PARAMETER ESTIMATION AND TRACKING OF SINUSOIDAL SIGNAL USING VARIABLE STEP-SIZE LMS ALGORITHMS

Submitted towards the partial fulfilment of requirement for the award of degree of

Master of Engineering

In

Wireless Communication

Submitted by:

ROHIT GARG

(801463021)

Under the guidance of:

Dr. Amit Kumar Kohli

(Associate Professor, ECED)

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

(Established under the section 3 of UGC Act, 1956)

PATIALA – 147004, PUNJAB, INDIA

JULY 2016

DECLARATION

I, **Rohit Garg**, hereby declare that the work, which is being presented in the thesis entitled "*Parameter Estimation and Tracking of Sinusoidal Signal using Variable-Step-Size LMS Algorithms*" by me in partial fulfilment of the requirements for the award of degree of M.E. (Wireless Communication) at Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of **Dr. Amit Kumar Kohli**, Associate Professor, Electronics and Communication Engineering Department.

The matter presented in this thesis has not been submitted to any other University/Institute for the award of any other degree.

Date: 05/July/2016

Rohit Garg

Rohit Garg

(801463021)

It is certified that the above statement made by the student is correct to the best of my knowledge and belief.

Date:

Dr. Amit Kumar Kohli

Associate Professor, ECED

Thapar University, Patiala

12/7/16

Countersigned by:

(Dr. Sanjay Sharma)

Professor and Head, ECED

Thapar University, Patiala

(Dr. S. S. Bhatia)

Dean of Academic Affairs

Thapar University, Patiala

ACKNOWLEDGMENT

It is my proud privilege and delectableness to express the deepest sense of gratitude to several people for their indispensable role that made completion of this work possible.

First and foremost, I express my heart full indebtedness and owe a deep sense of gratitude to my guide **Dr. Amit Kumar Kohli**, Associate Professor (P.G. Coordinator, Electronics and Communication Engineering Department) for the guidance, motivation and inspiration that triggered me for the work. Without their invaluable cognitive and rational guidance and support, this work would not have reached upto such extent.

I am thankful to **Dr. Sanjay Sharma**, Professor and Head of the Department, Electronics and Communication Engineering Department for availing me with the infrastructure and best faculty needed for this research work. I would like to express my utmost regards for the ECE department of my institute for helping me in my lean patches.

I would like to thank Shiv Sharma, Sonali and Shivani Sehrawat for their helpful suggestions.

Last but not the least I owe my deep regards to my parents for their unyielding love and encouragement.

This research work enhances my knowledge and avenues related to signal processing.

ROHIT GARG
Roll No. : 801463021

ABSTRACT

The current advancements in technologies like voice prediction, noise filtering, and system identification are possible only because of variety of tools present in discrete-time-signal-processing (DSP). But if the system is designed with some particular conditions, then it is not always possible that the system will respond in the same way under different conditions. So, we need to make it adaptive using some factor so that it can adapt itself accordingly. As in the case of filtering, if the coefficients of the filter are predefined then the system response remains constant in particular conditions and we cannot directly implement it in the real world scenario as the conditions change on regular basis. Taking into account these restrictions, it is recommended to use some adaptable system that can adapt itself according to the real world conditions. In these systems, the coefficients are made adaptable. In this research work, system identification by using signal received at two wireless sensors is carried out. The problem of adaptively estimating the time delay of sinusoidal signals received at two spatially separated sensors is a bit different from real scenario. Thus, relative amplitude and delay of the sinusoidal signal are needed to be estimated vis-a-vis a reference sinusoidal signal using finite-impulse-response (FIR) filter. There is always a tradeoff between convergence rate and misadjustment. To eliminate this dilemma, variable-step-size (VSS) methods are proposed. The VSS algorithm has a big step-size at the beginning for a maximum convergence speed and a much smaller step-size after the convergence for a minimum residual error.

In this thesis, I present the performance evaluation of different VSS algorithms, which are used for parameter estimation of sinusoidal signal. The VSS in least-mean-square (LMS) algorithm is not only varied based on the error and input signal correlations, but also it can be varied by using sigmoid function as an alternative approach. The performance of VSS-LMS algorithm is appraised on the basis of convergence and tracking characteristics, in terms of mean-squared-error (MSE), when delay, as well as amplitude, need to be estimated. Simulation results are presented to demonstrate the efficiency and efficacy of both types of VSS criteria in combination with LMS algorithm.

Keywords: Adaptive filters, adaptation characteristics, delay estimation, amplitude estimation, misadjustment, signal-to-noise-ratio (SNR).

TABLE OF CONTENT

LIST OF ACRONYMS AND ABBREVIATIONS

AGC	Automatic-Gain-Control
AVSS	Aboulnasr-Variable-Step-Size
ASDF	Average-Squared-Difference-Function
AWGN	Additive-White-Gaussian-Noise
CC	Cross-Correlation
DTFT	Discrete-Time-Fourier-Transform
DSP	Discrete-Time-Signal-Processing
FIR	Finite-Impulse-Response
FSS	Fixed-Step-Size
KVSS	Kwong-Variable-Step-Size
LMS	Least-Mean-Square
MAVSS	Modified-Aboulnasr-Variable-Step-Size
ML	Maximum-Likelihood
MSE	Mean-Square-Error
MSVSS	Modified-Sigmoid-Variable-Step-Size
NLMS	Normalized-LMS
PHAT	Phase-Transform
RLS	Recursive-Least-Square
SCOT	Smoothed-Coherence-Transform
SNR	Signal-to-Noise-Ratio
SVSS	Sigmoid-Variable-Step-Size
QDE	Quadrature-Delay-Estimator

LIST OF FIGURES

Chapter 1

INTRODUCTION

This chapter includes the introduction of adaptive systems and its parameters, various applications of adaptive system like system identification and modeling, prediction, echo cancellation for long-distance communication, adaptive noise cancelling etc. Different performance measures for adaptive systems are also reviewed.

1.1 Introduction

Discrete-time-signal-processing (DSP) has developed manyfold in recent times due to significant advancements in the digital and fabrication technologies. Signal processing is the representation and interpretation of signal that carries information and transformation of the signal to make it usable. So, nowadays, sophisticated complex signal processing tasks and circuits can be implemented by writing algorithms. The major advantage of using digital system is the degree of flexibility in algorithms that one can change according to its requirement. One of the commonly used digital systems is filter that processes the input and extracts output only we are interested in. Thus, filtering is simply the mapping of input signal to output signal according to some fixed criteria. So, if the specifications (transition bands, desired bands, passband ripple and stopband ripples) are known then, accordingly rational transfer function can be computed and selection of form and structure can be developed. But in the real scenario, either the specifications are not given or are time-variant then it is not possible to have desired output unless some changes are made to the digital filter. The only solution to this problem is to employ digital filter with adaptive coefficients i.e., adaptive systems that keep on updating its parameters (coefficients).

1.2 Characteristics of adaptive system

1. The system is self-optimized in changing environment.

2. The system can be used in time-varying or unknown environment but at the cost of complexity of the circuit.
3. These systems need not to be synthesized again and again as these are self-designing systems.

1.3 Applications of adaptive system

The adaptive systems are employed in communication systems, navigation systems, biomedical electronics, seismology and many more. According to the usage, these applications are classified as system identification and modeling, prediction, de-convolution, equalization, noise cancelling etc.

1.3.1 System identification and modeling

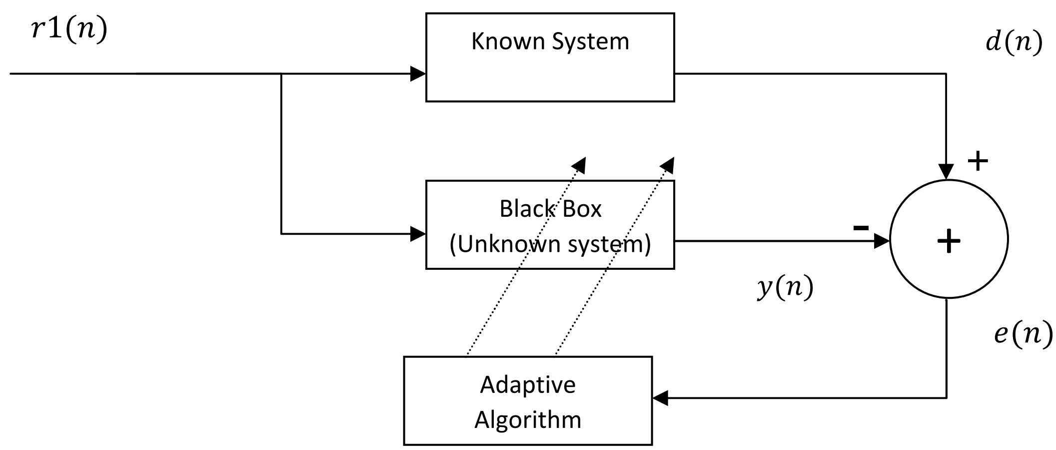

Fig. 1.1. System identification and modeling [1].

The input signal $r1(n)$ is applied simultaneously to known and unknown system. To reduce the value of $e(n)$, the adaptive algorithm emulate the unknown system coefficients. The coefficient value of the known system and black box may vary, but the transfer characteristics of both the systems are same. But in practical cases, the black box is affected with additive noise which is uncorrelated to the input signal. The adaptive filters have the flexibility to catch the dynamic response, thus, its output matches exactly when the mean-square-error (MSE) is reduced to minimum value.

This application is used in communication because when the signal is transmitted, multipath propagation takes place when it reaches the receiver side, thus, introduces echoes, delays, and interferences in the signal. Along with this, when the signal is affected by noise then it confuses the predictor circuit i.e., cross-correlator.

It is used in the exploration of gas and oil reservoirs. It uses seismic technology for detection. Generally speaking, the oil and gas fields are covered with clay or shale. The impulses are produced on earth surface that propagate through the water to the sea bottom and deep into the earth. As the geophysical material varies, the reflection takes place and is measured at the top to summarize the result and model a system whose response matches the desired result. The time-delay is measured in the case of SONAR and RADAR.

1.3.2 Prediction

Prediction is used to reduce the impact of noise and signal encoding. Prediction is basically the evaluation of current signal value based on past values. The delayed signal is passed through the unknown system with the adaptive algorithm and is compared with the input signal, therefore, try to reduce the error signal and predict the current value of input signal.

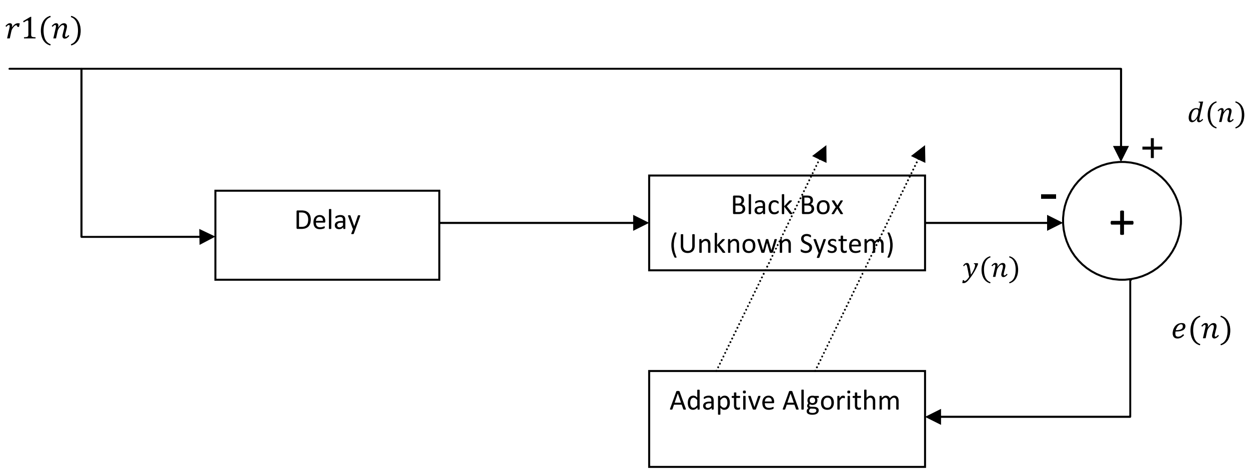

Fig. 1.2. Application of adaptive system as predictor [1].

