs. iteration number using MA method.



Bitrates vs. SNR after transmitting 75 symbols

Figure 2.5: Achievable bitrate vs. SNR using MA and instantaneous estimates.

slight decrease in bitrate for not using a threshold based criterion. This technique can be used in the proposed algorithms and the existing SAM and SLAM. But to avoid redundancy, only ALAM-DSS is simulated and the bitrate versus iteration numbers are shown in Fig. 2.4. In this figure as observed, the convergence with initial step size of 0.5 is achieved at around 400 block number, which is the 23rd DMT symbol. Note that the bitrate after the convergence becomes almost constant which means that the adaptive algorithm have stopped adapting, i.e., the TEQ algorithm entered a freezing state.

2.5 Conclusions

In this Chapter, a low complexity blind adaptive algorithm called ALAM for the TEQ to shorten the channel has been proposed. The proposed algorithm is based on a new criterion which is to minimize of the adjacent lag autocorrelation. The reduction in complexity is attained by simplified cost function and the validation of the proposed approach is verified with existing SAM algorithm using MA method to calculate the expectation. A decaying step size is used in ALAM and the existing SAM and SLAM algorithms to control the convergence and achieve steady bitrate performance. Comparative analysis in terms of computational complexities and performance has been performed. Simulation showed the effectiveness of the proposed low complexity approach with small degradation in convergence rate.

Chapter 3

Efficient Blind Adaptive TEQ Designs

3.1 Introduction

The new algorithms developed in Chapter 2 and the existing algorithms SAM [16] and SLAM [43] have some common limitations. These algorithms are slow in convergence as they update every block and are not suitable for fast time varying channels. This is due to the nature of the MA method used in those algorithms to estimate the correlation that would slow down the update of the adaptive algorithm. In addition, those algorithms require high computational complexity as they require an averaging over a window of samples for each update. In order to achieve fast convergence and low complexity, AR method is used to update the adaptive algorithms. However, this update depends on previous estimate while estimating the autocorrelation. Because of this, AR method it only suitable for slow time varying channels [16]. Moreover, the algorithms in Chapter 2 and in [16], [44] share the same drawback of drop down in bitrate which can be treated by freezing the TEQ, using stopping criterion, once it reaches the maximum shortening SNR as in SLAM [44]. Hence, they are not suitable for fast time varying environments. Another approach in Chapter 2 was proposed where the freezing of the TEQ taps is achieved using variable step size. In this approach, threshold value is not required which does not require a manual

setting and saves some amount of computations required for examining the TEQ taps while adapting. However, this approach also becomes only suitable for time invariant channel due to the freezing of the TEQ taps.

In order to minimize complexity, several researchers have used TEQ filter properties to reduce the amount of computation cost of the channel shortening algorithms. Symmetrical property of the TEQ filter have been exploited in non-correlating based algorithms [50], such as in MSSNR, MMSE and MERRY. In this Chapter, the proposed algorithms are focused on reducing the complexity while achieving performance similar or better than SAM-MA and SLAM-MA. The contributions of this Chapter are as follows: (*i*) propose low complexity algorithms that use computational efficient method for fast time varying channel to estimate the expectation to replace the MA or AR method used in SAM and SLAM algorithms [16], [43], [44], (*ii*) exploit the symmetrical property of TEQ [50] to reduce the amount of computation by 50% in ALAM, SAM, and SLAM algorithms with slight decrease in convergence time, and *iii*) a TEQ design is proposed that enforces equal TEQ taps while adapting to achieve maximum stable bitrate, this allows the algorithms to avoid the limitations of drop down in bitrate in ALAM, SAM, and SLAM.

3.2 Proposed Low Complexity Algorithms

3.2.1 Low-complexity ALAM, SAM, and SLAM

Recall from chapter 2, a stochastic gradient descent update is implemented using MA or AR estimate in [16] and [44]. Our focus in this Chapter is to simplify the MA method to reduce the complexity and update the adaptive algorithm every sample to achieve faster convergence. The MA method is defined by [16]

$$E[y(n)y(n-1)] = \sum_{n=1}^{N} \frac{y(n)y(n-1)}{N},$$
(3.1)

where N here is user defined window to perform averaging. This MA method performs the estimate over a predefined window of size N samples that leads to a slow convergence and update the adaptive algorithm every N samples. Our approach for calculating the expectation, on the other hand, is based on a simplified version of MA estimate. It is assumed that the output of IFFT is a stationary process due to the WSS Gaussian process as per the central limit theorem [51]. For this reason, the simplified low complexity, instantaneous estimate, is used in the adaptive algorithm which is approximated as

$$E[y(n)y(n-1)] \approx y(n)y(n-1), \qquad (3.2)$$

This approximation is equal to the MA estimate. If $E[y(n)y(n-l)] = R_{yy}(l)$ for all n, for a stationary process, then

$$E[y(n)y(n-1)] = \frac{1}{N} N y(n)y(n-1) = y(n)y(n-1).$$
(3.3)

In addition to the above, the justification for this approximation comes from the fact that this expectation is estimated using the stochastic gradient algorithm that performs a two-fold way of averaging [52]. If we use the MA method in the stochastic gradient algorithm as in Equations (2.5) and (2.6) of SAM and SLAM respectively, then it looks like a redundant procedure is being done by performing an average of averages. For more details about this approximation see Section 2.5 of [52].

Both MA and AR implementation methods have been used in the literature to estimate the expectation. In [16] MA and AR methods have been used with a complexity of $4NL_w(L_c - v)$ and $4L_w(L_c - v)$ multiplications/additions in each update, respectively. Thus, AR method is lower in complexity by N times compare to MA method. However, the AR method depends on the previous estimates which makes it unsuitable for fast time variation environments [16]. SLAM, on the other hand, also uses the MA and AR method with a complexity of $3NL_w$ [43] and $4L_w$ [44], respectively. However, SLAM has the drawback of being dependent on the previous estimate which makes it suitable for slow time varying channel.

In the proposed TEQ algorithm, the expectation is implemented by instantaneous and MA estimates with complexities of $2L_w$ and $2NL_w$ multiplications/additions per update, respectively. The coefficient update equation for the proposed low-complexity ALAM (LALAM) blind adaptive algorithm using instantaneous estimate implementation of Equation (3.2) is given by

$$\mathbf{w}^{k+1} = \mathbf{w}^k - 2\mu \{ y(n)\mathbf{r}_{n-1} \}.$$
 (3.4)

The proposed method of instantaneous estimate is also implemented on SAM-MA and SLAM-MA to lower their complexity by a factor of N. The coefficient update equation for the low-complexity SAM (LSAM) is given by

$$\mathbf{w}^{k+1} = \mathbf{w}^k - 2\mu \sum_{l=\nu+1}^{L_c} \{y(n)y(n-l)\}\{y(n)\mathbf{r}_{n-l} + y(n-l)\mathbf{r}_n\},$$
(3.5)

and for the low-complexity SLAM (LSLAM) is given by

$$\mathbf{w}^{k+1} = \mathbf{w}^k - 2\mu \{ y(n)y(n-l) \} \{ y(n)\mathbf{r}_{n-l} + y(n-l)\mathbf{r}_n \}.$$
 (3.6)

The proposed LALAM, LSAM, and LSLAM algorithms are of low complexity and they update every sample which makes them depend on the current settings; making them suitable for fast time varying channels. The complexity comparisons of the algorithms using MA estimate and the low complexity instantaneous estimate are shown in Table. 3.1.

3.2.2 Exploiting Symmetrical TEQ

This Section considers the symmetrical TEQ design to reduce the complexity of proposed ALAM algorithm and the existing SAM and SLAM algorithms. In literature

Algorithms	MA estimate	Inst. estimate
SAM	$4NL_w(L_c-v)$	$4L_w(L_c-v)$
SLAM	$3NL_w$	$3L_w$
ALAM	$2NL_w$	$2L_w$

 Table 3.1: Complexity comparison in terms of number of multiplications and addition per TEQ update

it has been shown that the optimum TEQ taps become symmetrical as the length of the TEQ tends to infinity [53], [54], [55]. Several researchers have utilized this property to force the TEQ to be perfectly symmetric in the case of a finite length TEQ in non-correlating based TEQ algorithms [50] to reduce the complexity by 50% compare to the conventional TEQ design. TEQ length can be even or odd, and it is noted that when the length is even the enforcing of symmetry is described by

$$\mathbf{w}^T = [\mathbf{f}^T \ (\mathbf{I}_c \mathbf{f})^T], \tag{3.7}$$

while when the TEQ length is odd, symmetry it is enforced by

$$\mathbf{w}^T = [\mathbf{f}^T \quad \gamma \quad (\mathbf{I}_c \mathbf{f})^T]. \tag{3.8}$$

where γ is a scalar initialized as 1, $\mathbf{f} = [w_0 \ w_1 \ w_2 \ \dots \ w_{\lfloor L_w/2 \rfloor}]^T$ where $\lfloor . \rfloor$ is the floor function, and \mathbf{I}_c is the cross diagonal identity matrix defined as

$$\mathbf{I}_{c} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & \cdots & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \end{bmatrix}$$
(3.9)

Compared to the non-symmetrical implementation of LALAM, LSAM, and LSLAM in Equations (3.4), (3.5), and (3.6), respectively, the number of multiplications is

reduced by 50% for Sym-LALAM, Sym-LSAM, and Sym-LSLAM where the adaptation of TEQ is performed only for $L_w/2$ taps and symmetry is enforced to form w. Without loss of generality, we only show the odd symmetrical LALAM coefficient update equation which is described by

$$\mathbf{f}^{k+1} = \mathbf{f}^k - 2\mu \{ y(n)\mathbf{r}_{n-1} \}.$$
 (3.10)

$$\mathbf{w}^T = [\mathbf{f}^T \ \gamma \ (\mathbf{I}_c \mathbf{f})^T]. \tag{3.11}$$

The proposed algorithms using symmetrical TEQ design reduces the complexity while maintaining similar bitrate performance compare to SAM and SLAM in [16] and [43] with slight decrease in convergence rate.

3.2.3 Adaptive Algorithm for TEQ Using Equal-Taps Constraint (ETC)

A new approach for adapting the TEQ algorithm using equal-taps constraint (ETC) is introduced. This approach avoids the limitations in the above proposed and existing algorithms which are the drop down in bitrate and choosing bad minima while converging blindly. An extensive research has been done by Balakrishnan, et al. [16], in which an investigation has been carried out and conclusion was drawn that blind channel shortening algorithms produce multiple minima and the cost function is invariant to the operation of flipping the TEQ taps w. Theorem I in [16] states that whenever there is a good minima at w_o , there is also another minima at the flipped w_o but this does not mean that flipped one is the best minima. Therefore, it is clear that there are multiple minima and to achieve the best minima in order to maximize the bitrate, a global search technique is needed. Toker and Altin [56] proposed a genetic algorithm to search for the global minima to select the best minima. The use of genetic algorithm is very costly, in terms of complexity and searching time, as it requires to search all the combinations of TEQ taps to reach the best minima. A new approach is used to update a single tap using our proposed methods which will produce an equal taps for a single update which reduces the complexity of the proposed algorithms by L_w . Using this approach we can achieve a suitable minima that gives higher stable bitrate than the existing algorithms which is demonstrated by experiments in Section 3.3.

