2019

Efficient Adaptive Filter Algorithms Using Variable Tap-length Scheme

Salem Alsaid
University of Windsor

EFFICIENT ADAPTIVE FILTER ALGORITHMS USING VARIABLE
TAP-LENGTH SCHEME

by

Salem Alsaid

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

2019

Efficient Adaptive Filter Algorithms Using Variable Tap-length Scheme

by

Salem Alsaid

APPROVED BY:

_____

F. Gebali, External Examiner
University of Victoria

_____

F. Ghrib
Department of Civil and Environmental Engineering

_____

N. Kar
Department of Electrical and Computer Engineering

_____

K. Tepe
Department of Electrical and Computer Engineering

_____

E. Abdel-Raheem, Advisor
Department of Electrical and Computer Engineering

January 31, 2019

Declaration of Co-Authorship / Previous Publication

I. Co-Authorship Declaration

I hereby declare that this dissertation incorporates material that is result of joint research, as follows: This dissertation incorporates the outcomes of joint research undertaken in collaboration with Dr. Khaled Mayyas and Dr. Mohammed Khaled under the supervision of professor Esam Abdel-Raheem.

The collaboration is covered through Chapters 3, 4, 5 and 6 of the dissertation. In all cases, the key ideas, primary contributions, experimental designs, data analysis and interpretation, were performed by the author, and the contribution of the collaborator was primarily through the provision of valuable suggestions for the representation of ideas, for the analysis of the results for the experiments carried out and editorial activities throughout the process of dissemination of the work.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my dissertation, and have obtained written permission from the only co-author to include the above material in my dissertation.

I certify that, with the above qualification, this dissertation, and the research to which it refers, is the product of my own work.

II. Declaration of Previous Publication

This thesis includes 4 original papers that have been previously published/submitted in peer reviewed journals and conferences, as follows:

Dissertation Chapter	Publication Title	Publication Status
Chapter 3	Salem Alsaid ; Esam Abdel-Raheem and Khaled Mayyas "variable order adaptation of LMS lattice predictors using forward residual erros" 2018 IEEE 61st International Midwest Symposium on Circuits and Systems (MWSCAS) ©2018	published
Chapter 4	Salem Alsaid ; Esam Abdel-Raheem ; Mohammed Khalid and Khaled Mayyas "Variable tap-length algorithm using lattice structured LMS adaptive filters" 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE) ©2017	published
Chapter 5	Salem Alsaid ; Esam Abdel-Raheem and Khaled Mayyas "Efficient Lattice RLS adaptive Filters Using Variable Tap-length Scheme"	Submitted
Chapter 6	Salem Alsaid ; Esam Abdel-Raheem and Khaled Mayyas "New Constant Modulus Blind Equalizer with Optimum Tap-length for QAM Signals"	Submitted

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor. I certify that, to the best of my knowledge, my dissertation does not infringe upon anyones 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 dissertation, published or otherwise, are fully acknowledged in accordance with the standard referencing practices.

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

Abstract

Today the usage of digital signal processors has increased, where adaptive filter algorithms are now routinely employed in mostly all contemporary devices such as mobile phones, camcorders, digital cameras, and medical monitoring equipment, to name few. The filter tap-length, or the number of taps, is a significant structural parameter of adaptive filters that can influences both the complexity and steady-state performance characteristics of the filter. Traditional implementation of adaptive filtering algorithms presume some fixed filter-length and focus on estimating variable filter's tap-weights parameters according to some pre-determined cost function. Although this approach can be adequate in some applications, it is not the case in more complicated ones as it does not answer the question of filter size (tap-length). This problem can be more apparent when the application involves a change in impulse response, making it hard for the adaptive filter algorithm to achieve best potential performance. A cost-effective approach is to come up with variable tap-length filtering scheme that can search for the optimal length while the filter is adapting its coefficients. In direct form structure filtering, commonly known as a transversal adaptive filter, several schemes were used to estimate the optimum tap-length. Among existing algorithms, pseudo fractional tap-length (FT) algorithm, is of particular interest because of its fast convergence rate and small steady-state error. Lattice structured adaptive filters, on the other hand, have attracted attention recently due to a number of desirable properties.

The aim of this research is to develop efficient adaptive filter algorithms that fill the gap where optimal filter structures were not proposed by incorporating the concept of pseudo fractional tap-length (FT) in adaptive filtering algorithms. The contribution of this research include the development of variable length adaptive filter scheme and hence optimal filter structure for the following applications:

(1) lattice prediction; (2) Least-Mean-Squares (LMS) lattice system identification; (3) Recursive Least-Squares (RLS) lattice system identification; (4) Constant Modulus Algorithm (CMA) blind equalization. To demonstrate the capability of proposed algorithms, simulations examples are implemented in different experimental conditions, where the results showed noticeable improvement in the context of mean square Error (MSE), as well as in the context of convergence rate of the proposed algorithms with their counterparts adaptive filter algorithms. Simulation results have also proven that with affordable extra computational complexity, an optimization for both of the adaptive filter coefficients and the filter tap-length can be attained.

to my parents

"And lower to them the wing of humility out of mercy and say, My Lord, have
mercy upon them as they brought me up when I was small."
True are the words of Allah

Acknowledgments

All praise is due to Allah, the Lord of all creations for his guidance. There are several people who deserve my sincere thanks for their generous contributions to this project. This people have supported me unconditionally and unlimitedly to make the possibility of finishing this job a real one. I would like to express my sincere gratitude to my advisor Dr. Esam Abdel-Raheem for the continuous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. My sincere thanks also goes to Dr. Khaled Mayyas who provided me with valuable advice and precious comments through co-authoring me in several publications. A very special gratitude goes out to all my thesis committee members: Dr. F. Ghrib, Dr. N. Kar, Dr. K. Tepe and Dr. M. Khalid for their insightful comments and encouragement that promoted me to widen my research from various perspectives. I thank the external examiner, Dr. F. Gebali, for all his suggestions and help in improving the quality of this dissertation. I would like to express my deep thanks to the ECE department graduate secretary, Ms. Andria Ballo, for her great help and consistent support during my studies at the university of Windsor. And finally, last but by no means least, I am grateful to all my family members, who have been with me always with prays and support. My wife and my children for their unconditional love and patience , I thank my parents for everything they did to me and ask Allah to rest them in heaven.

Table of Contents

List of Tables

List of Figures

List of Abbreviations/Symbols

$\boldsymbol{E}\{\cdot\}$	Expectation operator.
$\text{sgn}(\cdot)$	Real valued sign operator.
$\lvert \cdot \rvert$	Absolute value.
$\lfloor \cdot \rfloor$	Floor operator.
$\forall$	For all.
$(\cdot)^T$	Transportation.
$(\cdot)^*$	Complex conjugation.
$(\cdot)^H$	Transportation and conjugation.
$(\cdot)_R$	Real component of a complex number.
$(\cdot)_I$	Imaginary component of a complex number.
α	Leakage parameter.
γ	Variable length step size.
δ	Small positive integer.
Δ	Positive integer.
κ	Partial correlation coefficients.
ϵ	Instability prevention positive constant.
λ	Forgetting factor.

μ	Step size.
μ_p	Step size of lattice predictor.
$\xi^{ff}(n)$	Lattice RLS forward prediction auto-correlation.
$\xi^{bb}(n)$	Lattice RLS backward prediction auto-correlation.
$\xi^{ee}(n)$	Lattice RLS joint process estimator auto-correlation.
$\xi^{be}(n)$	Lattice RLS backward joint process estimator cross-correlation.
$\psi_m(n)$	Lattice RLS conversion factor.
$\mathbf{a}$	Forward predictor tap coefficient vector.
$b_m(n)$	Backward prediction error for mth stage.
$\mathbf{c}$	Backward predictor tap coefficient vector.
$e(n)$	Error signal.
$f_m(n)$	Forward prediction error for mth stage.
$l(n)$	Fractional tap-length.
$L(n)$	Tap-length update.
P_m	Power of mth stage of lattice predictor.
T	Symbol period.
$v(n)$	Additive white Gaussian noise.
$\mathbf{w(n)}$	Filter tap coefficient vector.

x(n)	Regressor vector of filter input samples.
$\hat{x}$	The estimated value of 'x'.
ASIC	Application Specific Integrated Circuit
CMA	Constant modulus algorithm.
DSP	Digital Signal Processor
EMSE	Excess mean square error.
FIR	Finite impulse response.
FO-LPF	Fractional order lattice prediction filter algorithm.
FPGA	Field gate programmable arrays
FT	Fractional tap-length algorithm.
FT-LLMS	Fractional tap-length lattice LMS algorithm.
FT-LRLS	Fractional tap-length lattice RLS algorithm.
GSA	Generalized Sato algorithm.
IEEE	Institute of Electrical and Electronics Engineers.
i.i.d.	Independent and identically distributed.
I/O	Input/Output.
ISI	Inter symbol interference.
LMS	Least mean squares algorithm.
LLMS	Lattice least mean squares algorithm.
LRLS	Lattice recursive least squares algorithm.

MMA	Multimodulus algorithm.
MSE	Mean squares error.
MMSE	Minimum mean squares error.
NLMS	Normalized least mean ssquares algorithm.
RLS	Recursive least squares algorithm.
SNR	Signal to noise ratio.
TDL	Tapped delay line.
VLSI	Very large scale integration.
VL-CMA	Variable length constant modulus algorithm.

Chapter 1

Introduction

1.1 Structure adaptation

1.1.1 Evolution

Structure adaptation is a term used in literature to refer to optimizing the filter structure via variable tap-length techniques [1–4], and hence improving adaptive filters efficiency by searching for the filter's optimum length [5]. In direct form structure filters, commonly known as transferal adaptive filter, several schemes were utilized to pursue this task. According to the analysis in [6] and [7], underestimating the filter length leads to an extra steady-state mean-square-error (MSE), on the contrary, [1], [8] and [9] stated that, overestimating the tap-length can increase the computational complexity and eventually result in higher excess mean square error (EMSE). Thus, a variable tap-length is needed to find the *optimum* filter length that best balances between the filter's steady-state performance and complexity.

1.2 Motivation

Adaptive filtering plays a vital role in enormous number of applications ranging from digital and wireless communications to biomedical systems, to name a few [10–12]. Improving the performance adaptive filter algorithms necessitate not only bringing the algorithms up and running but also optimizing it in all aspects. One important aspect that has a direct impact on adaptive filter algorithms implementation in hardware such as VLSI, ASIC, FPGA ...etc, is the filter size. This brings the question of

"how long the adaptive filter should be", into attention. In practice there is no general solution for this question and researchers used different approaches to solve it. A cost-effective solution is to come up with a variable tap-length scheme that can search for the optimal length while the filter adapting. Because of the advantages of Fractional Tap-length (FT) algorithm, which will be discussed in more details in this thesis, this research utilizes FT variable tap-length scheme to introduce new variable adaptation strategies in widely used signal processing applications. Lattice structured adaptive filters are well known by some characteristics including [13–15]:

- Stability

- Modularity

- Fast convergence rate

In the constant modulus algorithm (CMA) blind equalization algorithm [5,13], which is used to compensate for signal distortion attributed to Inter-symbol Interference (ISI) without restoring to training sequence. It is noted that, his capability comes with high computational load which can make the equalizer alone is sufficient to drive the design of quadratic amplitude modulation (QAM) signals demodulators. Therefore, a new variable length CMA (VL-CMA) algorithm is proposed using a pseudo-FT concept to estimate the optimal equalizers weights and tap-length simultaneously and hence enhancing the equalization process in blind mode with extra efficiency.

1.3   Research purpose and challenges

The aim of this research is to investigate available variable tap-length strategies and find an adaptive tap-length solution for the following applications:

- Adaptive prediction (LMS lattice structure)

- System identification (LMS & RLS lattice structure)
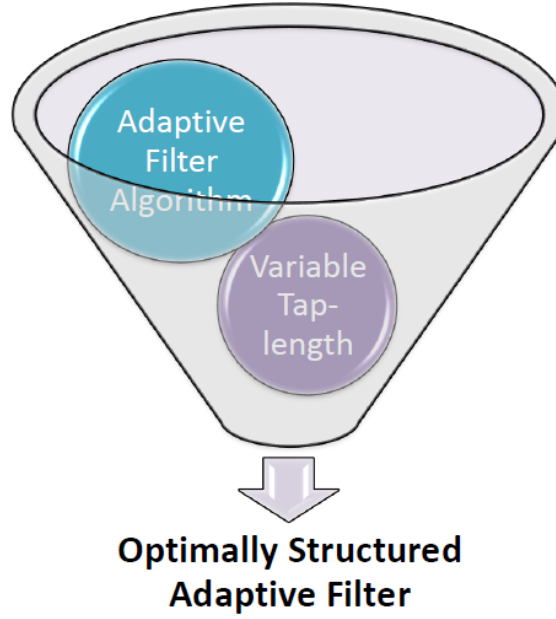
Figure 1.1 – Proposed combination of structure adaptation.

- Blind equalization (transversal structure)

To achieve this goal, two dimensional adaption algorithm, that optimizes the tap-length and tap-weight is needed. This requires adjusting the proposed algorithms to accommodate the change in filtering tap-length during iterations, which is achieved by combining a variable tap-length algorithm with the adaptive filter algorithm in the application under investigation Fig. 1.1 .

The parameters that make adaptation process challenging are the following:

- Adaptive filter application parameters

- Variable length algorithm parameters

- Associated application modeling

The last parameters, i.e., the associated application modeling, can include any sudden change in the channel. The four major challenges burdening the realization of the proposed scheme adaptation frameworks are as follows:

- Parameters selection is an application dependent.

- Concurrency of coefficients and structure adaptation processes.

- Identifying the error signal that contains characteristics suitable to update fractional tap-length in different applications of this dissertation.

- Formulation of variable tap-length blind equalization modeling to accommodate the multirate system.

1.4 Organization of the dissertation

Fig. 1.2, shows the proposed road map for this dissertation, as it deals with two kinds of adaptive filtering structures, namely, lattice form and direct form. Fig. 1.2 shows the road map of this work. Two adaptive filtering structures are dealt with, namely, direct-form, and lattice structures. LMS and RLS algorithms are re-implemented using variable tap-length in the applications of adaptive prediction, system identification and blind equalization and bring about structurally optimized adaptive filter algorithms. The remainder of this dissertation is organized as follows. Chapter 2 provides a brief literature review of variable tap-length algorithms in the adaptive filtering applications of this research. Adaptive prediction using direct form and reconstructing it using lattice form and optimizing this final structure using forward and backward adaptive prediction are discussed in Chapter 3. Variable tap-length adaptive filters using LMS and RLS algorithms in lattice realization of system identification setup will be discussed in Chapter 4 and Chapter 5 respectively, where a comparison between the two adaptation schemes will be discussed at the end of the Chapter 5. The proposed optimum tap-length CMA blind equalizer for QAM signals is discussed in Chapter 6. Finally, Chapter 7 will summarize the dissertation with the presentation of conclusions and the suggestions of future works.
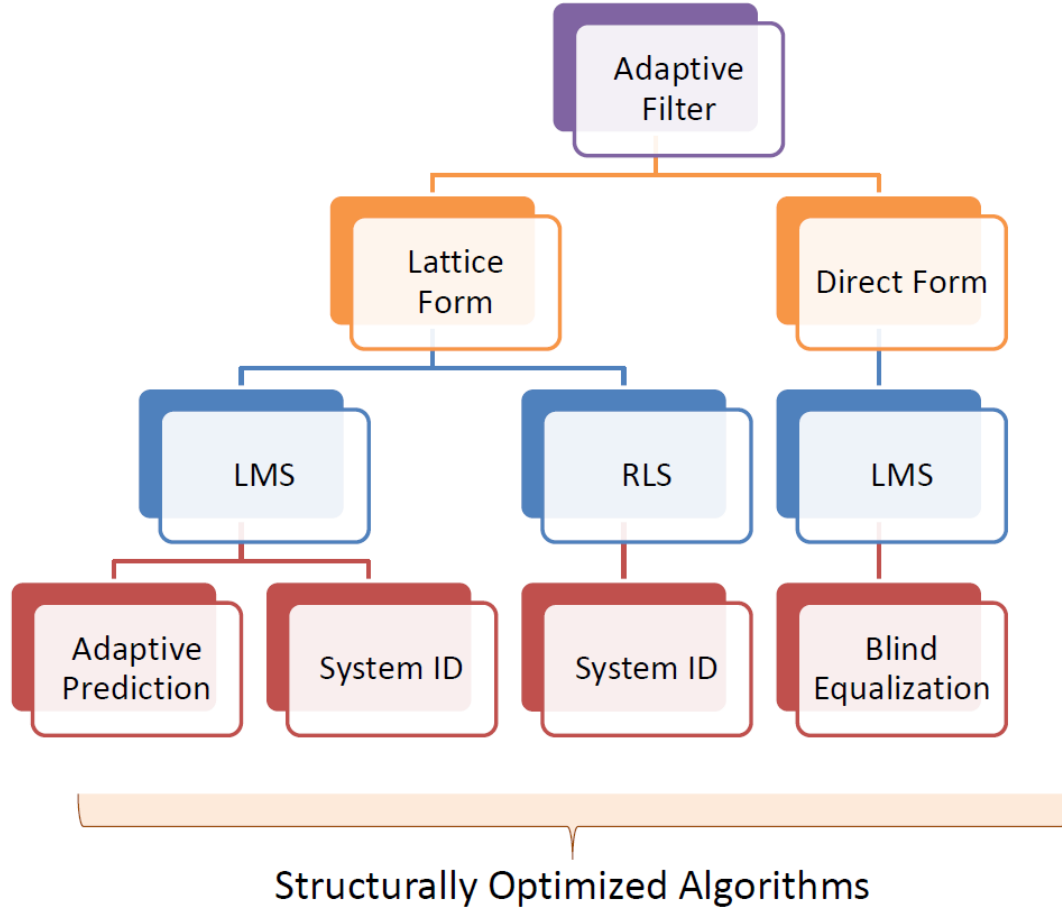
Figure 1.2 – Targeted adaptive filter structures and algorithms

1.5 Research contribution

The major contribution of this dissertation is to present new algorithms for optimizing the adaptive filter in a variety of applications and structures using variable tap-length scheme. The proposed algorithms are:

- Fractional Order Lattice Prediction Filter (FO-LPF)

- Fractional Tap-length Lattice LMS Filter (FT-LLMS)

- Fractional Tap-length Lattice RLS Filter (FT-LRLS)

- Variable Length CMA Blind Equalizer (VL-CMA)

References

[1] Y. Gong and C. F. N. Cowan, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July. 2005.

[2] N. Li, Y. Zhang, Y. Hao, and Y. Zhao, "A new variable tap-length lms algorithm with variable error width," in *2008 9th International Conference on Signal Processing*, Oct. 2008, pp. 276–279.

[3] D. Xu, B. Yin, W. Wang, and W. Zhu, "Variable tap-length lms algorithm based on adaptive parameters for tdl structure adaption," *IEEE Signal Processing Letters*, vol. 21, no. 7, pp. 809–813, July 2014.

[4] Y. Zhang, J. A. Chambers, S. Sanei, P. Kendrick, and T. J. Cox, "A new variable tap-length lms algorithm to model an exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 263–266, April 2007.

[5] P. K. Mohapatra, P. K. Jena, S. K. Bisoi, S. K. Rout, and S. P. Panigrahi, "Channel equalization as an optimization problem," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Oct. 2016, pp. 1158–1163.

[6] Y. Gu, K. Tang, H. Cui, and W. Du, "Convergence analysis of a deficient-length lms filter and optimal-length sequence to model exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 10, no. 1, pp. 4–7, Jan. 2003.

[7] K. Mayyas, "Performance analysis of the deficient length lms adaptive algorithm," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2727–2734, Aug. 2005.

[8] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, Feb. 2008.

[9] Y. Gong and C. F. N. Cowan, "Structure adaptation of linear mmse adaptive filters," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 151, no. 4, pp. 271–277, Aug. 2004.

[10] D. Chang and F. Chu, "Feedforward active noise control with a new variable tap-length and step-size filtered-x lms algorithm," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 542–555, Feb. 2014.

[11] M. Z. U. Rahman, R. A. Shaik, and D. R. K. Reddy, "Efficient sign based normalized adaptive filtering techniques for cancelation of artifacts in ecg signals: Application to wireless biotelemetry," *Signal Processing*, vol. 91, no. 2, pp. 225 – 239, 2011.

[12] Y.-P. Li, T.-S. Lee, and B.-F. Wu, "A variable step-size sign algorithm for channel estimation," *Signal Processing*, vol. 102, pp. 304 – 312, 2014.

[13] H. F. Lee, J. T. Yuan, and T. C. Lin, "Iir lattice-based blind equalization algorithms," in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, Sept. 2012, pp. 1–5.

[14] M. Setareh, M. Parniani, and F. Aminifar, "Ambient data-based online electromechanical mode estimation by errorfeedback lattice rls filter," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 3745–3756, July 2018.

[15] S. Alsaid, E. Abdel-Raheem, M. Khalid, and K. Mayyas, "Variable tap-length algorithm using lattice structured lms adaptive filters," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, April. 2017, pp. 1–5.

Chapter 2

Literature Review

2.1 Introduction

In the context of adaptive filtering, the filter tap-length problem is simply how to optimize the conflicting requirements of filter's parameters to determine the optimum tap-length that balances performance and complexity that vary with scenarios [1–3]. FIR filter which is usually implemented using a tapped-delay-line (TDL) structure with tap coefficients recursively updated by adaptive algorithms, such as least mean squares (LMS) and recursive least squares (RLS) algorithms [4]. The number of taps, is considered to be an important parameter that can influence the adaptive filter's performance [5, 6]. This is because, the tap-length needs to be long enough to accomplish the desired performance and on the other front, the tap-length cannot be too long, as this can cause the adaptation noise to become too high, because the adaptive filter usually converges to a MSE level higher than the MMSE as a result to the adaption noise [7]. Even without adaption noise, the filter should not be too long as this will increase the computational complexity of the adaptive filter [8]. Various variable length adaptive schemes are available in the literature; the most popular ones are the segmented filter (SF) algorithm [9], the gradient descent (GD) algorithm [10] and the fractional tap-length (FT) algorithm [6, 7]. By dividing the filter into k segments and assigning each segment with Δ coefficients, the SF algorithm compares the difference between the accumulated square errors from the last two segments and based on that, the filter tap-length is adjusted by adding or subtracting one segment [9]. Thus, the GD algorithm is more flexible than SF algorithm. However,

the GD algorithm suffers from "wandering" problem, where the adaptive filter tap-length keeps hovering in a range larger than the optimum tap-length [11].

2.2 Variable tap-length algorithms comparison

Table 2.1 provides a summary of comparison between the most popular variable length algorithms in the literature.

Table 2.1 – Variable tap-length algorithms.

Algorithms	Segmented Filter (SF)	Gradient Descent (GD)	Fractional Tap-length (FT)
Summary	The filter is partitioned into k segments each with length Δ.	No filter segmentation, as the tap length is updated using gradient descent approach.	Filters tap-length increment in each iteration is not restricted to fixed value, and hence Fractional Tap (FT).
Cons	Inflexible	Wandering problem, in which the filter tap-length hovers in a range larger than the optimum.	With little extra complexity, (SF) and (GD) disadvantages can be overcome
Pros	Easy to implement	Flexible	Easy to implement and flexible.

FT algorithm uses the fractional tap-length during the instantaneous tap-length update, and the integer value of the fractional tap-length remains unchanged until the increment of the fractional tap-length accumulates to a certain extent compared with the integer value [1, 6]. Therefore, the FT algorithm is considered to be an efficient variable tap-length alternative that exhibits better convergence performance than the previously mentioned algorithms [6, 11]. Fractional tap-length is nominated to be the guiding structure adaption strategy in this dissertation due to its advantages when compared to other schemes. Therefore, in the following subsections FT algorithm is discussed in details

2.2.1 Optimum tap-length

Based on the previous Section, It can be concluded that, there exists an *optimum* tap-length that balances the conflicting concerns of performance and complexity. References [12] and [13] defined the the optimum tap-length to be the smallest integer N_0 that fulfills the following inequality

$$\xi_{N-1} - \xi_N \leq \varepsilon \qquad \forall \qquad N \geq N_0 \tag{2.1}$$

where ξ_N is the steady-state mean square error (MSE) when the tap-length is N, and ε is some small positive value that is predetermined based on system requirements.