1.3.3 Echo cancellation for long-distance transmission

In circuit theory, impedance matching is mandatory to transmit maximum power from source to load. If in case, the impedance is not matched then reflection takes place, thus, reduces the power delivered to the load. In a communication system, these reflections produce the echo effect. To compensate this echo effect, the impedance of the system must be made adaptable.

The long distance transmission in wired communication induces delay with different values when receiver side is an array of antennas. Adaptive systems can be used to estimate the delay value to increase the gain and clarity of voice at the receiver side.

1.3.4 Adaptive noise cancelling

Whenever the signal is transmitted with wireless mode, the signal is corrupted with different additive noise. Thus, we have two signals at hand, one signal with noise and another is correlated noise. When both inputs are compared and adaptive algorithm is applied, then the actual signal that is transmitted can be extracted out.

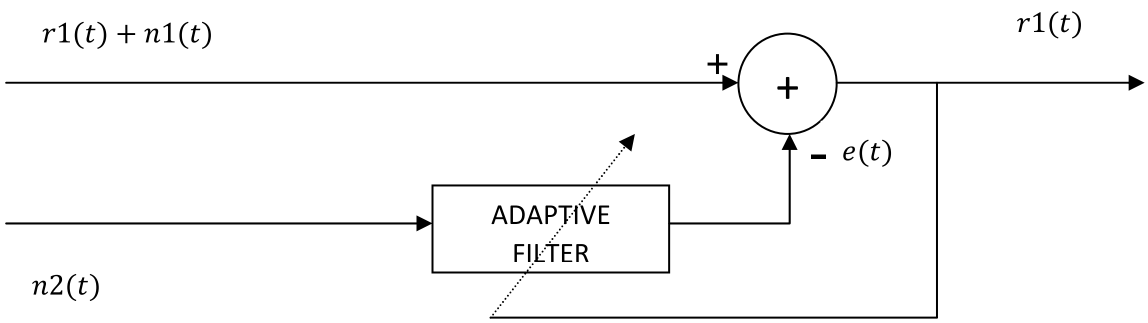

Fig. 1.3. Application of adaptive system as noise cancelling [2].

1.4 Performance measures in adaptive system

Performance measures in adaptive filters are convergence rate, minimum MSE, computational complexity, stability, robustness, and filter length.

- **Convergence Rate**: The convergence rate characterises the speed at which the filter converges to its resultant state. It depends on the nature of performance surface. Usually, a faster convergence rate is the desired characteristic of an adaptive system. It is inversely proportional to time constant that determines the time taken by the algorithm to decay to $1/e$ of the starting value. There is always a tradeoff between convergence rate and misadjustment. If the convergence rate is increased, then the steady state error also increased. Likewise, if the minimum steady state error is prime requirement, then the convergence rate has to be reduced.
- **Minimum Mean Square Error (MSE)**: It determines how accurately the system is modeled and predicted. The squared difference between the output of modeled system and optimal solution is known as MSE. Large value of MSE indicates that either the adaptive filter is not modeled accurately or the initial values of the system are not adequate to cause the adaptive filter to converge. The MSE curve can be divided into two parts i.e., transient response and Tracking. Transient response corresponds to that part of curve, where the output of the system response reaches the minimum permissible error. Tracking part starts after transient response. It indicates the ability of the system to how quickly the system responds to the change in input and reflects back in output.
- **Robustness**: Robustness is a performance measure that indicates how well the system can resist both input and quantization noise. It is directly proportional to stability of the system.
- **Filter Length**: Filter length directly influences the ability of system in terms of convergence rate, stability etc. If the filter length is increased, then the computational complexity also increased because the number of calculations at each iteration increased. Thus, the convergence rate reduces. Conversely, if the length of filter is decreased then the number of calculations decreased but steady state error increased manyfold. By increases the length of filter, the designer adds poles and zeros in the response that

increases the stability but at the cost of decrease in convergence rate. The increase in filter length increases the accuracy of the system.

- **Computational Complexity**: The hardware limitation always affects the implementation of real time system. As the complexity level of algorithm is increased, there is proportional increase in the hardware resource requirement. Thus, computational complexity is of immense importance performance measure for designer.
- **Stability**: Stability for adaptive system can be defined as the ability of system to provide bounded output for bounded input. There are very few completely asymptotically adaptive stable systems that can be practically realized. In most cases, the systems that are implemented are marginally stable, i.e., the system is verse on stability. The designer has to specify the initial values, environment condition etc in case of marginally stable system for their accurate functionality.

1.5 Motivation

Estimation and detection theory plays a vital role in study and development of wireless technology. The theory deals with the extraction of information from signal buried deep in noise. In wireless communication, the channel is free space i.e., air, so unwanted and stochastic noise comes into play. Estimation theory provides with approximate inferences from which one can estimate system parameters to check the performance. Time-delay estimation is in the same league.

Considering the case of RADAR, the pulsed electromagnetic signal is transmitted and measures the time for the first echo received at the receiver end. This time-delay between transmitted and received signal determines the pulse rate for radar. If the pulse rate is higher than the inverse of time-delay, then false alarm condition occurs. These parameters in radar estimate the position and velocity of the target. Moreover, in a communication system, the space diversity technique is used to increase the gain at the receiver side. In this diversity technique, the signal is summed with proper phase from two or more antennas.

The problem doesn’t end here. Once the parameters are known, the system is designed according to some specific set of conditions. Thus, the system is unable to work in worse conditions. So to eliminate this, adaptive systems are modeled in which parameters can adapt itself according to the environmental conditions, so there is no need to redesign the whole system.

Practically, it is impossible to check the system capability and response in different environmental conditions. Simulators are the solution to this limitation. Simulator impersonates the real process and helps the researcher to analyze the results without an actual field test.

1.6 Thesis objective

- To discuss the different adaptive step-size LMS (least-mean-square) algorithms developed with the passage of time to increase the convergence rate vis-a-vis reduction in misadjustment in comparison with fixed step-size LMS. To estimate the parameters of the sinusoidal signal at two spatially separated sensors under different SNRs. To analyze the tracking ability of FSS (fixed-step-size), KVSS (Kwong-variable-step-size), AVSS (Aboulnasr-variable-step-size), MAVSS (modified-Aboulnasr-variable-step-size), SVSS (Sigmoid-variable-step-size) and MSVSS (modified-Sigmoid-variable-step-size) using LMS algorithm with simulation results.

1.7 Organization of thesis

- **Chapter 2: Literature Survey:** It entails the literature survey for different VSS algorithm and time-delay estimation techniques that can be exploited in the adaptive system.
- **Chapter 3: Time-Delay Estimation:** It is the concise description of time-delay estimation and techniques developed so far for estimation. Block diagram for these techniques and formulae are discussed. It introduces adaptive filter algorithm, its formulation with different gradient search methods and parameters to check the capability of the system. In the end, LMS algorithm is reviewed.
- **Chapter 4: Adaptive Delay and Amplitude Estimation using LMS algorithm:** This chapter includes the system identification for amplitude and delay estimation with the adaptive algorithm and various VSS algorithms that can enhance the system estimation accuracy.

- **Chapter 5: Adaptive Delay and Amplitude Estimation using VSS-LMS algorithms:** It is the core of this research work. The proposed methodology for system identification paradigm of delay and amplitude estimation using VSS algorithms is discussed.
- **Chapter 6: Simulation Results and Discussion:** It addresses the performance of different algorithm discussed in chapter 5 with the help of simulation results in MATLAB. MSE of delay and amplitude along with tracking ability of system is taken into consideration.
- **Chapter 7: Conclusion and Future Scope:** It is the conclusion of this work and future scope that can be carried out to uplift this research work.

Chapter 2

LITERATURE SURVEY

This chapter entails the literature survey for different VSS algorithms and time-delay estimation techniques that can be exploited in the adaptive system.

In [3], Chakraborty has proposed the adaptive filter to estimate the mean delay and amplitude of the signal received at two spatially separated sensors. Convergence analysis for the mean delay and amplitude is carried out. Stability condition for the algorithm is computed. Simulation results with different SNR are discussed.

In [4], Kwong and Johnston have modeled a new adaptive filter with VSS (variable-step-size) algorithm to increase the convergence rate while maintaining the level of misadjustment same as in the case of FSS. Initially, the step-size is large so that the system can reach steady state in lesser iterations. As the result leads to steady state, the step-size is going to decrease, so misadjustment remains the same. The algorithm is comparatively less sensitive to non-stationary conditions so the effect on misadjustment reduces. The authors have also computed the approximate formulae for stationary and non-stationary conditions. The simulation results obtained are approximately equal to theoretical results.

In [5], Aboulnasr and Mayyas have reviewed VSS algorithm that reduces the sensitivity of the system to uncorrelated noise disturbance. The authors use autocorrelation of the error signal to compute the step-size of next iteration. The simulation results indicate that the proposed algorithm works well in low and high SNR (signal-to-noise-ratio) condition with high convergence rate while maintaining same excess MSE. In non-stationary conditions, the algorithm response is more or less same.

In [6], Kun and Xiubing have modeled a new VSS algorithm with good tracking and anti-noise ability. The algorithm depends on error autocorrelation that is considered to be independent of additive-white-Gaussian-noise (AWGN). Along with this, the instantaneous error power controls the tracking rate of the system. As the error power reduces, the value of step-size reduces. The author compares the result of proposed algorithm with VSS algorithm proposed in [4] and [5].

In [7], Chen *et al.* have established a non-linear relation between the step-size and error signal based on the translational transformation of the sigmoid function. The parameters for selection of step-size are computed. The effect of interference reduces to a great extent due to the non-linear relation.

In [8], Carter has discussed the maximum-likelihood (ML) estimation as a family member of generalized cross-correlation. The performance of algorithm is synthesized taking into account low and high SNR value. The parameters required for time-delay estimation are reviewed.

In [9], Knapp and Carter have studied the ML estimator for time-delay by evaluating the correlation maximum in time argument. At low SNR, the results are comparatively same with respect to Eckart filter.

In [10], the time domain representation of Roth and Scot algorithm is derived. These algorithms are well suited for time-varying environments. The noise assumed to be Gaussian.

In [16], So has presented two algorithms to evaluate the phase delay estimation by taking into consideration all the phase and quadrature component of the received signal. This method eliminates the bias introduced in quadrature-delay-estimator (QDE). The second algorithm utilizes the difference between the discrete-time-fourier-transform (DTFT) of two signals to estimate the phase. The second algorithm has large operational range of SNR.

In [17], Maskell and Woods have examined the technique to estimate the time-delay of the periodic signal by using the analog quadrature method. This technique can be employed in RADAR to measure the grouping delay. Simulation results indicate that the response of proposed techniques is outperforming other estimators with little bias and exhibit excellent performance when SNR varies from medium to high.

In [18], Nandi has discussed the common problem that occurs in correlation based time-delay estimators due to misidentification of extremum. The author has proposed two algorithms MSX and MXS that can be used in narrowband signal for the estimation of subsample time-delay. The algorithm computes the correlation and autocorrelation at different time lags then recombine to estimate the accurate delay value. The proposed algorithm performs better in terms of bias and variance at low SNR as compared to average-squared-difference-function (ASDF) and direct-correlator (DC) estimator. The simulation results with ultrasonic echoes at different SNR verify the theoretical results.

In [29], Dass and Chakraborty have reviewed the convergence of sparse adaptive filter using proportionate normalized-least-mean-square (NLMS) algorithm. The authors use transform based model and the random vector is angularly discretized thus eliminate the assumption of white noise. So the evaluation results for MSE are universal. Also, the relation between gain and corresponding tap value is derived.