The scalar adaptive algorithm using the equal-taps constraints for LALAM (ETC-LALAM) can be formulated as

$$f^{k+1} = f^k - 2\mu \{ y(n)r_{n-1} \}, \tag{3.12}$$

and the equal TEQ taps are enforced in the above adaptive algorithm by the vector $\mathbf{w}^T = [f \dots f]$ of length L_w .

Similarly, the ETC approach is applied in LSAM and LSLAM algorithms to reduce their complexity and achieve better stable bitrates, the resulting algorithms are labeled as ETC-LSAM and ETC-LSLAM, respectively.

3.3 Simulation Results

In this Section the simulation setup/parameters of Chapter 2 are used to test the performance of the proposed efficient algorithms. The step sizes chosen based on extensive simulation are 0.01, 0.05, and 7 for LALAM, LSAM, and LSLAM, respectively.

3.3.1 Validation of LALAM Algorithm.

The proposed LALAM algorithm, and the modified SAM and SLAM algorithms are validated by comparing them with the conventional SAM in [16], SLAM in [43], MSSNR solution, and the matched filter bound (MFB). The metric used to validate our results are bitrates, TEQ taps, and shortened channel. To validate the low complexity LALAM algorithm, the adaptive algorithm of Equation (3.4) is implemented using the low complexity, instantaneous method to calculate correlation estimate.



Figure 3.1: Effective (shortened) channel impulse response using instantaneous method.



Figure 3.2: TEQ taps obtained using instantaneous method.

The shortened channel and the TEQ taps of the low-complexity algorithms using instantaneous estimate are demonstrated in Fig. 3.1 and Fig. 3.2, respectively, where it is shown that the optimization of TEQ taps to shorten the channel can be achieved using instantaneous estimate instead of the high complexity averaging method i.e., MA method. The achievable bitrate vs. the iteration number (sample number) is shown in Fig. 3.3. This figure shows that the convergence is achieved at approximately 5000, 2500, and 7500 samples for LALAM, LSAM, LSLAM, respectively, (i.e., in terms of DMT symbols the convergence is achieved at approximately 9, 5, and 13 symbols, respectively). In comparison with the simulation of ALAM algorithm using MA method shown in Fig. 2.4 of Chapter 2, it is shown that LALAM, LSAM, and LSLAM are faster than ALAM-MA, LSAM-MA, and LSLAM-MA by 9, 10, and 45 symbols, respectively. These figures confirm that the proposed adaptive algorithm and the modified SAM and SLAM using instantaneous estimate attain faster convergence rates while the algorithm update per sample which make them suitable for fast time varying channel. The simulation in Fig. 3.4 shows the achievable bitrate versus SNR for MSSNR, SAM-MA, SLAM-MA and the proposed algorithms. The bitrate at each SNR is obtained after transmitting 75 symbols. Observe that the low complexity algorithms, LALAM, LSAM, and LSLAM achieve similar results as the original SAM-MA and SLAM-MA algorithm for SNRs up to 40dBs, however at higher SNRs LALAM and LSAM perform better than LSLAM.

3.3.2 Symmetrical TEQ Taps Design.

This Section presents simulation of Sym-LALAM, Sym-LSAM, and Sym-LSLAM algorithm. Fig. 3.5 shows the shortened channel after running symmetrical based adaptive algorithms that obtains bit rate performance shown in Fig. 3.6. This figure shows the bitrate versus iteration number for algorithms imposing symmetry on the TEQ. The symmetrical algorithms, which are represented by solid lines, are compared with the low complexity algorithms LALAM, LSAM, and LSLAM which are represented



Figure 3.3: Achievable bitrate vs. iteration number using instantaneous method.



Bitrates vs. SNR after transmitting 75 symbols

Figure 3.4: Achievable bitrate vs. SNR using MA and instantaneous estimates.

as dotted lines. It is clear that the symmetrical based algorithms can achieve similar performance as non-symmetrical based algorithms with slight decrease in convergence rate while reducing the complexity by approximately 50%.

3.3.3 Equal-Taps TEQ Design.

In this Section, simulation is conducted for the adaptive algorithms derived in Section 3.2. Fig. 3.7 shows the obtained effective shortened channel for the proposed algorithms using the TEQ adaptive algorithm with ETC. The convergence in terms of bitrate vs. iteration number is shown in Fig. 3.8. As observed from this figure, the achievable bitrate of the ETC based algorithms outperform the non-ETC based algorithms with slight decrease in convergence rate. As a consequence, ETC based algorithms avoids the severe drop down in bitrate as in the other proposed and existing SAM and SLAM algorithms. The achievable bitrate versus SNR for LALAM, LSAM, and LSLAM algorithms, which are represented by dotted line, are compared with ETC based algorithms, represented by solid line is shown in Fig. 3.9. As observed from this figure, the ETC based algorithms outperform the symmetrical based algorithms. The ETC based algorithms outperform the symmetrical based algorithms. The ETC based algorithms outperform the symmetrical based algorithms. The ETC based algorithms outperform the symmetrical based algorithms. The ETC based algorithms outperform the symmetrical based algorithms. The ETC based algorithms outperform the symmetrical based algorithms the TEQ to shorten the channel for ADSL environment. The complexity for equal TEQ taps is reduced by L_w multiplications/dditions per update.

3.4 Conclusions

In this Chapter, a low complexity blind adaptive algorithm called LALAM for the TEQ is proposed to shorten the channel. This algorithm is an extension of ALAM algorithm proposed in Chapter 2. In this algorithm, instantaneous estimate is implemented to the proposed ALAM, SAM and SLAM to lower their complexities by a factor of N multiplication/addition per update. Using instantaneous estimate, the adaptive algorithm updates every sample which makes it suitable for fast time vary-



Figure 3.5: Effective (shortened) channel impulse response for symmetrical ALAM, SAM, and SLAM.



Figure 3.6: Achievable bitrate vs. iteration number using symmetrical TEQs.



Figure 3.7: TEQ taps obtained for equal-taps constrained ALAM, SAM, and SLAM.



Figure 3.8: Achievable bitrate vs. iteration number using Equal-taps constrained ALAM, SAM, and SLAM.



Figure 3.9: Achievable bitrate vs. SNR using symmetrical TEQs and equal-taps constrained.

ing channels. Symmetrical property of the TEQ is utilized to reduce the complexity of the algorithms to half of the non-symmetrical based algorithms. Finally, in order to minimize the complexity of all algorithms and achieve suitable minima, ETC is enforced to ALAM, SAM, and SLAM algorithms to obtain better performance. Simulation results show comparable results of the proposed algorithms with slight degradation in convergence time.

Chapter 4

FPGA Implementation of a Low Complexity Time Domain Equalizer for ADSL Systems

4.1 Introduction

DSP algorithms are implemented using application-specific integrated circuits (ASICs) to obtain high operating frequencies, reduce power consumption and size. However, with the development of modern field programmable gate array (FPGA) and tools that optimize area, speed, and power consumption. They have been used more frequently to provide an ideal solution for implementing high speed signal processing circuitry [58], [59], [60]. In addition, the availability of many resources that support DSP algorithms such as embedded multipliers, multiply accumulate units (MAC), and intellectual property (IP) cores make the task of implementing signal processing algorithms on FPGAs convenient.

The process of implementation involves many challenges in compiling the software code into pure hardware [61]. Some of the challenges are to avoid arbitrary division and multiplications to reduce the circuit size. In addition, DSP algorithms developed using software codes like Matlab are performed using floating-point numbers with infinite precision which is not always feasible in hardware implementation. Therefore, an extensive and iterative process is performed to convert the very precise floatingpoint numbers to less precise fixed-point numbers that can be easily implemented on the hardware. The conversion process is a very difficult step for the designer as the size of the circuit and the quantization accuracy is inversely proportional. This conversion is an iterative process, which is achieved using quantization in Matlab.

This chapter discusses the design and the FPGA implementation of a blind adaptive time-domain equalizer intellectual property (IP) core for ADSL receivers. The design can be configured to implement the adjacent lag auto-correlation minimization (ALAM) algorithm to shorten the channel. The calculation of the expectation in ALAM algorithm uses MA estimate [43] and the low complexity method introduced in Chapter 2. The feasibility of implementing the low complexity channel shortening algorithm is shown in this chapter and the complexity is compared with ALAM algorithm using MA estimate introduced in [16]. The IP core is implemented using the Xilinx Vertix II Pro XC2VP7-FF672-5 and ADSL receivers that operates at a maximum frequency of 27 MHz and 38 MHz for ALAM-MA and LALAM proposed in Chapter 2, respectively.

4.2 Standard FPGA Design Flow

FPGA implementation is performed in multiple stages starting from algorithm development to the programming of the FPGA device. The flow of the design stages are illustrated in Fig. 4.1. As illustrated in this figure, the first step is the development of the DSP algorithm which is performed and analyzed using high level programming languages such as C/C++ or Matlab. In high level programming, the computation is performed using floating-point numbers with infinite precisions. Since the hardware specification requires fixed-point numbers, the floating-point numbers are converted to fixed-point numbers. The floating-point simulations are then matched with the



Figure 4.1: FPGA design flow.

fixed-point simulation to obtain minimum quantization errors. The next step is to perform register transfer level (RTL) simulations using fixed-point numbers. The RTL is used to model a sequential circuit that describes the flow of data between registers. To confirm the functionality of RTL model, the simulations at RTL level are compared with the floating-point and fixed-point simulations. Furthermore, Xilinx synthesizer tool is used to translate Verilog code to device netlist. The netlist contains a complete description of the circuit that includes logic elements such as gates, flip flops, etc. FPGA vender place and route process consist of three steps: translate, map, and place-and-route. The translate process, which is also called physical synthesis, uses the netlist to assign the ports to the physical elements of the target device and specifies the timing constraints. The map process divides the whole circuit to sub blocks to fit into the targeted FPGA logic blocks. The place-and-route (PAR) process places the logic blocks in the target FPGA device and connects them according to the constraints. Gate level simulation is performed to validate the functionality of the circuit in the FPGA and is compared with the simulation of previous step. Finally a bit file is generated to program the FPGA.

The FPGA design flow is an iterative process that requires careful analysis at different steps of Fig. 4.1. To ease this process, several electronic design automation (EDA) tools are available such as the one offered by Xilinx (ISE) and Altera (Quartus II). The Xilinx tool is being used here to develop an IP core that can be integrated to within the ADSL receiver.

4.3 TEQ Design and Objectives

In this section, the implementation of the low complexity proposed ALAM algorithm in Chapter 2 is discussed and modified to be implementable on FPGA. The time domain equalizer using ALAM algorithm is configurable and can achieve tap coefficient adaptation to shorten the channel using two different methods: low complexity method proposed in Chapter 2 and the existing method of MA in [16]. The primary goals of this implementation are:

- 1. To design a generic IP core of adaptive equalizer using ALAM algorithm and implement this core on Xilinx Vertix II Pro XC2VP7-FF672-5 FPGA.
- 2. Use FPGA implementation to validate the low complexity method used in Chapter 2 and compare with the high complexity method proposed in [16].
- 3. Implement the ALAM algorithm using low complexity unit tap constraint (UTC) to void all zero solution [17].



Figure 4.2: Fixed-point representation.

4. To target the IP core for ADSL receivers to achieve data rates comparable to recent algorithms for channel shortening.