2.2.2 FT algorithm cost function

In LMS transversal adaptive filter, if $\mathbf{w}_N$ and $\mathbf{x}_N$ are the N-length corresponding steady state tap-vector and regressor input vector, respectively. Then, the segmented steady-state error can be defined as [6, 14, 15]

$$e_G^{(N)} = d(n) - \mathbf{w}_N^T(1:G)\mathbf{x}(1:G) \tag{2.2}$$

where N is the assumed tap-length, $1 \leq G \leq N$, $\mathbf{w}_N(1:G)$, $\mathbf{x}_N(1:G)$ are the vectors corresponding to first G coefficients of steady-state tap-vector and regressor-vector, respectively, and $d(n)$ is the desired signal. The segmented steady state MSE was defined in [11] as $\xi_G^{(N)} = E|e_G^{(N)}|^2$ and ideally $\xi_{N-\Delta}^{(N)} \geq \xi_N^{(N)}$, where Δ is a positive integer. Therefore the improved cost function was constructed by [6] as

$$\min\{N \mid \xi_{N-\Delta}^{(N)} - \xi_N^{(N)} \leq \varepsilon'\} \tag{2.3}$$

which have been used by [1, 7, 14–17] to search for optimal tap-length in different adaptive filter applications. It can be noted that, in (2.3), if $\varepsilon' = \varepsilon$, the optimum tap-length from cost function could be overestimated which means $N_0' \geq N_0$, and because in practice, ξ_N and $\xi_G^{(N)}$ are unknown in advance, the cost function of cost function (2.3) gives a biased solution [6]. Because of its advantages that related to the fractional variable tap-length, the cost function of (2.3) will be used throughout the rest of this research.

2.3 Variable length blind equalization

The tap-length of the equalizer was discussed in the literature, and because no general solution have answered the question of the equalizer's tap-length, in practice different approaches were proposed in attempt to solve it. In the first approach, a prototype of the equalizer is built and then tested against a variety of actual channels [18]. The second method applies rules of thumb that appears intuitively reasonable for length selection depending on the type of communication channel and the transmitted signal sampling rate [19], [18]. Both approaches are costly and does not deal with changes in the channel behaviors which can compromise the equalizer's performance. A cost-effective approach is to come up with a variable tap-length scheme that can search for the optimal length while the equalizer is adapting its coefficients. In [9], a segmented

filter (SF) variable tap-length algorithm is employed to the equalizer. In SF algorithm the equalizer is subdivided into k segments, each with fixed coefficients. Then, based on the difference between the accumulated squared errors from the last two segments, the tap-length of the filter is modified by adding or subtracting one segment, which makes SF an inflexible option. Authors in [1] and [16] used a more flexible and robust variable length technique that employs the fraction tap-length (FT) algorithm, for trained mode adaptive equalizer. In the literature, authors in [16] , [9], and [1] utilized the fractional variable tap-length toward linear equalizer. Therefore, the novelty of this work is to search for the optimal tap-length of the blind equalizer using the CMA algorithm with the FT technique. This is done by modifying a non-linear error $e(n)$ of the CMA equalizer output $y(n)$ to update FT iterations.

References

[1] Z. Liu, "Variable tap-length linear equaliser with variable tap-length adaptation step-size," *Electronics Letters*, vol. 50, no. 8, pp. 587–589, April 2014.

[2] Y. Zhang, J. A. Chambers, S. Sanei, P. Kendrick, and T. J. Cox, "A new variable tap-length lms algorithm to model an exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 263–266, April 2007.

[3] D. Xu, B. Yin, W. Wang, and W. Zhu, "Variable tap-length lms algorithm based on adaptive parameters for tdl structure adaption," *IEEE Signal Processing Letters*, vol. 21, no. 7, pp. 809–813, July 2014.

[4] X. Ma, K. Shi, and G. T. Zhou, "A variable step size and variable tap length lms algorithm for impulse responses with exponential power profile," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on(ICASSP)*, 04 2009, pp. 3105–3108.

[5] N. Li, Y. Zhang, Y. Hao, and Y. Zhao, "A new variable tap-length lms algorithm with variable error width," in *2008 9th International Conference on Signal Processing*, Oct. 2008, pp. 276–279.

[6] Y. Gong and C. F. N. Cowan, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[7] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, Feb. 2008.

[8] P. Diniz, "Adaptive filtering: Algorithms and practical implementation. springer," *New York, NY, USA*, 2008.

[9] F. Riera-Palou, J. M. Noras, and D. G. M. Cruickshank, "Linear equalisers, with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, pp. 1553–1554, Dec. 2001.

[10] Y. Gu, K. Tang, and H. Cui, "Lms algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, March 2004.

[11] Y. Gong and C. Cowan, "Structure adaptation of linear mmse adaptive filters," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 151, no. 4, pp. 271–277, 2004.

[12] Y. Gong and C. F. N. Cowan, "A novel variable tap-length algorithm for linear adaptive filters," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, May 2004, pp. ii–825.

[13] Y. Gong and C. F. N. Cowan, "A novel variable tap-length algorithm for linear adaptive filters," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, May 2004.

[14] N. Li, Y. Zhang, Y. Zhao, and Y. Hao, "An improved variable tap-length lms algorithm," *Signal Processing*, vol. 89, no. 5, pp. 908–912, 2009.

[15] K. Mayyas and H. A. AbuSeba, "A new variable length nlms adaptive algorithm," *Digital Signal Processing*, vol. 34, pp. 82–91, 2014.

[16] Y. Gong, X. Hong, and K. F. Abu-Salim, "Adaptive mmse equalizer with optimum tap-length and decision delay," in *Sensor Signal Processing for Defence (SSPD 2010)*, Sept. 2010, pp. 1–5.

[17] S. Alsaid, E. Abdel-Raheem, M. Khalid, and K. Mayyas, "Variable tap-length algorithm using lattice structured lms adaptive filters," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, April 2017, pp. 1–5.

[18] J. R. Treichler, I. Fijalkow, and C. R. Johnson, "Fractionally spaced equalizers," *IEEE Signal Processing Magazine*, vol. 13, no. 3, pp. 65–81, May 1996.

[19] J. R. Treichler, M. G. Larimore, and J. C. Harp, "Practical blind demodulators for high-order qam signals," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1907–1926, Oct. 1998.

Chapter 3

Variable Tap-length Adaptive Lattice Prediction

3.1 Introduction

In classic implementation of adaptive filtering applications, the filter's length is kept fixed, however, in several practical circumstances the optimal tap-length is unknown and/or variable with time. According to the analysis in [1] and [2], underestimating the filter length leads to an extra steady-state mean-square-error (MSE), on the contrary, [3], [4] and [5] stated that, overestimating the tap-length can increase the computational complexity and eventually result in higher excess mean square error (EMSE). Thus, there exists an *optimum* tap-length that best trades off between the filter's steady-state performance and complexity. Consequently, a variable tap-length algorithm is needed to find the optimal filter's length. Various variable length adaptive schemes are available in the literature, the most popular ones are the Segmented Filter (SF) [6], the Gradient Descent (GD) [7], and the Fractional Tap-length (FT) algorithm [5] [3] and [4]. Because of its robustness and efficiency FT algorithm is of particular interest because. This algorithm utilizes some fractional tap-length value during the instantaneous tap-length update until the increment of the fractional tap-length accumulates to a certain extent compared with the length's integer value [3,8].

Adaptive prediction is the application in which a model for the future (forward) or previous (backward) value of the filter input sequence is estimated using forward or backward predictors respectively. Lattice structure has most commonly been uti-

lized for implementing linear predictors in the context of speech processing applications [9, 10]. This is mainly because of some appealing characteristics that lattice structure acquires over other filtering forms, such as, modularity, fast convergence rate and orthogonalization transformation [11, 12]. However, traditional implementation of adaptive prediction assumes some predefined filter's tap-length, which does not necessarily meet the optimal criterion of filter's length. This paper takes advantage of the forward and backward residual errors of lattice predictors to develop a novel variable tap-length lattice predictor algorithm, that is, the Fractional Tap-length Lattice Predictor Filter (FT-LPF) [13, 14].

3.2 Adaptive prediction

Using *transversal* or *tapped delay line* filter structure shown in figure [3.1] below [11, 12]
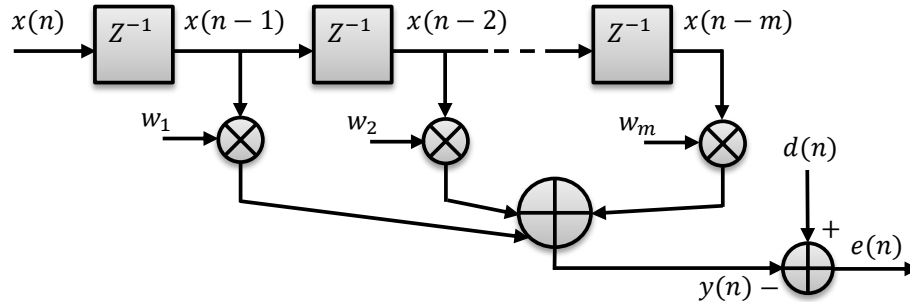
Figure 3.1 – Transversal structured adaptive filter [15].

If the input sequence $x(n)$ is a realization of stationary stochastic process, then two distinguished prediction schemes can be defined [16]:

3.2.1 Forward prediction

In this filter, the future value of the input process is predicated using the past values of same process. Figure 3.2 shows the transversal implementation of an mth-order forward predictor. If the tap-input vector is $\mathbf{x}(n-1) = x(n-1), x(n-2), \ldots, x(n-m)$ then a prediction of the present value $x(n)$ can be obtained by optimizing the filter's tap-weight vector $a_{m,1}, a_{m,2}, \ldots, a_{m,m}$ in the mean-square sense according to the Wiener theory [11, 12, 17].
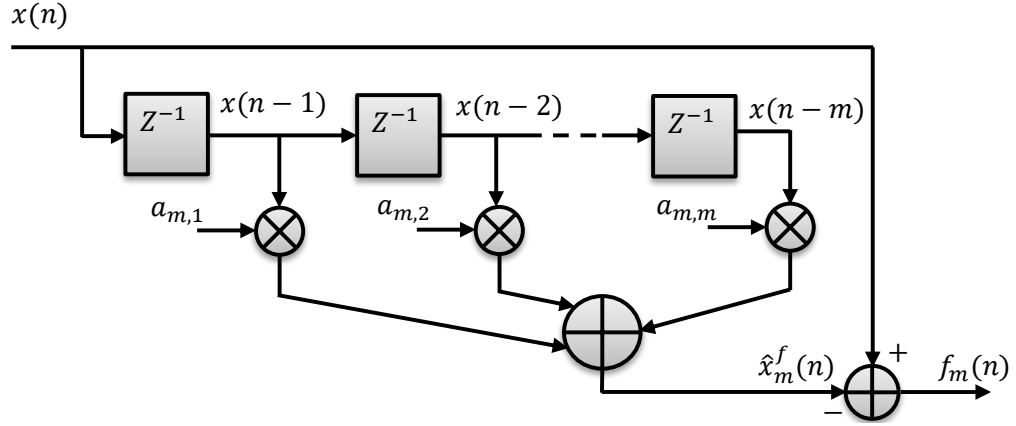
Figure 3.2 – Transversal forward predictor [12].

Thus, the optimum forward predictor tap-weights are obtained by minimizing the function

$$P_m^f = \mathbf{E}\left[f_m^2(n)\right] \tag{3.1}$$

where

$$f_m(n) = x(n) - \hat{x}_m^f(n) \tag{3.2}$$

is the forward prediction error and

$$\hat{x}_m^f = \sum_{i=1}^{m} a_{m,i} x(n-i) = \mathbf{a}_m^T \mathbf{x}_m(n-1) \tag{3.3}$$

18

is the mth-order forward prediction of the sample $x(n)$. This is a conventional Weiner filtering problem with the input vector $\mathbf{x}_m(n-1)$ and the desired output $x(n)$. Hence the Weiner-Hopf equation is obtained by [11], [12]

$$\mathbf{R}\mathbf{a}_{m,o} = \mathbf{r} \tag{3.4}$$

where $\mathbf{R} = \mathbf{E} \left[\mathbf{x}_m(n-1)\mathbf{x}_m^T(n-1)\right]$, $\mathbf{r} = \mathbf{E} \left[x(n)\mathbf{x}_m(n-1)\right]$ and $\mathbf{a}_{m,o}$ denotes the optimum value of $\mathbf{a}_m$. When the predictor tap weights are optimized according to (3.4), P_m^f is minimized and the Weiner forward minimum mean-square error can be obtained by

$$P_m^f = \mathbf{E} \left[x^2(n)\right] - \mathbf{r}^T \mathbf{a}_m = \mathbf{E}[x^2(n)] - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} \tag{3.5}$$

Based on that the forward-prediction error filter, shown in Fig. 3.3, can be defined as the one has $x(n)$ as the input and the forward prediction error as the output [11].
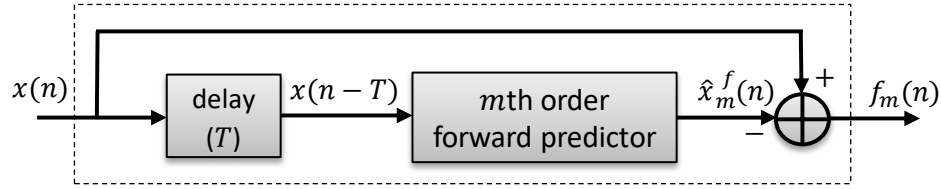
Figure 3.3 – Forward prediction filter [11].

3.2.2 Backward prediction

Figure 3.4 shows the transversal implementation of an mth-order backward predictor. If the tap-input vector is $\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-m+1)]$, then a prediction of the input sample $x(n-m)$ can be obtained by optimizing the filter's tap-weight vector is $c_{m,1}, c_{m,2}, \ldots, c_{m,m}$ in the mean-squares sense according to the Wiener theory [11, 12]. Thus, the optimum backward predictor tap-weights are obtained by minimizing the function

$$P_m^b = \mathbf{E} \; b_m^2(n) \tag{3.6}$$

19
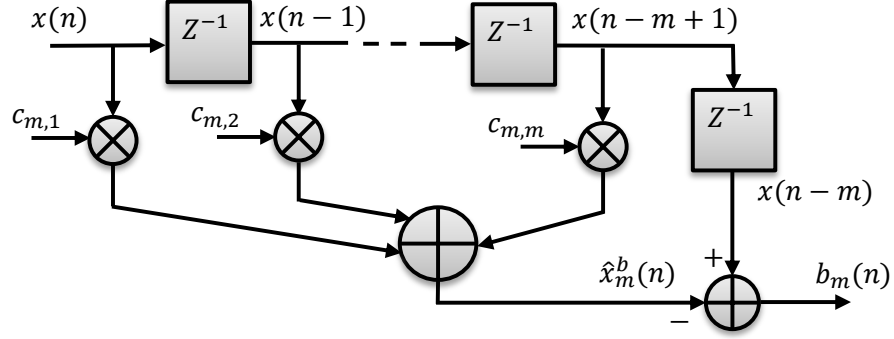
Figure 3.4 – Transversal backward predictor [12].

where

$$b_m(n) = x(n - m) - \hat{x}_m^b(n) \tag{3.7}$$

is the backward prediction error and

$$\hat{x}_m^b = \sum_{i=1}^{m} c_{m,i} x(n - i + 1) = \mathbf{c}_m^T \mathbf{x}_m(n) \tag{3.8}$$

is the mth-order forward prediction of the sample $x(n)$. This is a conventional Weiner
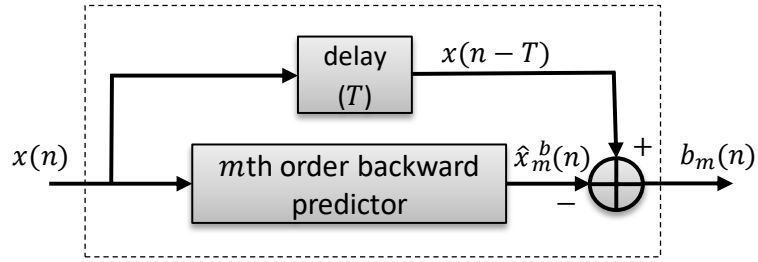
Figure 3.5 – Backward prediction filter [11].

filtering problem with the input vector $\mathbf{x}_m(n)$ and the desired output $x(n)$. Hence the Weiner-Hopf equation is obtained by [11, 12]

$$\mathbf{R}_b \mathbf{c}_{m,o} = \mathbf{r}_b \tag{3.9}$$

20

where $\mathbf{R}_b = \mathbf{E}\left[\mathbf{x}_m(n-1)\mathbf{x}_m^T(n-1)\right]$, $\mathbf{r}_b = \mathbf{E}\left[x(n)\mathbf{x}_m(n-1)\right]$ and $\mathbf{c}_{m,o}$ denotes the optimum value of $\mathbf{c}_m$. When the predictor tap weights are optimized according to (3.9), P_m^b is minimized and the transversal Weiner backward minimum mean-square error can be obtained using

$$P_m^b = \mathbf{E}\left[x^2(n)\right] - \mathbf{r}^T\mathbf{a}_m = \mathbf{E}\left[x^2(n)\right] - \mathbf{r}^T\mathbf{R}^{-1}\mathbf{r} \tag{3.10}$$

Based on that the backward-prediction error filter that has $x(n)$ as the input and the backward prediction error as the output can be defined.
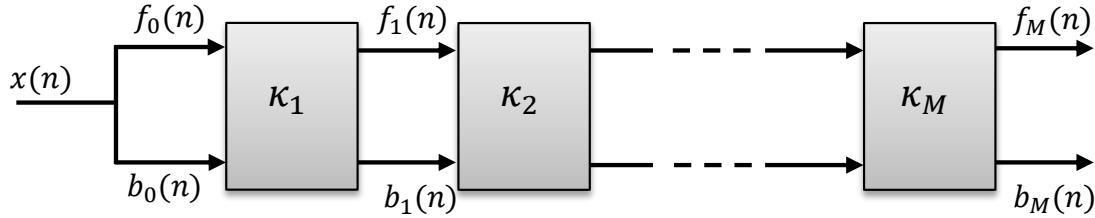
Figure 3.6 – Overall lattice structure [11].

3.3   Lattice structure

Lattice structure is formulated around the basic building block shown in Fig. 3.7. The the input-output relation of such a structure is characterized by a single parameter, known as, the Partial Correlation (PARCOR) Coefficient $\kappa_m(n)$. The order-update equations for forward and backward residual prediction errors are recursively specified by [11, 18].

$$\begin{bmatrix} f_m(n) \\ b_m(n) \end{bmatrix} = \begin{bmatrix} 1 & -\kappa_m(n) \\ -\kappa_m(n) & 1 \end{bmatrix} \begin{bmatrix} f_{m-1}(n) \\ b_{m-1}(n-1) \end{bmatrix} \tag{3.11}$$

where $m = 1, 2, \ldots, M$, $\kappa_m(n)$ is the partial coefficient (PARCOR) at the Mth stage and time n and $f_m(n)$ and $b_m(n)$ are forward and backward prediction errors respectively. The PARCOR superscript of * denotes to the complex conjugation. Figure 3.6 shows the overall lattice structure of an M-order forward-backward predictor. Each stage receives the forward and backward prediction errors from the previous stage as its input and produces the forward and backward prediction errors of one order higher as output.
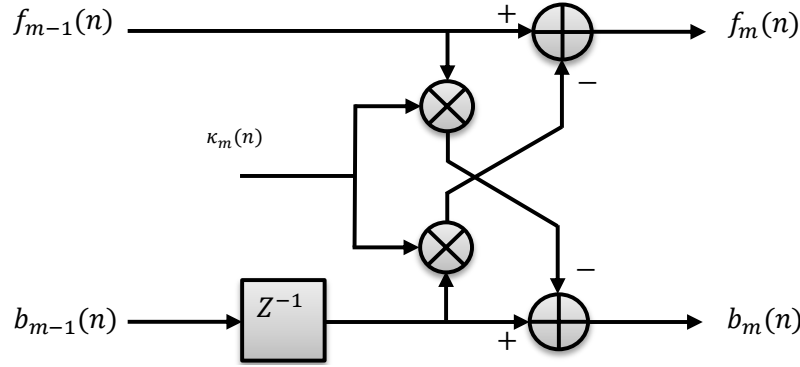
Figure 3.7 – Lattice building block [11].

To initialize the adaptation [11]:

$$f_0(n) = b_0(n) = x(n) \tag{3.12}$$

The optimum (PARCOR) coefficient κ_m of the mth stage of lattice predictor is obtained by minimizing the cost function:

$$\xi_m = \mathbf{E}\left[f_m^2(n) + b_m^2(n)\right] \tag{3.13}$$

$$\kappa_m(n+1) = \kappa_m(n) - \mu_{p,m}(n)\frac{\partial \hat{\xi}_m(n)}{\partial \kappa_m} \tag{3.14}$$

22

where $\mu_m(n)$ is the step-size. An estimate of the cost function ξ_m, based on the most recent samples of the forward and backward prediction errors, is given by

$$\hat{\xi}_m = f_m^2(n) + b_m^2(n) \tag{3.15}$$

Substituting (3.13) in (3.14) and using (3.11), yields

$$\kappa_m(n+1) = \kappa_m(n) + 2\mu_m(n). \left[f_m(n)b_{m-1}(n-1) + b_m(n)f_{m-1}(n) \right] \tag{3.16}$$

The convergence rate can be accelerated by normalizing the step-size $\mu_{p,m}(n)$ by the signal power at the mth stage of the predictor, which is estimated by the iteration

$$P_{m-1}(n) = \beta P_{m-1}(n-1) + (1-\beta).[(f_{m-1}^2(n) + b_{m-1}^2(n-1))] \tag{3.17}$$

Hence, the normalized step-size is given by

$$\mu_m(n) = \frac{\mu_p}{P_{m-1}(n) + \epsilon} \tag{3.18}$$

where μ_p is the constant step-size and ϵ is a small positive value to avoid algorithm instability. Given the PARCOR coefficients of lattice predictor, the corresponding transversal structure can be calculated using Levinson-Durbin algorithm [19–21]. Algorithm 1 outlines the lattice forward prediction filter [11], in which as an input, the algorithm receives the present values of PARCOR coefficients, the backward prediction error vector, the power estimates vector and the most recent input sample $x(n)$. Then the algorithm updates these parameters and returns the forward residual errors $[f_1(n), f_2(n), \ldots, f_M(n)]$. In a similar manner, the lattice backward prediction filter [11] receives the present values of PARCOR coefficients, the backward prediction error vector, the power estimates vector and the most recent input sample $x(n)$. However, as a lattice backward predictor filter, the algorithm in this case updates the-

ses parameters according to equations demonstrated in Algorithm 2 and returns the backward residual errors $[b_1(n), b_2(n), \ldots, b_M(n)]$. The following two algorithms summarize the forward and backward LMS lattice structured adaptive filters respectively.

Algorithm 1: LMS algorithm for adaptive lattice forward predictor.

Input: $x(n)$, $\kappa_1(n), \kappa_2(n). \ldots, \kappa_M(n)$,

$\mathbf{b}(n-1) = [b_0(n-1), b_1(n-1), \ldots, b_M(n-1)]^T$,

$\mathbf{f}(n-1) = [f_0(n-1), f_1(n-1), \ldots, f_M(n-1)]^T$,

$P_0(n-1), P_1(n-1), \ldots, P_M(n-1)$.

Output: $\mathbf{f}(n) = [f_1(n), f_2(n), \ldots, f_M(n)]^T$,

$\kappa_1(n+1), \kappa_2(n+1). \ldots, \kappa_M(n+1)$,

$P_0(n), P_1(n), \ldots, P_M(n)$.

1 Initialize $f_0(n) = b_0(n) = x(n)$

2 $P_0(n) = \beta P_0(n-1) + (1-\beta)\left[f_0(n)^2 + b_0(n-1)^2\right]$

3 **for** $m = 1, 2, \ldots, M$ **do**

4 $\quad f_m(n) = f_{m-1}(n) - \kappa_m(n)b_{m-1}(n-1)$

5 $\quad b_m(n) = b_{m-1}(n-1) - \kappa_m(n)f_{m-1}(n)$

6 $\quad \kappa_m(n+1) = \kappa_m(n) + \frac{2\mu_p}{P_{m-1}(n)+\epsilon}\left[f_{m-1}(n)b_m(n) + b_{m-1}(n-1)f_m(n)\right]$

7 $\quad P_m(n) = \beta P_m(n-1) + (1-\beta)[f_m(n)^2 + b_m(n-1)^2]$

8 **end for**

9 Return $f_M(n)$

24

Algorithm 2: LMS algorithm for adaptive lattice backward predictor.