In [30], Lee *et al.* have proposed the distributed estimation using diffusion VSS-LMS algorithm. The algorithm computes the suboptimal step-size for network application at each node, thus, reducing the dependency on central fusion center. This can be employed in different node profile network and is insensitive to link failure.

In [31], Paziatis and Constantinides have modeled algorithm using kurtosis of estimation error that reduces the sensitivity to strong noise to a great extent. There is an increase in circuit complexity. Comparison with FSS-LMS and another VSS-LMS algorithm is carried out. The modeled algorithm increases the tracking ability of the system.

In [32], Harris *et al.* have discussed version for LMS algorithm with new adaptation procedure of step-size. In the proposed method, the authors use the sign of gradient to check the direction of adaptation. The upper and lower bounds for step-size are computed. Thus, the convergence rate remarkably increases.

In [33], Gelfand *et al.* have reviewed the stability of VSS algorithm in uncorrelated noise. Entire priori and posteriori step-size sequences are focused on analyzing stability. The authors demonstrated the general assumption of step-size that the variable step can be truncated when the step-size leads to fixed step-size are wrong. It is much more complex than assumed.

In [34], Foley and Boland have presented an analysis of adaptive filter with the Gaussian input signal and investigated the root bounds and compared the results with conditions suggested by Horowitz and Senne that uses fourth-order moment.

In [35], Mayyas has discussed the practical situation that occurs in LMS (finite-impulse-response) FIR filters where the length of the filter is not equal to the impulse response of the unknown system. The mean square analysis in transient and steady state is developed. When the input signal is assumed to be white then the system response is same when compared with filter of sufficient length. An exact expression for computing the excess MSE from update equation is obtained.

In [36], Mathews *et al*. have presented the LMS with gradient descent algorithm in order to reduce the square error estimation. The change in weight coefficient is based on the negative gradient of squared error so step-size is time varying, thus decreases the sensitivity of the algorithm to the change in parameters. In non-stationary conditions, the filter approaches to the best possible performance of LMS and recursive-least-square (RLS) algorithm. Analytical and simulation results are discussed for the same.

In [37], the automatic-gain-control (AGC) scheme that is widely used in communication systems is discussed. The system with adaptive gain reduces the vulnerability to noise. Basically, the correlation is computed for input and error signal, if the cross-correlation is high, the gain value comes out to be high and considered the algorithm is in active state. But, when this cross-correlation is low or zero then the algorithm is in asleep mode and gain is zero. The author has reviewed the scheme with RLS algorithm.

Chapter 3

TIME-DELAY ESTIMATION

This chapter is the concise description of time-delay estimation and techniques developed so far for estimation. Block diagram of these techniques and formulas are discussed. This chapter is dedicated to adaptive filter algorithm, its formulation with different gradient search methods and parameters to check the capability of the system. In the end, LMS algorithm is also reviewed.

3.1 Introduction

Time-delay estimation [8] problem is mainly found in control engineering and signal processing. A lot of variations in this problem can exist like the signal may be real or complex, narrowband or wideband etc. When the noise assumed is zero, then it is considered as active time-delay estimation and is generally used in automatic control and range estimation. The enormous amount of research is going on in the various approximations of active time-delay estimation like time-domain approximation, frequency-domain approximation, Laguerre-domain approximation etc.

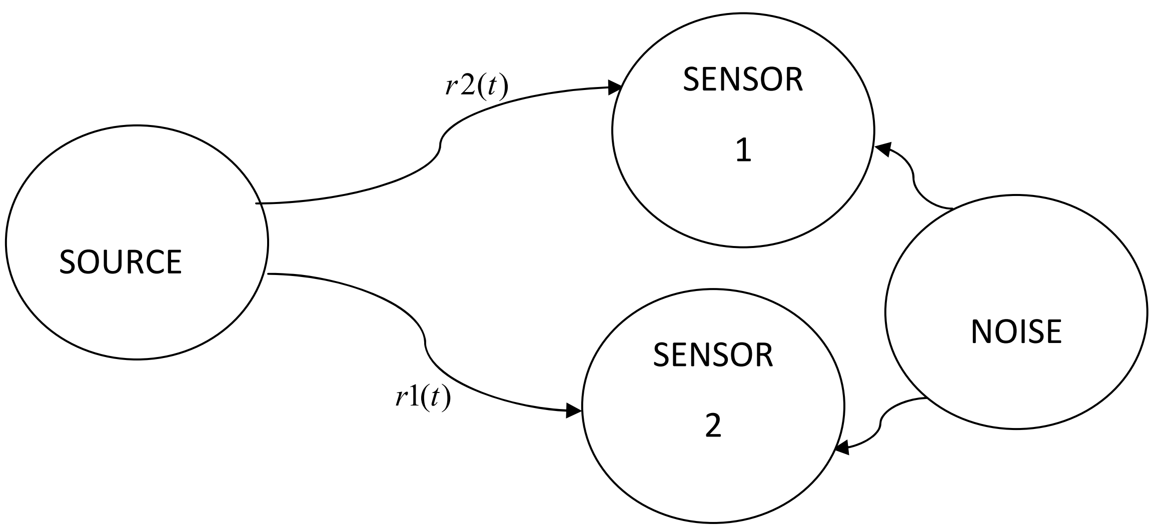

Fig. 3.1. Time-delay estimation problem [8].

When the noise is assumed to be non-zero, then it is classified as passive time-delay estimation. In this case, the signal is assumed to travel different distances for the two sensors. The signal received at two spatially separated sensors is assumed as $r1(t)$ and $r2(t)$.

3.2 Time-delay estimation techniques

3.2.1 Cross-Correlation Method [9]: As the name suggests, the cross-correlation (CC) between the two received signals is calculated and considering the time argument which indicates the maximum peak of the output signal is estimated time-delay.

$$CC(d) = E[r1(t).r2(t-D)] \tag{3.1}$$

$$Delay(d) = \arg\max[CC(d)] \tag{3.2}$$

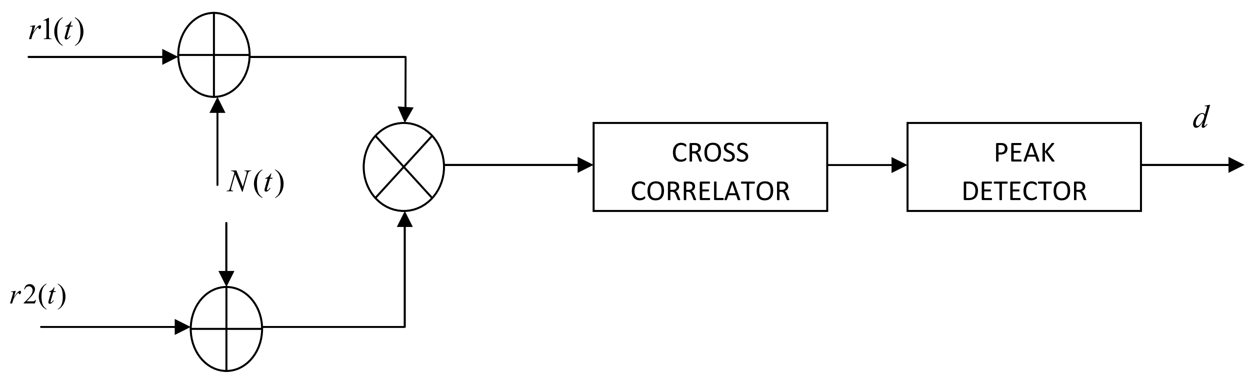

Fig. 3.2. Cross-correlation method [9].

3.2.2 Generalized cross-correlation [9]: This algorithm is modeled by Knapp and Carter in 1976. In this method, the input signal is whitened by using some weighting function. It avoids the correlation peak spreading. The weighting factor can have different values according to the algorithm used such as Smoothed-coherence-transform (SCOT), Phase-transform (PHAT), Eckart filter etc.

$$GCC(d) = \int_{-\infty}^{\infty} \delta(f) C_{r1r2}(f) e^{j2\pi fd} df \tag{3.3}$$

$$Delay(d) = \arg\max[GCC(d)] \tag{3.4}$$

where, $C_{r1r2}(f)$ is cross-spectrum of signal.

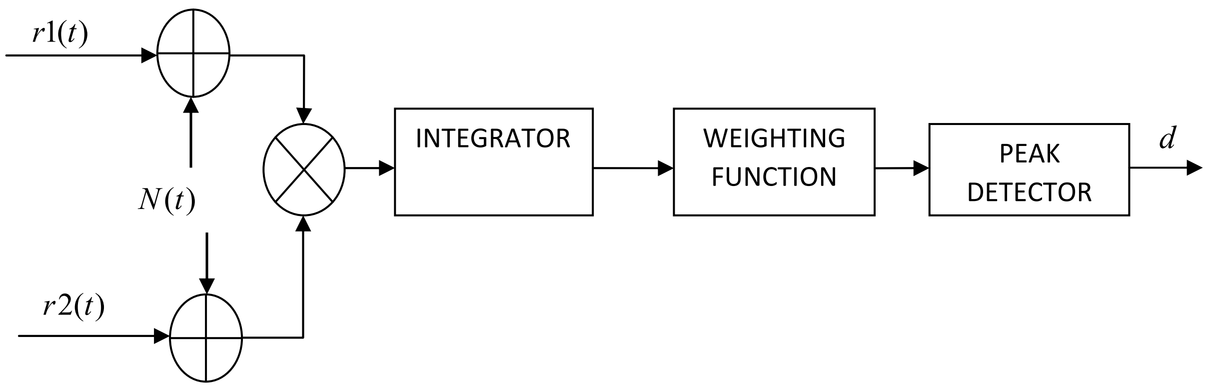

Fig. 3.3. Generalized cross-correlation method [9].

Weighting function

ROTH weighting function [10],

$$\delta(f) = \frac{1}{C_{r1r2}(f)} \tag{3.5}$$

where, $C_{r1r2}(f)$ is cross-spectrum of signal.

SCOT weighting function [10],

$$\delta(f) = \frac{1}{\sqrt{C_{r1r1}(f).C_{r2r2}(f)}} \tag{3.6}$$

where, $C_{r1r1}(f)$ *and* $C_{r2r2}(f)$ is autocorrelation of $r1(t)$ and $r2(t)$ signal.

PHAT weighting function [11],

$$\delta(f) = \frac{1}{|\, C_{r1r2}(f) \,|} \tag{3.7}$$

where, $C_{r1r2}(f)$ is cross-spectrum of signal.

Cps-m weighting function [9],

$$\delta(f) = \frac{1}{\sqrt[m]{C_{r1r1}(f).C_{r2r2}(f)}} \tag{3.8}$$

where, $C_{r1r1}(f)\ and\ C_{r2r2}(f)$ is autocorrelation of $r1(t)$ and $r2(t)$ signal.

Hannon and Thomson weighting function [12],

$$\delta(f) = \frac{(|\ C_{r1r2}(f)\ |)^2}{[|\ 1 - (C_{r1r2}(f)\ |]^2} \tag{3.9}$$

where, $C_{r1r1}(f)\ and\ C_{r2r2}(f)$ is autocorrelation of $r1(t)$ and $r2(t)$ signal and $C_{r1r2}(f)$ is cross-spectrum.

Eckart weighting function [13],

$$\delta(f) = \frac{|\ C_{r1r2}(f)\ |}{[|\ C_{r1r1}(f) - C_{r1r2}(f)\ |][|\ C_{r2r2}(f) - C_{r1r2}(f)\ |]} \tag{3.10}$$

where, $C_{r1r1}(f)\ and\ C_{r2r2}(f)$ is autocorrelation of $r1(t)$ and $r2(t)$ signal and $C_{r1r2}(f)$ is cross-spectrum.

The basic difference between the cross-correlation and generalized cross-correlation is that the calculation part is transformed to the frequency domain in case of generalized cross-correlation that enhances the accuracy and reliability of the system and reduces the computational cost.