To achieve these goals, FPGA design flow in Fig. 4.1 is followed. The DSP algorithm developed in Chapter 2 is revisited where UTC is used instead of UNC to avoid all zero solution. When UTC is used in adaptive algorithms it achieves superior performance as shown in [17] and complexity is reduced to make it suitable for hardware implementation.

4.4 Fixed-point Analysis

After developing the DSP algorithm according to the FPGA design flow in Fig. 4.1, an extensive simulation analysis is performed to re-design the DSP algorithm using fixed-point numbers instead of floating-point numbers. Using simulation analysis, a process to identify a proper precision size in terms of word-length (WL) and fractional word-length (FWL) is performed. This process is performed delicately due to the tradeoffs involved with precision size; for example, increase in WL to achieve higher accuracy results in higher hardware complexity. The problem is resolved by optimizing the precision size while minimizing the cost of area in terms of logic elements (LEs) [62]. The fixed-point number representation in Fig. 4.2 shows that the most significant bit

(MSB) is the sign bit and all bits to the right of the radix point are the fractional word-length. Since the internal signal values are smaller than 1 in adaptive filtering, the FWL is kept to it is maximum length allowing only 1 bit for integer word-length (IWL) that is the sign bit in our implementation. Note that, since we are using UTC where the one tap of the filter coefficient is always 1, therefore it is required to have at least 2 bits IWL for the filter coefficients and the rest of the bits for FWL.

The WL of the I/O signals and the intermediate signals should be determined carefully to achieve the best performance of adaptive filtering with acceptable hardware complexity. The word length of the internal circuit, (i.e., including intermediate and inter-blocks signals), is critical in reducing the rounding effects and hardware complexity. Therefore, the WL of the multipliers output is chosen to be higher than the word length of the input. Rounding is applied to the input of the multiplier to limit the word length to a pre-specified value. This results in eliminating the least significant bits of the input to the multipliers. An overflow can occur which is treated by saturation instead of wrap [63]. The WL of the I/O signals is considered for stable operation and low quantization error as the output of the filter is fed back to the closed loop system. The impact of quantization error on the performance due to the transformation from floating-point to fixed-point is shown in Fig. 4.3 and Fig. 4.4. A simulation is conducted by setting the WL to 14-bit for the internal signals and examining three different sets of WL for the I/O signals as illustrated in Fig. 4.3. With WL of 14-bit for internal signals and 10-bit for the I/O signals, the performance increases gradually but the convergence is slow. This slow convergence is due to the shorter WL of the internal signals because increasing of I/O signals WL to 14-bit does not improve the convergence as shown in Fig. 4.3. Therefore, a larger WL for the internal signals is analyzed by simulating higher WLs at a fixed WL of 10-bit for the I/O signals. The results in Fig. 4.4 show that using 16-bit and 10-bit for the internal and I/O signals, respectively, achieves the optimum performance. Note that 10-bit instead of 14-bit WL for the I/O signals is being used to reduce the complex-



Figure 4.3: Impact of the I/O WL on the performance using 14-bit WL for internal signals.



Figure 4.4: Impact of the internal signals WL on the performance using 10-bit WL for I/O signals.

ity, as the performance with 14-bit WL for the I/O signals does not show significant improvement over 10-bit WL for the I/O signals.

In the rest of the design implementation stage, 10-bit WL of the I/O signals and 16-bits WL of the internal signals is used. To further validate the ALAM algorithm using fixed-point simulation, Fig 4.5, Fig 4.6, and Fig 4.7 illustrate the effectiveness of the WL of internal signals and I/O signals used for the FPGA implementation. The fixed-point simulation results are compared with the floating-point algorithm that shows almost identical results.

4.5 FPGA Implementation of ALAM

The direct form realization of the adaptive equalizer using ALAM algorithm in 4.1 is shown in Fig. 4.8 which depicts a structural view of such an algorithm implemented using FIR filter. As shown in this figure, the main components of the adaptive filter consist of L_w unit delay registers T, $L_w + 1$ filter weight updates, and calculation of the gradient of the expectation. The weight-update component updates the TEQ taps according to 4.1 whereas the unit delay registers are simply D Flip-Flops. The correlation between the input and output of the filter is calculated using instantaneous estimate and MA estimate [16]. Note that the MA estimate used in SAM and SLAM algorithms is implemented to compare the complexity of our low-complexity proposed algorithm.

4.5.1 Weight Update Block

Recalling the ALAM adaptive equation proposed in Chapter 2 given by

$$\mathbf{w}^{k+1} = \mathbf{w}^k - \mu \nabla_w E[y(n)y(n-1)]$$

=
$$\mathbf{w}^k - 2\mu E[y(n)\mathbf{r}_{n-1}]$$
 (4.1)



Figure 4.5: Shortened channel using LALAM Fixed-point simulations.



Figure 4.6: TEQ taps using LALAM Fixed-point simulations.



Figure 4.7: Achievable bit rate using LALAM Fixed-point simulations.

This adaptive algorithm is the core of the ALAM algorithm to update the TEQ taps to achieve channel shortening implemented in Fig. 4.9. The figure illustrates, the correlation term $E[y(n)\mathbf{r}_{n-1}]$ and the step size μ used to control the TEQ taps update. UNC is used to avoid all zero solution of the TEQ taps during update, however it adds high complexity as discussed in Chapter 2. Whereas UTC is used to minimize the complexity and increase the performance [17], that also avoids the all zero solution. UTC is used by constraining the middle single tap of the TEQ vector \mathbf{w} to 1 [64]. The correlation term in the adaptive Equation 4.1, is the gradient of the adjacent autocorrelation derived in Chapter 2.

4.5.2 Calculation of Expectation-Term Block

The expectation used in the adaptive Equation 4.1 is calculated in this implementation using the two methods discussed previously in Chapter 2. The MA estimate used in SAM [16] and SLAM [43] algorithms, allows the adaptive algorithm to be updated every block of size N. However, this results in slow convergence slow which



Figure 4.8: Realization of adaptive equalizer using ALAM algorithm.



Figure 4.9: Realization of update equation using ALAM algorithm.

is only suitable for slow varying channel. The MA method is given by [16]

$$\nabla \boldsymbol{J} = E[\boldsymbol{y}(n)\mathbf{r}_{n-1}] = \sum_{n=1}^{N} \frac{\boldsymbol{y}(n)\mathbf{r}_{n-1}}{N}, \qquad (4.2)$$

The realization of this estimator to calculate the gradient of the cost function is shown in Fig. 4.10 (a). In this figure, N samples are multiplied N times and the multiplication results are stored in delay registers. These stored values are summed and multiplied with 1/N to estimate the gradient used in the adaptive Equation 4.1. From this realization, it is clear that MA estimate requires a block of N-samples to update a single TEQ tap with the complexity of N number of multiplications and additions. To avoid such complexity and slow convergence, a simplified estimator is used in the proposed algorithm in Chapter 2. This simplification is achieved to



Figure 4.10: Realization of calculating the gradient of the cost function using (a) MA estimate (b) Instantaneous estimate.

avoid dual averaging that is performed using the stochastic gradient algorithm if MA estimate is used [52]. The simplified method to estimate the expectation used in the adaptive algorithm is defined by

$$\nabla \boldsymbol{J} = E[\boldsymbol{y}(n)\mathbf{r}_{n-1}] = \boldsymbol{y}(n)\mathbf{r}_{n-1}.$$
(4.3)

The realization of the simplified estimator is shown in Fig. 4.10 (b), to calculate the gradient of the cost function. The expectation is found using a single multiplier to update single TEQ tap. With this simplified approach two advantages are achieved over the MA estimator: (i) the complexity is low as it uses single multiplier for TEQ tap, and (ii) the approach updates the TEQ taps every sample which is suitable for fast varying channel. The complexity comparison using the implementation result reports are shown in the following Section in order to verify the low complexity of the proposed algorithm using instantaneous estimate.

4.6 Implementation Results

In this section, functionality of the IP core at the RTL level and gate level is verified to match the fixed-point simulations performed in Section 4.4. The low complexity


Figure 4.11: TEQ taps using LALAM RTL simulations.



Figure 4.12: Achievable bit rate using LALAM RTL simulations.



Figure 4.13: TEQ taps using ALAM-MA RTL simulations.



Figure 4.14: Achievable bit rate using ALAM-MA RTL simulations.

method used in the proposed ALAM method is verified to have lower complexity than the MA method. The IP core is implemented on Xilinx Verix II Pro XC2VP7-FF672-5.

In both RTL level and gate level simulation, the design is tested by creating a set of fixed-point input vectors to be used at the RTL level and gate level. Similarly, the simulated outputs are saved in a stimulus file in the fixed-point format. Using Matlab tool box, a script is written to convert the fixed-point numbers to floating point numbers in order to compare them with the expected results. The RTL level simulations for the IP core are illustrated in Fig. 4.11 and Fig. 4.12 for the LALAM algorithm that shows the TEQ taps and bit rate vs. the no. of iterations, respectively. Similarly, RTL level simulation in terms of TEQ taps and bit rate for the ALAM-MA algorithm are shown in Fig. 4.13 and Fig. 4.14. As illustrated in these figures, these results are almost identical to the floating-point Matlab simulations, and similar to the fixed-point simulations performed in the previous Section.

The last level of verification of the FPGA implementation as per the standard design flow discussed in Section 4.2 is to perform synthesis in order to create the device netlist. This is performed using Xilinx synthesizer tool along with place-and-route according to the target device specified in the synthesizer tool. The target device selected for this implementation is Vertix II Pro XC2VP7-FF672-5 which has a capacity of 4928 configurable logic blocks (CLBs), 44 embedded multipliers of 18×18 -bit multiplier blocks, and 396 bounded I/O blocks.

The MA method to calculate the expectation was previously used for SLAM and SAM algorithms in [43] and [16]. However, in ALAM algorithm the expectation is implemented using both, the instantaneous estimate and MA method. In order to investigate the complexity of these methods, an FPGA implementation of ALAM algorithm using both methods is conducted and measurements are taken. The physical synthesis report of this implementation is illustrated in Table 4.1 which provides comparative analysis between LALAM and ALAM-MA algorithms. It also provides

Vertix II Pro XC2VP7-FF672-5	ALAM-MA	LALAM		
Area (slices + multipliers)	$2097 \ slices + 33 \ multipliers$	$737 \ slices + 3 \ multipliers$		
Max. Clock Frequency (MHz)	27	38		
Relative Area	2.87	1.00		

Table 4.1: Virtex implementation (post-layout) results for clock frequency and area

the post-layout timing information generated using Xilinx design software tool. The obtained results for implementation using automation tools show that the system performance for the LALAM is higher than ALAM-MA by 11 MHz. The low complexity method used in LALAM utilizes only 3 multipliers and 737 slices whereas ALAM-MA uses 33 multipliers and 2097 slices. Therefore, ALAM-MA occupies 2.87 times more area than LALAM. This is due to the fact that the MA estimate requires an averaging over a window of size N for each update of the TEQ equalizer taps, which adds additional multipliers and slices. Hence, it requires larger area and runs slower than the proposed LALAM algorithm that uses instantaneous estimate.