Input: $x(n)$, $\kappa_1(n), \kappa_2(n), \ldots, \kappa_M(n)$,

$\qquad \mathbf{b}(n-1) = [b_0(n-1), b_1(n-1), \ldots, b_M(n-1)]^T$,

$\qquad \mathbf{f}(n-1) = [f_0(n-1), f_1(n-1), \ldots, f_M(n-1)]^T$,

$\qquad P_0(n-1), P_1(n-1), \ldots, P_M(n-1)$.

Output: $\mathbf{b}(n) = [b_1(n), b_2(n), \ldots, b_M(n)]^T$,

$\qquad \kappa_1(n+1), \kappa_2(n+1), \ldots, \kappa_M(n+1)$,

$\qquad P_0(n), P_1(n), \ldots, P_M(n)$.

1 Initialize $f_0(n) = b_0(n) = x(n)$

2 $P_0(n) = \beta P_0(n-1) + (1-\beta) \left[f_0(n)^2 + b_0(n-1)^2 \right]$

3 for $m = 1, 2, \ldots, M$ **do**

4 $\qquad f_m(n) = f_{m-1}(n) - \kappa_m(n) b_{m-1}(n-1)$

5 $\qquad b_m(n) = b_{m-1}(n-1) - \kappa_m(n) f_{m-1}(n)$

6 $\qquad \kappa_m(n+1) = \kappa_m(n) + \frac{2\mu_p}{P_{m-1}(n)+\epsilon} \left[f_{m-1}(n) b_m(n) + b_{m-1}(n-1) f_m(n) \right]$

7 $\qquad P_m(n) = \beta P_m(n-1) + (1-\beta)[f_m(n)^2 + b_m(n-1)^2]$

8 $\quad$ **end for**

9 Return $b_M(n)$

3.4 Fractional tap-length LMS algorithm

Using system identification setup, the weight update of the adaptive filter in the FT-LMS algorithm is given by:

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e_{L(n)}^{(L(n))}(n)\mathbf{x}_{L(n)}(n) \tag{3.19}$$

where $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ are the weight update and input vectors respectively, μ is the step size, $L(n)$ is the variable tap-length and $e_{L(n)}^{(L(n))}(n)$ is defined in [3] to be the segmented steady-state error that is calculated by the equation

$$e_G^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:G}^T(n)\mathbf{x}_{L(n);1:G}(n) \tag{3.20}$$

where $1 \leq G \leq L(n)$, $d(n)$ is the desired signal, and $\mathbf{w}_{L(n);1:G}(n)$ and $\mathbf{x}_{L(n);1:G}(n)$ are vectors consisting of the first G elements of the vectors $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ respectively.

By defining $l_f(n)$ as the pseudo fractional tap-length, the update equation of the FT-LMS was proposed in [3] as follows:

$$l_f(n+1) = l_f(n) - \alpha - \gamma[(e_{L(n)}^{(L(n))}(n))^2 - (e_{L(n)-\Delta}^{(L(n))}(n))^2] \tag{3.21}$$

where γ is the step size for the tap-length adaptation, α is a positive leakage parameter and Δ is a positive integer. Then, the updated tap-length, which will be used in the next iteration, is calculated from the fractional tap-length $l_f(n)$ by:

$$L(n+1) = \begin{cases} \lfloor l_f(n) \rfloor & \text{if } |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases} \tag{3.22}$$

where $\lfloor . \rfloor$ is the floor operator and δ is a small integer. When a fixed Δ is employed, the FT algorithm is required find a compromise between convergence speed and the

bias from the optimum tap-length [4].

3.5   Proposed algorithm

Authors in [22] proposed lattice structured variable tap-length algorithm in system identification setup, and for this purpose the direct error signal of joint process estimator was utilized to update the tap-length recursions. Distinctively, in this work, the forward errors $f_0, f_1, \ldots, f_m$ are employed to update the errors $e_{L(n)}^{L(n)}$ and $e_{L(n)-\Delta}^{L(n)}$ in tap-length recursions of (3.20), (3.21) and (3.22) with $f_{L(n)}^{L(n)}$ and $f_{L(n)-\Delta}^{L(n)}$ respectively [14, 23]. Algorithm 3, summarizes the proposed LMS fractional order lattice predictor filter (FO-LPF) algorithm.

Algorithm 3: Proposed fractional order lattice prediction filter (FO-LPF).

Input: $x(n)$, $l(n)$, $L(n)$, $\kappa_1(n), \kappa_2(n)\ldots,\kappa_M(n)$,

$\quad\quad\quad \mathbf{b}(n-1) = [b_0(n-1), b_1(n-1), \ldots, b_M(n-1)]^T$,

$\quad\quad\quad \mathbf{f}(n-1) = [f_0(n-1), f_1(n-1), \ldots, f_M(n-1)]^T$,

$\quad\quad\quad P_0(n-1), P_1(n-1), \ldots, P_M(n-1)$.

Output: $\mathbf{b}_{(}n) = [b_1(n), b_2(n), \ldots, b_M(n)]^T, \mathbf{f}(n) = [f_1(n), f_2(n), \ldots, f_M(n)]^T$,

$\quad\quad\quad \kappa_1(n+1), \kappa_2(n+1)\ldots, \kappa_M(n+1),\ P_0(n), P_1(n), \ldots, P_M(n)$,

$\quad\quad\quad L(n+1)$.

1 Initialize $f_0(n) = b_0(n) = x(n)$, $l(0)$ & $L(0)$

2 for $n = 0, 1, \ldots,$ **do**

3 $\quad f_0(n) = b_0(n) = x(n)$

4 $\quad P_0(n) = \beta P_0(n-1) + (1-\beta)\left[f_0(n)^2 + b_0(n-1)^2\right]$

5 $\quad$ **for** $m = 1, 2, \ldots, M,$ **do**

6 $\quad\quad f_m(n) = f_{m-1}(n) - \kappa_m(n)b_{m-1}(n-1)$

7 $\quad\quad b_m(n) = b_{m-1}(n-1) - \kappa_m(n)f_{m-1}(n)$

8 $\quad\quad \kappa_m(n+1) = \kappa_m(n) + \frac{2\mu_p}{P_{m-1}(n)+\epsilon}\left[f_{m-1}(n)b_m(n) + b_{m-1}(n-1)f_m(n)\right]$

9 $\quad\quad P_m(n) = \beta P_m(n-1) + (1-\beta)[f_m(n)^2 + b_m(n-1)^2]$

10 $\quad$ **end for**

11 $\quad e_{L(n)}^{L(n)}(n) = f_{N-1}\ ;\ e_{L(n)-\Delta}^{L(n)}(n) = f_{N-\Delta-1}$

12 $\quad l_f(n+1) = (l_f(n) - \alpha) - \gamma\left[\left(e_{L(n)}^{(L(n))}\right)^2 - \left(e_{L(n)-\Delta}^{(L(n))}\right)^2\right]$

13 $\quad L(n+1) = \begin{cases} l_f(n) & \text{if} \quad |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases}$

14 $\quad$ **end for**

15 Return $[f_1, f_2, \ldots, f_M]$, $[b_1, b_2, \ldots, b_M]$ & $L(n+1)$

28

3.6   System simulation

In this experiment, a lattice structured forward predictor of Fig. 3.3 is used to predict a narrow band 100 Hz signal $cos(2\pi ft)$ superimposed on white noise input signal using the variable tap-length filter shown in Algorithm 3. A 200 samples snap shot of the input signal, prior to implementing FO-LPF algorithm, is depicted in Fig. 3.8 below.
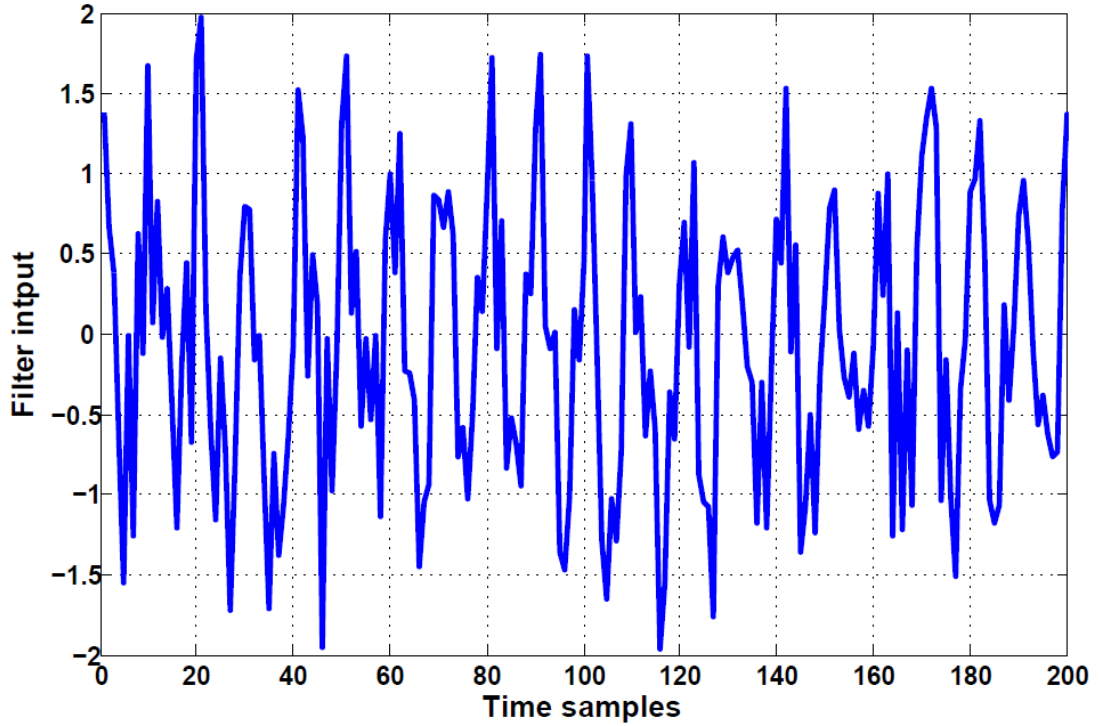
Figure 3.8 – Input signal.

Filter performance is evaluated by applying the input signal to the forward lattice structured predictor that is illustrated in Fig. 3.3 and the frequency response of an equivalent transversal predictor is determined. Theoretically, the predictor's frequency response, after convergence, should pass the narrow band signal of 100 Hz and depress all other frequency components. For this purpose Algorithm 3 is used to compute the frequency response of FO-LPF and the results is compared with different

29

fixed and predetermined filter lengths. Throughout the algorithm's implementation a step-size $\mu_p = 0.003$ was chosen and FO-LPF variable tap-length algorithm's parameters were selected according to [4] as the the following. δ should be a positive integer as small as $1 \leq \delta \leq 10$ [3] [4], the selected value of leakage parameter α is an application dependent, however, [4] considered a choice of α between 0.001 and 0.01 is generally a good choice. The parameter γ controls both convergence speed and fluctuation of the tap-length, a large γ leads to fast convergence of the tap-length but results in large fluctuation, consequently, a trade-off of γ should be considered [4] [3]. Hence, in this experiment parameters' choices are made as follows: $\Delta = 10$, $\alpha = 0.005$, $\delta = 2$ and $\gamma = 1$. 100 independent runs of system simulation were performed and
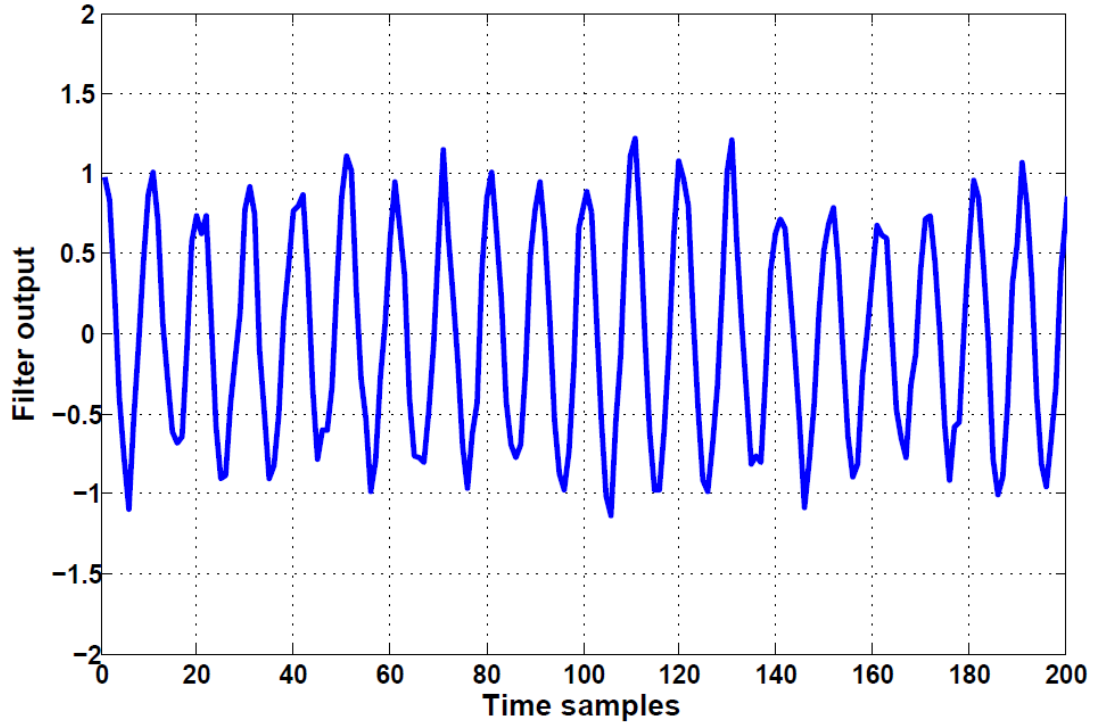
Figure 3.9 – Output signal.

the frequency response was calculated in every run. The results were averaged. A snap shot of the FO-LPF is shown in Fig. 3.9, where it can be seen clearly that the narrow-band signal is recognized using the variable tap-length algorithm. Fig. 3.10

30

shows the simulation results of computing the filter's frequency response of variable tap-length algorithm of FO-LPF against various fixed tap-lengths filters with lengths $L = 5, 10$ and 20 where FO-LPS algorithm manifests best frequency response properties of passing the narrow-band signal of $100Hz$ and attenuating all other frequencies with smoother ripples. This is mainly because of FO-LPF's ability to estimate the optimal filter size (number of taps) while predicting the filter coefficients.
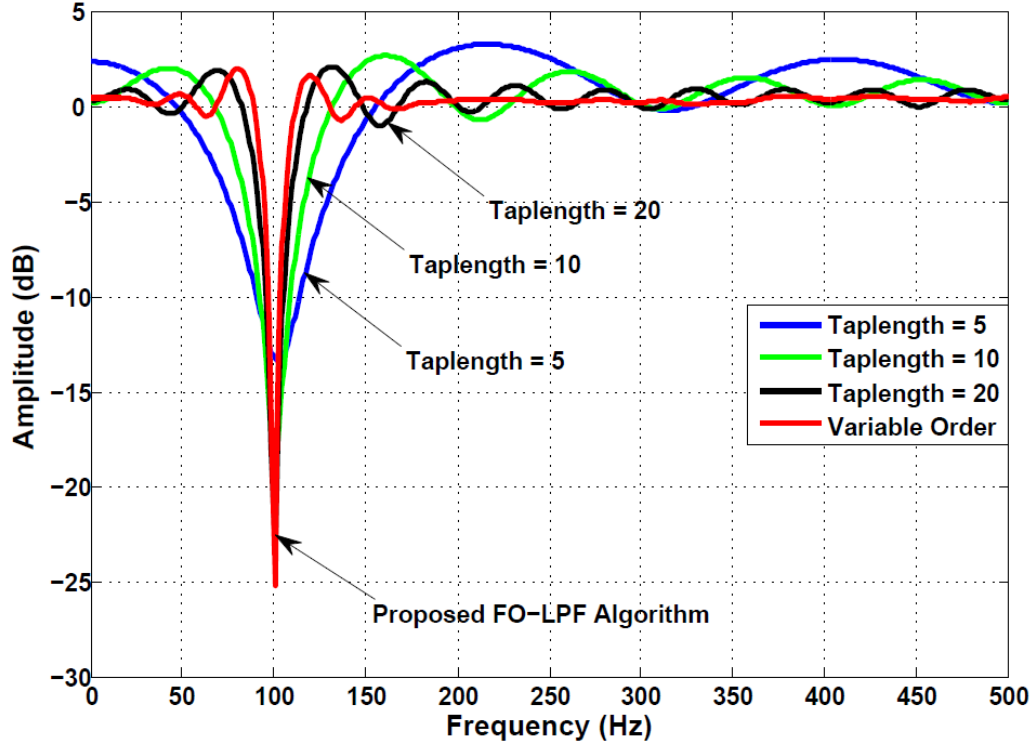
Figure 3.10 – Frequency response of proposed FO-LPF algorithm against different fixed filter lengths.

The expected value of FO-LPF filter tap-length is shown in Fig. 3.11. The simulation results showed an expected tap-length predictor filer of approximately 32 taps. Because FO-LPF algorithm uses a fixed in its length adaptation, the filter algorithm is supposed to have a bias of $\Delta = 10$ [3], consequently, an optimal tap-length of about 22 taps, which can be conceived from Fig. 3.10, as we notice the fixed filter of 20 taps is closer to the FO-LPF variable tap-length algorithm frequency

response characteristics.
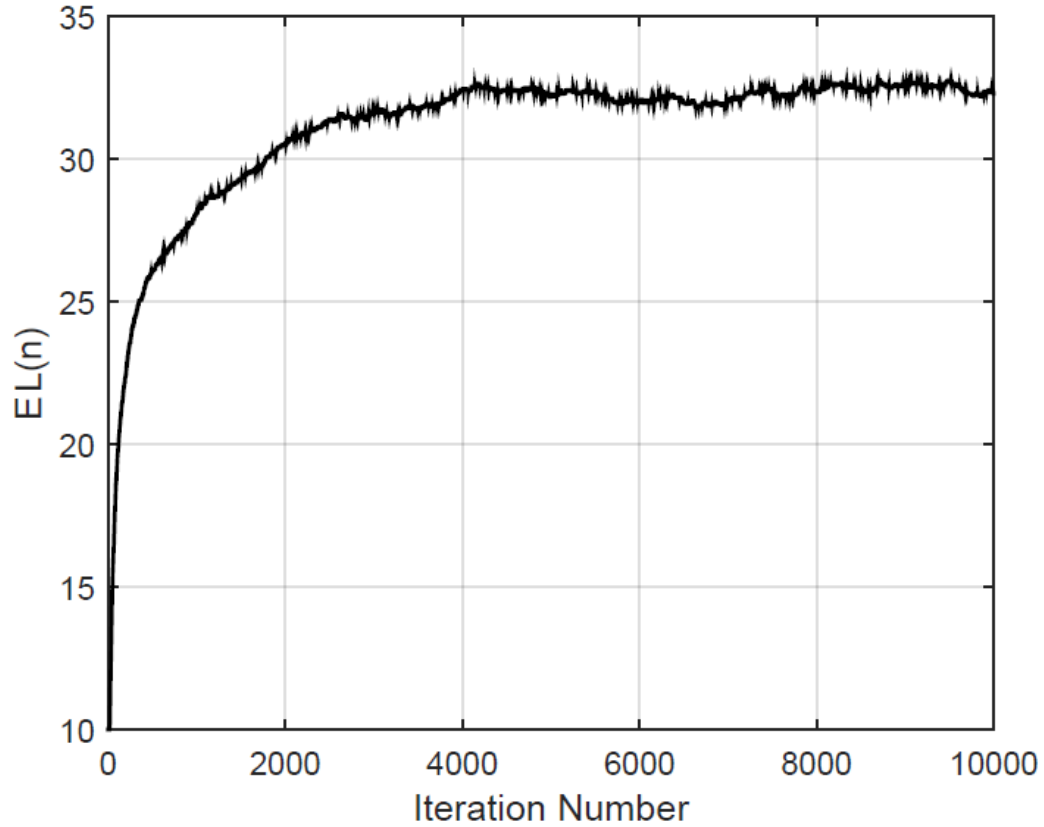
Figure 3.11 – Tap-length's expected value of FO-LPF algorithm.

3.7 Conclusion

A new variable tap-length algorithm for lattice structured adaptive predictor was proposed. This algorithms utilizes the forward residual errors to find the optimal tap-length adaptively. Simulation results of the filter frequency response showed that the proposed algorithm can predict the narrow-band message signal efficiently even in the presence of noise. The suggested algorithm's frequency response showed superior characteristics when compared with predetermined lengths of lattice predictors.

References

[1] Y. Gu, K. Tang, H. Cui, and W. Du, "Convergence analysis of a deficient-length lms filter and optimal-length sequence to model exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 10, no. 1, pp. 4–7, Jan. 2003.

[2] K. Mayyas, "Performance analysis of the deficient length lms adaptive algorithm," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2727–2734, Aug. 2005.

[3] Y. Gong and C. F. N. Cowan, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[4] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, Feb. 2008.

[5] Y. Gong and C. F. N. Cowan, "Structure adaptation of linear mmse adaptive filters," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 151, no. 4, pp. 271–277, Aug. 2004.

[6] F. Riera-Palou, J. M. Noras, and D. G. M. Cruickshank, "Linear equalisers, with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, pp. 1553–1554, Dec 2001.

[7] Y. Gu, K. Tang, and H. Cui, "Lms algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, March 2004.

[8] P. Nathen, M. Haussmann, M. Krause, and N. Adams, "Adaptive filtering for the simulation of turbulent flows with lattice boltzmann methods," *Computers Fluids*, vol. 172, pp. 510 – 523, 2018.

[9] H. Yeh and C. Rangel-Ruiz, "Fixed-point implementation of cascaded forward-backward adaptive predictors," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 103–107, Jan. 2012.

[10] M. T. Akhtar and A. Nishihara, "Lattice adaptive filtering-based method for acoustic feedback cancellation in hearing aids with robustness against sudden changes in the feedback path," in *2017 Eighth International Workshop on Signal Design and Its Applications in Communications (IWSDA)*, Sep. 2017, pp. 59–63.

[11] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications, Second Edition.* John Wiley and Sons, 4 2013.

[12] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation.* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

[13] J. Bird, M. Santer, and J. Morrison, "Compliant kagome lattice structures for generating in-plane waveforms," *International Journal of Solids and Structures*, vol. 141-142, pp. 86 – 101, 2018.

[14] D. Buneag and D. Piciu, "A new approach for classification of filters in residuated lattices," *Fuzzy Sets and Systems*, vol. 260, pp. 121 – 130, 2015, theme: Algebraic Structures.

[15] S. Haykin *et al.*, "Adaptive filtering theory," *Englewood Cliffs, NJ: Prentice-Hall*, 1996.

[16] https://www.dspalgorithms.com.

[17] H. Yeh and C. Rangel-Ruiz, "Fixed-point implementation of cascaded forward-backward adaptive predictors," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 103–107, Jan. 2012.

34

[18] C. Paleologu, S. Ciochina, A. A. Enescu, and C. Vladeanu, "Gradient adaptive lattice algorithm suitable for fixed point implementation," in *2008 The Third International Conference on Digital Telecommunications (icdt 2008)*, June 2008, pp. 41–46.

[19] N. Levinson, "The wiener (root mean square) error criterion in filter design and prediction," *Journal of Mathematics and Physics*, vol. 25, no. 1-4, pp. 261–278, 1946.

[20] J. Markel and A. Gray, *Linear Prediction of Speech*, ser. Communication and Cybernetics. Springer Berlin Heidelberg, 2013.

[21] C. N. Papaodysseus, E. B. Koukoutsis, and C. N. Triantafyllou, "Error sources and error propagation in the levinson-durbin algorithm," *IEEE Transactions on Signal Processing*, vol. 41, no. 4, pp. 1635–1651, April 1993.

[22] S. Alsaid, E. Abdel-Raheem, M. Khalid, and K. Mayyas, "Variable tap-length algorithm using lattice structured lms adaptive filters," in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2017, pp. 1–5.

[23] P. He, X. Xin, and J. Zhan, "On derivations and their fixed point sets in residuated lattices," *Fuzzy Sets and Systems*, vol. 303, pp. 97–113, 2016, theme: Algebra.

Chapter 4

Variable Tap-length LMS Adaptive Lattice Filters Applied to System Identification

4.1 Introduction

Tap-Length is one of the important parameters that significantly affects the performance of adaptive filters. An overestimated tap-length increases the filter computational complexity and reduces the convergence rate, while underestimating it leads to an extra steady state Mean-Square Error (MSE). Most applications assume some predefined tap-length, which does not necessarily provide optimal adaptation results. Therefore, adjusting the filter tap-length to reach the optimal filter length has gained more attention in recent years [1], [2]. The variable length algorithm should converge fast to the optimum tap-length with small steady state fluctuations.