3.2.3 Maximum-likelihood method [12]: It uses weighting function proposed by Hannon. Its popularity is due to relative simplicity and optimal solution in appropriate conditions. The idea behind this algorithm is to attenuate the signal where the SNR is lowest.

$$ML(d) = \int_{-\infty}^{\infty} \delta(f).\, C_{r1r2}(f) e^{j\pi f d} df \tag{3.11}$$

$$\delta(f) = \frac{(|\ \gamma_{r1r2}(f)\ |)^2}{|\ C_{r1r2}(f)\ |\ [|\ 1 - (\gamma_{r1r2}(f)\ |]^2} \tag{3.12}$$

$$Delay(d) = \arg\max[ML(d)] \tag{3.13}$$

$$(|\ \gamma_{r1r2}(f)\ |)^2 = \frac{(|\ C_{r1r2}(f)\ |)^2}{C_{r1r1}(f).\, C_{r2r2}(f)} \tag{3.14}$$

where, $C_{r1r1}(f)\ and\ C_{r2r2}(f)$ is autocorrelation of $r1(t)$ and $r2(t)$ signal and $C_{r1r2}(f)$ is cross-spectrum.

The weight functions are adjusted in accordance with variance, where the variance is least with respect to cross-spectral phase estimation.

3.2.4 Average-square-difference-function method (ASDF)*:* The delay is calculated by estimating the minimum MSE position and is assumed as the estimated delay. ASDF performs equally well when there is no noise, but in the case of correlation, there are some estimation errors. This algorithm doesn't need any multiplication, so circuit complexity reduces to a great extent. There is no need of information regarding input spectra.

$$C(d) = \frac{1}{N}\sum_{n=1}^{N}[r1(n) - r2(n-d)]^2 \tag{3.15}$$

$$Delay(d) = \arg\min[C(d)] \tag{3.16}$$

where, $C(d)$ is correlation.

3.2.5 Quadrature-delay-estimator (QDE): This method estimates the delay by taking into consideration all the phase and quadrature phase of the signal. This estimator is mainly employed in RADAR, SONAR, and digital communication.

3.2.6 Adaptive filter algorithm: This algorithm is based on an adaptive system, where the input signal is compared with some desired signal. The difference between the output from the filter and desired signal is known as an error signal. This error signal is then applied to adaptive algorithm, which may be LMS, RLS etc. Accordingly, the weights of the filter are varied at each iteration in order to reach the optimal solution. When the error signal moves to steady state, the weight factor is the delay estimated. The main advantage of using this method is that no prior knowledge of input signal spectra is required. Initially, the weight function is equated zero so error signal is maximum, but the error reduces with each iteration until the steady state is reached.

3.3 Adaptive filters

3.3.1 Introduction: An adaptive filter is a time-variant filter that can operate in unknown and time-varying environment. It is self-modifying digital filter that can tackle the problem of unknown conditions while designing the filter. There is no need of signal statistics in past, so lesser computational complexity. The parameters are evaluated in an iterative manner by modeling the relation between two signals. As the signal statistics are not available to the designer, the reference signal is selected to compare with error signal. The main features of adaptive filters are

1. **Filter structure**: It defines the relation between the signals, type and number of coefficients that are altered to synchronize the output with desired signal. Generally, FIR transversal filter or tapped delay line is used because of simplicity and efficacy.
2. **Adaptive algorithm**: It formulates problem mathematically tractable and the steps involved in the computation of the coefficient according to some optimization technique and modifies the coefficients in an iterative manner.
3. **Criteria for performance**: This choice is made by designer by taking in account the complexity, circuitry, system response, speed etc. Different criteria that can be used are: MSE, regularized MSE, Li norm criteria, least mean fourth etc. Minimum MSE is simple and elegant as it uses second-order statistics.

3.3.2 Adaptive filtering problem: In a single input filtering problem, the adaptive circuitry can be implemented using linear combiner and delay elements. The input signal is vector for n samples and is represented by [14]

$$\boldsymbol{R_{in}} = [r_{in}(0) \quad r_{in}(1) \quad r_{in}(2) \quad \ldots\ldots\ldots\ldots\ldots \quad r_{in}(n)] \tag{3.17}$$

The transversal filter is used for the sake of ease. The number of delay elements in a circuit depends on the order of tapped filter. It directly influences the accuracy of the system. Bias weights are adaptable that vary according to the adaptive algorithm used. Thus, the output is

$$y(n) = \sum_{l=0}^{n} w(l).r_{in}(n-l) \tag{3.18}$$

where, $\boldsymbol{W} = [w(0) \quad w(1) \quad w(2) \quad \ldots\ldots\ldots.. \quad w(l)]$.

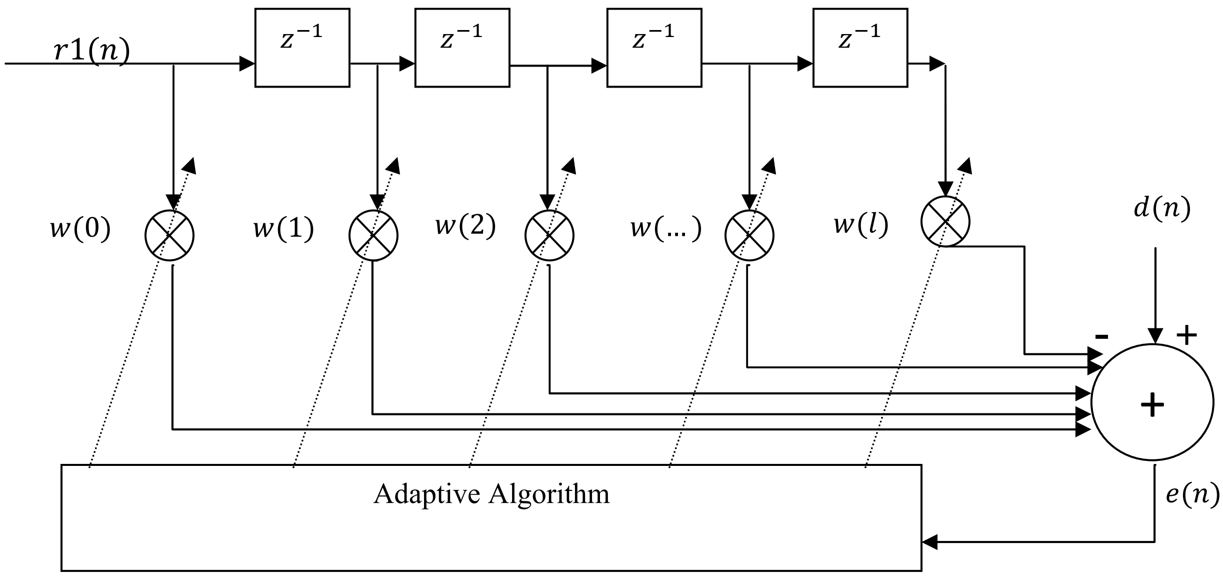

Fig. 3.4. Adaptive filter with n tapped delay [14].

The adaptive filters are closed loop systems, so there must be some other signal with which the output from the filter is compared and the error signal is generated in order to update the filter coefficients at each iteration. This signal is desired or training signal. The error signal produced is given by

$$e(n) = d(n) - y(n) \tag{3.19}$$

where, $y(n)$ is given by

$$y(n) = \boldsymbol{W}^T . \boldsymbol{R}_{in} \tag{3.20}$$

Thus, the error signal can be written as

$$e(n) = d(n) - \boldsymbol{W}^T . \boldsymbol{R}_{in} \tag{3.21}$$

Cost function used is minimum MSE. So, squaring both sides

$$e^2(n) = (d(n) - \boldsymbol{W}^T . \boldsymbol{R}_{in})^2 \tag{3.22}$$

$$e^2(n) = d^2(n) - 2d(n)\boldsymbol{R}_{in}^T . \boldsymbol{W} + \boldsymbol{W}^T \boldsymbol{R}_{in} \boldsymbol{R}_{in}^T \boldsymbol{W} \tag{3.23}$$

Taking expectation on both sides

$$E[e^2(n)] = E[d^2(n)] - 2E[d(n)\boldsymbol{R}_{in}^T] . \boldsymbol{W} + \boldsymbol{W}^T E[\boldsymbol{R}_{in} \boldsymbol{R}_{in}^T] \boldsymbol{W} \tag{3.24}$$

$$E[\boldsymbol{R}_{in}\boldsymbol{R}_{in}^T] = \boldsymbol{E}\begin{bmatrix} r_{in}(0)r_{in}(0) & r_{in}(0)r_{in}(1) & r_{in}(0)r_{in}(2) & \dots\dots. & r_{in}(0)r_{in}(l) \\ r_{in}(1)r_{in}(0) & r_{in}(1)r_{in}(1) & r_{in}(1)r_{in}(2) & \dots\dots.. & r_{in}(1)r_{in}(l) \\ r_{in}(.)r_{in}(0) & r_{in}(.)r_{in}(1) & r_{in}(.)r_{in}(2) & \dots\dots. & r_{in}(.)r_{in}(l) \\ r_{in}(l)r_{in}(0) & r_{in}(l)r_{in}(1) & r_{in}(l)r_{in}(2) & \dots\dots. & r_{in}(l)r_{in}(l) \end{bmatrix} \tag{3.25}$$

where, $E[\boldsymbol{R}_{in}\boldsymbol{R}_{in}^T]$ is auto-correlation of input signal.

The above-mentioned matrix is input correlation matrix where the principle diagonal elements are the mean square and other elements are cross-correlation among the input signal $\boldsymbol{R_{in}}$ and is represented by R [14].

$$E[d(n)\boldsymbol{R}_{in}^T] = \boldsymbol{E}[d(n)r_{in}(0) \quad d(n)r_{in}(1) \quad d(n)r_{in}(2) \quad \dots\dots. \quad d(n)r_{in}(l)] \tag{3.26}$$

where, $E[d(n)\boldsymbol{R}_{in}^T]$ is cross-correlation of input signal and desired signal.

This matrix represents the cross-correlation between input and desired signal and is denoted by P. Thus the Eq. (3.24) can be represented as

$$E[e^2(n)] = E[d^2(n)] - 2P^T.\boldsymbol{W} + \boldsymbol{W}^T R\boldsymbol{W} \tag{3.27}$$

From Eq. (3.27), it can be concluded that the MSE is a quadratic function of weight. When the MSE is drawn, it seems like a hyper paraboloid with the y-axis representing the MSE and x-axis is weight values. The bowl-shaped curve is the performance surface with concave upward indicating that all values of MSE are positive. The optimal weight is a point, where the value of MSE is minimum. As the MSE equation is a quadratic function, there is only one optimal solution with no local minima. The performance surface for stationary signals and invariant statistical properties is rigid in the coordinate system, thus the process starts from some point on the performance surface and proceeds to downhill to the minimum neighborhood and stay there. On the other hand, if signal properties are varying and non-stationary, then the performance surface is fuzzy in its coordinate system, so the point is not only moving to downhill minimum neighborhood and track the minimum as it moves.

The performance surface for MSE is quadratic function of weights. But in practical scenario, this performance surface is unknown. Systematic procedure and algorithm is computed by averaging the error signal over a period of time. There are two main algorithms used for finding the performance surface using the gradient estimate: Newton method and Steepest descent method. In Newton's method of gradient search, the components are varied at each iteration cycle. These changes are always pointing to the minimum value of performance surface. While in the case of Steepest descent, the changes are according to negative gradient of performance surface. The complexity of Steepest descent is less as compared to Newton's method [1].

$$w(n+1) = w(n) + \mu\big(-\nabla(n)\big) \tag{3.28}$$

$$\nabla(n) = \frac{de}{dw} = 2 * \lambda\big(w(n) - w_{opt}\big) \tag{3.29}$$

where, w_{opt} is optimal weight coefficient.