To validate the correct operation of the design, gate-level simulations are performed where the fixed-point test vectors are fed into design under test (DUT) and the output results are saved in a file as a fixed-point format. This data is converted to floating-point format and plotted using Matlab along with the floating-point simulation for comparisons. These gate-level simulation results are illustrated for LALAM in Fig. 4.15 and Fig. 4.16 and for ALAM-MA in Fig. 4.17 and Fig. 4.18. The results are almost identical to their corresponding fixed-point and RTL results in Section 4.4 and this Section, respectively.

4.7 Conclusion

In this chapter, the IP core is developed for the low complexity, LALAM, algorithm for channel shortening equalizer to be implemented on an FPGA. Additionally, this



Figure 4.15: TEQ taps using LALAM gate-level simulations.



Figure 4.16: Achievable bit rate using LALAM gate-level simulations.



Figure 4.17: TEQ taps using ALAM-MA gate-level simulations.



Figure 4.18: Achievable bit rate using ALAM-MA gate-level simulations.

implementation was extended to include the implementation of the MA estimate for the ALAM algorithm referred as ALAM-MA. For implementation on FPGA, the ALAM algorithm is modified to use UTC instead of UNC to avoid all zero solution for the TEQ taps. This modification was performed to reduce the complexity of the ALAM algorithm while updating the TEQ. Since using UNC requires normalization at each update (i.e., square root and divide), which is very expensive in terms of hardware implementation as compared to UTC which constraints unity at each update. Simulations at all levels were shown including fixed-point, RTL, and Gate-level simulation to approximately match the floating-point simulation. A comparison in terms of the complexity is discussed and verified between the using instantaneous estimate in LALAM and MA estimate in ALAM-MA to calculate the expectation. The implemented IP core on Xilinx Verix II Pro XC2VP7-FF672-5 for ADSL receivers has shown that it can operate at a maximum frequency of 27 MHz and 38 MHz for ALAM-MA and LALAM as proposed in Chapter 2, respectively.

Chapter 5

QAM Equalization and Symbol Detection in OFDM Systems using Extreme Learning Machine

5.1 Introduction

Channel estimation and equalization techniques are employed to mitigate the effects of ISI and ICI in an OFDM receiver. Those techniques can be divided into three categories based on how they operate: blind, semi-blind, and trained. Blind techniques use the statistical information of the transmitted signals to estimate the channel response, hence they do not use training sequence. Using training sequence decreases overall throughput but reduces the receiver complexity. However, semi-blind techniques first use training sequence to estimate the channel response then reverts to blind adaptation [70]- [78]. The majority of those techniques employ algorithms based on linear equalization using least-square (LS), zero-forcing (ZF), or MMSE criterion [73], [77].

Nonlinear classifications, such as Bayesian decision theory, offer promising solution for equalization and symbol detection problems [3], [21]. Artificial neural networks (ANN), such as ANNs based on multilayer feed-forward neural networks and radial basis function (RBF), have been used for system identification and noise cancelation problems to recover transmitted data [22]. Learning based techniques that employ NNs can process a complex signal utilizing two independent real-valued multilayer perceptrons (MLP) or a split complex activation function [79]. MLP have been successfully used in channel estimation/equalization without symbol detection [80] - [82], in symbol detection [83], and in quadrature amplitude modulation (QAM) demodulation [84]. Complex-valued radial basis function (CRBF) network in [81] and complex-valued minimal resource allocation network (CMRAN) developed in [36] use stochastic gradient approach for parameter adjustments in channel equalization. CMRAN is a complex-valued version of the real valued minimal resource allocation network (MRAN), which is based on sequential learning with ability to prune and grow hidden neurons in order to achieve superior performance. CMRAN requires shorter training time and data for model learning than RBF [38]. On the other hand, a fully complex activation function was deployed in complex back propagation (CBP) in [80] as an extension to the traditional back propagation learning. Both CRBF and CMRAN can work with complex signals by deploying a split-complex activation function, which comprises of two real-valued activation functions, one for real and one for imaginary part of the input signal. Learning-based equalization schemes require manual tuning of the learning rate and epochs, and have limited success due to slow convergence to local minima. A single hidden layer feed forward neural network, called extreme learning machine (ELM), transforms learning paradigm into a simple linear solution [85]. A true complex-valued ELM (C-ELM), an extension of ELM, was proposed in [38] for channel equalization with complex input data. Because both CBP and C-ELM process complex values, they have higher computational complexities. A receiver structure that combines a decision feedback equalizer and a self organizing map (SOM) as symbol slicer was proposed in [84]. Later, a receiver structure that combines recurrent neural network (RNN) equalizer with SOM detector to estimate QAM symbols was proposed in [86]. Those structures require two separate systems, one for equalization and one for symbol detection, and converge slowly due to the

use of traditional neural networks that require manual tuning of control parameters, such as epochs and learning rate.

In this Chapter, a framework that uses real-valued ELM which solves the combined problem of equalization and symbol detection is proposed. This framework converges much faster than traditional NNs and jointly solves equalization and detection problems. In the joint solution, the framework does not need an additional QAM slicer circuit, which is required in C-ELM, CMRAN, CRBF, and CBP equalizers. In addition to this, the proposed framework employs a fully real-valued processing scheme where QAM constellation points are transformed to a real- valued vector of 2-tuple, which are labeled with integer values (i.e., class number). Using real-valued vectors eliminates complex valued processing and reduces computational complexity of the receiver. This leads faster convergence. Another advantage of the proposed framework is that it achieves higher accuracy by associating a 2-tuple pattern to one of the M possible classes, which corresponds to an individual QAM constellation point. The advantages of the proposed framework are demonstrated via simulation studies.

5.2 Preliminaries

5.2.1 OFDM System Model

OFDM system model, that is tested in this experiment, is illustrated in Fig. 5.1. In this model, data bits are mapped to one of the symbols of QAM and then N-point fast inverse Fourier transform (IFFT) is applied to produce orthogonal subcarriers. In order to simplify the simulations, the broadband carrier modulation scheme is omitted in the model. The output of IFFT block is transmitted in time-domain where N base-band symbols, x(n) are uncorrelated to each other and are obtained by the relation

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{j(2\pi kn/N)}, \quad n = 0, 1, 2, ..., N-1.$$
(5.1)



Figure 5.1: OFDM system model.

A guard interval called cyclic prefix (CP) of length v is inserted at the start of each OFDM symbol which is a copy of the last v samples of the OFDM symbols to mitigate inter-symbol interference (ISI). The resulted signal with guard interval $x_g(n)$ is transmitted over a frequency selective time varying fading channel. The signal, $x_g(n)$, which is white and wide-sense stationary (W.S.S), is transmitted through the linear time varying finite-impulse response (FIR) channel whose impulse response is h(n, p). The output of the channel (i.e., received signal), $y_g(n)$, is given by

$$y_g(n) = \sum_{l=0}^{p-1} h(n, p) x_g(l) + v(n), \qquad (5.2)$$

where n is the sample index and v(n) is a zero-mean additive white Gaussian noise (AWGN) sample with variance of σ_v^2 ; h(n, p) is the impulse response of the sampled time-varying channel, and p is the number of propagation paths. Each path has an amplitude of a complex Gaussian distribution and the power spectrum. The power spectrum of the channel is determined by Doppler frequency shift of f_D . The guard interval, at the receiver, is removed from $y_g(n)$ and DFT is applied on y(n), which is given by

$$Y(k) = \sum_{n=0}^{N-1} y(n) e^{-j(2\pi kn/N)}, \quad k = 0, 1, 2, ..., N-1.$$
 (5.3)

In a conventional OFDM system, the channel estimation is performed using pilot symbols. The channel transfer function after extraction of pilot symbols and estimation is denoted by H(k). The transmitted QAM symbols can be recovered as

$$\hat{X}(k) = \frac{Y(k)}{\hat{H}(k)},\tag{5.4}$$

where $\hat{H}(k)$ is an estimate of H(k). The signal $\hat{X}(k)$ is fed through QAM symbol slicer to detect the actual transmitted data which is transformed into binary sequence using a demapper.

5.2.2 Extreme Learning Machine

Feed forward neural networks (FNN) have been used to solve nonlinear problems in different applications because of their approximating ability in nonlinear mappings. The major bottlenecks of FNN are slow learning speeds and the possibility of converging to a local minima due to poor adjustments of input weights and biases using gradient descend approaches. An analytical learning called extreme learning machine, which is based on a fast single-hidden layer feed forward neural network (SLFNN), was proposed by Huang et al. in [85]. ELM can exactly learn S distinct observations by using majority of nonlinear activation functions and maximum of Shidden neurons. Fig. 5.2 illustrates ELM architecture. In ELM, the input weights



Figure 5.2: ELM architecture.

and hidden layer biases are generated randomly instead of being tuned as in the

traditional FNN. The learning speed of ELM is extremely fast and results in good generalized performance for activation functions f(.) that can infinitely differentiable. Thus, a nonlinear problem can be transformed into a linear problem where output weights are calculated analytically by performing a generalized inverse operation of hidden layer weight matrices. ELM outperforms traditional learning frameworks in terms of learning speed and improved generalization performance with a minimum training error. Such properties and enhanced performance of ELM can allow us to deploy in real-time applications. Given S arbitrary distinct samples of $(\mathbf{x}_i, \mathbf{d}_i)$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{ip}]^T \in \mathbf{R}^p$ and $\mathbf{d}_i = [d_{i1}, d_{i2}, \ldots, d_{im}]^T \in \mathbf{R}^m$, where both column vectors are of length p input neurons and m output neurons, respectively. ELM [85] with activation function of f(x) and L hidden neurons is mathematically modeled as

$$\sum_{i=1}^{L} \beta_i f(\mathbf{w}_i \cdot \mathbf{x}_l + b_i) = \mathbf{o}_l, \ l \in \{1, 2, \dots, S\},$$

$$(5.5)$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{ip}]^T$ represents the weight vectors connecting the input nodes to an *i*th hidden node and $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{im}]^T$ represents the connection between the *i*th hidden node and the output nodes. $\mathbf{w}_i \cdot \mathbf{x}_l$ represents the inner product of \mathbf{w}_i and \mathbf{x}_l , and b_i is the threshold for the *i*th hidden node. The ELM can approximate S arbitrary samples with zero error as

$$\sum_{l=1}^{L} \|\mathbf{o}_{l} - \mathbf{d}_{l}\| = 0,$$
(5.6)

and the output weights, β_i , are obtained using the relation

$$\sum_{i=1}^{L} \beta_i f(w_i \cdot x_l + b_i) = d_l, \ l \in \{1, 2, \dots, S\}.$$
(5.7)

Using the above model, the nonlinear training problem is transformed into a linear system and is formulated as $H\beta = D$ where

$$\mathbf{H} = \begin{bmatrix} f(\mathbf{w}_1.\mathbf{x}_1 + b_1) & \dots & f(\mathbf{w}_L.\mathbf{x}_1 + b_L) \\ \vdots & \dots & \vdots \\ f(\mathbf{w}_1.\mathbf{x}_S + b_1) & \dots & f(\mathbf{w}_L.\mathbf{x}_S + b_L) \end{bmatrix}_{S \times L},$$
(5.8)

$$\boldsymbol{\beta} = [\beta_1^T, \beta_2^T, \dots, \beta_L^T]_{L \times m}^T, \quad and \quad \mathbf{D} = [\mathbf{d}_1^T, \mathbf{d}_2^T, \dots, \mathbf{d}_S^T]_{S \times m}^T.$$
(5.9)