Most popular variable length adaptive algorithms are the segmented filter (SF) algorithm [3], the gradient descent (GD) algorithm [4], and the fractional tap-length (FT) algorithm [1, 2, 5]. In the SF algorithm, the filter is subdivided into k segments, each with fixed Δ coefficients. Then, based on the difference between accumulated squared errors from the last two segments, the tap-length of the filter is modified by adding or subtracting one segment [3]. The GD algorithm does not divide the filter into segments nor does it constraint the tap-length step-size update to Δ [4]. Thus, the GD algorithm is more flexible than SF algorithm. However, the GD suffers from "wandering" problem, where the adaptive filter tap-length keeps hovering in a range

larger than the optimum tap-length [5].

The FT algorithm has less computational complexity and better convergence performance than the previously mentioned algorithms. It uses the fractional tap-length during the instantaneous tap-length update, and the integer value of the fractional tap-length remains unchanged until the increment of the fractional tap-length accumulates to a certain extent compared with the integer value [5], [6].

This paper is organized as follows. In Section 4.2 the Fractional Tap-Length FT-LMS algorithm is described within a system identification model and using FIR filter structure. Lattice structure will be introduced in Section 4.3. In Section 4.4, a new Fractional Tap-Length lattice structured LMS algorithm (FT-LLMS) is presented. Simulation results will be shown in Section 4.5 to verify the system performance. Finally, Section 4.6 will conclude the paper.

4.2   FT-LMS algorithm

In the context of transversal filter (FIR) system identification model, the role of FT-LMS algorithm is to identify the unknown filter coefficients as well as the tap-length L_{opt} of the unknown filter $\mathbf{w}$. The desired signal $d(n)$ is represented as in [7]

$$d(n) = \mathbf{x}_{L_{opt}}^{T}(n)\mathbf{w}_{L_{opt}} + z(n) \tag{4.1}$$

where $\mathbf{x}_{L_{opt}}(n)$ is the input vector given by $[x(n), x(n-1) \ldots x(n-L_{opt}+1)]^{T}$, $\mathbf{w}_{L_{opt}}(n)$ $= [w_1(n), w_2(n) \ldots w_{Lopt}(n)]^{T}$, is the optimum coefficient vector, L_{opt} is the optimum tap-length and $z(n)$ is a stationary zero-mean uncorrelated noise signal that is independent of $x(n)$.

In the FT-LMS algorithm, the weight update recursion of the adaptive filter is given by

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e_{L(n)}^{(L(n))}(n)\mathbf{x}_{L(n)}(n) \tag{4.2}$$

37

where $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ are the weight update and input vectors respectively, μ is the step size, $L(n)$ is the variable tap-length and $e_{L(n)}^{(L(n))}(n)$ is defined in [5] to be the segmented steady-state error that is calculated by the equation

$$e_M^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:M}^T(n)\mathbf{x}_{L(n);1:M}(n) \tag{4.3}$$

where $1 \leq M \leq L(n)$, and $\mathbf{w}_{L(n);1:M}(n)$ and $\mathbf{x}_{L(n);1:M}(n)$ are vectors consisting of the first M elements of the vectors $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$, respectively.

By defining $l_f(n)$ as the pseudo fractional tap-length, the update equation of the FT-LMS was proposed in [5] as follows:

$$l_f(n+1) = l_f(n) - \alpha - \gamma[(e_{L(n)}^{(L(n))}(n))^2 - (e_{L(n)-\Delta}^{(L(n))}(n))^2] \tag{4.4}$$

where γ is the step size for the tap-length adaptation, α is a positive leakage parameter and Δ is a positive integer.

Then, the updated tap-length, which will be used in the next iteration, is calculated from the fractional tap-length $l_f(n)$ by:

$$L(n+1) = \begin{cases} \lfloor l_f(n) \rfloor & \text{if } |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases} \tag{4.5}$$

where $\lfloor . \rfloor$ is the floor operator and δ is a small integer. When a fixed Δ is employed, the FT algorithm is required find a compromise between convergence speed and the bias from the optimum tap-length [1, 8].

4.3 Lattice structure

Adaptive system identification filter can be realized using lattice form, which is formulated around the lattice basic structure shown in Fig. 4.1. The input-output relation

38

of such a building block is characterized by a single parameter, known as, the Partial
Correlation (PARCOR) Coefficient $\kappa_m(n)$. The order-update equations for forward
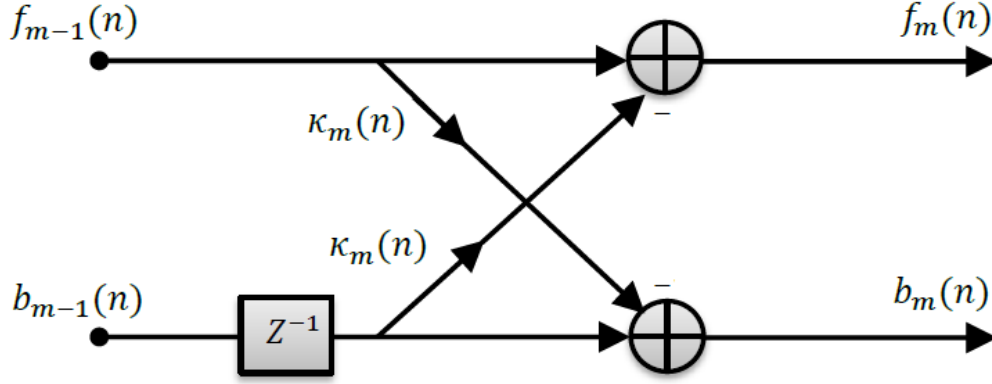
Figure 4.1 – Lattice building block [9].

and backward prediction errors are recursively specified by [9–11]

$$f_m(n) = f_{m-1}(n) - \kappa_m(n)b_{m-1}(n-1) \tag{4.6}$$

$$b_m(n) = b_{m-1}(n-1) - \kappa_m(n)f_{m-1}(n) \tag{4.7}$$

where $m = 1, 2, \ldots, M$, $\kappa_m(n)$ is the partial coefficient at the mth stage and time n
and both of $f_m(n)$ and $b_m(n)$ are forward and backward prediction errors respectively.
To initialize the adaptation, the zeroth-order forward and backward perdition errors
are set as

$$f_0(n) = b_0(n) = x(n) \tag{4.8}$$

The optimum (PARCOR) coefficient κ_m of the mth stage of lattice predictor is ob-
tained by minimizing the following cost function:

$$\xi_m = \mathbf{E}\left[f_m^2(n) + b_m^2(n)\right] \tag{4.9}$$

The LMS algorithm for minimizing this cost function is implemented according to the following recursion

$$\kappa_m(n+1) = \kappa_m(n) - \mu_m(n)\frac{\partial \xi_m(n)}{\partial \kappa_m} \tag{4.10}$$

where $\mu_m(n)$ is the step-size. An estimate of the cost function ξ_m, based on the most recent samples of the forward and backward prediction errors, is given by

$$\hat{\xi}_m = f_m^2(n) + b_m^2(n) \tag{4.11}$$

Substituting (4.11) in (4.10) and using (4.6) and (4.7), yields

$$\kappa_m(n+1) = \kappa_m(n) + 2\mu_m(n).\left[f_m(n)b_{m-1}(n-1) + b_m(n)f_{m-1}(n)\right] \tag{4.12}$$

A faster convergence can be obtained by normalizing the step-size $\mu_{p,m}(n)$ by the signal power at the mth stage of the predictor, which is estimated by the recursion

$$P_{m-1}(n) = \beta P_{m-1}(n-1) + (1-\beta)(f_{m-1}^2(n) + b_{m-1}^2(n-1)) \tag{4.13}$$

Hence, the normalized step-size is given by

$$\mu_m(n) = \frac{\mu_p}{P_{m-1}(n) + \epsilon} \tag{4.14}$$

where μ_p is the constant step-size and ϵ is a small positive constant to prevent algorithm instability. The LMS algorithm can then be used to adapt w_i coefficient of the
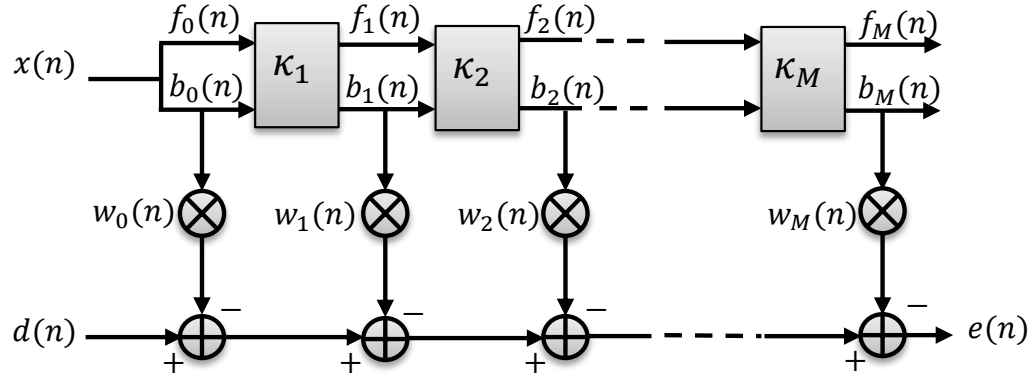
Figure 4.2 – Lattice joint process estimator [12].

linear combiner in the lattice joint process estimator in Fig. 4.2 as [12–14]

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu_{\mathbf{c}}(n)e(n)\mathbf{b}(n) \tag{4.15}$$

$$e(n) = d(n) - \mathbf{w}(n)^T \mathbf{b}(n) \tag{4.16}$$

$$\mu_c(n) = \frac{\mu}{P_m(n) + \epsilon} \quad , \text{for} \quad m = 0, 1, \ldots, M \tag{4.17}$$

where μ is the unnormalized step-size that is different from μ_p and the vector $\mathbf{b}(n) = [b_0(n)b_1(n)\ldots b_{N-1}(n)]^T$ is backward prediction error . Algorithm 4 summarizes lattice LMS joint processor estimator.

Algorithm 4: Lattice LMS algorithm for adaptive joint process estimator.

Input: $x(n)$, $d(n)$, $\kappa_1(n)$, $\kappa_2(n)\ldots,\kappa_M(n)$,

$\mathbf{w}(n-1) = [w_0(n), w_1(n), \ldots, w_M(n)]^T$,

$\mathbf{b}(n-1) = [b_0(n-1), b_1(n-1), \ldots, b_M(n-1)]^T$,

$P_0(n-1), P_1(n-1), \ldots, P_M(n-1)$.

Output: $\kappa_1(n+1), \kappa_2(n+1)\ldots,\kappa_M(n+1)$,

$\mathbf{w}(n-1) = [w_0(n), w_1(n), \ldots, w_M(n)]^T$,

$\mathbf{b}(n) = [b_0(n), b_1(n), \ldots, b_M(n)]^T$, $P_0(n), P_1(n), \ldots, P_M(n)$.

1 **for** $n = 0, 1, \ldots,$ **do**

2   Initialize $f_0(n) = b_0(n) = x(n)$,

      $P_0(n) = \beta P_0(n-1) + (1-\beta)\left[f_0(n)^2 + b_0(n-1)^2\right]$

3   **for** $m = 1, 2\ldots, M,$ **do**

4        $f_m(n) = f_{m-1}(n) - \kappa_m(n)b_{m-1}(n-1)$

5        $b_m(n) = b_{m-1}(n-1) - \kappa_m(n)f_{m-1}(n)$

6        $\kappa_m(n+1) = \kappa_m(n) + \frac{2\mu_p}{P_{m-1}(n)+\epsilon}\left[f_{m-1}(n)b_m(n) + b_{m-1}(n-1)f_m(n)\right]$

7        $P_m(n) = \beta P_m(n-1) + (1-\beta)[f_m(n)^2 + b_m(n-1)^2]$

8   **end for**

9     $y(n) = \mathbf{w}^T(n)\mathbf{b}(n)$

10    $e(n) = d(n) - y(n)$

11    $\boldsymbol{\mu_c}(n) = \mu\,\mathrm{diag}((P_0(n)+\epsilon)^{-1}..(P_{N-1}(n)+\epsilon)^{-1})$

12    $\mathbf{w}(n+1) = \mathbf{w}(n) + 2\boldsymbol{\mu}_c(n)e(n)\mathbf{b}(n)$

13 **end for**

4.4 Proposed algorithm

The proposed algorithm brings together the Fractional-Tap (FT) strategy and LMS lattice structured filter, to form a new fractional-tap lattice LMS (FT-LLMS) filter. Algorithm 5 outlines the proposed FT-LLMS filter. The computational complexity of the proposed FTLLMS filter and a comparison with its counterpart standard lattice LMS (LLMS) is shown in Table 4.1.

Table 4.1 – Computational complexity for LLMS and FTLLMS filters.

Operation	Addition	Multiplication	Division
LLMS Filter	$9L$	$14L$	L
FTLLMS Filter	$10L + 3$	$15L + 3$	L

The standard lattice LMS filter requires $8L$ additions, $12L$ multiplications and L divisions, where L is the steady-state tap-length of the algorithm. The FT lattice LMS, on the other hand, adds $L+3$ additions and $L+3$ multiplications to the standard algorithm [15], [16].

Algorithm 5: Proposed fractional tap lattice LMS (FTLLMS) algorithm.

Input: $x(n)$, $d(n)$, $l(n)$, $L(n)$ $\kappa_1(n), \kappa_2(n)\ldots, \kappa_M(n)$,

$\qquad \mathbf{w}(n-1) = [w_0(n), w_1(n), \ldots, w_M(n)]^T$,

$\qquad \mathbf{b}(n-1) = [b_0(n-1), b_1(n-1), \ldots, b_M(n-1)]^T$,

$\qquad P_0(n-1), P_1(n-1), \ldots, P_M(n-1)$.

Output: $L(n+1)$, $\mathbf{b}(n) = [b_0(n), b_1(n), \ldots, b_M(n)]^T$,

$\qquad \kappa_1(n+1), \kappa_2(n+1)\ldots, \kappa_M(n+1)$,

$\qquad \mathbf{w}(n+1) = [w_0(n+1), w_1(n+1), \ldots, w_M(n+1)]^T$,

$\qquad P_0(n), P_1(n), \ldots, P_M(n)$.

1 **for** $n = 0, 1, \ldots,$ **do**

2 $\quad$ Initialize $L(0), l(0), f_0(n) = b_0(n) = x(n)$,
$\qquad P_0(n) = \beta P_0(n-1) + (1-\beta)\left[f_0(n)^2 + b_0(n-1)^2\right]$

3 $\quad$ **for** $m = 1, 2 \ldots, M,$ **do**

4 $\qquad f_m(n) = f_{m-1}(n) - \kappa_m(n)b_{m-1}(n-1)$

5 $\qquad b_m(n) = b_{m-1}(n-1) - \kappa_m(n)f_{m-1}(n)$

6 $\qquad \kappa_m(n+1) = \kappa_m(n) + \frac{2\mu_p}{P_{m-1}(n)+\epsilon}\left[f_{m-1}(n)b_m(n) + b_{m-1}(n-1)f_m(n)\right]$

7 $\qquad P_m(n) = \beta P_m(n-1) + (1-\beta)\left[f_m(n)^2 + b_m(n-1)^2\right]$

8 $\quad$ **end for**

9 $\quad e_{L(n)}^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:L(n)}^T(n)\mathbf{b}_{L(n);1:L(n)}(n)$

10 $\quad e_{L(n)-\Delta}^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:L(n)-\Delta}^T(n)\mathbf{b}_{L(n);1:L(n)-\Delta}(n)$

11 $\quad \boldsymbol{\mu_c}(n) = \mu \, \text{diag}\left((P_0(n)+\epsilon)^{-1}, \ldots, (P_{N-1}(n)+\epsilon)^{-1}\right)$

12 $\quad \mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + 2\boldsymbol{\mu}_c(n)e_{L(n)}(n)\mathbf{b}_{L(n)}(n)$

13 $\quad l_f(n+1) = (l_f(n) - \alpha) - \gamma\left[\left(e_{L(n)}^{(L(n))}\right)^2 - \left(e_{L(n)-\Delta}^{(L(n))}\right)^2\right]$

14 $\quad L(n+1) = \begin{cases} l_f(n) & \text{if} \quad |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases}$

15 $\quad$ **end for**

4.5 System simulations

4.5.1 Lattice structure vs. direct form structure

The main objective of this experiment is to identify an unknown system characterized by the impulse response shown in Fig. 4.3, which is a car cabin truncated to 100 samples, using FIR structured and lattice structured LMS adaptive filters. The experimental setup is similar to those in [1] and [2], where the input signal is white Gaussian with zero-mean and unity variance and the results are obtained by averaging multiple independent runs of the simulation.
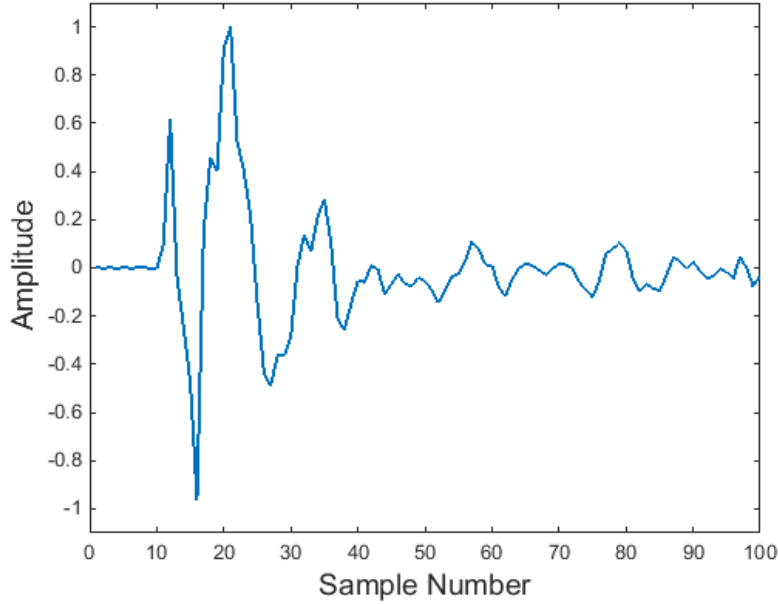
Figure 4.3 – Car cabin impulse response of length 100 samples.

Filter length in both structures are fixed to ($L_{opt} = 100$), the step-size for FIR LMS adaptive filter and and the linear combiner part of lattice structure LMS adaptive filter is ($\mu = 0.0005$), and the step-size for the predictor part of lattice structure is ($\mu_p = 10^{-7}$). Simulation results for Mean Square Error (MSE) are shown in Fig. 4.4 and Fig. 4.5 which are related to low SNR of 10 dB and high SNR of 30 dB

respectively. In both cases the two adaptive filter structures show approximately same error level, nevertheless, lattice structured adaptive filter has a faster convergence rate.
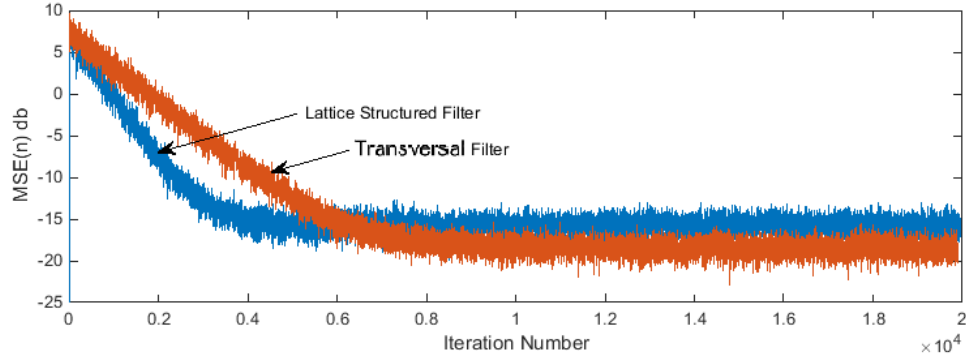
Figure 4.4 – MSE(n) for lattice and direct form filter, SNR = 10 dB.
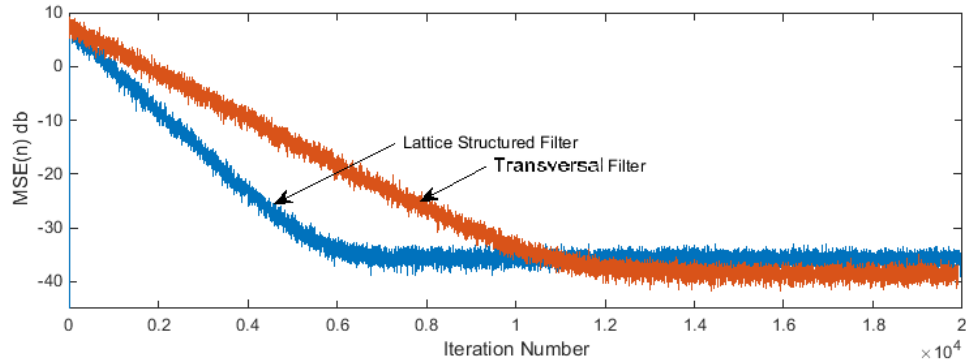
Figure 4.5 – MSE(n) for lattice and direct form filter, SNR = 30 dB.

4.5.2 Proposed algorithm simulations

Tracking capability of FTLLMS algorithm

Here, the proposed FTLLMS algorithm in Section 4.4 is used to identify an unknown system that consists of two parts, the first impulse response of it is shown in Fig. 4.3 and then an abrupt change occurs to the examined system's impulse response at

approximately half of the iteration period where all coefficients are multiplied by -1 and the length of the new unknown system becomes 200 samples, as shown in Fig. 4.6. Algorithm parameters are chosen according to [8] as follows below. The value δ should be a small positive integer, $1 \leq \delta \leq 10$ [5] [8] [2], therefore, it is selected as $\delta = 2$ in this paper. The selected value of leakage parameter α is an application dependent, however, it was stated in [8] that values of α between 0.001 and 0.01 would lead to proper performance, hence, $\alpha = 0.005$ is selected in simulation. The parameter γ controls both convergence speed and fluctuation of the tap-length, a large γ leads to fast convergence of the tap-length but results in large fluctuation, consequently, a trade-off choice of $\gamma = 1$ is selected. Simulation results are obtained by averaging 100 independent runs which provided the results shown in Fig. 4.7 for high SNR of 30 dB and in Fig. 4.8 for low SNR of 10 dB.
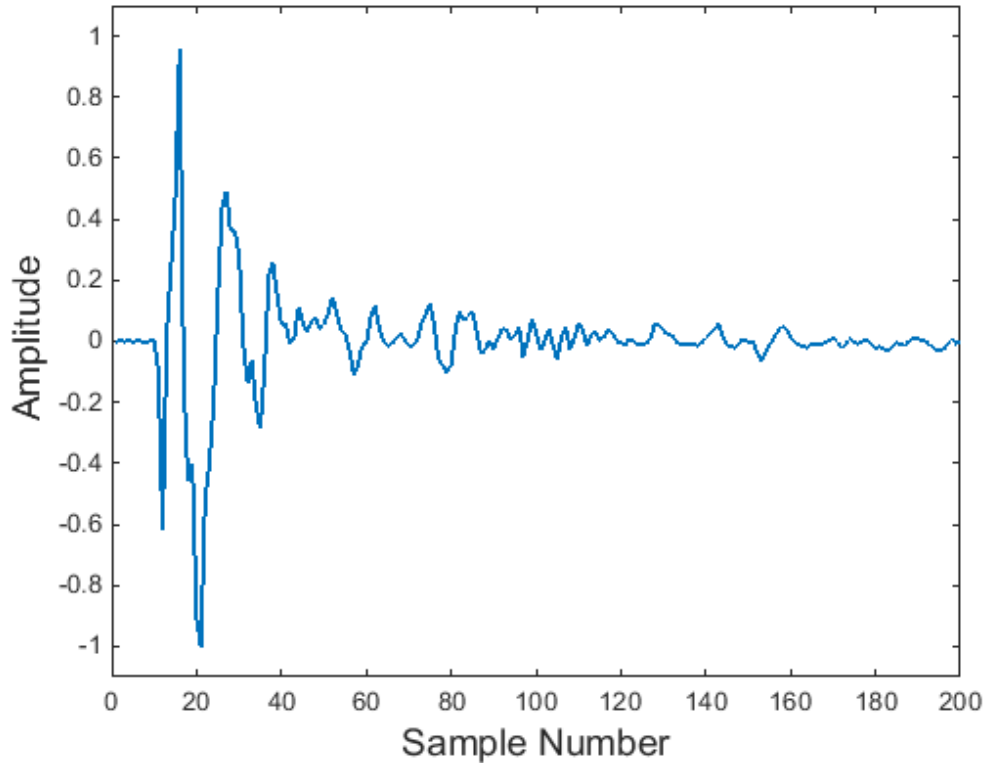
Figure 4.6 – Car cabin impulse response of length 200 samples.
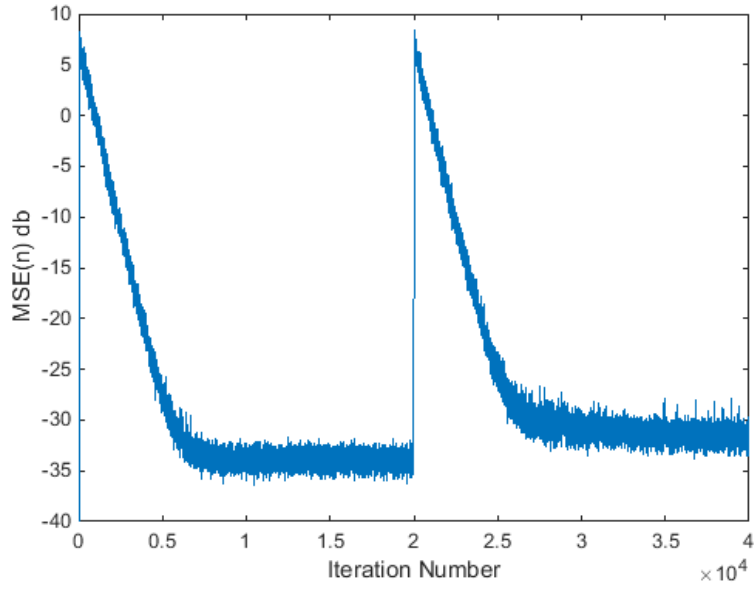
Figure 4.7 – MSE(n) for a combined impulse response system of Fig. 4.3 and Fig. 4.6, (SNR = 30 dB).
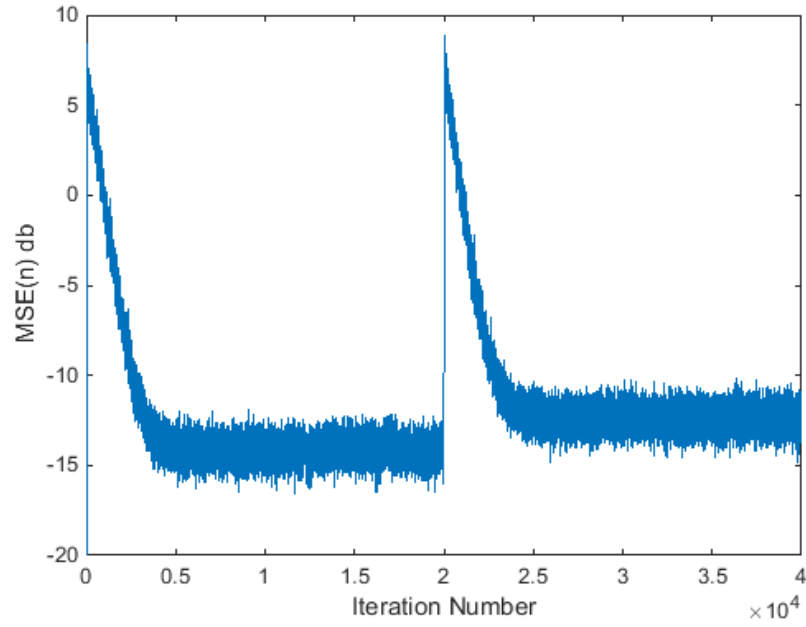
Figure 4.8 – MSE(n) for a combined impulse response system of Fig. 4.3 and Fig .4.6, (SNR = 10 dB).