Substituting Eq. (3.29) in Eq. (3.28)

$$w(n+1) = w(n) + \mu\left(-2 * \lambda\big(w(n) - w_{opt}\big)(n)\right) \tag{3.30}$$

$$w(n+1) = (1 - 2\mu\lambda)w(n) + 2\mu\lambda w_{opt} \tag{3.31}$$

Thus,

$$w(1) = (1 - 2\mu\lambda)w(0) + 2\mu\lambda w_{opt} \tag{3.32}$$

$$w(2) = (1 - 2\mu\lambda)^2\, w(0) + 2\mu\lambda w_{opt}\, [(1 - 2\mu\lambda) + 1] \tag{3.33}$$

$$w(n) = (1 - 2\mu\lambda)^n\, w(0) + 2\mu\lambda w_{opt} \sum_{k=0}^{n-1} (1 - 2\mu\lambda)^n \tag{3.34}$$

$$w(n) = w_{opt} + (1 - 2\mu\lambda)^n\big(w(0) - w_{opt}\big) \tag{3.35}$$

Also,

$$e(n) = e_{min} + \lambda\big(w(n) - w_{opt}\big)^2 \tag{3.36}$$

where, e_{min} is the minimum error.

Substituting this,

$$e(n) = e_{min} + \lambda\big(w(0) - w_{opt}\big)^2 (1 - 2\mu\lambda)^{2n} \tag{3.37}$$

So, it is evident to say that the error value moves from zero value to minimum. Learning curve is used to indicate the minimization of MSE value at each iteration**.**

***3.3.3 Gradient search by Newton's method [1]**:* In this method, the roots of a polynomial are calculated i.e., finding the zeros of the function. So, the value is optimized in one step. It uses Newton-Raphson method to find the root given by [1].

$$w(1) = w(0) - \frac{f(w(0))}{f'w(0)} \tag{3.38}$$

where, $f'w(0)$ is first derivative.

$$f'\big(w(0)\big) = \frac{f\big(w(0)\big)}{w(0) - w(1)} \tag{3.39}$$

$$f'\big(w(n)\big) = \frac{f\big(w(n)\big) - f\big(w(n-1)\big)}{w(n) - w(n-1)} \tag{3.40}$$

and

$$w(n) = w(n-1) - \frac{f\big(w(n-1)\big)(w(n-1) - w(n-2))}{f(w(n-1) - f(w(n-2)} \tag{3.41}$$

In this method, the convergence depends on the initial values considered i.e., $w(0)$. Initially $f'w(0)$ is assumed to be zero to find the minima.

Fig. 3.5 addressed that the convergence rate of Newton method is fast. The calculation for R^{-1} is very complex and the system may became unstable if the initial value is not chosen wisely.

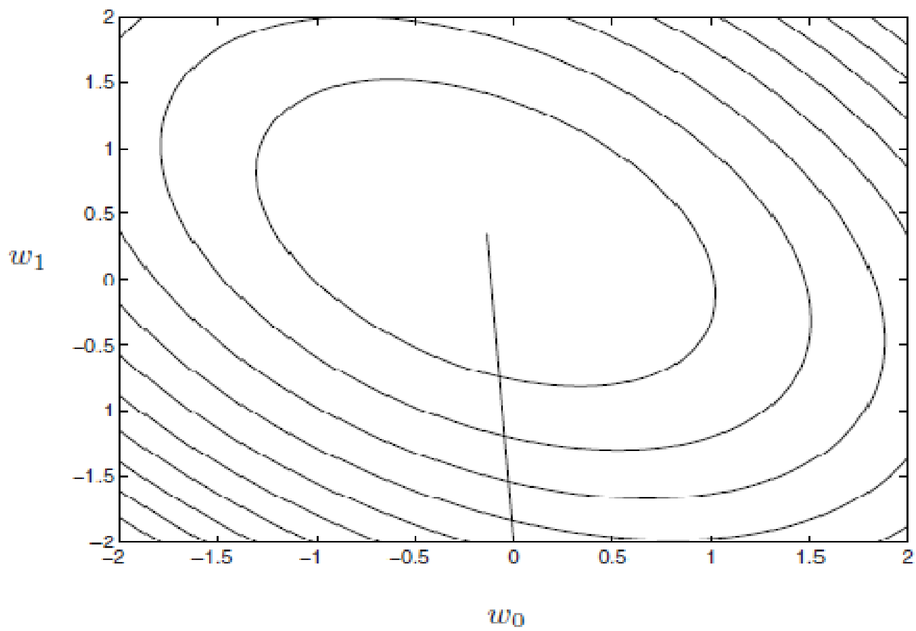

Fig. 3.5. Convergence rate with Newton method [15].

So, Steepest descent method is used in practical systems. As in most cases, the conditions are time-varying; so if at some instant, the value of noise is increased to threshold level, then the system parameters are completely changed and the direction to minima is turned around.

3.3.4 Gradient search by Steepest descent method [1]: As discussed earlier, this method follows negative gradient to reach the optimal weight. So the noise doesn't affect much to the system. It is a recursive process, in which the weights are varied step by step by computation of feedback system [1].

$$J\big(w(n)\big) < J\big(w(n-1)\big) \tag{3.42}$$

where, J is cost function.

$$w(n) = w(n-1) + \mu\big(-\nabla J(w)\big) \tag{3.43}$$

$$w(n) = w(n-1) - 2\mu R\left(w_{opt} - w(n-1)\right) \tag{3.44}$$

$$w(n) = (I - 2\mu R)w(n-1) + 2\mu R w_{opt} \tag{3.45}$$

$$w(n) = w_{opt} + (1 - 2\mu\lambda)^n \left(w(0) - w_{opt}\right) \tag{3.46}$$

The stability condition is satisfied when

$$lim_{k\to\infty}(1 - 2\mu\lambda)^2 = 0 \tag{3.47}$$

$$0 < \mu < 1/\lambda_{max} \tag{3.48}$$

where, λ is the eigen value of R and λ_{max} is the largest value.

Therefore, only correlation matrix on input signal and cross-correlation of input and desired signal is computed to adapt the system parameters.

Fig. 3.6 indicates the convergence procedure and learning curve of Steepest descent method.

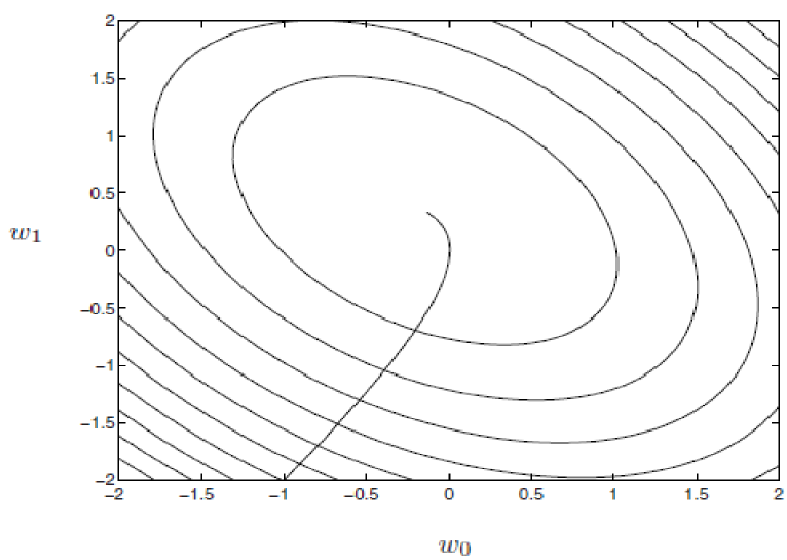

Fig. 3.6. Convergence rate with Steepest descent method [15].

- ***Excess MSE:*** Noise adversely affects the system response. So, in the presence of noise, the system is unable to reach the minimum point in MSE. There is always some difference between the exact value and steady state solution. Also, the solution oscillates around the minimum point. This difference is known as excess MSE.

Thus, excess MSE is $E[e(n) - e_{\min}]$.

Excess mean square error for Newton's method is [1],

$$\frac{(L+1)e_{min}\lambda_{av}(\frac{1}{\lambda})_{av}}{8\,NP\tau} \tag{3.49}$$

while for Steepest descent method is [1],

$$\frac{(L+1)^2 e_{min}}{8P}\left(\frac{1}{T_{MSE}}\right)_{av} \tag{3.50}$$

Where, L is number of weights, e_{min} is minimum MSE, P is normalized value of error by estimation, N is number of observations, τ is time constant of weight adjustment, λ is eigen value, T_{MSE} is time constant for adaptation.

- ***Misadjustment**:* It is defined as the ratio of excess MSE and minimum MSE. It is a normalized and dimensionless measure of the cost of adaptability [14].

$$M = \frac{excess\ MSE}{e_{min}} \tag{3.51}$$

Misadjustment (M) for Newton's method [1] is

$$\frac{(L+1)\lambda_{av}(\frac{1}{\lambda})_{av}}{8\,NP\tau} \tag{3.52}$$

while for Steepest descent method [1] is

$$\frac{(L+1)^2}{8P}\left(\frac{1}{T_{mse}}\right)_{av} \tag{3.53}$$

From the above-said expressions, we can conclude that M decreases as perturbation value increases. Moreover, if the eigen values of the autocorrelation matrix of input signal are equal then the misadjustment with both algorithm is same and performance surface indicates circular symmetry.

3.4 LMS algorithm

LMS also depends on performance surface and use linear combiner estimation of the gradient. It is comparatively easy to use.

To find the optimal value of weight vector, MSE is differentiated with respect to weight function and is denoted by ∇(e).

$$\nabla = \frac{\partial e}{W} = \left[\frac{\partial e}{\partial w_0} \quad \frac{\partial e}{\partial w_1} \quad \frac{\partial e}{\partial w_2} \quad \dots \dots . \quad \frac{\partial e}{\partial w_l}\right] \tag{3.54}$$

$$\nabla = \frac{\partial e(n)}{W} = 2RW - 2P \tag{3.55}$$

Equating Eq. (3.55) with zero, thus

$$0 = \frac{\partial e}{W} = 2RW - 2P \tag{3.56}$$

$$2RW = 2P \tag{3.57}$$

Hence, $W_{opt} = R^{-1}P$

Now substituting this optimal value of weight in above equation,

$$E[e^2(n)] = E[d^2(n)] - 2P^T.W_{opt} + W_{opt}^T R W_{opt} \tag{3.58}$$

$$E[e^2(n)] = E[d^2(n)] - 2P^T.R^{-1}P + [R^{-1}P]^T R R^{-1} P \tag{3.59}$$

Here, RR^{-1} is identity matrix, also $[R^{-1}P]^T = P^T R^T$

Thus,

$$E[e^2(n)] = E[d^2(n)] - 2P^T.R^{-1}P + P^T R^{-1} P \tag{3.60}$$

$$E[e^2(n)] = E[d^2(n)] - P^T.R^{-1}P \tag{3.61}$$

$$E[e^2(n)] = E[d^2(n)] - P^T W_{opt} \tag{3.62}$$

The excess MSE in LMS is given by [14],

$$Excess\ MSE = \mu e_{min} tr[R] \tag{3.63}$$

where, $tr[R]$ is trace of input signal.

while, misadjustment is given by [14],

$$Misadjustment = \mu tr[R] \tag{3.64}$$

3.5 Differences between traditional digital filters and adaptive filters

An adaptive filter differs from a traditional digital filter in the following ways

- A traditional digital filter has only one input signal $x(n)$ and one output signal $y(n)$. An adaptive filter requires an additional input signal $d(n)$ and returns an additional output signal $e(n)$.
- The filter coefficients of a traditional digital filter do not change over time. The coefficients of an adaptive filter change over time. Therefore, adaptive filters have a self-learning ability that traditional digital filters do not have.

3.6 Advantages of adaptive filters

Compared to traditional digital filters, adaptive filters have the following advantages

- Adaptive filters can complete some signal processing tasks that traditional digital filters cannot. For example, you can use adaptive filters to remove noise whose power spectrum changes over time.
- Adaptive filters can complete real-time modeling tasks.