H is the hidden layer output matrix of ELM. For example, *i*th column of **H** is the output of *i*th hidden node with respect to inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_S$. When the activation function f is infinitely differentiable, the number of hidden nodes are such that $L \ll S$. Training of ELM [85] requires minimization of the cost function, J, which is given by

$$J = \sum_{l=1}^{S} (\sum_{i=1}^{L} \beta_i g(\mathbf{w}_i \mathbf{x}_l + b_i) - \mathbf{d}_l)^2.$$
 (5.10)

Unknown **H** is estimated using a gradient-descent based scheme to search the minimum of $||\mathbf{H}\beta - \mathbf{D}||$ using an adaptive algorithm stated such as

$$\mathbf{W}_{k} = \mathbf{W}_{k-1} - \mu \frac{\partial J(\mathbf{W})}{\partial \mathbf{W}},\tag{5.11}$$

where the weight vector W is a combination of \mathbf{w}_i , β_i , and bias parameters b_i . In Equitation (5.11), the learning rate μ significantly affects the learning speed and accuracy. A large value of μ causes divergence and instability, and a small value slows the speed of convergence. To overcome these limitations, Huang et al. [85] proposed minimum least-square (LS) solution for ELM. The solution is to randomly choose the input weights and hidden layer biases and to analytically determine the hidden layer output matrix **H**, instead of tuning the entire neural network parameters. The learning of ELM is accomplished by finding least-square solution of:

$$\|\mathbf{H}(\hat{\mathbf{w}}_1,\ldots,\hat{\mathbf{w}}_L,\hat{b}_1,\ldots,\hat{b}_L)\hat{\beta} - \mathbf{D}\| = \min_{\mathbf{w}_i,b_i,\beta} \|\mathbf{H}(\mathbf{w}_1,\ldots,\mathbf{w}_L,b_1,\ldots,b_L)\beta - \mathbf{D}\|.$$
(5.12)

H can be a non-square matrix for a number of hidden nodes $L \ll S$, the norm least- square solution of linear system, $\mathbf{H}\beta = \mathbf{D}$, which forces the analytic solution to be represented as $\hat{\beta} = \mathbf{H}^*\mathbf{D}$, where \mathbf{H}^* is the moore-penrose generalized inverse of a matrix **H** for non-square matrix where as the solution is straightforward for S = L. The smallest training error can be achieved by this special solution:

$$\|\mathbf{H}\hat{\boldsymbol{\beta}} - \mathbf{D}\| = \|\mathbf{H}\mathbf{H}^*\mathbf{D} - \mathbf{D}\| = \min_{\boldsymbol{\beta}} \|\mathbf{H}\boldsymbol{\beta} - \mathbf{D}\|.$$
(5.13)

It has been theoretically analyzed and experimentally demonstrated by Huang et al. [85] that ELM can obtain good generalization performance with extremely fast learning speed.

5.3 Proposed Algorithm

This section explains the proposed framework which jointly solves the problem of equalization and symbol detection using fully real-valued ELM. The block diagram of the proposed framework is shown in Fig. 5.3. OFDM signals are transmitted over a frequency selective fading channel and AWGN noise is added. The received OFDM signal is disturbed due to several factors such as multipath fading, Doppler frequency shift, and local oscillator frequency drift. The QAM signal is reconstructed using FFT, which re-generates M-QAM symbols transmitted using N subcarriers. Reconstructed QAM signals are fed into the proposed joint equalization and symbol detection module after splitting them into training and testing data. The training part of the data is used to learn a generalized network model. After analytically adjusting ELM parameters, the classifier processes the real data. The number of output neurons is analogous to the value of M representing QAM mode, whereas input layer consists of two neurons, one assigned to the real part, and the other for the imaginary part of the complex input signal.

Equalization and symbol detection problem is solved using real-valued ELM by establishing a mapping between 2-tuple real input vector to a single complex symbol that correspond to an integer. However, existing approaches perform equalization using regression through a single mapping between a pair of input and output values [38], which compromises accuracy and requires additional QAM slicer. Fig. 5.4 shows the process of 4-QAM constellation where the received QAM signal Y(k) is transformed to 2-tuple real data vector, which is used as an input into a fully realvalued ELM, $\mathbf{x}_i = [re(Y_k), Im(Y_k)]^T$, and the target or desired output is the level



Figure 5.3: Proposed block diagram of OFDM system for joint equalization and symbol detection using ELM.

that corresponds to one of the known QAM constellation points shown in Fig. 5.5. The equalization is performed analytically and the symbols are decided based on ELM Max-pool decision rule that determines the winner output neuron [85]. As shown in right side of Fig. 5.4, the index of a winner output neuron represents an integer value (also called 'level'); it is important to state that that each level corresponds to one QAM symbol of the equivalent 4-QAM constellation in Fig. 5.4. The relationship between output levels of the proposed framework and 4- and 16 -QAM are shown in Fig. 5.5. The number of levels is always equal to the number of classes or distinct symbols of the QAM being equalized. In addition, the proposed scheme can automatically identify the QAM mode/size based on training information, therefore, it builds up knowledge about the transmitted data for any QAM based receiver.

The CBP and C-ELM are based on complex inputs, weights and activation functions which cause additional computational complexity. In contrast, the proposed framework utilizes real- valued classifier, i.e., ELM [85] whose training is performed analytically using least-square solution. Using inputs and activation function belongs to real domain reduces computational cost of the proposed framework and allows us to use large number of nonlinear infinitely differentiable activation functions. The proposed algorithm can be described as follows:



Figure 5.4: Equalization and symbol detection using ELM for 4-QAM.

- 1. Using the training data, the QAM mode is identified and the QAM symbols are mapped to different levels according to Fig. 5.5.
- 2. Transformation of complex values to a 2-tuple vector of real values. This step is performed as follows: Given a set of complex training data $(\mathbf{Y}_i, \mathbf{X}_i)$ where \mathbf{Y}_i is the received QAM symbol (complex value) and \mathbf{X}_i is the expected QAM symbol (complex value). Note that *i* here corresponds to *k*th QAM symbol index. The corresponding input-output of the ELM is formed by $\mathbf{x}_i = [Re(Y_i), Im(Y_i)]^T$ and $\mathbf{d}_i = [d_{i1}, d_{i2}, \ldots, d_{im}]^T$, $i = 1, 2, 3, \ldots, S$, and *m* represents the number of classes (number of output neurons) formed according to quadrature distribution of complex plane (as shown in Fig. 5.5).
- 3. Randomly assign real values input weight w_i and bias b_i , i = 1, 2, ..., L.
- 4. Calculate the hidden layer matrix H.
- 5. Calculate the output weight matrix $\hat{\beta} = \mathbf{H}^* \mathbf{D}$
- 6. Using the estimated output weight $\hat{\beta}$, the equalization is performed and the correct class/level is chosen based on Max-pool decision rule of ELM.
- 7. The selected class/output neuron(which is one of the quadrant or subquadrant of Fig. 5.5) is mapped back to the QAM symbol (complex value).

In the proposed framework, the complex values are split for processing as real values in a similar fashion like CMRAN and CRBF. However, the proposed framework works complectly in the real domain as the classification problem is treated as mapping between a pair of real data to single class which requires only single activation function whereas CMRAN and CRBF require two activation functions [38]. The proposed scheme offers improved scalability through automated QAM mode selection, and use of variety of activation functions from real as well as complex domain. This adds the advantage of lowering the complexity due real valued processing, scalability in terms of automated identification of QAM mode, and use of most of activation function of our proposed method over the above mentioned techniques.

			,	h	#		-
			16	н	10	15	
	3		12	4	3	9	
		Ĩ.	5	1	2	8	7~
1 2		13	6	,	- 14		
4-QAM constell	ation labels	•	16- QA	M coaste	lation Ial	oels	-

Figure 5.5: 4 and 16 QAM quadrant distribution.

5.4 **Results and Discussion**

Wireless OFDM systems with time varying Raleigh fading channel and three different QAM modes are constructed in the experimental setup in order to test the performance of the proposed framework. Simulated OFDM system parameters and assumptions of the experiments are provided in Table 5.1. The time varying Raleigh fading channel simulator described in [87] is used as a channel model. It is assumed that the system does not provide any *a priori* knowledge of the channel being estimated. The presented results are averaged values of 10 runs of the same experiment with random selection of training data. The proposed scheme is trained and tested using OFDM signal of same SNR, which is analogous to a real-life scenario, where as in C-ELM [38]

Simulation parameters				
Modulation	4QAM, 16QAM, 64QAM			
Symbol Rate	250kSymbol/sec			
Number of Subcarriers	52			
IFFT and FFT points	64			
Guard interval	1/4 of OFDM symbol			
Signal-to-Noise Ratio (SNR)	1 to 15 dB			
Doppler Frequency	[200, 250, 300, 350, 400, 450, 500]			
OFDM packet Length (QAM symbols/packet)	[64, 100, 150, 200, 250, 325]			
Channel Model	Raleigh fading channel [87]			

Table 5.1: Simulation parameters

the system is trained at higher SNRs to improve the accuracy, and the trained system is tested with data transmitted at lower SNRs. Furthermore, all experiments are executed in Matlab environment on an Intel Core 2 Duo processor at 2.0 GHz clock speed and 3GB RAM. The proposed framework is compared with other learning-based equalization schemes, namely C-ELM, CRBF, CMRAN, k-nearest neighbor (k-NN) [88], back propagation (BP) neural network [82], and stochastic gradient boosting (SG-Boosting) [89]. In the following subsection, experiments and their results are explained in detail.

5.4.1 Optimal Parameter Selection

There are mainly three parameters that require consideration to effectively deploy the proposed framework. These parameters are: 1) activation functions, 2) amount of training data, and 3) the number of hidden neurons. The variations in normalization factor of an input signal do not severely degrade the performance of the proposed scheme, therefore, there analysis are not included in the experiments.

- Activation Functions: Activation functions play a pivotal role in correct clas-• sification and are mainly divided into real and complex domain based on their operability. We split the complex numbers into two real values that are considered the inputs to an ELM. This approach has two advantages, namely: 1) the ability to apply real-valued activation functions and 2) the ability to reduce computational complexity. Representation of complex signal in real domain allows us to exploit variety of activation functions in the proposed scheme which is not possible in the complex domain such as in C-ELM. It is important to state that the proposed framework presents equalization and symbol detection problem as a classification task. Traditionally, classification results are presented in terms of accuracy where a higher accuracy corresponds to a lower symbol error rate (SER). The maximum accuracy is represented as 1 which is equal to 0 SER, (i.e., accuracy = 1 - SER). For 4-QAM, an accuracy analysis is presented using 8 different activation functions namely sine, sigmoid, hard limit, tribas, radial basis, asinh, and atanh. Fig. 5.6 provides results for accuracy vs. different activation functions. These activation functions are operable on real and complex-valued signals as well. In this experiment, the number of hidden neurons, amount of training data, and normalization factor are kept constant for all activation functions in order to compare them precisely. The accuracy comparison for various activation functions using the proposed scheme and C-ELM are presented in Fig. 5.6. From the figure, the proposed scheme using given activation functions for real-valued inputs outperforms C-ELM. Amongst the given activation functions, the performance of Hardlim(.) was the worst with accuracy around 90% whereas sig(.) generated the best accuracy followed by sin(.) and asinh(.).
- Amount of training data: In another set of trials, the performance of the proposed framework is tested in terms of correct classification, accuracy, using data



Figure 5.6: Performance analysis for 4-QAM for different activation functions using the proposed scheme and C-ELM.