48

Figure 4.7 and Fig. 4.8 show the FTLLMS algorithm's ability to identify the unknown system in presence of an abrupt change in its impulse response, in low and high SNR environments. The filter length expected value of the FTLLMS algorithm, in SNR = 30 dB and SNR = 10 dB environments are shown in Fig. 4.9 and Fig. 4.10 respectively. Because FTLLMS algorithm uses a fixed Δ, the filter length of FTLLMS algorithm a bias of $\Delta = 50$ [5], this bias is more noticeable in higher SNR case.
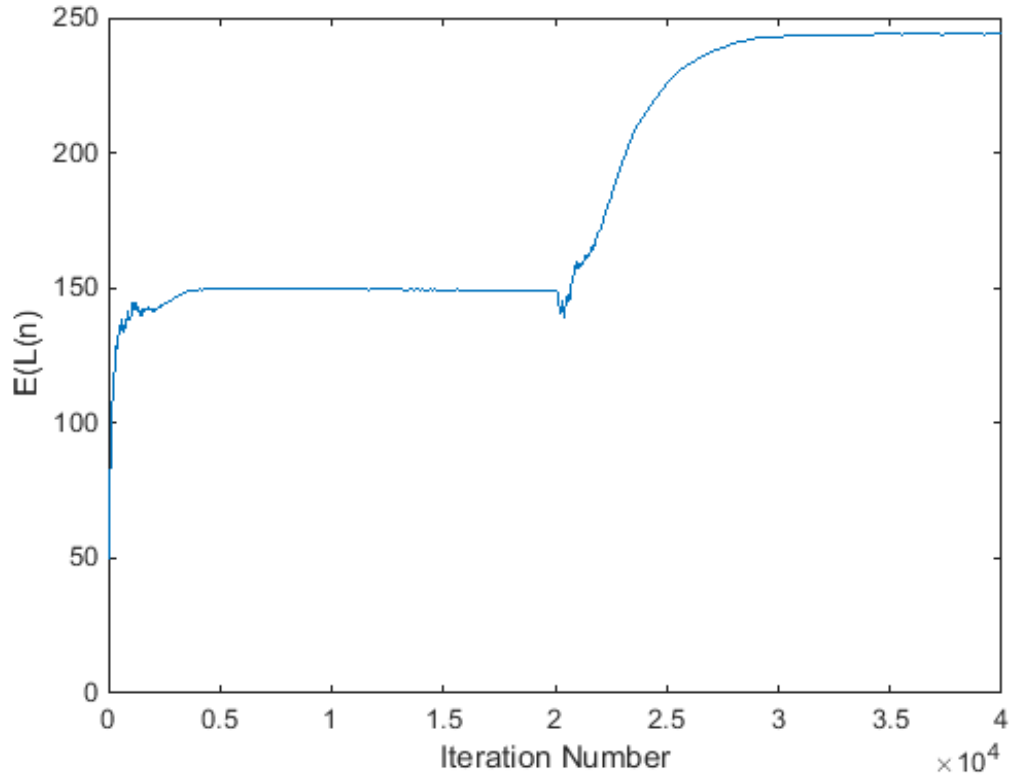
Figure 4.9 – The expected value of the proposed algorithm (FTLLM) tap-Length $\mathbf{E}(L(n))$, (SNR= 30 dB).
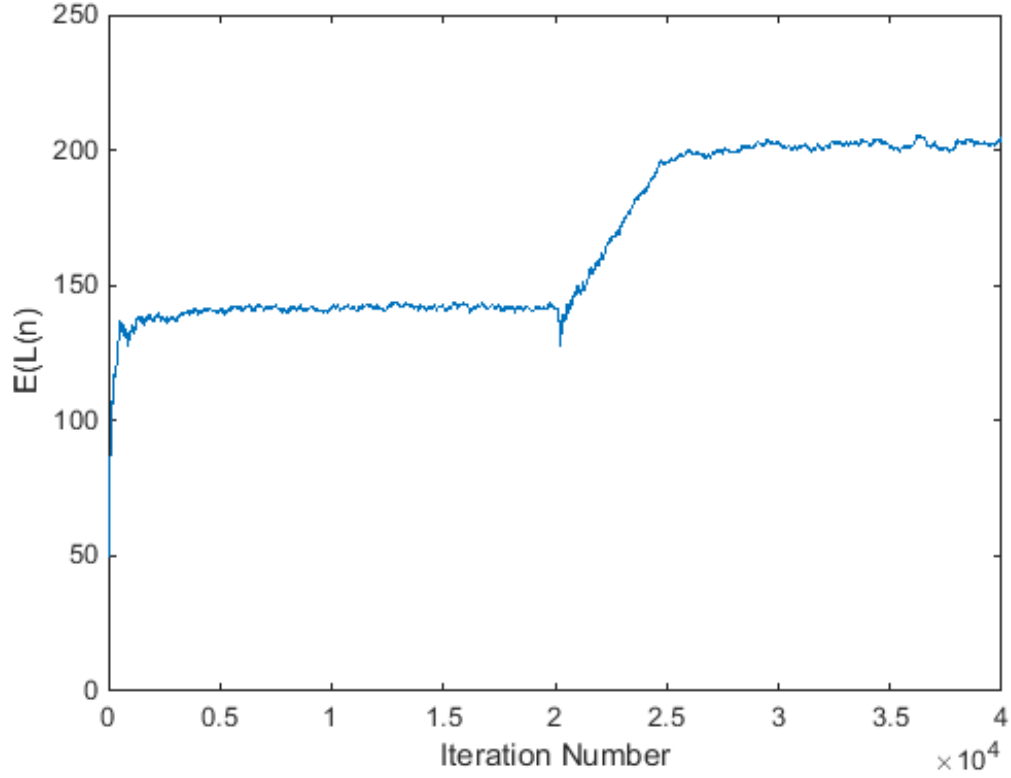
Figure 4.10 – The expected value of the proposed algorithm (FTLLM) tap-Length $\mathbf{E}(L(n))$, (SNR= 10 dB).

Comparison with fixed lengths lattice LMS filters

Using same algorithm parameters to that employed in the previous simulations, FTLLMS algorithm is compared to standard Lattice LMS with lengths $L = 60, 120$, and 180, to identify the system shown in Fig. 4.6. Over 100 Monte Carlo trials of the same experiment using SNR=30 and SNR=10 were performed and averaged. In high SNR environment Fig. 4.11, the FTLLMS showed best error properties and convergence rate, while in low SNR environment Fig. 4.12, the proposed variable tap-length FTLLMS and a fixed tap-length of near to optimum ($L = 180$) tap-length showed close results.
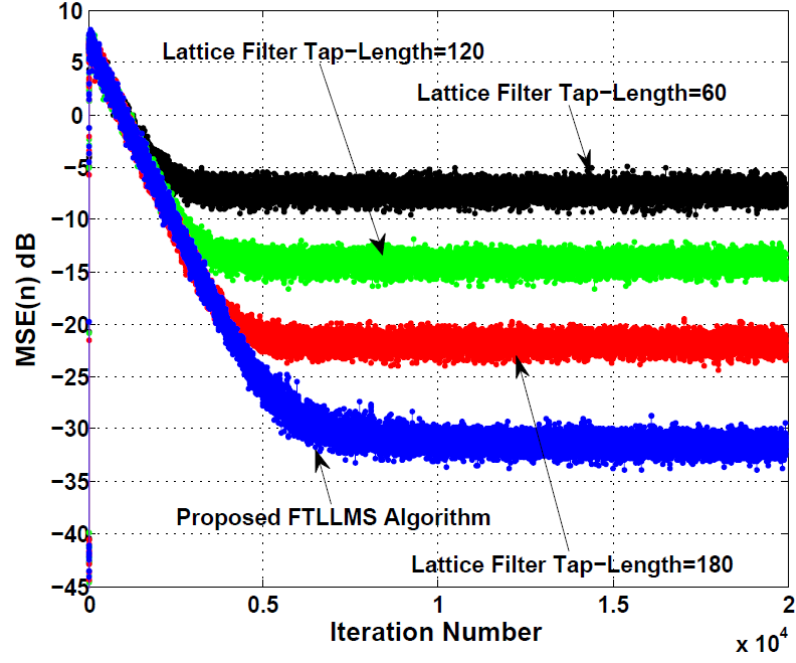
Figure 4.11 – MSE(n) of proposed FTLLMS and different lattice LMS filter lengths (SNR= 30 dB).
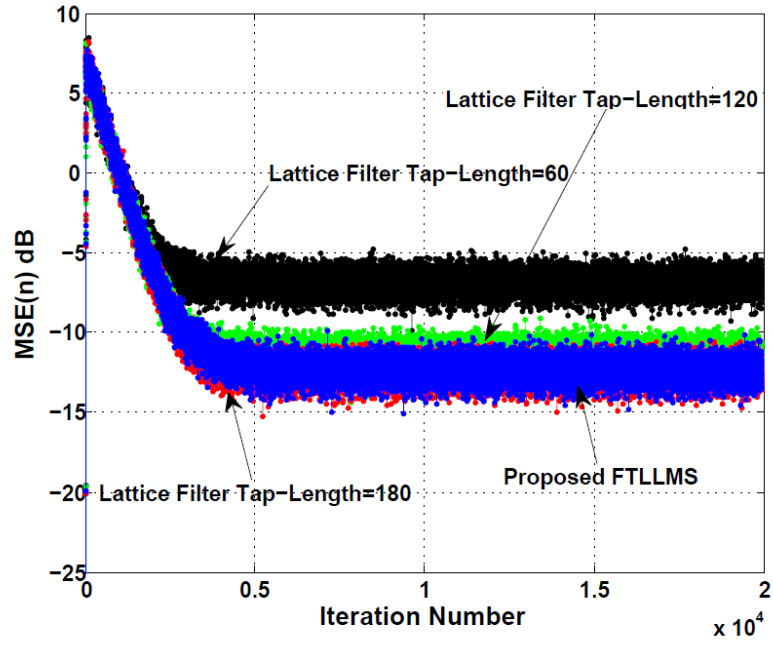
Figure 4.12 – MSE(n) of proposed FTLLMS and different lattice LMS filter lengths (SNR= 10 dB).

51

4.6 Conclusion

A novel variable tap-length algorithm that uses lattice structure adaptive filter was proposed in a system identification setup. Simulation results showed that the new algorithm is capable of identifying unknown systems even in the presence of sudden change in the length of the unknown impulse response. Improved convergence rate and error properties of the proposed algorithm were also shown by simulations as compared with the fixed length Lattice adaptive LMS filters.

References

[1] K. Mayyas and H. A. AbuSeba, "A new variable length nlms adaptive algorithm," *Digital Signal Processing*, vol. 34, pp. 82–91, 2014.

[2] N. Li, Y. Zhang, Y. Zhao, and Y. Hao, "An improved variable tap-length lms algorithm," *Signal Processing*, vol. 89, no. 5, pp. 908–912, 2009.

[3] F. Riera-Palou, J. M. Noras, and D. Cruickshank, "Linear equalisers with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, p. 1, 2001.

[4] Y. Gu, K. Tang, and H. Cui, "Lms algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, 2004.

[5] Y. Gong and C. F. N. Cowan, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[6] Y. Gong and C. Cowan, "Structure adaptation of linear mmse adaptive filters," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 151, no. 4, pp. 271–277, 2004.

[7] S. Haykin *et al.*, "Adaptive filtering theory," *Englewood Cliffs, NJ: Prentice-Hall*, 1996.

[8] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, 2008.

[9] P. Diniz, "Adaptive filtering: Algorithms and practical implementation. springer," *New York, NY, USA*, 2008.

[10] T. Togami and Y. Iiguni, "Convergence analysis of the adaptive lattice filter for a mixed gaussian input sequence," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 151, no. 5, pp. 428–433, Oct. 2004.

[11] B. Friedlander, "Lattice filters for adaptive processing," *Proceedings of the IEEE*, vol. 70, no. 8, pp. 829–867, Aug. 1982.

[12] B. Farhang-Boroujeny, *Adaptive filters: theory and applications.* John Wiley & Sons, 2013.

[13] M. Honig, "Convergence models for lattice joint process estimators and least squares algorithms," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 2, pp. 415–425, April 1983.

[14] H. F. Lee, J. T. Yuan, and T. C. Lin, "Iir lattice-based blind equalization algorithms," in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, Sep. 2012, pp. 1–5.

[15] Y. Sun, "Adaptive echo cancellation analysis," *Ann Arbor*, vol. 1050, pp. 48 106–1346.

[16] C. Paleologu, S. Ciochina, A. A. Enescu, and C. Vladeanu, "Gradient adaptive lattice algorithm suitable for fixed point implementation," in *Digital Telecommunications, 2008. ICDT'08. The Third International Conference on.* IEEE, 2008, pp. 41–46.

Chapter 5

Variable Tap-length RLS Adaptive Lattice Filters Applied to System Identification

5.1 Introduction

Although standard recursive least squares (RLS) algorithms converge faster than their least mean squares (LMS) counterparts, this advantage comes with a burden on the filter size [1]. The computational complexity of RLS algorithm grows proportional to the square of the filter length [2, 3]. For long filter tap-length, this can become costly and hence unacceptable. There have been several attempts in the literature to solve this drawback of RLS filters. Those solutions, whose computational complexity grows proportional to the length of the filter, are commonly referred to as fast RLS algorithms [2, 4]. All the fast RLS algorithms employ lattice order-update and time-update equations [5–7], which means that those algorithms combine the concept of prediction and filtering to form RLS computationally efficient implementation. Most popular filters that use this efficient employment are known as Fast Transversal RLS filter (FTF) and lattice RLS filter (LRLS) [4]. The main advantage of the FTF algorithm is its reduced computational complexity as compared with other available RLS solutions. However, in the case of FTF filters this significant reduction in the complexity comes with high sensitivity to roundoff error accumulation which renders the algorithm to be instable [2, 4]. LRLS filter leads to more robust implementation [8], and consequently have been utilized in a variety of signal and image processing applications such as linear prediction [4,9], noise cancellation [5], system identification

[10, 11] and channel equalization [7, 12]. Variable tap-length algorithm is a technique used to search for the filter optimal structure [13–15]. Among a variety of algorithms, the fractional tap-length (FT) [16, 17] filter is a recommended strategy due to its improved convergence performance and efficiency. Moreover, the FT algorithm resolve the problems that other variable tap-length algorithms suffer from such as inflexibility and wondering issues [18, 19]. In this work, Section 5.2 will first introduce the concept of LRLS joint process estimator used for system identification. Section 5.3 will then describe the algorithm utilized in this work as a method to search for the *optimum* tap-length, namely fractional tap-length FT algorithm. In Section 5.4, the FT algorithm is incorporated in LRLS joint process estimator to form the new proposed lattice filter. System simulations that verify the analysis of the proposed algorithm are going to be presented in Section 5.5. Finally, Section 5.6 will summarize the work.

5.2   LRLS joint process estimator

In the RLS method, at any time instant $n > 0$, the adaptive filter tap weights are calculated so that the quantity

$$\zeta(n) = \sum_{k=1}^{n} \lambda^{n-k} e_n^2(k) \tag{5.1}$$

is minimized, where $0 < \lambda_n(k) < 1$, is the forgetting factor. The error, $e_n(k)$, for $k = 1, 2, \ldots, n$ is computed at time n that depends on a set of filter parameters, and hence RLS are related to type of filtering known as deterministic framework as opposed to statistical framework such as LMS filters. Fig. 5.1 shows lattice RLS joint process estimator which is used to estimate a process $d(n)$ from another correlated process $x(n)$. The two parts of the filter are the lattice predictor part and the linear combiner part. To optimize the coefficients of lattice RLS joint process estimator,

56
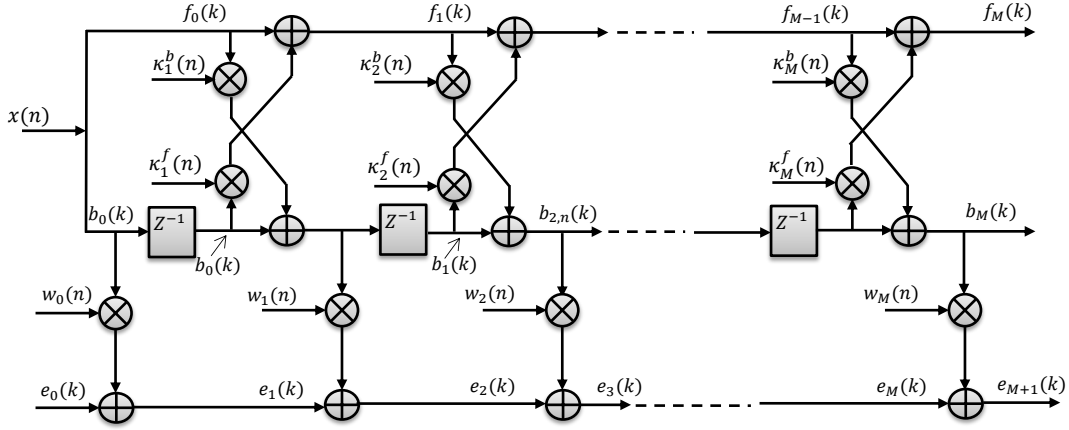
Figure 5.1 – Lattice RLS joint process estimator [4].

the following sums are minimized simultaneously [2, 4]:

$$\xi_m^{ff}(n) = \sum_{k=1}^{n} \lambda^{n-k} f_m^2(k), \qquad m = 1, 2, .., M, \tag{5.2}$$

$$\xi_m^{bb}(n) = \sum_{k=1}^{n} \lambda^{n-k} b_m^2(k), \qquad m = 1, 2, .., M, \tag{5.3}$$

$$\xi_m^{ee}(n) = \sum_{k=1}^{n} \lambda^{n-k} e_m^2(k), \qquad m = 1, 2, .., M + 1, \tag{5.4}$$

where $f_m(n)$ and $b_m(n)$ are the *a posteriori* forward and backward estimation prediction errors receptively, and $e_m(n)$ is the *a posteriori* estimation error of the length M joint process estimator. The adaptive joint process estimator adjusts all the forward PARCOR coefficients $\kappa_i^f(n), i = 1, 2, ...M$, the backward PARCOR coefficients $\kappa_i^b(n), i = 1, 2, ...M$, and the linear combiner coefficients $w_i(n), i = 0, 1, ..., M$, simultaneously. The PARCOR coefficients are adjusted to minimize the forward and backward prediction errors, while the linear combiner coefficients are adjusted to minimize the error signal $e(n)$ in the RLS sense and according to the following equations

which define the forward and backward PARCOR coefficients [4].

$$\kappa_m^f(n) = \frac{\sum_{k=1}^n \lambda^{n-k} f_{m-1}(k) b_{m-1}(k-1)}{\sum_{k=1}^n \lambda^{n-k} b_{m-1}^2(k-1)}, \tag{5.5}$$

and

$$\kappa_m^b(n) = \frac{\sum_{k=1}^n \lambda^{n-k} f_{m-1}(k) b_{m-1}(k-1)}{\sum_{k=1}^n \lambda^{n-k} f_{m-1}^2(k)}, \tag{5.6}$$

whereas the linear combiner coefficients are obtained according to

$$w_m(n) = \frac{\sum_{k=1}^n \lambda^{n-k} e_m(k) b_m(k)}{\sum_{k=1}^n \lambda^{n-k} b_m^2(k)} \tag{5.7}$$

for $m = 0, 1 \ldots, M$, where the summation in the numerator of (5.5) and (5.6) are defined as the deterministic cross-correlation between the forward and backward prediction errors $f_{m-1}(k)$ and $b_{m-1}(k-1)$ respectively, and is given by

$$\xi_m^{fb}(n) = \sum_{k=1}^n \lambda^{n-k} f_{m-1,n}(k) b_{m-1,n-1}(k-1) \tag{5.8}$$

Similarly, the summation in the numerator of (5.7) is labelled as $\xi_m^{be}(n)$, the cross-correlation between the backward prediction error $b_m(k)$ and the joint process estimation error $e_m(k)$. Algorithm 6 outlines the lattice RLS joint process estimator algorithm [2, 4].

58

Algorithm 6: Lattice RLS algorithm for adaptive joint process estimator.

Input: $x(n)$, $d(n)$, $b_m(n-1)$, $w_m(n)$, $\kappa_m^b(n)$, $\kappa_m^f(n)$, $\xi_m^{ff}(n-1)$, $\xi_m^{bb}(n-1)$, $\xi_m^{fb}(n-1)$, $\xi_m^{fe}(n-1)$, $\psi_m(n-1)$.

Output: $b_m(n)$, $w_(n+1)$, $\kappa_m^b(n+1)$, $\kappa_m^f(n+1)$, $\xi_m^{ff}(n)$, $\xi_m^{bb}(n)$, $\xi_m^{fb}(n)$, $\xi_m^{fe}(n)$, $\psi_m(n)$.