Chapter 4

ADAPTIVE DELAY AND AMPLITUDE ESTIMATION USING LMS ALGORITHM

This chapter includes the system identification for amplitude and delay estimation with adaptive algorithm and various VSS algorithms that can enhance the system estimation accuracy.

Parameter estimation of the signal received at two spatially separated sensors plays an important role in acoustics, direction finding, speed sensing, positioning etc. In [3], Chakraborty has proposed the algorithm in which the signal sampling rate is four times of signal received at two sensors. The parameters are estimated by updating the coefficients of the filter at iterations. Thus, it is system identification problem where the input is noisy.

$$x_1(t) = r_1(t) + \eta_1(t) \tag{4.1}$$

$$x_2(t) = Ar_1(t - D) + \eta_1(t) \tag{4.2}$$

where, $r_1(t) = cos(\Omega t + \phi)$ received signal at sensor, $\eta_1(t)$ is zero mean AWGN independent of random phase and received signal, A is the gain factor, $r_1(t - D)$ is delayed version of received signal.

The signal is sampled at four times of frequency of received signal.

$$x_1(n) = cos(\Omega_s n + \phi) + \eta_1(n) \tag{4.3}$$

$$x_2(n) = A\, cos(\Omega_s n + \phi)cos(\Omega_s D) + Asin(\Omega_s n + \phi)sin(\Omega_s D) \tag{4.4}$$

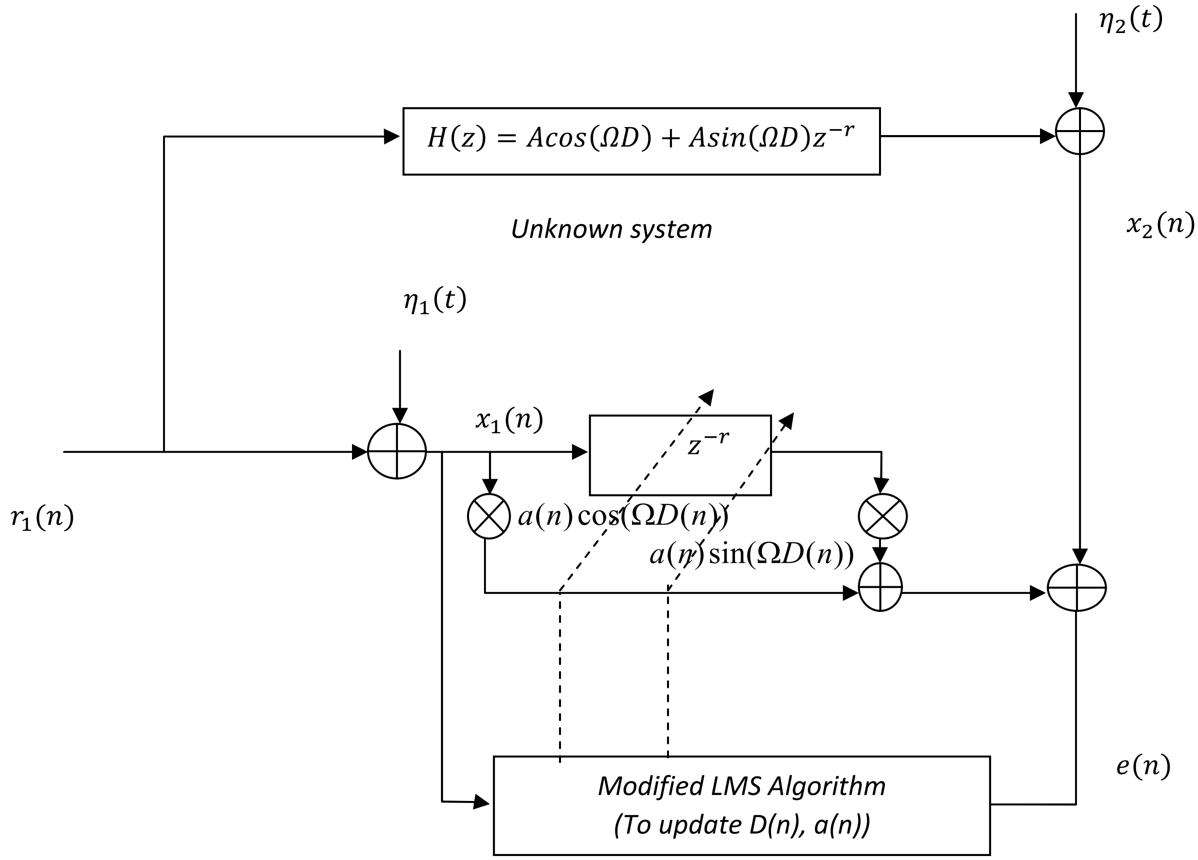

Fig. 4.1. System identification model for estimation of amplitude and delay [3].

The filter coefficients used are $[cos(\Omega_s D), sin(\Omega_s D)]^T$. Two different step-sizes are used to update the parameter coefficients namely μ_1, μ_2 given by $\mu = \begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix}$.

The generalized update equation for LMS is

$$W(n+1) = w(n) + 2\mu\varepsilon\boldsymbol{x} \tag{4.5}$$

Thus, the corresponding LMS update can be computed as [3],

$$\theta(n+1) = \theta(n) - 2\mu\nabla_\theta e(n)e(n) \tag{4.6}$$

$$\nabla_\theta = \begin{bmatrix} \dfrac{\partial(.)}{\partial D(n)} \\ \dfrac{\partial(.)}{\partial a(n)} \end{bmatrix} \quad (4.7)$$

$$\theta(n+1) = \theta(n) - 2\mu \begin{bmatrix} \dfrac{\partial e(n)}{\partial D(n)} \\ \dfrac{\partial e(n)}{\partial a(n)} \end{bmatrix} e(n) \quad (4.8)$$

$$\theta(n+1) = \theta(n) + 2\begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix} \begin{bmatrix} \dfrac{\partial e(n)}{\partial D(n)} \\ \dfrac{\partial e(n)}{\partial a(n)} \end{bmatrix} e(n) \quad (4.9)$$

$$\theta(n+1) = \theta(n)$$
$$+2\begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix} \begin{bmatrix} [-\Omega_s a(n) sin(\Omega_s D(n)) \quad a(n)\Omega_s cos(\Omega_s D(n))] \begin{bmatrix} x1(n) \\ x1(n-1) \end{bmatrix} \\ [\cos(\Omega_s D(n)) \quad sin(\Omega_s D(n))] \begin{bmatrix} x1(n) \\ x1(n-1) \end{bmatrix} \end{bmatrix} e(n) \quad (4.10)$$

$$\theta(n+1) = \theta(n)$$
$$+2\begin{bmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{bmatrix} \begin{bmatrix} -\Omega_s a(n) sin(\Omega_s D(n)) x(n) + a(n)\Omega_s \cos(\Omega_s D(n)) x1(n-1) \\ \cos(\Omega_s D(n)) x(n) + sin(\Omega_s D(n)) x1(n-1) \end{bmatrix} e(n) \quad (4.11)$$

$$\theta(n+1) = \theta(n)$$
$$+2\begin{bmatrix} -\Omega_s a(n) \sin(\Omega_s D(n)) x1(n)\mu_1 + a(n)\Omega_s cos(\Omega_s D(n)) x1(n-1)\mu_1 \\ \cos(\Omega_s D(n)) x1(n)\mu_2 + sin(\Omega_s D(n)) x1(n-1)\mu_2 \end{bmatrix} e(n) \quad (4.12)$$

$$\theta_d(n+1)$$
$$= \theta_d(n) + \left(-2(\Omega_s a(n) sin(\Omega_s D(n)) x1(n) + a(n)\Omega_s cos(\Omega_s D(n)) x1(n-1)\mu_1)\right) \quad (4.13)$$

$$\theta_a(n+1) = \theta_a(n) + 2e(n)\left(cos(\Omega_s D(n)) x1(n)\mu_2 + sin(\Omega_s D(n)) x1(n-1)\mu_2\right) \quad (4.14)$$

Eq. (4.6) to Eq. (4.14) are derived with fixed step-size.

There are several criteria to make the step-size variable like an instantaneous error, squared prediction error, change of sign at adjacent samples, the correlation between successive samples etc.

As Kwong [4] proposed VSS algorithm that can reduce the number of iterations for the same misadjustment. The motivation behind same misadjustment is that when the difference between the optimal solution and estimated value is larger, then the step-size is increased in order to speed up the system to achieve the result but as this difference reduces to a considerable level then step-size reduces to reduce the fluctuation around the mean value. The step-size in [4] is given by

$$\mu_{KVSS}(n+1) = \alpha_{KVSS}\mu_{KVSS}(n) + \gamma_{KVSS}\varepsilon^2_{KVSS}(n) \tag{4.15}$$

where, α_{KVSS} is selected in between 0 to 1 in order to achieve the exponential forgetting. Typically, $\alpha_{KVSS} = 0.97$ works well. γ_{KVSS} is used in conjunction with α_{KVSS} to achieve minimum permissible misadjustment. Typically its value is small in range of 10^{-4}. The values of α_{KVSS} and γ_{KVSS} are selected using formula [4].

$$0 < \frac{3\,\gamma_{KVSS}\,\varepsilon_{min}}{1 - \alpha_{KVSS}{}^2} < \frac{1}{3\,tr(R)} \tag{4.16}$$

$$\mu_{KVSS}(n+1) = \begin{Bmatrix} \mu_{max} & if\ \mu'_{KVSS}(n > \mu_{max} \\ \mu_{min} & if\ \mu'_{KVSS}(n) < \mu_{min} \\ \mu'_{KVSS}(n+1) & otherwise \end{Bmatrix} \tag{4.17}$$

For bounded MSE, the μ_{max} is given by [4],

$$\mu_{max} \leq \frac{2}{3\,tr(R)} \tag{4.18}$$

$\mu_{KVSS}(n)$ is restricted in between μ_{min} and μ_{max} to guarantee the stability. Also, the non-stationarity does not affect much in performance in case of VSS as compared to FSS. The VSS is sensitive to ε_{min}, the misadjustment is directly influenced by ε_{min} while in case of FSS, misadjustment is independent of it. γ_{KVSS} is decreased to reduce misadjustment.

Aboulnasr [5] proposed new VSS algorithm in which step-size is made adaptable according to the autocorrelation of adjacent samples of error. This eliminated the problem of system sensitivity to high noise.

$$e_{AVSS}(n) = d_{AVSS}(n) - X(n)W_{AVSS}(n) \tag{4.19}$$

$$p_{AVSS}(n) = \beta_{AVSS}\, p_{AVSS}(n-1) + (1-\beta_{AVSS})e_{AVSS}(n)e_{AVSS}(n-1) \tag{4.20}$$

$$\mu_{AVSS}(n+1) = \alpha_{AVSS}\mu_{AVSS}(n) + \gamma_{AVSS}\, p^2_{AVSS}(n) \tag{4.21}$$

The error autocorrelation is efficient parameter to check the quality of the estimation because it rejects the noise effects in the adaptation procedure. Initially, the value of $\gamma_{AVSS}\, p^2_{AVSS}(n)$ is large, so the step-size is very large but as the iteration number is increased, the value of $\gamma_{AVSS}\, p^2_{AVSS}(n)$ keeps on decreasing resulting in the minimum misadjustment. β_{AVSS} is selected on the basis of environment. If stationary environment is assumed then it is good practice to use β_{AVSS} near to 1 as this decreases the misadjustment. While for non-stationary environment, β_{AVSS} is chosen to be close to 0 to cope with time varying conditions and increase tracking ability. Thus, it can be deduced that there is tradeoff between misadjustment and tracking speed for choosing β_{AVSS}.