Figure 5.7: Performance analysis for 4-QAM for varying percentage of training data using proposed framework.

at different SNRs (dB). A learner is rated based on its ability to learn a model using minimum training time and amount of data. The data is divided into training and testing parts. For varying SNRs, the percentage of training data is increased at equal intervals between 1% and 19%. The correct classification of the proposed scheme for varying size of training data is presented in Fig. 5.7, it is noticeable that higher accuracy is obtained for data with rising SNRs. Except for training data of size 1%, the proposed framework generates accuracy starting from 99.8% and more for increasing SNRs. High accuracy is noticeable for training data of size 10% and more to learn the channel being investigated. As observed from Fig. 5.7, there is no significant improvement for exploiting higher percentage of information during model learning. Therefore, in all of experiments 10% of the data are utilized for training and rest of the data for testing purpose.



Figure 5.8: Performance analysis for varying number of hidden neurons of ELM in the proposed framework using 4-QAM.



Figure 5.9: Performance analysis for varying number of hidden neurons of ELM in the proposed framework using 16-QAM.

• Number of hidden neurons: In learning-based frameworks, nonlinearity of an equalization scheme is mainly dependent on the number of hidden layers and quantity of neurons in each layer. Using only one hidden layer of neurons in the proposed framework results in an efficient way of processing QAM signals to attain higher accuracy. Efficiency of the proposed framework is attributed to analytic training in real-domain where a single complex number is represented by a 2-tuple vector for improved separability and correct classification. Training in an analytic fashion has a limitation of growing computational complexities with increasing number of neurons during computation of Moore-Penrose generalized inverse of a matrix. For that reason, the performance analysis of the framework tested with changing number of neurons in the hidden layer of ELM using data of two different modes of QAM, namely 4-QAM and 16-QAM. For both modes, data is generated using four different SNRs ranging from 2 dB to

8 dB. Generally, equalization and symbol detection becomes more complicated for higher mode QAM where the QAM constellation points become closer to each others. Therefore, an algorithm that performs well with high order QAM is needed, and the proposed framework can effectively equalize and detect the symbols, which is considered as an enhancement over the existing schemes. ELM with only one hidden layer of neurons restricts the options of improving the equalization capacity of the classifier by only changing the number of hidden neurons. It is not recommended to increase the number of neurons since large number of hidden neurons contributes to a higher computational complexity and increase in approximation errors. Additionally, such an increase in neurons does not guarantee proportional improvement in accuracy. For example, Fig. 5.8 and Fig. 5.9 presents performance analysis for changing number of hidden neurons using data acquired for two different modes. The number of hidden neurons are varied from 6 - 20 and 10 - 70 for 4-QAM and 16-QAM, respectively. It is observed that the performance of the proposed framework is gradually improving with increasing SNR, furthermore, higher number of hidden neurons are required for 16-QAM channel equalization compared to 4-QAM channel equalization due to the increase in QAM constellation points. The proposed framework results in smaller variations and stable behavior for data with higher SNRs and for $L \geq 2M$ where L and M represent the number of hidden neurons and the mode of investigated QAM, respectively. Besides, a rippling behavior of the graph lines for various SNRs and modes of QAM is also spotted for small number of hidden neurons. A similar trend in accuracy is found for 64-QAM data; based on evidence from the experiments, it is recommended for hidden layer of an ELM to satisfy $L \ge 2 * M$ criterion for a minimum number of neurons to achieve improved equalization and symbol detection.

5.4.2 Varying Packet Length and Doppler Frequency

For time varying channels, equalization schemes are heavily dependent on the change in Doppler frequency and OFDM packet length. SER is directly proportional to the length of its input packets and Doppler frequency. For better understanding, a set of experiments in time-varying channel for increasing Doppler frequency and packet length are performed for 4-QAM. In these experiments, SNR is fixed to 4dB. The channel with Doppler frequencies ranging from 64Hz to 325Hz are used to test the performance of SER in seven different schemes and the results of these tests are shown in Fig. 5.10. In this figure, the SER using the proposed framework at 100Hz Doppler frequency is approximately 100 times better than kNN, SG-Boosting, and CRBF. The proposed framework outperforms all other equalizers by generating minimum SER, and 5 times on average better than C- ELEM, CMRAN, and BP. Fig. 5.11 illustrates the SER versus varying packet length which ranges from 200 OFDM symbols to 500 OFDM symbols. The figure shows that at packet length of 300 OFDM symbols, the SER using kNN, SG-boosting, and CRBF is worse than the proposed by approximately 100 times where as using CMRAN, CELM and BP worse than the proposed by approximately 5 times on average. Overall, the proposed framework outperforms all other equalizers in achieving minimum SER under varying Doppler frequencies and packet lengths.

5.4.3 Performance Analysis of 4-, 16-, and 64-QAM

The performance comparison of the proposed framework against existing equalization schemes is presented in Figs. 5.12-5.14 for M-QAM signals, namely 4-, 16-, and 64-QAM. For all three experiments, QAM with varying SNR ranging from 1 dB to 15 dB is used where each experiment utilizes equal amount of training samples. SER for 4-QAM is presented in Fig. 5.12 where seven equalization schemes are divided into three noticeable groups based on SER performances. The best group among the



Figure 5.10: SER vs. Doppler frequency using packet length of 200 symbols 4-QAM.



Figure 5.11: SER vs packet length using 500 Hz Doppler frequency for 4-QAM.



Figure 5.12: Error probability of various scheme for 4-QAM.

three is the proposed framework and BP. The group consisted of SG-Boosting, kNN, and CRBF did not perform well. SER performances of C-ELM and CMRAN lie in the middle where a slow decrease in SER is observed for increasing SNR.

The performance comparison for 16-QAM is presented in Fig. 5.13, it is shown that all the schemes show poor performance except the proposed framework and C-ELM. For the proposed framework, a steep and consistent drop of SER ascertains superior performance whereas C-ELM is the second best equalizer with widening gap in performance against the proposed framework. No improvement in SER are observed for the rest of the schemes which perform consistently high SERs as observed in Fig. 5.13 and Fig. 5.14.

For 64-QAM, Fig. 5.14 presents results that establishes superior realization of the proposed framework with consistently improved performance. The performance of C-ELM is significantly degraded in current set of experiments with a very small improvement for increasing SNR. As observed from Fig. 5.13 and Fig. 5.14 increase



Figure 5.13: Error probability of various scheme for 16-QAM.



Figure 5.14: Error probability of various scheme for 64-QAM.

in the SER for the rest of the equalizers is attributed towards rising of the QAM constellation points as in 16- and 64-QAM. Clearly, the lowest SER for the proposed framework is consistent throughout the set of experiments on different settings.

5.4.4 Computational Comparison

The architecture of the proposed framework is very much alike to C-ELM and other learning- based equalization schemes. However, presentation of equalization and symbol detection problem in a classification domain removes inherent limitation to use a decision slicer to identify a symbol. On the other hand, C-ELM offers regression based equalization in complex domain where an additional decision slicer is required for symbol detection; whereas the proposed framework operates in a real domain by splitting complex signal into two real-valued inputs. Moreover, analytic training and equalization in complex domain is computationally intensive compare to the real domain operations. In order to check the computational cost of various methods, a set of experiments are performed using data of 16-QAM and 64-QAM. The higher mode QAM are being used to test the performance of various methods for nonlinear channel equalization in terms of computational time (seconds) during training and testing phases. The computational time for channel estimation and testing is presented in Table 5.2. Deploying seven different techniques named proposed framework using ELM, kNN, BP, SG-Boosting, C-ELM, CRBF, and CMRAN. Naturally, the computational time for training is much higher than the testing phase due to optimization constraints to learn a model or channel under investigation. Note that the training time for kNN scheme is equal to zero since the centers of sub-quadrants as the means of clusters are provided, to associate an incoming input to a symbol using Euclidean distance. The size of training and testing data is set to equal for all schemes for a fair comparison. The computational time for learning phase in 16-QAM and 64-QAM are presented in Table 5.2, where the training time for SG-Boosting and CMRAN are the highest amongst all techniques whereas the proposed framework

16-QAM 64-QAM				
Algorithms	Training time (s)	Testing time (s)	Training time (s)	Testing time (s)
Proposed using ELM	0.0156	0.2031	0.0625	0.2656
kNN	N/A	0.4062	N/A	0.4375
BP	2.530	0.7308	2.112	0.6702
SG-Boost	241.6	0.65625	419.0	0.8125
C-ELM	0.0937	0.6875	0.4218	1.140
CRBF	0.7656	2.625	0.7656	2.578
CMRAN	44.60	7.125	40.73	7.5

Table 5.2: Complexity time comparison of training and testing for different algorithms

has the minimal training time and then follows C-ELM, CRBF and BP. In addition, schemes like BP require a careful selection of number of epochs and layers, and learning parameters since these are implicit characteristics of gradient-descent methods. The training using proposed framework runs at least 6 times faster than C-ELM and the rest of the algorithms. Moreover, the computational time for various methods during testing phase for equalization and symbol detection are shown in Table. 5.2 for both 16-QAM and 64-QAM. The proposed framework consumes minimum resources, whereas kNN is the second inline of the most prudent methods in the trials. On other hand, CRBF and CMRAN schemes consumes the highest amount of CPU time for equalization and symbol identification. Based on computational analysis, it is affirmed that the proposed framework requires minimum computational time amongst different learning-based equalization schemes

5.5 Conclusions

The problem of equalization and symbol detection is presented as an optimum classification task. The use of a classification framework removes inherent limitation in symbol detection schemes to use an additional decision slicer followed by equalization step. In channel estimation, analytic learning approach finishes training at faster than traditional gradient-descent schemes. The proposed framework performs joint equalization and symbol detection in real- domain by transforming a complex signal into a single 2--tuple real-valued vector. Such transformation offers equalization in real domain with minimum computational load and high accuracy. The proposed work shows significant improvement in terms of SERs, and has lower complexity than existing algorithms. One of the benefits of operating in real domain is to use large number of nonlinear and infinitely differentiable activation functions.