1 Initialize $\xi_m^{ff}(0)$, $\xi_m^{bb}(0)$, $\xi_m^{fb}(0)$, $\xi_m^{be}(0)$.

2 **for** $n = 0, 1, \ldots$, **do**

3 $\quad f_0(n) = b_0(n) = x(n)$

4 $\quad e_0(n) = d(n)$

5 $\quad \psi_0(n) = 1$

6 $\quad$ **for** $m = 1, 2, \ldots, M$, **do**

7 $\quad\quad \xi_m^{ff}(n) = \lambda \xi_m^{ff}(n-1) + \frac{f_m^2(n)}{\psi_m(n-1)}$

8 $\quad\quad \xi_m^{bb}(n) = \lambda \xi_m^{bb}(n-1) + \frac{b_m^2(n)}{\psi_m(n)}$

9 $\quad\quad \xi_m^{fb}(n) = \lambda \xi_m^{fb}(n-1) + \frac{f_m(n)b_m(n-1)}{\psi_m(n-1)}$

10 $\quad\quad \kappa_{m+1}^f(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{bb}(n-1)}$

11 $\quad\quad \kappa_{m+1}^b(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{ff}(n)}$

12 $\quad\quad f_{m+1}(n) = f_m(n) - \kappa_{m+1}^f(n)b_m(n-1)$

13 $\quad\quad b_{m+1}(n) = b_m(n-1) - \kappa_{m+1}^b(n)f_m(n)$

14 $\quad\quad \xi_m^{be}(n) = \lambda \xi_m^{be}(n-1) + \frac{e_m(n)b_m(n)}{\psi_m(n)}$

15 $\quad\quad w_m(n) = \frac{\xi_m^{be}(n)}{\xi_m^{bb}(n)}$

16 $\quad\quad e_{m+1}(n) = e_m(n) - w_m(n)b_m(n)$

17 $\quad\quad \psi_{m+1}(n) = \psi_m(n) - \frac{b_m^2(n)}{\xi_m^{bb}(n)}$

18 $\quad\quad$ **end for**

19 $\quad$ **end for**

5.3 Optimal structure using FT algorithm

Using system identification setup, the weight update of the adaptive filter in the FT-LMS algorithm is given by:

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e_{L(n)}^{(L(n))}(n)\mathbf{x}_{L(n)}(n), \tag{5.9}$$

where $\mathbf{w}_{L(n)}(n) = [w_0(n)w_1(n)\dots w_{L(n)}(n)]^T$ and $\mathbf{x}_{L(n)}(n) = [x_0(n)x_1(n)\dots x_{L(n)}(n)]^T$ are the weight update and input vectors respectively, μ is the step size, $L(n)$ is the variable tap-length and $e_{L(n)}^{(L(n))}(n)$ is defined in [17] to be the segmented steady-state error that is calculated by the equation

$$e_G^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:G}^T(n)\mathbf{x}_{L(n);1:G}(n), \tag{5.10}$$

with $1 \leq G \leq L(n)$, where $d(n)$ is the desired signal, $\mathbf{w}_{L(n);1:G}(n)$ and $\mathbf{x}_{L(n);1:G}(n)$ are vectors consisting of the first G elements of the vectors $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ respectively. By defining $l_f(n)$ as the pseudo fractional tap-length, the update equation of the FT-LMS is proposed as in [17]

$$l_f(n+1) = l_f(n) - \alpha - \gamma[(e_{L(n)}^{(L(n))}(n))^2 - (e_{L(n)-\Delta}^{(L(n))}(n))^2], \tag{5.11}$$

where γ is the step size for the tap-length adaptation, α is a positive leakage parameter and Δ is a positive integer. Then, the updated tap-length, which will be used in the next iteration, is calculated from the fractional tap-length $l_f(n)$ by:

$$L(n+1) = \begin{cases} \lfloor l_f(n) \rfloor & \text{if } |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases}, \tag{5.12}$$

where $\lfloor . \rfloor$ is the floor operator and δ is a small integer. When a fixed Δ is employed, the FT algorithm is required to find a compromise between convergence speed and

60

the bias from the optimum tap-length [13].

5.4   Fractional tap-length lattice algorithm

In this section, the fractional tap-length (FT) strategy is combined with lattice RLS joint process estimators to form a new optimally structured lattice filter, namely, the fractional tap-length lattice recursive least squares FT-LRLS. For the RLS-based lattice joint process algorithm, the fractional tap-length equation in (5.11) is slightly modified to cope with the minimization principle of the RLS that suppresses the effect of past solution using a forgetting factor as [20]

$$l_f(n+1) = \lambda l_f(n) - \gamma [(e_{L(n)}^{(L(n))}(n))^2 - (e_{L(n)-\Delta}^{(L(n))}(n))^2] \qquad (5.13)$$

where λ is the exponential weighting-factor or forgetting-factor, which is set close 1. Then the integer tap-length $L(n)$ is updated in similar manner according to (5.12). The proposed algorithm and its computational complexity are shown in Algorithm 7 and Table 5.1 respectively. In Algorithm 7, the auto-correlations $\xi_m^{ff}(0)$ and $\xi_m^{bb}(0)$ are initialized to a small positive number to prevent numerical difficulties [4]. The cross-correlations $\xi_m^{fb}(0)$ and $\xi_m^{be}(0)$ are initialized to zero. On the other hand, the fractional $l(n)$ and the integer $L(n)$ tap-lengths are initialized to some selected value [13, 17].

Algorithm 7: Proposed fractional tap lattice RLS (FT-LRLS) algorithm.

Input: $x(n)$, $d(n)$, $b_m(n-1)$, $w_m(n)$, $\kappa_m^b(n)$, $\kappa_m^f(n)$, $\xi_m^{ff}(n-1)$, $\xi_m^{bb}(n-1)$,
$\xi_m^{fb}(n-1)$, $\xi_m^{fe}(n-1)$, $\psi_m(n-1)$, $l(n)$, $L(n)$.

Output: $b_m(n)$, $w_{(}n+1)$, $\kappa_m^b(n+1)$, $\kappa_m^f(n+1)$, $\xi_m^{ff}(n)$, $\xi_m^{bb}(n)$, $\xi_m^{fb}(n)$, $\xi_m^{fe}(n)$,
$\psi_m(n)$, $L(n)$.

---

1 Initialize $\xi_m^{ff}(0)$, $\xi_m^{bb}(0)$, $\xi_m^{fb}(0)$, $\xi_m^{be}(0)$, $l(0)$, $L(0)$.

2 **for** $n = 0, 1, \ldots,$ **do**

3 $\quad$ $f_0(n) = b_0(n) = x(n)$

4 $\quad$ $e_0(n) = d(n)$

5 $\quad$ $\psi_0(n) = 1$

6 $\quad$ **for** $m = 1, 2, \ldots, M$, **do**

7 $\quad\quad$ $\xi_m^{ff}(n) = \lambda \xi_m^{ff}(n-1) + \frac{f_m^2(n)}{\psi_m(n-1)}$

8 $\quad\quad$ $\xi_m^{bb}(n) = \lambda \xi_m^{bb}(n-1) + \frac{b_m^2(n)}{\psi_m(n)}$

9 $\quad\quad$ $\xi_m^{fb}(n) = \lambda \xi_m^{fb}(n-1) + \frac{f_m(n)b_m(n-1)}{\psi_m(n-1)}$

10 $\quad\quad$ $\kappa_{m+1}^f(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{bb}(n-1)}$

11 $\quad\quad$ $\kappa_{m+1}^b(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{ff}(n)}$

12 $\quad\quad$ $f_{m+1}(n) = f_m(n) - \kappa_{m+1}^f(n)b_m(n-1)$

13 $\quad\quad$ $b_{m+1}(n) = b_m(n-1) - \kappa_{m+1}^b(n)f_m(n)$

14 $\quad\quad$ $\xi_m^{be}(n) = \lambda \xi_m^{be}(n-1) + \frac{e_m(n)b_m(n)}{\psi_m(n)}$

15 $\quad\quad$ $w_m(n) = \frac{\xi_m^{be}(n)}{\xi_m^{bb}(n)}$

16 $\quad\quad$ $e_{m+1}(n) = e_m(n) - w_m(n)b_m(n)$

17 $\quad\quad$ $\psi_{m+1}(n) = \psi_m(n) - \frac{b_m^2(n)}{\xi_m^{bb}(n)}$

18 $\quad\quad$ $l_f(n+1) = (\lambda l_f(n)) - \gamma \left[\left(e_{L(n)}^{(L(n))} \right)^2 - \left(e_{L(n)-\Delta}^{(L(n))} \right)^2 \right]$

19 $\quad\quad$ $L(n+1) = \begin{cases} l_f(n) & \text{if} \quad |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases}$

20 $\quad\quad$ **end for**

21 $\quad$ **end for**

Table 5.1 – Computational complexity of the proposed FT-LRLS algorithm.

| Equations	$+$	$\times$	$\div$
(1) $\xi_m^{ff}(n) = \lambda\xi_m^{ff}(n-1) + \frac{f_{m,n}^2(n)}{\psi_m(n-1)}$	M	$2M$	M
(2) $\xi_m^{bb}(n) = \lambda\xi_m^{bb}(n-1) + \frac{b_{m,n}^2(n)}{\psi_m(n)}$	M	$2M$	M
(3) $\xi_m^{fb}(n) = \lambda\xi_m^{fb}(n-1) + \frac{f_{m,n}(n)b_{m,n-1}(n-1)}{\psi_m(n-1)}$	M	$2M$	M
(4) $\kappa_{m+1}^f(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{bb}(n-1)}$	$-$	$-$	M
(5) $\kappa_{m+1}^b(n) = \frac{\xi_m^{fb}(n)}{\xi_m^{ff}(n)}$	$-$	$-$	M
(6) $f_{m+1,n}(n) = f_{m,n}(n) - \kappa_{m+1}^f(n)b_{m,n-1}(n-1)$	M	M	$-$
(7) $b_{m+1,n}(n) = b_{m,n-1}(n-1) - \kappa_{m+1}^b(n)f_{m,n-1}(n)$	M	M	$-$
(8) $\xi_m^{be}(n) = \lambda\xi_m^{be}(n-1) + \frac{e_{m,n}(n)b_{m,n}(n)}{\psi_m(n)}$	M	$2M$	M
(9) $c_m(n) = \frac{\xi_m^{be}(n)}{\xi_m^{bb}(n)}$	$-$	$-$	M
(10) $e_{L(n)}^{L(n)}(n)e_{m+1,n}(n) = e_{m,n}(n) - c_m(n)bm,n(n)$	M	M	$-$
(11) $\psi_{m+1}(n) = \psi_m(n) - \frac{b_{m,n}^2(n)}{\xi_m^{bb}(n)}$	M	M	M
Tap-length Update: (12) $l_f(n+1) = (\lambda l_f(n)) - \gamma\left[\left(e_{L(n)}^{(L(n))}\right)^2 - \left(e_{L(n)-\Delta}^{(L(n))}\right)^2\right]$	2	3	$-$
(13) $L(n+1) = \begin{cases} l_f(n) & \text{if} \quad \|L(n) - l_f(n)\| > \delta \\ L(n) & \text{otherwise} \end{cases}$	$-$	$-$	$-$
Total	$8M + 2$	$12M + 3$	$8M$
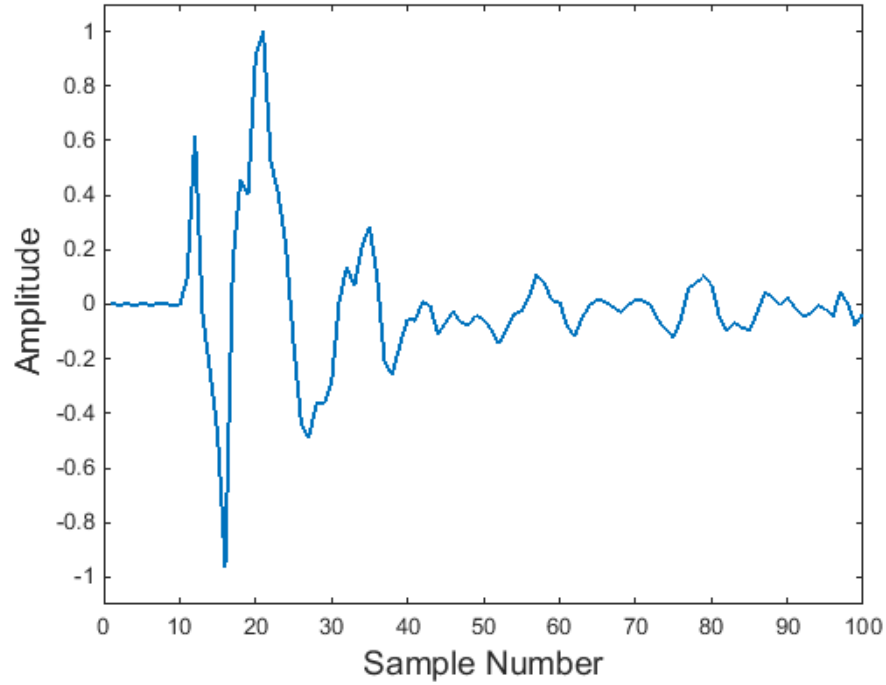
Figure 5.2 – Car cabin impulse response of length 100 samples.

5.5 System simulations

The main objective of the simulation in this section is to identify an unknown system characterized by the impulse response shown in Fig. 5.2, which is a car cabin truncated to 100 samples. The setup for simulation is similar to those in [14] and [21], where the input signal is white Gaussian noise with zero-mean and unity variance and the results are obtained by averaging multiple independent runs of the simulation of the proposed algorithms.

5.5.1 Lattice filters vs. transversal filters

This simulation shows a comparison between lattice LMS and RLS filters and their counterpart of transversal filters to identify the unknown system of Fig. 5.2. Simu-

lation is implemented using high-signal-to-noise ratio SNR = 30. Filter length in all structures are fixed to $L = 100$. The step-size for transversal LMS adaptive filter and and the linear combiner part of lattice structure LMS adaptive filter is $\mu = 0.0005$, and the step-size for the predictor part of lattice structure is $\mu_p = 10^{-7}$, and the RLS forgetting value $\lambda = 0.999$. Figure 5.3 shows the learning curves for the respective algorithms. Lattice LMS adaptive filter shows superiority in convergence rate with slightly less MSE value as compared to the LMS transversal filter, whereas the lattice RLS manifests superiority in both of convergence rate as well as the MSE properties as compared with the transversal RLS filter.
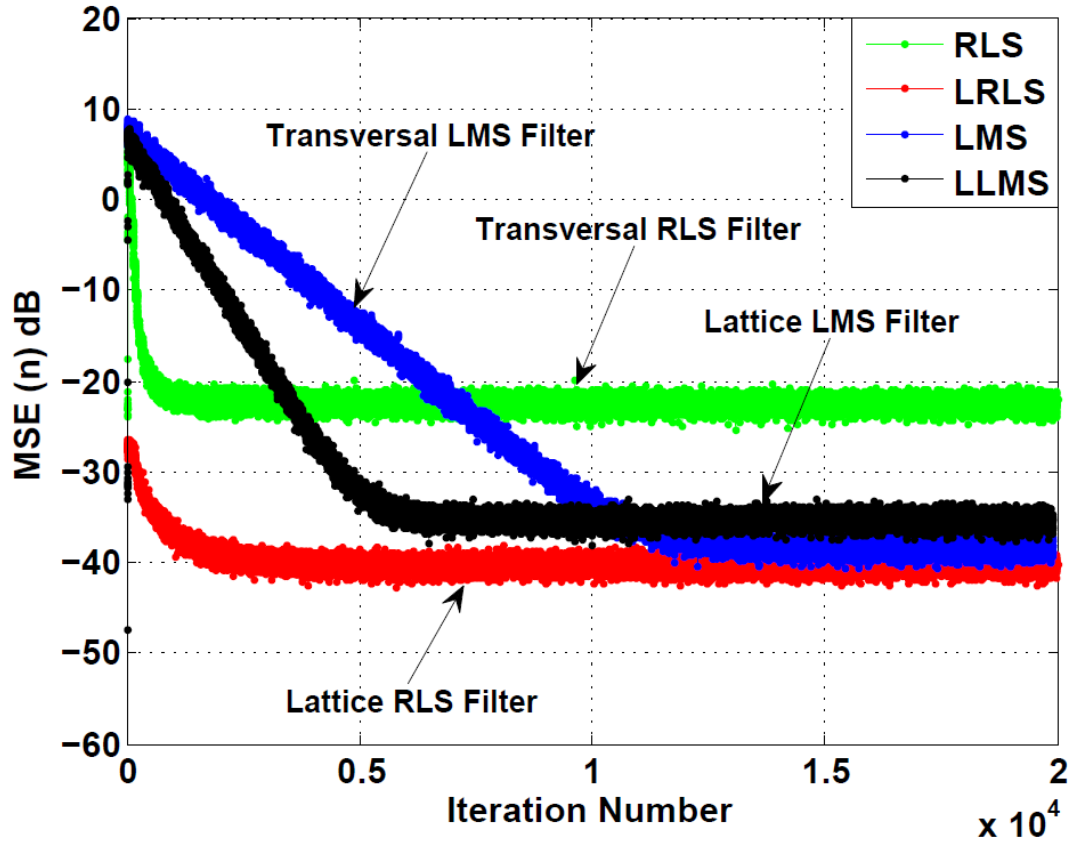

Figure 5.3 – Comparison of MSE of fixed length lattice and transversal filters

65

5.5.2 Parameters' selection

Here, the proposed joint process FT-LRLS algorithm uses the *a posteriori* estimation errors as previously discussed in Algorithm 7. The forgetting factor of $\lambda = 0.999$ is selected which gave less weight to older error samples [20]. In each iteration, the algorithm begins with some initial values of forward $f_0(n)$, backward $b_0(n)$ and *a posteriori* $e_0(n)$ errors and the conversion factor $\psi_0(n)$, as inputs to the first stage and proceeds to update the consecutive stages of Algorithm 7. According to [4], the auto-correlations of $\xi_m^{ff}(n)$, $\xi_m^{bb}(n)$ can theoretically be initiated to zero. However, because such initialization results in division by zero in first iterations of Algorithm 7, it is recommended in [4] to start with a small positive number, and therefore an initialization of $\xi_m^{ff}(0) = \xi_m^{bb}(0) = 0.001$, for $m = 0, 1, \ldots, M$ is selected. The cross-correlations $\xi_m^{fb}(0)$ and $\xi_m^{be}(0)$ are initialized to zero. As a result, all remaining recursion are written in terms of the *a posteriori* estimation error. These recursions are then used to to calculate the forward and backward PARCOR coefficients of $\kappa_{m+1}^f(n)$ and $\kappa_{m+1}^b(n)$ respectively as in Algorithm 7. Whereas fractional tap-length algorithm parameters are chosen according to [13]. The value δ should be a small positive integer $1 \leq \delta \leq 10$ [13, 14, 17], therefore, it was chosen $\delta = 2$ in this paper. The selected value of leakage parameter α is an application dependent, however, it was stated in [13] that values of α between 0.001 and 0.01 are generally good choices, hence, $\alpha = 0.005$ was selected in simulation. The parameter γ controls both convergence speed and fluctuation of the tap-length, a large γ leads to fast convergence of the tap-length but results in large fluctuation, consequently, a trade-off choice of $\gamma = 3$ is selected. The parameter Δ plays an important part to the proposed algorithms as it controls convergence speed and bias from the optimum tap-length, which requires a compromise between them due to its fixed value [13, 17, 21]. Therefore, $\Delta = 50$ is selected throughout the system simulation section.

### 5.5.3 Variable length LRLS vs. fixed length LRLS filters

The adaptive filtering FT-LRLS algorithm is tested here using algorithm parameters given in Section 5.5.2, and then compared to standard adaptive filtering lattice RLS with lengths $L = 60, 120$, and 180 respectively using SNR = 30 dB and SNR = 10 dB environments. Figure 5.4 shows that FT-LRLS exposed noticeable low MSE level in both of high and low SNR environments. In high SNR, Fig. 5.4a the MSE goes as low as -50 dB, whereas in low SNR Fig. 5.4b the MSE is around -20 dB.

5.5.4 FTLRLS algorithm vs. FTLLMS algorithm

Figure 5.5 shows the learning curves and the expected value of tap-length of the adaptive lattice filtering using FTLRLS and FTLLMS algorithms. Figure 5.5a shows better MSE error level and faster convergence speed of FTLRLS algorithm over FTLLMS algorithm. In Fig. 5.5b, both algorithms converged to the same value of about 150 taps, which indicates that the FT algorithm has a bias of around 50. This matches the analysis of the FT algorithm [17] since it uses a fixed $\Delta = 50$, where the expected bias according to analysis is approximately the $E[L(\infty - 50)]$. Figure 5.6 shows similar performance using low SNR of 10 dB.
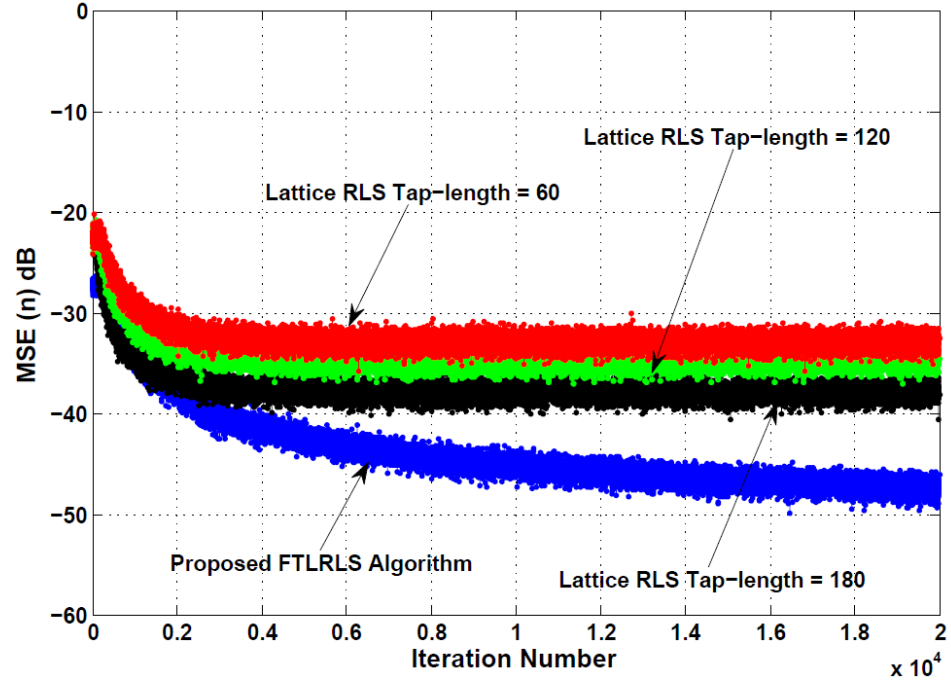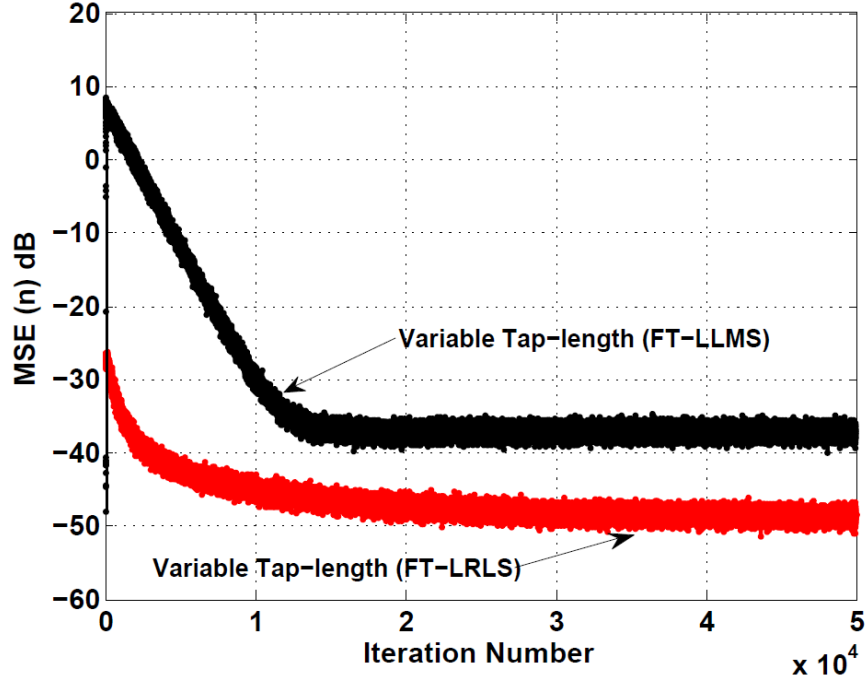
(a) (SNR = 30 dB).
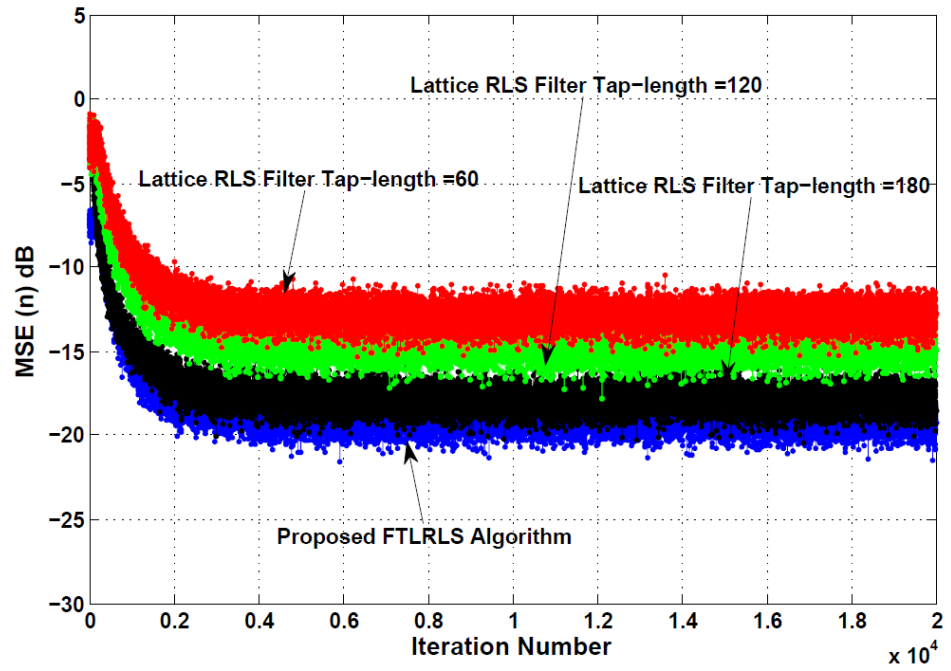
(b) (SNR = 10 dB).