MAVSS can be considered as the combination of [4] and [5]. Thus in tracking it is as good as VSS and in anti-noise ability as good as AVSS. The updating equations in MAVSS [6] are:

$$\beta_{MAVSS}(n+1) = \begin{Bmatrix} \beta_{MAVSSMAX} & if\, \beta_{MAVSS}(n) > \beta_{MAVSSMAX} \\ \beta_{MAVSSMIN} & if\, \beta_{MAVSS}(n) < \beta_{MAVSSMIN} \\ \eta_{MAVSS}\beta_{MAVSS}(n) + \lambda_{MAVSS}e^2_{MAVSS}(n) & otherwise \end{Bmatrix} \tag{4.22}$$

$$p_{MAVSS}(n+1) = (1-\beta_{MAVSS}(n))p_{MAVSS}(n) + \beta_{MAVSS}(n)e_{MAVSS}(n)e_{MAVSS}(n-1) \tag{4.23}$$

$$\mu_{MAVSS}(n+1) = \begin{Bmatrix} \mu_{MAVSSMAX} & if\, \mu_{MAVSS}(n) > \mu_{MAVSSMAX} \\ \mu_{MAVSSMIN} & if\, \mu_{MAVSS}(n) < \mu_{MAVSSMIN} \\ \alpha_{MAVSS}\mu_{MAVSS}(n) + \gamma_{MAVSS}p^2_{MAVSS}(n) & otherwise \end{Bmatrix} \tag{4.24}$$

where, $\alpha_{MAVSS} > 0,\ \eta_{MAVSS} < 1,\ \gamma_{MAVSS} > 0\ and\ \lambda_{MAVSS} > 0$.

β_{MAVSS} is controlling parameter for the sensitivity of $p_{MAVSS}(n+1)$. Thus, the correlation is controlled by error signal power. So, when the error signal power is increased then the correlation is enlarged.

Sigmoid VSS algorithm [7] formulates the non-linear relationship between the step-size and weight coefficients. This decreases the steady state error along with the reduction of interference at the output. But the simple sigmoid function oscillates around the steady state so it is not as efficient as theory result presents, the modified sigmoid is calculated by varying the γ factor to stabilize the learning curve at steady state.

$$\mu_{MSVSS}(n+1) = \beta_{MSVSS}\left|\frac{1}{1+\exp(-\alpha_{MAVSS}(e_{MAVSS}(n)-\gamma_{MAVSS}))}-\frac{1}{1+\exp(+\alpha_{MAVSS}(e_{MAVSS}(n)+\gamma_{MAVSS}))}\right| \tag{4.25}$$

The shape of the function is controlled by α_{MAVSS} and γ_{MAVSS}. The steady state error is limited by the value of γ_{MAVSS}.

Chapter 5

ADAPTIVE DELAY AND AMPLITUDE ESTIMATION USING VSS-LMS ALGORITHMS

It includes the core of this research work. The proposed methodology for system identification paradigm of delay and amplitude estimation using VSS algorithms is discussed.

Time-delay and amplitude estimation is an important field of research in the domain of wireless sensor networks. The sensors are located separately at a particular distance from each other; therefore, the same signal corrupted by noise is received at two or more different sensors with different time-delays at the same carrier frequency [8]. The cross-correlation function based approach can be used to handle stationary time-delay [9], but it suffers due to the resolution of time-delay estimate at certain sampling rates. Some other methods include discrete-Fourier-transform [16], quadrature delay [17] and correlations [18].

The problem arises, when this time-delay appears to be time-variant due to the motion of transmitter and/or receiver sensor [19]. The adaptive filtering approach is a remedial solution to estimate the time-delay between the received copies of sinusoidal signal of known frequency [20]. However, the case of simultaneous estimation of "time-delay and amplitude" of sinusoidal signals is considered in [8]. An LMS-type adaptive algorithm is proposed by Chakraborty in [3] to estimate and track both the time-delay and the relative amplitude of the delayed sinusoidal signal by identifying the two-tap filter coefficients, which is based on the choice of the particular sampling rate. Here, two separate FSS coefficients $\mu \rightarrow \{\mu_1, \mu_2\}$ are required in the LMS-type adaptive algorithm for the estimation of time-delay and amplitude variables. But, FSS-LMS algorithms suffer due to lag-misadjustment in the time-varying system identification problem, and the gradient-misadjustment is inevitable due to the finite value of FSS [21].

However, a number of variable-step-size criteria have been proposed so far, which can be incorporated into LMS algorithm framework under the non-stationary environment [22]. Kwong and Johnston have proposed a VSS-LMS (KVSS-LMS) adaptive algorithm in [4] for the tracking of the time-varying first-order autoregressive channel, in which the step-size is adjusted by the square of prediction error. Subsequently, Aboulnasr and Mayyas have presented a variable-step-size LMS (AVSS-LMS) adaptive algorithm in [5], in which the step-size of the algorithm is adjusted according to the square of time-averaged estimate of the autocorrelation of present/instantaneous estimation error $e(n)$ and the past estimation error $e(n-1)$. This scheme is modified to give MAVSS-LMS algorithm in [6] to expedite the convergence rate. In another approach [7], the time-varying SVSS-LMS is one of the tractable solutions to expedite the convergence rate and to fine tune the tracking process in case of LMS algorithm. This technique is further improved to develop MSVSS-LMS algorithm in [7].

In this research work, we emphasis on the performance evaluation of FSS-LMS [14], KVSS-LMS [4], AVSS-LMS [5], MAVSS-LMS [6] and SVSS-LMS [7], MSVSS-LMS [7] algorithms in the estimation and tracking of both the time-delay and the relative amplitude of two delayed versions of sinusoidal signal by identifying the two-tap filter coefficients.

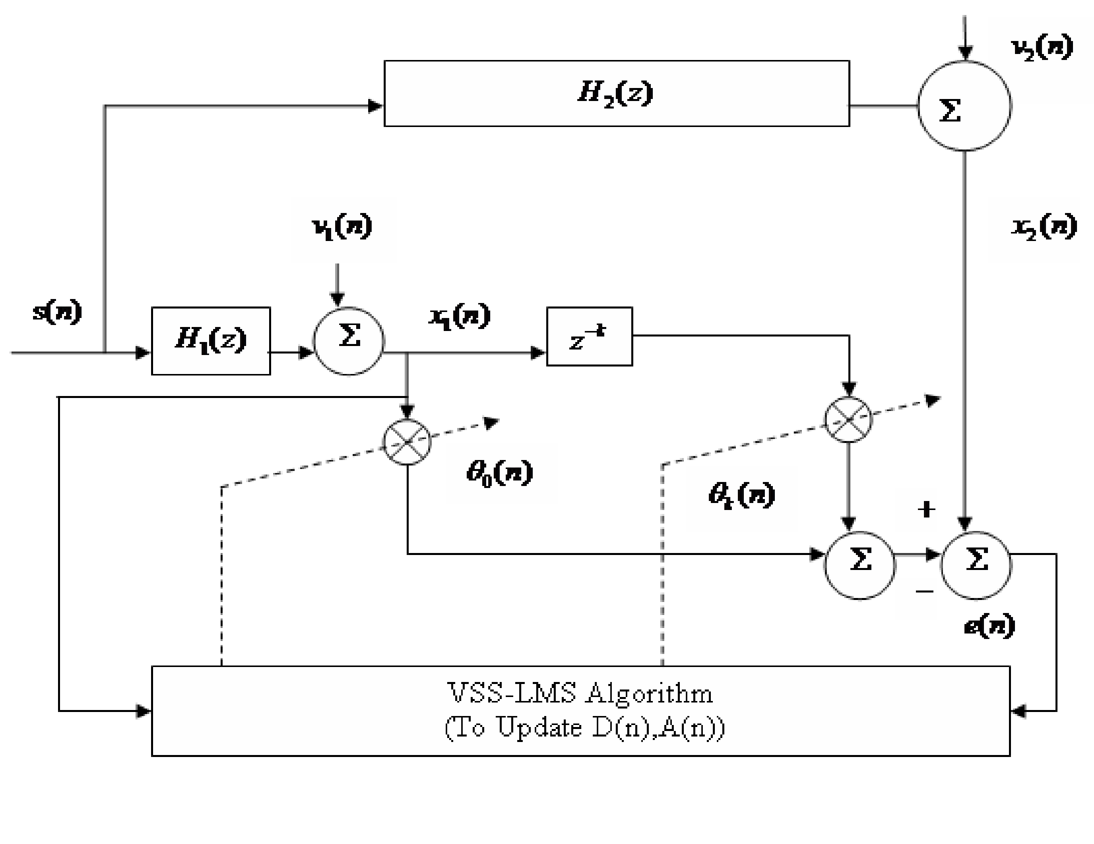

Fig. 5.1. System identification paradigm for delay and amplitude estimation using VSS-LMS algorithms.

Let the signals received at two sensors located at the different positions be

$$x_{a1}[t] = s_a[t] + v_1[t] \tag{5.1}$$

$$x_{a2}[t] = As_a[t-D] + v_2[t] \tag{5.2}$$

where, $s_a[t] = \cos(\Omega t + \varphi)$ is a sinusoidal signal with known angular frequency Ω and uniformly distributed random phase φ [20] in the range $[0, 2\pi)$. Here, D is the time-delay variable, A is the amplitude variable in Eq. (5.1) and Eq. (5.2); v_1 and v_2 are additive white Gaussian noise components with zero-means and variances σ_{v1}^2 and σ_{v2}^2 respectively. The sampling interval is considered to be T_s, which must satisfy relationship $\Omega T_S = \pi/2k$ with $k \to \{1, 2, 3, ...\}$. The corresponding angular sampling frequency is $\Omega_S = 4k\Omega$. After sampling, Eq. (5.1) and Eq. (5.2) can be written as

$$x_{a1}[nT_S] = s_a[nT_S] + v_1[nT_S] \tag{5.3}$$

$$x_{a2}[nT_S] = As_a[nT_S - D] + v_2[nT_S] \tag{5.4}$$

Now, Eq. (5.3) and Eq. (5.4) can be rewritten as

$$x_1(n) = s(n) + v_1(n) \tag{5.5}$$

$$x_2(n) = A\cos(\Omega D)s(n) + A\sin(\Omega D)s(n-k) + v_2(n) \tag{5.6}$$

If we consider $\sigma_{v1}^2 = \sigma_{v2}^2 = 0$, then system response based on Eq. (5.5) is $H_1(z) = 1$, and the system response based on Eq. (5.6) is

$$H_2(z) = A\cos(\Omega D) + A\sin(\Omega D)z^{-k} \tag{5.7}$$

As time-delay D and amplitude A may be time-varying in nature, therefore, Eq. (5.6) can be modified as

$$x_2(n) = \theta_0(n)s(n) + \theta_k(n)s(n-k) + v_2(n) \tag{5.8}$$

where, $\theta_0(n) = A(n)\cos(\Omega D(n))$, $\theta_k(n) = A(n)\sin(\Omega D(n))$

The corresponding two tap-coefficients of adaptive filter with weights are $w_0(n)$ and $w_k(n)$, where

$w_0(n) = A(n)\cos(\Omega D(\mathrm{n}))$, $w_k(n) = A(n)\sin(\Omega D(\mathrm{n}))$

Now, time-delay can be estimated using following equation

$$D(n) = \frac{1}{\Omega}\arctan\left\{\frac{w_k(n)}{w_0(n)}\right\} \tag{5.9}$$

For adaptive filtering application in aforementioned framework for the time-delay and amplitude estimation, we consider

$$\bar{\mathrm{W}}(n) = [\cos(\Omega D(\mathrm{n})),\ \sin(\Omega D(\mathrm{n}))]^T \tag{5.10}$$

$$\vec{W}(n) = A(n)\bar{W}(n) \tag{5.11}$$

$$\hat{W}(n) = [-\sin(\Omega D(n)),\ \cos(\Omega D(n))]^T \tag{5.12}$$

where, $(.)^T$ is the matrix transposition operator. Using, the concepts of system identification [3], [14], [20] as shown in Fig. 5.1, the error term is defined as

$$e(n) = x_2(n) - \vec{W}^T(n)\overrightarrow{X}(n) \tag{5.13}$$

where,

$\overrightarrow{X}(n) = [x_1(n),\ x_1(n-k)]^T$ and $\vec{\theta}(n) = [D(n),\ A(n)]^T$

Based on the theory and concepts of VSS-LMS algorithm [3]-[7],

$$\vec{\theta}(n+1) = \vec{\theta}(n) - \vec{\mu}(n)\nabla_\theta e^2(n) \tag{5.14}$$

Where, $\nabla_\theta(.) = \left[\partial(.)/\partial D(n),\ \partial(.)/\partial A(n)\right]^T$,

$\dfrac{\partial e(n)}{\partial D(n)} = -\Omega A(n)\hat{W}^T\overrightarrow{X}(n)$, $\dfrac{\partial e(n)}{\partial A(n)} = -\bar{W}^T(n)\overrightarrow{X}(n)$ and $\vec{\mu}(n) = \begin{bmatrix}\mu_1(n) & 0 \\ 0 & \mu_2(n)\end{bmatrix}$.