Chapter 6

Conclusion and Future Work

6.1 Concluding Remarks

This dissertation investigates efficient equalization algorithms for multicarrier modulation systems. A low complexity blind adaptive algorithm called ALAM for the TEQ has beep proposed to shorten the channel. The proposed algorithm has employed a new criterion that minimizes of the adjacent lag autocorrelation. The reduction in complexity has been attained by simplified cost function. This new cost function has been verified with SAM [16] algorithm using MA method in the calculation of the expectation. A decaying step size is used in ALAM, SAM, and SLAM algorithms to control the convergence and to achieve steady bitrate performance. Since the proposed ALAM algorithm has used MA estimate, the adaptive algorithm is updated every block which makes the algorithm not suitable for fast time varying channels. A low complexity estimator that updates every sample has been used. This method is called instantaneous estimate and has been implemented in the proposed ALAM, SAM and SLAM to lower the complexity by a factor of N multiplication/addition per update. Using instantaneous estimate, the adaptive algorithm updates every sample and hence the algorithm is suitable for fast time varying channels. Symmetrical property of the TEQ has been utilized to further reduce the complexity of

the algorithms by 50% with respect to the non-symmetrical algorithms. The bitrate drops down when it reaches its maximum. This problem of drop down in bitrate is solved by enforcing ETC on the adaptive algorithm while updating. Only single tap is updated, if ETC is used, and the rest of the taps are forced to be equal to the adapting tap. This resulted in higher bitrates while the complexity reduced by the length of filter length, L_w , with respect to non-ETC based algorithm. An IP core is developed to implemented LALAM on Xilinx Vertix II Pro FPGA in order to show if the proposed LALAM algorithm is feasible for hardware implementation and is lower complexity than the existing algorithms. Moreover, the implementation has been extended to include the implementation of the MA estimate for the ALAM algorithm, (i.e., ALAM-MA). Before direct implementation on FPGA, the ALAM algorithm was slightly modified to use UTC instead of UNC to avoid all zero solution for the TEQ taps. UTC is replaced with UNC which constraints unity at each update and reduces the complexity of the ALAM algorithm while updating the TEQ. Simulations at all levels were shown including fixed-point, RTL, and Gate-level to match the floatingpoint simulations. A comparison in terms of the complexity has been discussed and verified between the use of instantaneous estimate in LALAM and MA estimate in ALAM-MA. The implemented IP core on Xilinx Vertix II Pro XC2VP7-FF672-5 for ADSL receivers can operate at a maximum frequency of 27 MHz and 38 MHz for ALAM-MA and LALAM algorithms, respectively. Once the channel is shortened (or is already short), FEQ equalizer is used to recover the QAM signal which could be distorted due to several reasons such as multipath fading, Doppler frequency shift, or local oscillator frequency drift. A joint problem of equalization and symbol detection in OFDM systems has been presented as an optimum classification task to recover the QAM signals. The use of a classification framework removes inherent limitation in symbol detection schemes to use an additional decision slicer followed by equalization step. In channel estimation, the proposed analytic learning approach using ELM performs training at faster rate than traditional gradient-descent schemes. The proposed framework performs joint equalization and symbol detection in real-domain by transforming a complex signal into a single 2-tuple real-valued vector. Such transformation offers equalization in real domain with minimum computational load and high accuracy. The proposed framework using ELM has shown significant improvement in terms of SERs while maintaining lower complexity compare to existing algorithms. Another advantage of operating in real domain is to use large number of nonlinear and infinitely differentiable activation functions.

6.2 Future Work

Although this dissertation have provided several algorithms to solve the problem of equalization in multicarrier modulation systems, there are still some outstanding research problems that can be considered in the future which are:

- Investigation of ETC-TEQ adaptive algorithm to achieve faster convergence rate and further improvement in bitrates to reach MSSNR performance.
- Investigation of computational efficient methods to implement efficient *multipliers* involved in adaptive algorithms on an FPGA.
- The implementation of proposed framework using ELM require future investigation, such as implementation issues involving matrix inverse, and channel equalization issues for fast time varying channels. Also, pilot symbols that exist in the OFDM systems can be exploited to further enhance the performance of the proposed framework.

References

- B. R. Saltzberg, "Performance of an efficient parallel data transmission system," IEEE Trans. Commun. Technol., vol. 15, no. 6, pp. 805-811, Dec. 1967.
- [2] R. W. Chang, "Synthesis of band-limited orthogonal signals for multichannel data transmission," Bell Syst.Technol. Journal, vol. 45, pp. 1775-1796, Dec. 1966.
- [3] J. G. Proakis, Digital communications, 3rd ed. New York, McGraw-Hill, 1995.
- [4] J. A. C. Bingham, "Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come," IEEE Communications Magazine, vol. 28, pp. 5-14, May 1990.
- [5] J. W. Cooley and J.W. Tukey, "An algorithm for the machine calculation of complex Fourier Series, Mathematics of Computation," vol. 19, no. 90, pp. 297-301, 1965.
- [6] W. Henkel and G. Taubock, "The cyclic prefix of OFDM/DMT an analysis," Proc. IEEE Int. Zurich Seminar on Broadband Commun. Access-Transmission-Networking, pp. 22.1-22.3, Feb. 2002.
- [7] J. S. Chow and J. M. Cioffi, "A cost-effective maximum likelihood receiver for multicarrier systems," IEEE Trans. Signal Commun., vol. 2, pp. 948-952, 1992.
- [8] R. Prasad, OFDM for Wireless Communications Systems, Artech House, 2004.

- [9] D. D. Falconer and F. R. Magee, "Adaptive Channel Memory Truncation for Maximum Likelihood Sequence Estimation," Bell Sys. Technol. Journal, pp. 1541-1562, Nov. 1973.
- [10] J. S. Chow, J. M. Cioffi, and J. A. C. Bingham, "Equalizer Training Algorithms for Multicarrier Modulation Systems," Proc. IEEE Int. Conf. on Commun., (Geneva, Switzerland), pp. 761-765, May 1993.
- [11] N. Al-Dhahir and J. M. Cioffi, "Efficiently Computed Reduced-Parameter Input-Aided MMSE Equalizers for ML Detection: A Unified Approach," IEEE Trans. on Info. Theory, vol. 42, pp. 903-915, May 1996.
- [12] P. Melsa, R. Younce, and C. Robs, "Impulse Response Shortening for Discrete Multitone Transceivers," IEEE Trans. Signal Commun., vol. 44, no. 12, pp. 1662-1672, 1996.
- [13] R.K. Martin, K. Vanbleu, M. Ding, G. Ysebaert, M. Milosevic, B. Evans, M. Moonen, and C.R. Johnson, Jr, "Unification and evaluation of equalization structures and design algorithms for discrete multitone modulation systems," IEEE Trans. Signal Process., vol. 53, no. 10, pp. 3880-3894, 2005.
- [14] R.K. Martin, J. Balakrishnan, W.A. Sethares, and C.R Johnson, Jr, "A blind adaptive TEQ for multicarrier systems,"" IEEE Signal Process. Letters., vol. 9, no. 11, pp. 341-343, 2002.
- [15] R. K. Martin, J. Balakrishnan, W. A. Sethares, and C. R. Johnson, Jr., "Blind, Adaptive Channel Shortening for Multicarrier Systems," Proc. IEEE Asilomar Conf. on Signals, Systems, and Computers, Pacific Grove, CA, pp. 372-376, Nov. 2002.
- [16] J. Balakrishnan, R.K. Martin, and C.R. Johnson, Jr, "Blind, Adaptive
Channel Shortening by Sum-squared Auto-correlation Minimization (SAM)," IEEE Trans. Signal Process., vol. 51, no. 12, pp. 3086-3093, 2003.

- [17] R. K. Martin "Fast-Converging Blind Adaptive Channel-Shortening and frequency-Domain Equalization," IEEE Trans. on Signal Process., vol. 55, no. 1, pp. 102-110, Jan. 2007.
- [18] C. R. Johnson, Jr., P. Schniter, T. J. Endres, J. D. Behm, D. R. Brown, and R. A. Casas, "Blind equalization using the constant modulus criterion: A review," Proc. IEEE, vol. 86, no.10, pp. 1927-1950, Oct. 1998.
- [19] J. Yang, J.-J. Werner, and G. A. Dumont, "The multimodulus blind equalization algorithm," Proc. Int. Conf. on Digital Signal Processing, pp. 127-130, 1997.
- [20] J. Yang, J.-J. Werner, and G. A. Dumont, The multimodulus blind equalization and its generalized algorithms, IEEE Journal on selected areas in commun., vol. 20, no. 5, pp. 997-1015, 2002.
- [21] S. Haykin, Digital communication, New York, Wiley, 1998.
- [22] G. Kechriotis, E. Zervas, and E. S. Manolakos, "Using Recurrent Neural Networks for Adaptive Communication Channel Equalization," IEEE Trans. Neural Network, vol. 5, no. 2, pp. 267-277, 1994.
- [23] C. Stergiou and D. Siganos, NEURAL NETWORKS, [Online] Available: http://www.doc.ic.ac.uk/ nd/surprise_96/journal/vol4/cs11/report.html.
- [24] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," D.E. Rumelhart and J.L. McClelland (Eds.), Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1, Cambridge, MA: MIT Press.

- [25] B. Widrow and M.A. Lehr, "Backpropagation and its Applications," Proc. of the INNS Summer Workshop on Neural Network Computing for the Electric Power Industry, Stanford, pp. 21-29, August 1992.
- [26] B. Widrow and S. D. Stearns, Adaptive Signal Processing, Prentice Hall, 1985.
- [27] T. Kohonen, Self-organization and Associative Memory, Springer-Verilag, Berlin, 1984.
- [28] S. Bang, S. H. Sheu, and J. Bing, "Neural network for detection of signals in communication," IEEE Trans. Circuits Syst. I: Fundamental Theory and Applications, vol. 43, no. 8, pp. 644-655, Aug. 1996.
- [29] S. K. Nair and J. Moon, "A theoretical study of linear and nonlinear equalization in the nonlinear magnetic storage channels," IEEE Trans. Neural Network, vol. 8, no. 5, pp. 1106-1118, Sep. 1997.
- [30] C. P. Lim and R. F. Harrison, "Online pattern classification with multiple neural network systems: An experimental study," IEEE Trans. Syst., Man, and Cybern. Part C: Applic. and Reviews., vol. 33, no. 2, pp. 235-247, 2003.
- [31] B. Widrow and M. A. Lehr, "30 years of adaptive neural networks: Perceptron, madaline and backpropagation," Proc. IEEE, vol. 78, no. 9, pp. 1415-1442, Sep. 1990.
- [32] N. Benvenuto and F. Piazza, "On the complex backpropagation algorithm," IEEE Trans. Signal Process., vol. 40, no. 4, pp. 967-969, Apr. 1992.
- [33] S. Chen, B. Mulgrew, and S. McLaughlin, "Adaptive Bayesian decision feedback equalizer based on a radial basis function network," Proc. IEEE Int. Conf. Commun., Chicago, IL, 1992, pp. 343.3.1-343.3.5.

- [34] S. Chen, B. Mulgrew, and S. McLaughlin, "Adaptive Bayesian equalizer with decision feedback," IEEE Trans. Signal Process.," vol. 41, no. 9, pp. 2918-2927, Sep. 1993.
- [35] S. Chen, B. Mulgrew, and P. M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," IEEE Trans. Neural Network., vol. 4, no. 4, pp. 570-590, Jul. 1993.
- [36] D. Jianping, N. Sundararajan, and P. Saratchandran, "Communication channel equalization using complex-valued minimal radial basis function neural networks," IEEE T. Neural Network, vol. 13, no. 3, pp. 687-696, 2002.
- [37] L. Yingwei, N. Sundararajan, and P. Saratchandran, "Performance evaluation of a sequential minimal radial basis function (RBF) neural network learning algorithm," IEEE Trans. Neural Network., vol. 9, no. 2, pp. 308-318, Mar. 1998.
- [38] M. Li, G. Huang, P. Saratchandran, and N. Sundararajan, "Fully complex extreme learning machine," Neurocomputing, vol. 68, pp. 306-314, 2005.
- [39] G.-B. Huang, Q.-Y. Zhu, K. Z. Mao, C.-K. Siew, P. Saratchandran, and N. Sundararajan, "Can threshold networks be trained directly?," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 157, no. 9, pp. 644-655, Mar. 2006.
- [40] G.-B. Huang, M.-B. Li, L. Chen, and C.-K. Siew, "Incremental extreme learning machine with fully complex hidden nodes," Neurocomputing, vol. 71, pp. 576-583, 2008.
- [41] R.K. Martin, and C.R. Johnson, Jr, "Adaptive equalization transitioning from single carrier to multicarrier systems," IEEE Signal Process. Mag., vol. 22, no. 6, pp. 108-122, 2005.
- [42] R. Nawaz and J.A. Chambers, "Robust blind channel shortening in impulsive noise environments," European Signal Process. Conf., Vienna, Austria, 2004.