Figure 5.4 – Learning curves of the proposed FT-LRLS and different lattice RLS filter lengths.

(a) Comparison of MSE(n) of the proposed algorithms.
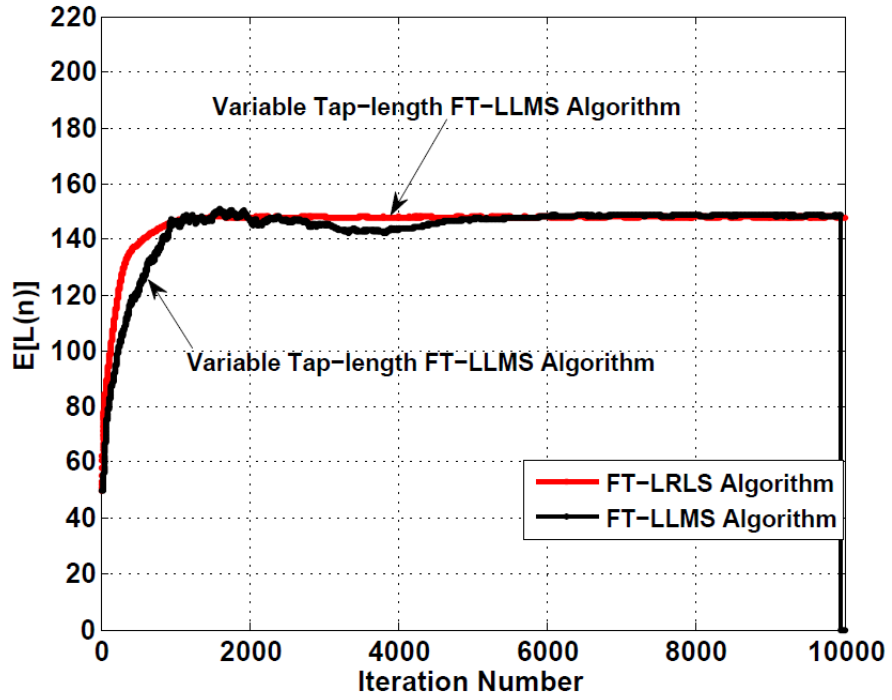
(b) Comparison of $E[L(n)]$ of the proposed algorithms.

Figure 5.5 – MSE and expectation of tap-length of proposed algorithms in high SNR environment

(a) Comparison of MSE(n) of the proposed algorithms.

(b) Comparison of $E[L(n)]$ of the proposed algorithms.

Figure 5.6 – MSE and expectation of tap-length of proposed algorithms in low SNR environment

70

5.6   Conclusion

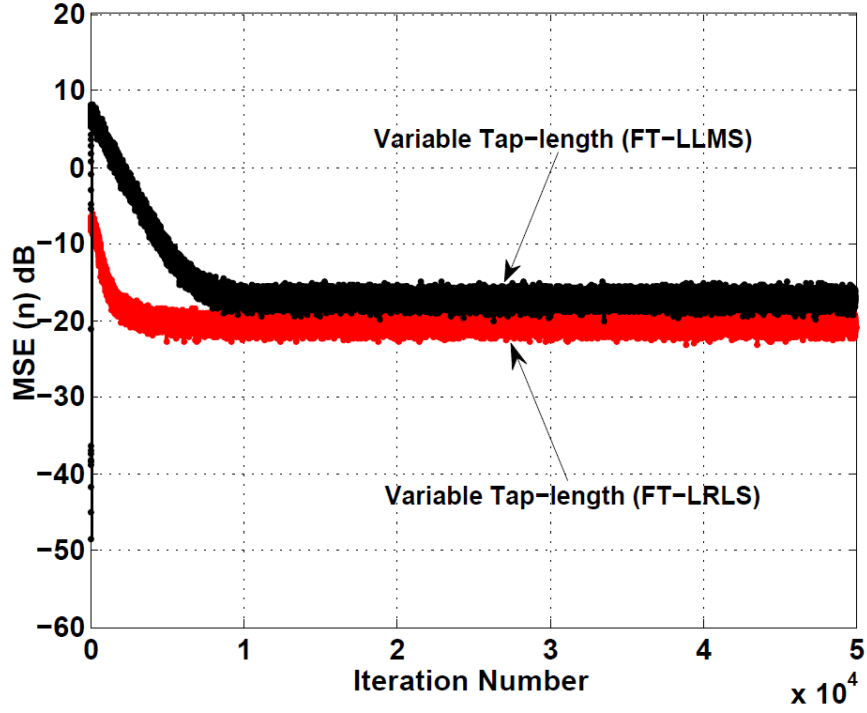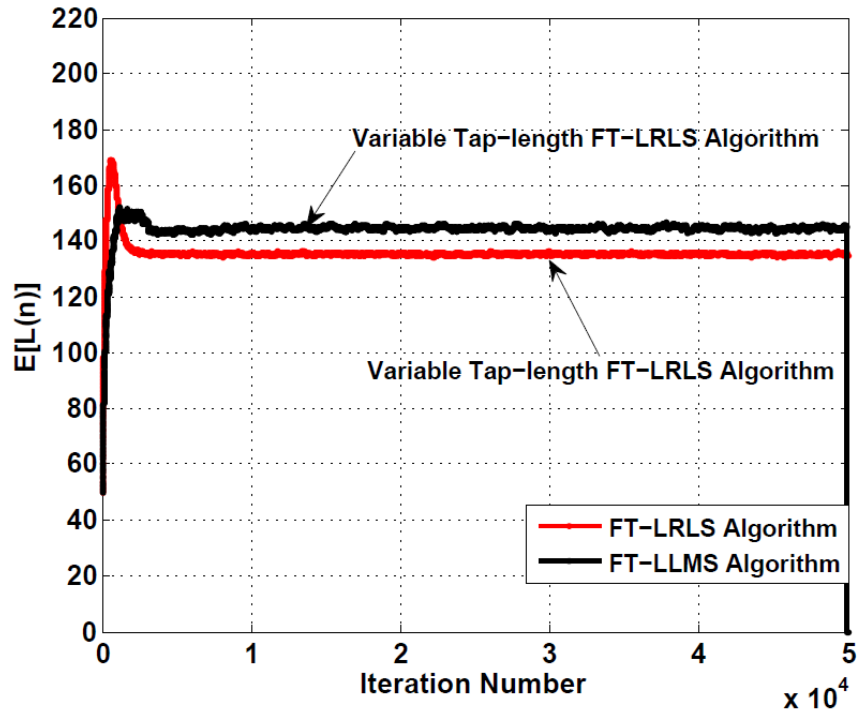In this work, we proposed a new variable tap-length algorithm for lattice RLS joint process estimator filter. The fractional tap-length was used to search for the optimal length and hence improve lattice structure adaptive filter performance. System simulations of the proposed algorithms were carried out in a system identification setup and the simulation results have shown that the new algorithms are capable of identifying unknown systems in high SNR as well as in low SNR. Improved convergence rate and error properties of the proposed algorithms were also shown by simulations as compared with the fixed length LMS and RLS lattice adaptive algorithms.

References

[1] S. Haykin *et al.*, "Adaptive filtering theory," *Englewood Cliffs, NJ: Prentice-Hall*, 1996.

[2] P. Diniz, "Adaptive filtering: Algorithms and practical implementation. springer," *New York, NY, USA*, 2008.

[3] M.-L. Leou, Y.-C. Liaw, and C.-M. Wu, "A simple recursive least square algorithm of spacetemporal joint equalizer," *Digital Signal Processing*, vol. 22, no. 6, pp. 1145 – 1153, 2012.

[4] B. Farhang-Boroujeny, *Adaptive Filters: Theory and Applications, Second Edition.* John Wiley and Sons, 4 2013.

[5] L. Griffiths, "An adaptive lattice structure for noise-cancelling applications," in *ICASSP '78. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, Apr. 1978, pp. 87–90.

[6] H. Qi, "Analysis and application of gradient adaptive lattice filtering algorithm," in *2008 International Conference on Computer and Electrical Engineering*, Dec. 2008, pp. 744–747.

[7] H. F. Lee, J. T. Yuan, and T. C. Lin, "Iir lattice-based blind equalization algorithms," in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, Sep. 2012, pp. 1–5.

[8] J. Cioffi and T. Kailath, "Fast, recursive-least-squares transversal filters for adaptive filtering," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 304–337, April 1984.

[9] J. Markel and A. Gray, *Linear Prediction of Speech*, ser. Communication and Cybernetics. Springer Berlin Heidelberg, 2013.

[10] B. Porat and T. Kailath, "Normalized lattice algorithms for least-squares fir system identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 1, pp. 122–128, Feb 1983.

[11] M. Setareh, M. Parniani, and F. Aminifar, "Ambient data-based online electromechanical mode estimation by errorfeedback lattice rls filter," *IEEE Transactions on Power Systems*, vol. 33, no. 4, pp. 3745–3756, July 2018.

[12] P. K. Mohapatra, P. K. Jena, S. K. Bisoi, S. K. Rout, and S. P. Panigrahi, "Channel equalization as an optimization problem," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Oct. 2016, pp. 1158–1163.

[13] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, Feb. 2008.

[14] N. Li, Y. Zhang, Y. Zhao, and Y. Hao, "An improved variable tap-length lms algorithm," *Signal Processing*, vol. 89, no. 5, pp. 908–912, 2009.

[15] H. Liu, X. Li, L. Ge, C. Rizos, and F. Wang, "Variable length lms adaptive filter for carrier phase multipath mitigation," *GPS Solutions*, vol. 15, no. 1, pp. 29–38, Jan 2011.

[16] Y. Gong and C. F. N. Cowan, "Structure adaptation of linear mmse adaptive filters," *IEE Proceedings - Vision, Image and Signal Processing*, vol. 151, no. 4, pp. 271–277, Aug. 2004.

[17] ——, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[18] F. Riera-Palou, J. M. Noras, and D. G. M. Cruickshank, "Linear equalisers, with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, pp. 1553–1554, Dec. 2001.

[19] Y. Gu, K. Tang, and H. Cui, "Lms algorithm with gradient descent filter length," *IEEE Signal Processing Letters*, vol. 11, no. 3, pp. 305–307, March 2004.

[20] S. Sitjongsataporn, "Variable tap-length mixed-tone rls-based per-tone equalisation with adaptive implementation," *ECTI Transactions On Electrical ENG., Electronics, and Communications*, vol. 10, no. 2, pp. 179–188, Aug. 2012.

[21] K. Mayyas and H. A. AbuSeba, "A new variable length nlms adaptive algorithm," *Digital Signal Processing*, vol. 34, pp. 82–91, 2014.

Chapter 6

Variable Tap-length Blind Equalization

6.1　Introduction

Adaptive equalizers are used to remove the signal distortion caused by multi-path effect within time dispersive channel [1–3]. When the scheme applied to achieve channel equalization does not include any training sequence from transmitter to the receiver, it is referred to as *blind equalization* algorithm [4,5]. Among several blind equalization algorithms in the literature, constant modulus algorithm (CMA) has become, since was discovered by Godard [6] and Treichler et al [7], the workhorse for blind equalization. This is due to CMA's capability to converge independent of phase recovery [8]. In the process of quadrature amplitude modulation (QAM), the equalizer's design is of special importance because it absorbs a large fraction of the computation complexity required in the receiver setup [9]. This necessitates answering the question of how long the equalizer should be. In practice, there is no general solution for this question and different approaches in literature were proposed in attempt to solve it. In the first approach, a prototype of the equalizer is built and then tested against a variety of actual channels [10]. The second method applies rules of thumb that appears intuitively reasonable for length selection depending on the type of commutation channel and the transmitted signal sampling rate [9], [10]. Both approaches are costly and does not deal with changes in the channel behaviors which can compromise the equalizer's performance. A cost-effective approach is to come up with a variable tap-length scheme that can search for the optimal length while the equalizer is adapting its coefficients. In [11], a segmented filter (SF) variable

tap-length algorithm was employed to the equalizer. In SF algorithm the equalizer is subdivided into k segments, each with fixed coefficients. Then, based on the difference between the accumulated squared errors from the last two segments, the tap-length of the filter is modified by adding or subtracting one segment, which makes SF an inflexible option. Authors [12] and [13] used a more flexible and robust variable length technique that employs the fractional tap-length (FT) algorithm, however, in the straightforward training mode equalization context. The novelty of this work is to search for the optimal tap-length of the CMA blind equalizer using the FT technique. This is done by modifying a non-liner error $e(n)$ of the CMA equalizer output $y(n)$ to update FT iterations within the initial operation of the equalizer i.e. in blind mode.

6.2 Constant modulus algorithm (CMA)

CMA accomplishes channel equalization by penalizing the dispersion of the output modulus, $|y(n)|$, from the constant γ_C. The cost function that is minimized by CMA is defined as following [4]:

$$J^{cma} = E[(|y(n)|^2 - \gamma_C^2)^2]$$ (6.1)

Minimizing this cost function can be thought of as fitting the signal constellation to a circle as shown in Fig. 6.1 [4].

The dispersion constant γ_C^2 is given by:

$$\gamma_C^2 = \frac{E[|s(n)|^4]}{E[|s(n)|^2]}.$$ (6.2)

A gradient-descent equalizer algorithm that minimizes the cost function of (6.1) is as follows

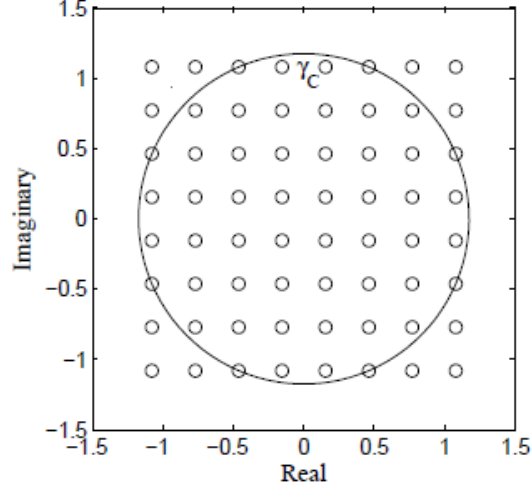$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu(-\nabla_w J^{cma})$$ (6.3)

76

Figure 6.1 – Graphical representation of CMA for 64-QAM.

Hence the CMA blind equalizer tap adjustment is given as

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu e^{cma}(n)\mathbf{x}^*(n) \tag{6.4}$$

where $\mathbf{x}(n)$ is the regressor input vector while $*$ denotes complex conjugation operation. The term $e^{cma}(n) = y(n)(\gamma_C^2 - |y(n)|^2)x^*(n)$, denotes the error of CMA blind equalizer filter [14], which is a complex signal consists of a real and imaginary parts

$$e^{cma}(n) = e_R^{cma}(n) + je_J^{cma}(n) \tag{6.5}$$

$$
\begin{aligned}
e_R^{cma}(n) &= y_R(n)[\gamma_C^2 - y_R^2(n) - y_I^2(n)] \\
e_I^{cma}(n) &= y_I(n)[\gamma_C^2 - y_R^2(n) - y_I^2(n)]
\end{aligned}
\tag{6.6}
$$

and hence, CMA equalizer update of (6.4) can be rewritten as

$$
\begin{aligned}
\mathbf{w}(n+1) = \mathbf{w}(n) &+ \mu(y_R(n)(\gamma_C^2 - y_R^2(n) - y_I^2(n)) \\
&+ jy_I(n)(\gamma_C^2 - y_R^2(n) - y_I^2(n)))\mathbf{x}^*(n)
\end{aligned}
\tag{6.7}
$$

77

6.3   Optimum tap-length

6.3.1   System identification

Within a system identification model and using FT variable tap-length algorithm to search for the optimal filter length, the LMS filter weight update is given by:

$$\mathbf{w}_{L(n)}(n+1) = \mathbf{w}_{L(n)}(n) + \mu e_{L(n)}^{(L(n))}(n)\mathbf{x}_{L(n)}(n) \tag{6.8}$$

where $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ are the weight update and input vectors respectively, μ is the step size, $L(n)$ is the variable tap-length and $e_{L(n)}^{(L(n))}(n)$ is defined in [15] to be the segmented steady-state error that is calculated by the equation

$$e_G^{L(n)}(n) = d(n) - \mathbf{w}_{L(n);1:G}^T(n)\mathbf{x}_{L(n);1:G}(n) \tag{6.9}$$

where $1 \leq G \leq L(n)$, $d(n)$ is the desired signal, and $\mathbf{w}_{L(n);1:G}(n)$ and $\mathbf{x}_{L(n);1:G}(n)$ are vectors consisting of the first G elements of the vectors $\mathbf{w}_{L(n)}(n)$ and $\mathbf{x}_{L(n)}(n)$ respectively.

The pseudo fractional tap-length $l_f(n)$ is updated using the following:

$$l_f(n+1) = (l_f(n) - \alpha) - \gamma[(e_{L(n)}^{(L(n))}(n))^2 - (e_{L(n)-\Delta}^{(L(n))}(n))^2] \tag{6.10}$$

where γ is the step size for the tap-length adaptation, α is a positive leakage parameter and Δ is a positive integer. Then, the updated tap-length, which will be used in the next iteration, is calculated from the fractional tap-length $l_f(n)$ by [15–17]

$$L(n+1) = \begin{cases} \lfloor l_f(n) \rfloor & \text{if } |L(n) - l_f(n)| > \delta \\ L(n) & \text{otherwise} \end{cases} \tag{6.11}$$

where $\lfloor . \rfloor$ is the floor operator and δ is a small integer. When a fixed Δ is employed, the FT algorithm is required find a compromise between convergence speed and the bias from the optimum tap-length [15].

6.3.2 CMA blind equalizer's optimal length

Here, we use the same concept of the previous section to estimate the CMA equalizer's optimum length. Assuming the CMA's tap-length to be variable in time as $L(n)$, then using (6.9) we define $e(n)_{L(n)}$ and $e(n)_{L(n)-\Delta}$ respectively, as follows:

$$
\begin{aligned}
e(n)_{L(n)}^{L(n)} &= |e_R^{cma}(n) + je_J^{cma}(n)|_{L(n)} \\
e(n)_{L(n)-\Delta}^{L(n)} &= |e_R^{cma}(n) + je_J^{cma}(n)|_{L(n)-\Delta}
\end{aligned}
\tag{6.12}
$$

Therefore using multirate system modeling of Fig. 6.2, the VL-CMA equalizer's tap weights are updated according to

$$
\mathbf{w}(n+1)_{L(n)} = \mathbf{w}(n)_{L(n)} + \mu |e^{cma}(n)|_{L(n)} \mathbf{x}_{L(n)}^*(n)
\tag{6.13}
$$

And finally the fractional and integer tap-length of the next iteration are given by (6.10) and (6.11), respectively.

6.4 System model

This section addresses CMA blind equalization in the context of single-input-single-output (SISO) systems. $T-$spaced equalizers are known to be sensitive the to symbol rate which makes a fractional spaced T/F equalizer [10] a more feasible alternative as it can achieve the desired equalization with a finite number of taps. To many designers, fractionally spaced equalizer with $F = 2$, i.e $T/2$, is considered a practical choice [18], which is going to be considered exclusively throughout this paper.
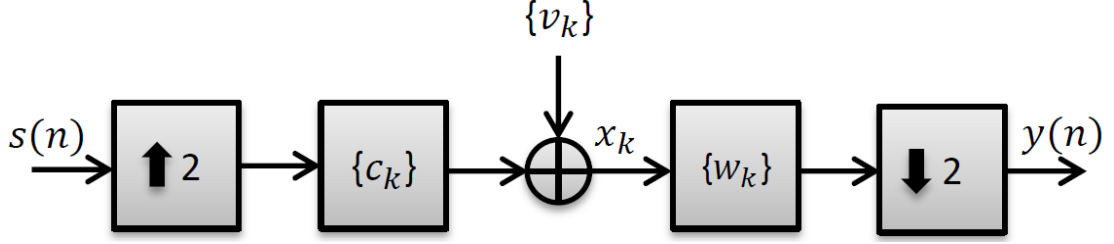
Figure 6.2 – Multirate system modeling.

Figure 6.2 shows the single input single output (SISO) multirate system that will employ the proposed VL-CMA algorithm. In this system model 'n' denotes the $T-$spaced quantities whereas 'k' is refereed to $T/2$ quantities. The source transmits $T-$spaced symbols through a pulse shaping filter and modulated onto a a microwave $T/2-$spaced channel. The source symbol is assumed to be an independent and identically distributed (i.i.d) random variable with a variance of $\sigma_s^2 = E[\mathbf{s}(n)^2]$. which is taken from a finite alphabet given by the set $\{\mathbf{s}_m = a_m + jb_m\}_{m=1}^M$ for $M-$QAM constellation. The received $T/2$ signal is corrupted by white Gaussian noise signal of $v(k)$ as well as ISI effect. In classical implementation, the baseband receiver consists of an $N-$tap $T/2-$spaced equalizer whose tap weights are specified by $N \times 1$ vector of $\mathbf{w}(n) = [w_1(n), w_2(n), .., w_N(n)]^T$. However, in this work the tap-length of the equalizer is made variable as $L(n)$ where the FT is used to search for the equalizer optimal length, and consequently the quantities of system model will become variable in length as follows. The regressor vector of the equalizer input sequence comprises the previous filter length's received $T/2$-space samples and is given by

$$\mathbf{x}_{L(n)}(n) = \mathbf{C}_{L(n)}^T \mathbf{s}_{L(n)}(n) + \mathbf{v}_{L(n)}(n) \tag{6.14}$$

where $\mathbf{C}_{L(n)}$ is a variable length $T-$spaced convolution matrix that has dimensions of $P(n) \times L(n)$, where $P(n) = K + L(n) - 1$, $\mathbf{s}_{L(n)}(n) = [s(n), s(n-1), .., s(n - P(n))]^T$ is a $P(n) \times 1$ vector of source symbols and the $\mathbf{v}_{L(n)}(n) = [v_1(n), v_2 n), ..., v_{L(n)}(n)]^T$ is the additive white Gaussian noise column vector of length $L(n)$. The convolution

matrix is formed by the odd rows of channel impulse response vector, that is [19]

$$
\begin{bmatrix}
c_1 & c_0 & & & & & \\
c_3 & c_2 & c_1 & c_0 & & & \\
\vdots & \vdots & c_3 & c_2 & \ddots & & \\
c_{K-1} & c_{K-2} & \vdots & \vdots & \ddots & c_1 & c_0 \\
& & c_{K-1} & c_{K-2} & & c_3 & c_2 \\
& & & & \ddots & \vdots & \vdots \\
& & & & & c_{K-1} & c_{K-2}
\end{bmatrix}
$$

The equalizer output is then decimated by a factor of two and is given by

$$
\begin{aligned}
y_{L(n)}(n) &= \mathbf{x}_{L(n)}^T(n)\mathbf{w}_{L(n)}(n) \\
&= \mathbf{s}_{L(n)}^T(n)\mathbf{C}_{L(n)}\mathbf{w}_{L(n)}(n) + \mathbf{v}_{L(n)}^T(n)\mathbf{w}_{L(n)}(n)
\end{aligned}
\tag{6.15}
$$

6.5 System simulation

In this section, the performance of proposed variable tap-length blind equalizer (VL-CMA) is tested using multirate system model shown in Fig. 6.2. The experimental setup is similar to that in [18], which consists of a $T/2-$spaced SPIB microwave channel [20] in cascade with a $T/2-$spaced finite impulse response (FIR) equalizer which is initialized by a unitary double center spike. A white Gaussian noise is added such that the final signal-to-noise ratio (SNR) is 30 dB. To validate the proposed algorithm performance, two experiments are carried out for $16-$QAM and $64-$QAM constellations.