- [43] R. Nawaz and J.A. Chambers, "Blind adaptive channel shortening by single lag autocorrelation minimization," Electronics Letters, vol. 40, no. 25, pp. 1609-1611, 2004.
- [44] R. Nawaz and J.A. Chambers, "A Novel Single Lag Auto-correlation Minimization (SLAM) Algorithm For Blind Adaptive Channel Shortening," IEEE Int. Conf. on Acoustics, Speech, and Signal Process., vol. 3, pp. 885-888, 2005.
- [45] T. Miyajima and Z. Ding, "Second-order statistical approaches to channel shortening in multicarrier systems," IEEE Trans. Signal Process., vol. 52, no. 11, pp. 3253-3264, 2004.
- [46] I.G. Muhammad, E. Abdel-Raheem, and K. Tepe, "Efficient Blind Adaptive Channel Shortening Algorithm for Multicarrier Modulation Systems," IEEE Int. Symp. on Signal Process. and Info. Technol., pp. 293-296, 2009.
- [47] G. Arslan, M. Ding, B. Lu, Z. Shen, and B. L. Evans. Equalizer Design Toolbox. Univ. Texas, Austin, TX. [Online] Available: http://users.ece.utexas.edu/ bevans/projects/adsl/ dmtteq/dmtteq.html
- [48] R.K. Martin Matlab Codes for Papers by R. K. Martin. [Online] Available:http://bard.ece.cornell.edu/matlab/martin/index.html
- [49] S. Haykin, Adaptive Filter Theory. Englewood Cliffs, NJ: Prentice-Hall, 1991.
- [50] R. K. Martin, M. Ding, B. L. Evans, and C. R. Johnson, Jr., "Efficient Channel Shortening Equalizer Design," EURASIP Journal on Applied Signal Process., vol. 2003, no. 13, pp. 1279-1290, 2003.
- [51] J. Beek, M. Sandell, and P. Borjesson, "ML estimation of time and frequency offset in OFDM systems," IEEE Trans. Signal Process., vol. 45, no. 7, pp. 1800– 1805, 1997.

- [52] O. Macchi, Adaptive Processing: The least mean squares approach with applications in transmission. Chichester, England: J. Wiley, 1995.
- [53] P. A. Regalia, "An Adaptive Unit Norm Filter with Applications to Signal Analysis and Karhunen-Loeve Transformations," IEEE Trans. on Circuits and Systems., vol. 37, no. 5, pp. 646-649, 1990.
- [54] C. Ribeiro, V. Silva, and P. S. R. Diniz., "Linear Phase Impulse Response Shortening for xDSL DMT Modems," IEEE Int. Telecommun. Symp., pp. 368-371, 2002.
- [55] R.K. Martin, M. Ding, B.L Evans, and C.R. Johnson, Jr., "Infinite length results and design implications for time-domain equalizers," IEEE Trans. Signal Process., vol. 52, no. 1, pp. 297-301, 2004.
- [56] C. Toker, and G. Altin, "Blind, Adaptive Channel Shortening Equalizer Algorithm Which Can Provide Shortened Channel State Information (BACS-SI)," IEEE Trans. Signal Process., vol. 57, no. 4, pp. 1483-1493, 2009.
- [57] V. Tarokh, New Directions in Wireless Communications Research, New York, USA: Springer, 2009.
- [58] J. Villasenor and B. Hutchings. "The flexibility of configurable computing," IEEE Signal Process. Magazine, pp. 67-84, Sep. 1998.
- [59] R. Tessier and W. Burleson, "Reconfigurable computing for digital signal processing: a survey," Journal of VLSI Signal Process., vol. 28, pp. 7-27, 2001.
- [60] C. Dick, "Re-discovering signal processing: a configurable logic based approach," Proc. Of Asilomar, pp. 1370-1374, Pacific Grove, CA, November 2003.
- [61] K.T. Gribbon, D.G. Bailey, and A. Bainbridge-Smith. "Development issues in using FPGAs for image processing", Proc. of Image and Vision Computing, pp. 217-222, Hamilton, New Zealand, December 2007.

- [62] W. C. Chew and B. Farhang-Boroujeny "FPGA Implementation of acoustic echo canceling," Proc. of the IEEE Region 10 Conference, Vol. 1, pp. 263-266, TEN-CON 1999.
- [63] G. Constantinides, P. Cheung, and W. Luk, "Synthesis of Saturation Arithmetic Architectures," ACM Trans. on Design Automation of Electronic Systems, vol. 8, pp. 334-354, 2003.
- [64] R.K Martin, "Unit Tap Constrained Adaptive Channel Shortening Equalization," Conf. Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers, pp. 777-781, Oct. 28 2005- Nov. 1 2005.
- [65] Y. Zhao and S. Haggman, "Intercarrier interference self cancelation scheme for OFDM mobile communication systems," IEEE Trans. Commun., vol. 49, no. 7, pp. 1185-1191, 2001.
- [66] L. Piazzo and P. Mandarini, "Analysis of phase noise effects in OFDM modems," IEEE Trans. Commun., vol. 50, no. 10, pp. 1696-1705, 2002.
- [67] J. Shentu, K. Panta, and J. Armstrong, "Effects of phase noise on performance of OFDM systems using an IC1 cancelation scheme," IEEE Trans. Broadcasting, vol. 49, no. 2, pp. 221-224, 2003.
- [68] K. Fazel and S. Kaiser, Multi-Carrier and Spread Spectrum Systems, John Wiley and Sons, 2003.
- [69] A. R. S. Bahai, B. R. Saltzberg, and M. Ergen, Multi Carrier Digital Communications: Theory and Applications of OFDM, Springer, 2004.
- [70] E. Panayirci, H. Senil, and H. V. Poor, "Joint Channel Estimation, Equalization, and Data Detection for OFDM Systems in the Presence of Very High Mobility," IEEE Trans. Signal Process. vol. 58, no. 8, pp. 4225-4238, 2010.

- [71] M. Muck, M. de Courville, and P. Duhamel, "A Pseudorandom Postfix OFDM ModulatorSemi-Blind Channel Estimation and Equalization," IEEE Trans. Signal Process., vol. 54 no. 3, pp. 1005-1017, 2006.
- [72] C. Wu, M. Shiue, and C. Wang, "Joint Carrier Synchronization and Equalization Algorithm for Packet-Based OFDM Systems Over the Multipath Fading Channel," IEEE Trans. Vehicular Technol., vol. 59, no. 1, pp. 248-260, 2010.
- [73] I. Barhumi, G. Leus, and M. Moonen, "Equalization for OFDM Over Doubly Selective Channels," IEEE Trans. Signal Process., vol. 54, no. 4, pp. 1445-1458, 2006.
- [74] T. Cui and C. Tellambura, "Joint Data Detection and Channel Estimation for OFDM Systems," IEEE Trans. Commun., vol. 54, no. 4, pp. 670-679, 2006.
- [75] H. Wu, X. Huang, Y. Wu, and X. Wang, "Theoretical Studies and Efficient Algorithm of Semi-Blind ICI Equalization for OFDM," IEEE Trans. Wireless Commun. vol. 7, no. 10, pp. 3791-3798, 2008.
- [76] F. Gao and A. Nallanathan, "Blind Channel Estimation for OFDM Systems via a Generalized Precoding," IEEE Trans. Vehicular Technol., vol. 56, no. 3, pp. 1155-1164, 2007.
- [77] X. G. Doukopoulos and G. V. Moustakides, "Blind Adaptive Channel Estimation in OFDM Systems," IEEE Trans. Wireless Commun., vol. 5, no. 7, pp. 1716-1725, 2006.
- [78] L. Rugini, P. Banelli, and G. Leus, "Simple Equalization of Time-Varying Channels for OFDM," IEEE Commun. Letters, vol. 9, no. 7, pp. 619-621, 2005.
- [79] T. Kim and T. Adali, "Approximation by fully complex multilayer perceptrons," Neural Computation, vol. 15, no. 7, pp. 1641-1666, 2003.

- [80] T. Kim and T. Adali, "Complex backpropagation neural network using elementary transcendental activation functions," Proc. of IEEE Inter. Conf. on Acoustics, Speech, Signal Process. (ICASSP), vol. 2, Salt Lake City, UT, pp. 1281-1284, May 2001.
- [81] I. Cha and S.A. Kassam, "Channel equalization using adaptive complex radial basis function networks," IEEE Journal. Sel. Area. Commun., vol. 13, no. 1, pp. 122-131, 1995.
- [82] M. Moustafa and S. El-Ramly, "Channel Estimation and Equalization Using Backpropagation Neural Networks in OFDM Systems," Proc. of IFIP Inter. Conf. on Wireless and Optical Commun. Networks, Cairo, Egypt, pp. 1-4, April 2009.
- [83] S. Lerkvaranyu, K. Dejhan, and Y. Miyanaga, "M-QAM demodulation in an OFDM system with RBF neural network," Proc. of 47th Midwest Symp. on Circuits and Systems (MWSCAS), pp. II-581-II-584, 2 July 2004.
- [84] K. Raivio, J. Henriksson and O. Simula, "Neural detection of QAM signal with strongly nonlinear receiver," Neurocomputing, vol. 21, 159-171, 1998.
- [85] G-B Huang, Q-Y Zhu, and C-K Siew, "Extreme Learning Machine: Theory and Applications," Neurocomputing, pp. 489-501, 2005.
- [86] X. Wang, J. Lu, H. Lin, N. Zhang, H. Sekiya and T. Yahagi, "Combining RNN Equalizer with SOM Detector," Proc. of 6th Inter. Conf. on signal process. (ICSP02), vol. 2, pp. 1291-1294, 2002.
- [87] H. Harada and R. Prasad, Simulation and software radio for mobile communication, Boston, Artech House, 2002.
- [88] T. Mitchell, Machine Learning, McGraw-Hill, 1997.

[89] J. H. Friedman, "Stochastic gradient boosting," Computational statistics and data analysis, vol. 38, pp. 367-378, 2002.

Vita Auctoris

Ishaq Gul Muhammad was born in Dubai, United Arab Emirates. He received his Bachelor of Science and Master of Science in Electrical & Computer Engineering from Oklahoma State University, U.S.A in December 1999 and May 2001, respectively. He worked as a member of scientific staff at Nortel Networks, Dallas, Texas working from Jan. 2001 and later moved to Computer College, Dubai, U.A.E as program manager/lecturer in Jan. 2002. From Jan. 2003 to Aug. 2007, he was with the department of electrical and computer engineering, American University in Dubai, U.A.E as a lab engineer. In spring 2007, he was an adjunct faculty in the electronics department at Higher Colleges of Technology, U.A.E. He joined Ph.D. program at the department of electrical and computer engineering, University of Windsor in Fall 2007. During Ph.D studies, he was awarded doctoral tuition scholarship and graduate teaching/research assistantship.