6.5.1 $16-$QAM

In this simulation, the source symbols are randomly taken from normalized $16-$QAM constellation. The received equalizer input samples are generated by convolving the

source sequence and the SPIB #10 microwave channel whose specifications are illustrated below [20].

Table 6.1 – Microwave radio channel SPIB #10.

Channel	SPIB Database Designator	# of Taps	Frequency
MCR-08	10	300	30 Mbaud

The magnitude of the channel impulse response is plotted in Fig. 6.3 where it is characterized by the finite series of $\{c_k\}_{k=1}^{K}$ where K is the channel length as shown in Table. 6.1. The algorithm step-size is $\mu = 2^{-12}$, and the VL-CMA algorithm's
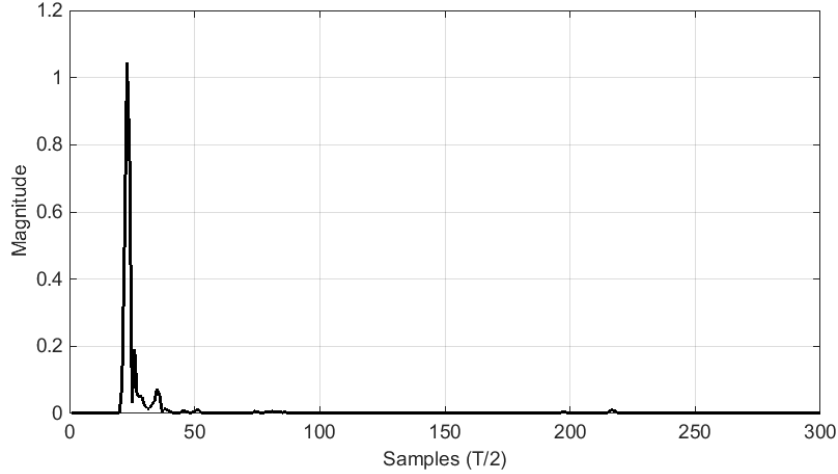

Figure 6.3 – Magnitude of impulse response for SPIB #10 microwave channel.

parameters are selected according to [21] with $\Delta = 3$, $\alpha = 0.005$, $\delta = 1$ and $\gamma = 0.015$. System simulation was performed by averaging over 100 independent runs. The ensemble mean squared error (MSE) of the VL-CMA algorithm was averaged and compared with those obtained from blind equalization algorithms of generalized Sato algorithm (GSA) [22, 23], multimodulus algorithm (MMA) [4, 24], and CMA [6, 19]. The GSA, and MMA algorithms are used with fixed length of 16 taps [18]. As can be seen from Fig. 6.4, the proposed variable tap-length equalizer showed much better

convergence rate than other fixed length algorithms while providing the same steady state MSE. The proposed algorithm was able to search for the optimal filter's length; this capability is shown in Fig. 6.5 which plots the expected value of the VL-CMA tap-length $E[L(n)]$.
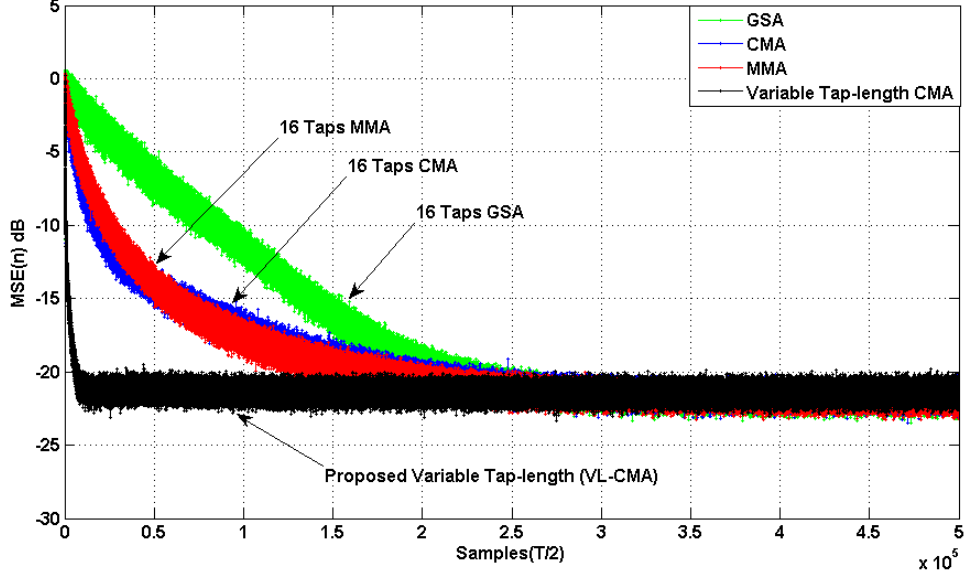

Figure 6.4 – Simulation results of MSE(n) of proposed algorithm against other blind equalization algorithms for 16 QAM.

The initial filter length was set to $L(0) = 16$, tap-length of proposed algorithm converged to the value of $L = 13$ and consequently the estimated optimal length is approximately $L_{opt} = 13 - \Delta = 10$ taps [21]. The proposed equalizer's output signal constellation for $16-$QAM at steady state is illustrated in Fig. 6.6 which indicates that the VL-CMA algorithm has achieved a recognizable constellation points in shorter time and lower number of taps than other algorithms.

6.5.2 $64-$QAM

In this simulation, $64-$QAM constellation is used to for source symbols and the received equalizer input samples are generated by convolving the source sequence
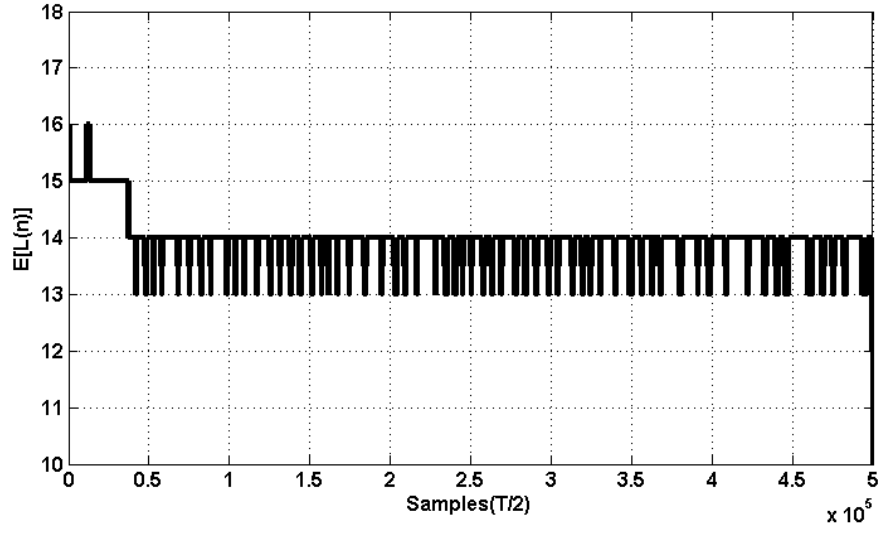
Figure 6.5 – The expected value of the proposed VL-CMA algorithm tap-length for 16-QAM.
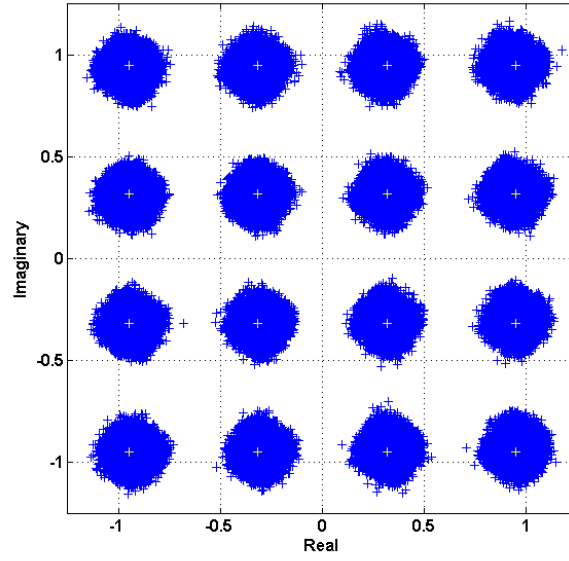

Figure 6.6 – Output signal constellation of the proposed VL-CMA algorithm for 16-QAM.

and the SPIB #12 microwave channel whose specifications are illustrated in Table 6.2 [20].

Table 6.2 – Microwave radio channel SPIB #12.

Channel	SPIB Database Designator	# of Taps	Frequency
MCR-10	12	227	22.5 Mbaud

The magnitude of the channel impulse response is plotted in Fig. 6.7 where it is again characterized by the finite series of $\{c_k\}_{k=1}^{K}$ where K is the channel length as shown in Table 6.2. The variable tap-length algorithm step-size in case of 64
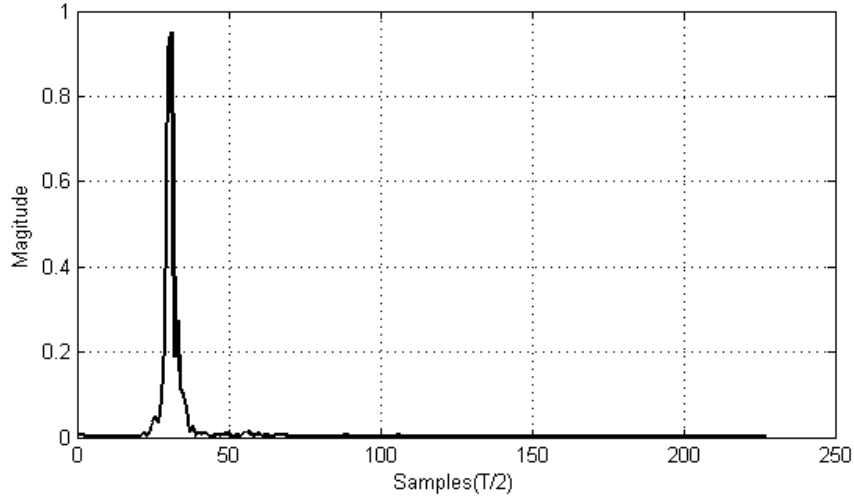
Figure 6.7 – Magnitude of impulse response for SPIB #12 microwave channel.

QAM is $\mu = 2^{-14}$, and the VL-CMA algorithm's parameters are selected according to [21] with $\Delta = 3$, $\alpha = 0.00025$, $\delta = 1$ and $\gamma = 0.0095$. The filter tap-length was initialized to $L(0) = 16$. As seen from Fig. 6.9, the the tap-length of proposed algorithm converged to the value of $L = 14$ and consequently the estimated optimal length is approximately $L_{opt} = 14 - \Delta = 11$ taps [21]. MSE of the proposed VL-CMa algorithms against fixed length GSA, CMA and MMA algorithm using 64-QAM is plotted in Fig. 6.8, where it shows a rapid convergence of the proposed algorithm to and approximately same MSE as compared with other algorithms. The proposed equalizer's output signal constellation for $64-$-QAM at steady state is illustrated in

85

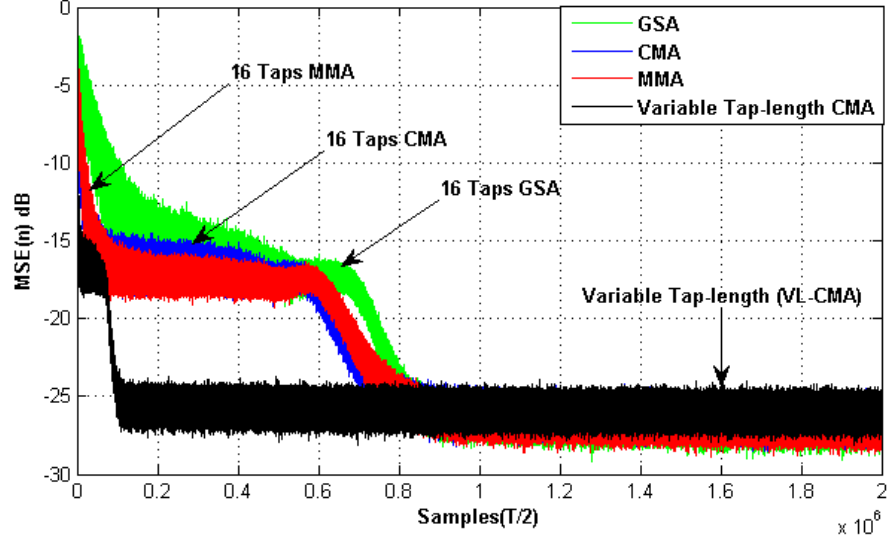**Figure 6.8** – Simulation results of MSE(n) of proposed algorithm against other blind equalization algorithms for 64-QAM.

Fig. 6.10, which shows that the VL-CMA algorithm has achieved a recognizable constellation points in shorter time and lower number of taps than other algorithms while attaining the similar misadjustment.
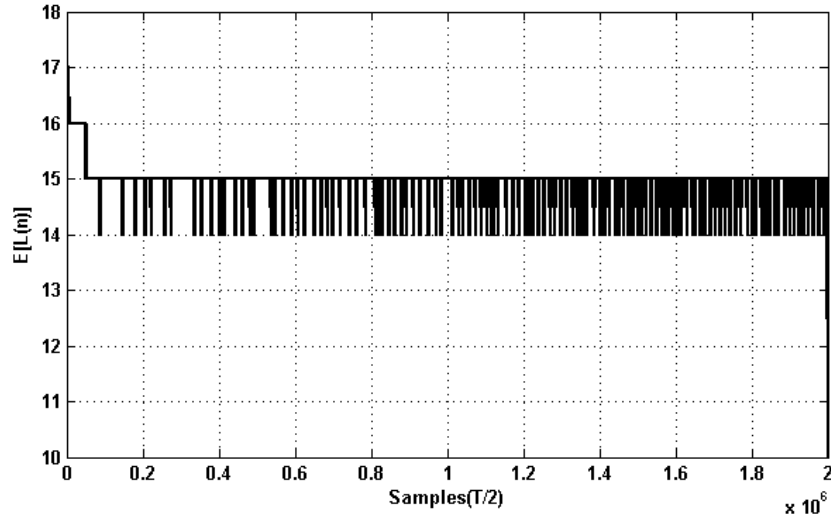

Figure 6.9 – The expected value of the proposed VL-CMA algorithm for 64-QAM.
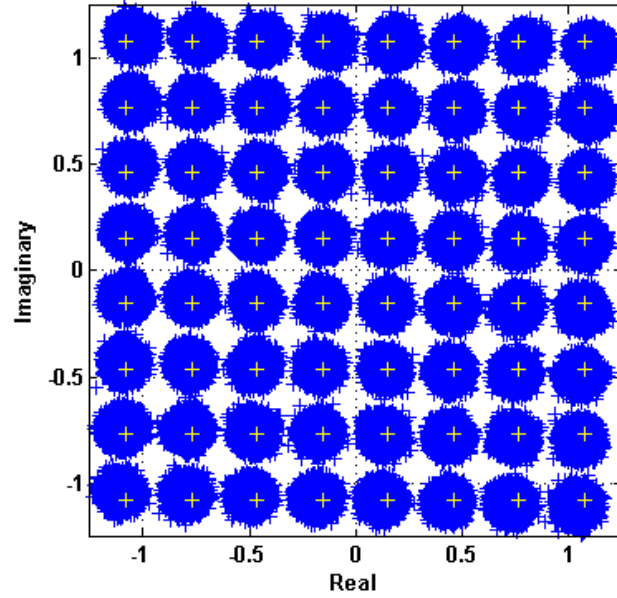
Figure 6.10 – Output signal constellation of the proposed VL-CMA algorithm for 64-QAM.

6.6 Conclusion

A novel variable tap-length CMA blind equalizer was proposed. This algorithm and with little extra complexity can be used to estimate the equalizer optimal length. The proposed algorithm was compared using 16-QAM and 64-QAM with some other blind algorithms using 16-QAM and 64-QAM constellations. The proposed algorithm has shown superior convergence properties as it was able to distinguish the sequence sent by the source and identify the constellation unambiguously.

References

[1] D. Kari, N. D. Vanli, and S. S. Kozat, "Adaptive and efficient nonlinear channel equalization for underwater acoustic communication," *Physical Communication*, vol. 24, pp. 83 – 93, 2017.

[2] S. J. Nanda and N. Jonwal, "Robust nonlinear channel equalization using wnn trained by symbiotic organism search algorithm," *Applied Soft Computing*, vol. 57, pp. 197 – 209, 2017.

[3] P. K. Mohapatra, P. K. Jena, S. K. Bisoi, S. K. Rout, and S. P. Panigrahi, "Channel equalization as an optimization problem," in *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, Oct. 2016, pp. 1158–1163.

[4] J. J. Werner, J. Yang, D. D. Harman, and G. A. Dumont, "Blind equalization for broadband access," *IEEE Communications Magazine*, vol. 37, no. 4, pp. 87–93, Apr. 1999.

[5] H. F. Lee, J. T. Yuan, and T. C. Lin, "Iir lattice-based blind equalization algorithms," in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, Sept 2012, pp. 1–5.

[6] D. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867–1875, Nov. 1980.

[7] J. Treichler and B. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 2, pp. 459–472, Apr. 1983.

[8] L. He, M. G. Amin, C. Reed, and R. C. Malkemes, "A hybrid adaptive blind equalization algorithm for qam signals in wireless communications," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 2058–2069, July 2004.

[9] J. R. Treichler, M. G. Larimore, and J. C. Harp, "Practical blind demodulators for high-order qam signals," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1907–1926, Oct. 1998.

[10] J. R. Treichler, I. Fijalkow, and C. R. Johnson, "Fractionally spaced equalizers," *IEEE Signal Processing Magazine*, vol. 13, no. 3, pp. 65–81, May 1996.

[11] F. Riera-Palou, J. M. Noras, and D. G. M. Cruickshank, "Linear equalisers, with dynamic and automatic length selection," *Electronics Letters*, vol. 37, no. 25, pp. 1553–1554, Dec. 2001.

[12] Z. Liu, "Variable tap-length linear equaliser with variable tap-length adaptation step-size," *Electronics Letters*, vol. 50, no. 8, pp. 587–589, April 2014.

[13] Y. Gong, X. Hong, and K. F. Abu-Salim, "Adaptive mmse equalizer with optimum tap-length and decision delay," in *Sensor Signal Processing for Defence (SSPD 2010)*, Sep. 2010, pp. 1–5.

[14] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation.* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2007.

[15] Y. Gong and C. F. N. Cowan, "An lms style variable tap-length algorithm for structure adaptation," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2400–2407, July 2005.

[16] N. Li, Y. Zhang, Y. Zhao, and Y. Hao, "An improved variable tap-length lms algorithm," *Signal Processing*, vol. 89, no. 5, pp. 908–912, 2009.

[17] Y. Zhang, J. A. Chambers, S. Sanei, P. Kendrick, and T. J. Cox, "A new variable tap-length lms algorithm to model an exponential decay impulse response," *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 263–266, Apr. 2007.

[18] K. Banovic, E. Abdel-Raheem, and M. A. S. Khalid, "A novel radius-adjusted approach for blind adaptive equalization," *IEEE Signal Processing Letters*, vol. 13, no. 1, pp. 37–40, Jan. 2006.

[19] R. Johnson, P. Schniter, T. J. Endres, J. D. Behm, D. R. Brown, and R. A. Casas, "Blind equalization using the constant modulus criterion: a review," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1927–1950, Oct. 1998.

[20] http://spib.rice.edu, "Signal processing information base," 2005.

[21] Y. Zhang, N. Li, J. A. Chambers, and A. H. Sayed, "Steady-state performance analysis of a variable tap-length lms algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 839–845, Feb. 2008.

[22] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Transactions on Communications*, vol. 23, no. 6, pp. 679–682, June 1975.

[23] G. K. C. K. C. John J. Shynk, Richard P. Gooch, "Comparative performance study of several blind equalization algorithms," pp. 1565 – 1565 – 16, 1991.

[24] and J. . Werner and G. A. Dumont, "The multimodulus blind equalization and its generalized algorithms," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 5, pp. 997–1015, June 2002.

Chapter 7

Conclusions and Future Recommendations

7.1　Conclusions

The major contribution of this dissertation is to present new and novel algorithms for optimizing adaptive filtering structure in a variety of applications using variable tap-length. The proposed algorithms are:

1. Fractional Order Lattice Prediction Filter (FO-LPF)

2. Fractional Tap-length Lattice LMS Filter (FT-LLMS)

3. Fractional Tap-length Lattice RLS Filter (FT-LRLS)

4. Variable Length CMA Blind Equalizer (VL-CMA)

In all demonstrated algorithms, pseudo fractional tap-length scheme was incorporated in lattice structured of LMS prediction, LMS system identification and RLS system identification. In LMS lattice prediction, the forward residual was used to update the fractional variable tap-length iterations and variable order lattice predictor method was suggested (FO-LPF), in which, the proposed filter's frequency repose showed superiority, in terms of passing the desired signal and attenuating all other frequencies with smoother ripples, when compared with other fixed length predictors and the optimal tap-length of the proposed predictor was estimated using system simulation and averaging the results. In lattice structured LMS and RLS system identification application, two new algorithms were proposed, FT-LLMS and FT-LRLS respectively, both are based on modifying the error signal to update variable length recursions. In

this application, the system simulation was performed to test the proposed algorithms in a variety of conditions, namely, low SNR of 10 dB, high SNR of 30 dB. The proposed algorithms showed noticeable improvement in MSE level as well as in the convergence rate properties. The proposed algorithms can be used to estimate the filter optimal tap-length and therefore enhance the efficiency of these applications. Finally, a new variable tap-length algorithm for CMA blind equalization was developed (VL-CMA) for QAM modulated signals. In which the non-linear error $e(n)$ of the CMA equalizer output $y(n)$ was modified to update FT iterations within the initial operation of the equalizer. In all proposed algorithms, structure adaptation added another dimension to adaptation process. This is because the coefficient's weight and tap-length are achieved simultaneously, which can be considered as addition to the adaptive filtering efficiency.

7.2   Future Recommendations

There are several ways to expand and develop the work of this thesis. The proposed FT lattice based algorithms can be used as a foundation to utilize it toward other lattice-form applications. This task can be achieved by studying the application under investigation and determine the suitable error signal, and its properties to employ it in variable tap-length adaptation formulation. Other adaptive filter forms such as IIR filters can also be examined for the structure adaptation optimization using variable tap-length. In the context of blind equalization, the proposed VL-CMA algorithm can be used as explorer to extend it to other blind equalization algorithms. Blind equalization can also be realized using lattice structure and optimizing such a realization by variable tap-length methods. Also, variable length tap algorithm can be applied to fully pipelined direct form filtering.

Appendix : Copyright permission

Dear Dr. E. Abdel-Raheem,

Dear Dr. Khaled Mayyas,

Dear Dr. M. Khalid,

I hope you receive this email and you are doing well. I was so overjoyed to work with you. I am writing to you today to ask for your permission to refer our publications in my Ph.D. thesis, according to graduate studies regulations which requires a statement of declaration from co-authors.

Thank you and best regards,

Salem Alsaid.

Dear Salem

Please go ahead. I provide my permission. regards

Esam Abdel-Raheem PhD, PEng

Professor

Dear Salem Alsaid,

This is to let you know that I give the permission for you to refer to our common publications in your Ph.D. thesis. Regards

Khaled Mayyas

Professor of Electrical Engineering

Dear Salem:

You have my permission.

Regards,

Khalid

Vita Auctoris

NAME	:	Salem Alsaid
BIRTH YEAR	:	1974
BIRTH PLACE	:	Libya
EDUCATION		
2019	:	**Doctor of Philosophy (Ph.D)**
		Electrical and Computer Engineering
		University of Windsor, Windsor, Canada.
2004	:	**Master (M.Sc.) of Computing**
		Electrical and Computer Engineering
		Bradford University, Bradford, UK.
1998	:	**Bachelor (B.Sc.) of Electronic Engineering**
		College of Electronic Engineering
		Nasser University, Libya.