Further simplification of Eq. (5.14) (as in [3]) leads to

$$\vec{\theta}(n+1)=\vec{\theta}(n)+2\vec{\mu}(n)\vec{\Omega}\vec{P}(n)\vec{X}(n)e(n) \tag{5.15}$$

where, $\vec{P}(n)=\begin{bmatrix} A(n)\hat{W}^T(n) \\ \vec{W}^T(n) \end{bmatrix}$ and $\vec{\Omega}=\begin{bmatrix} \Omega & 0 \\ 0 & 1 \end{bmatrix}$.

As per the bounds derived in [3], the VSS must be bounded as $0<\mu_1(n)<\frac{2}{\Omega^2 A^2}$ and $0<\mu_2(n)<2$. There is a tradeoff between convergence rate and misadjustment in case of FSS-LMS algorithm. Therefore, we use VSS-LMS algorithms to have high convergence rate in the initial mode of adaptation operation by using the high value of step-size and to have lower step-size in the tracking mode to reduce misadjustment. Here, we focus on KVSS-LMS, AVSS-LMS, MAVSS-LMS and SVSS-LMS, MSVSS-LMS to adjust the weights of adaptive filter in order to estimate time-delay and amplitude for the sinusoidal signal. In KVSS-LMS algorithm, the step-size is updated at each iteration as

$$\mu_{KVSS}(n+1)=\begin{cases} \mu_{KVSSMAX} & if\ \mu_{KVSS}(n)>\mu_{KVSSMAX} \\ \mu_{KVSSMIN} & if\ \mu_{KVSS}(n)<\mu_{KVSSMIN} \\ \alpha_{KVSS}\mu_{KVSS}(n)+\gamma_{KVSS}e^2{}_{KVSS}(n) & otherwise \end{cases} \tag{5.16}$$

where, $\alpha_{KVSS}<1$ and $\gamma_{KVSS}<1$ are adjusted to have minimum possible misadjustment in the tracking mode. However in AVSS-LMS algorithm, the step-size of algorithm is tuned according to the square of the time-averaged estimate of the autocorrelation of $e(n)$ and $e(n-1)$. It follows that

$$q_{AVSS}(n+1)=\beta_{AVSS}q_{AVSS}(n)+(1-\beta_{AVSS})e_{AVSS}(n+1)e_{AVSS}(n) \tag{5.17}$$

$$\mu_{AVSS}(n+1)=\begin{cases} \mu_{AVSSMAX} & if\ \mu_{AVSS}(n)>\mu_{AVSSMAX} \\ \mu_{AVSSMIN} & if\ \mu_{AVSS}(n)<\mu_{AVSSMIN} \\ \alpha_{AVSS}\mu_{AVSS}(n)+\gamma_{AVSS}q^2_{AVSS}(n) & otherwise \end{cases} \tag{5.18}$$

For non-stationary optimal coefficients, the time-averaging window must be small enough to permit for forgetting the deep past and adapting to the current statistics by using $\beta_{AVSS}<1$. By exploiting the features of KVSS-LMS and AVSS-LMS algorithms, MAVSS-LMS is proposed in [6], in which VSS is updated as

$$\beta_{MAVSS}(n+1)=\begin{cases}\beta_{MAVSSMAX} & if\,\beta_{MAVSS}(n)>\beta_{MAVSSMAX}\\ \beta_{MAVSSMIN} & if\,\beta_{MAVSS}(n)<\beta_{MAVSSMIN}\\ \eta_{MAVSS}\beta_{MAVSS}(n)+\lambda_{MAVSS}e^2_{MAVSS}(n) & otherwise\end{cases} \quad (5.19)$$

$$q_{MAVSS}(n+1)=(1-\beta_{MAVSS}(n))q_{MAVSS}(n)+\beta_{MAVSS}(n)e_{MAVSS}(n)e_{MAVSS}(n-1) \quad (5.20)$$

$$\mu_{MAVSS}(n+1)=\begin{cases}\mu_{MAVSSMAX} & if\,\mu_{MAVSS}(n)>\mu_{MAVSSMAX}\\ \mu_{MAVSSMIN} & if\,\mu_{MAVSS}(n)<\mu_{MAVSSMIN}\\ \alpha_{MAVSS}\mu_{MAVSS}(n)+\gamma_{MAVSS}q^2_{MAVSS}(n) & otherwise\end{cases} \quad (5.21)$$

Here, β_{MAVSS} is the time-average of the squared error signal, which is used to control the sensitivity of q_{MAVSS} to the instantaneous error correlation. In another approach, the sigmoid variable-step-size algorithm formulates a nonlinear relationship between the step-size and filter weight coefficients. This reduces the steady-state error along with the reduction of interference at the output [7]. In this LMS algorithm, the value of VSS is controlled as

$$\mu_{SVSS}(n+1)=\beta_{SVSS}\left(\left\{1+\exp(-\alpha_{SVSS}\left|e_{SVSS}(n)\right|)\right\}^{-1}-\gamma_{SVSS}\right) \quad (5.22)$$

To overcome the shortcoming of SVSS-LMS algorithm, MSVSS-LMS algorithm is presented in [7] to work in the stability region of operation to achieve minimum MSE in the convergence as well as tracking modes, such that

$$\mu_{MSVSS}(n+1)=\beta_{MSVSS}\times\left|\Gamma_{-}(n)-\Gamma_{+}(n)\right| \quad (5.23)$$

$$\Gamma_{-}(n)=\left\{1+\exp(-\alpha_{MSVSS}(e_{MSVSS}(n)-\gamma_{MSVSS}))\right\}^{-1} \quad (5.24)$$

$$\Gamma_{+}(n)=\left\{1+\exp(+\alpha_{MSVSS}(e_{MSVSS}(n)+\gamma_{MSVSS}))\right\}^{-1} \quad (5.25)$$

In Eq. (5.22) and Eq. (5.23), β_{SVSS} and β_{MSVSS} control the value ranges of the functions. Whereas α_{SVSS}, α_{MSVSS}, γ_{SVSS} and γ_{MSVSS} control the shape of the functions. In all aforementioned algorithms, the values of constant parameters are kept between 0 and 1. The variable-step-size is incorporated in Eq. (5.15) to update the LMS algorithm at each iteration according to KVSS, AVSS, MAVSS, SVSS and MSVSS to estimate time-delay and amplitude.

Chapter 6

SIMULATION RESULTS AND DISCUSSION

This chapter addresses the performance of different algorithm discussed in chapter 5 with the help of simulation results in MATLAB. MSE of delay and amplitude along with tracking ability of system is taken into consideration.

In adaptive signal processing, the solution for some particular problem is evaluated in performance surface determined by the cost function like MSE, regularized MSE, least mean fourth etc. Accordingly, the linear approximation of cost function is calculated by Steepest descent method while for quadratic approximation, Newton method is used.

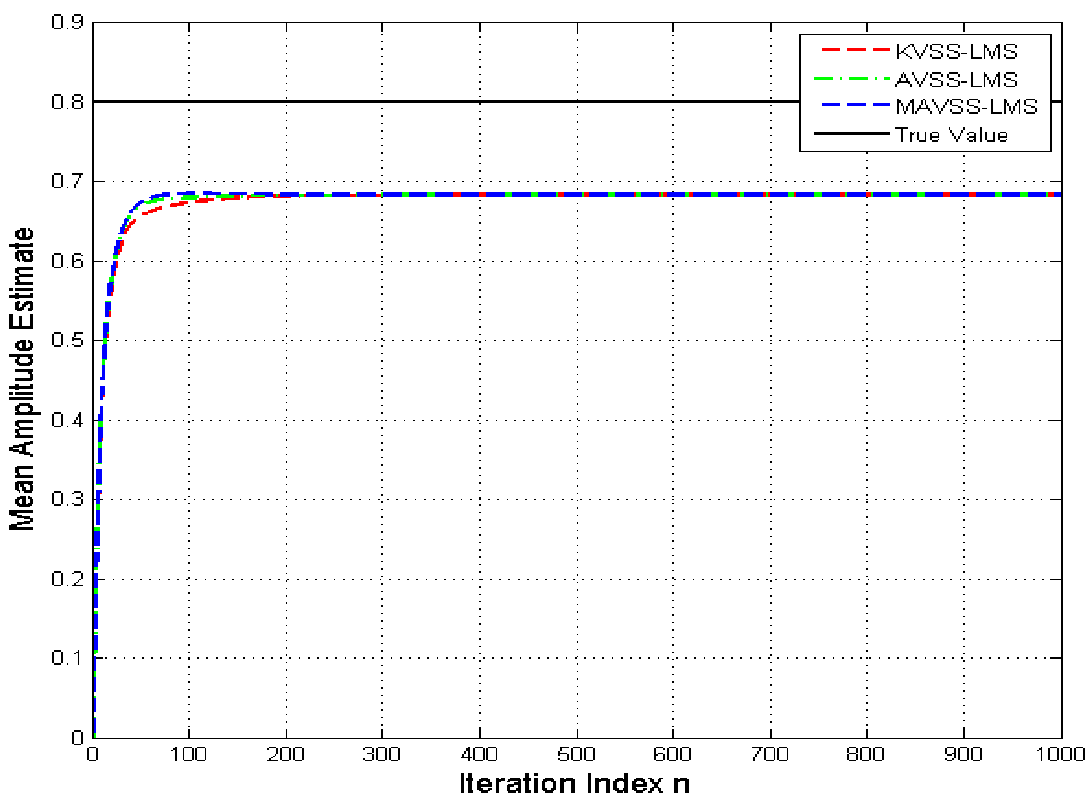

Fig. 6.1. Mean amplitude estimate vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

Moreover, adaptive filters have the capability to track signals in the time-varying environment but for the sake of ease, stationary SOE is assumed.

Fig. 6.1 depicts the trajectories of KVSS-LMS, AVSS-LMS, MAVSS-LMS algorithm for mean amplitude estimates in which amplitude A is held constant equal to 0.8. Step-size (μ) assumed is $\mu_{KVSSMIN} = \mu_{MAVSSMIN} = 0.0165$ and $\mu_{KVSSMAX} = \mu_{MAVSSMAX} = 0.1$ for KVSS-LMS and MAVSS-LMS, while for AVSS-LMS step-size μ_{AVSS} is 0.1. $\alpha_{KVSS} = \alpha_{AVSS} = \alpha_{MAVSS}$ and $\gamma_{KVSS} = \gamma_{AVSS} = \gamma_{MAVSS}$ is equated as 0.97 and 0.00048 respectively. β_{AVSS} is 0.99 in AVSS-LMS. η_{MAVSS} and λ_{MAVSS} is 0.97 and 0.0005 respectively in MAVSS-LMS. The mean amplitude estimate at steady state is 0.6832 and number of iterations required are 205, 150, 100 for KVSS-LMS, AVSS-LMS and MAVSS-LMS respectively.

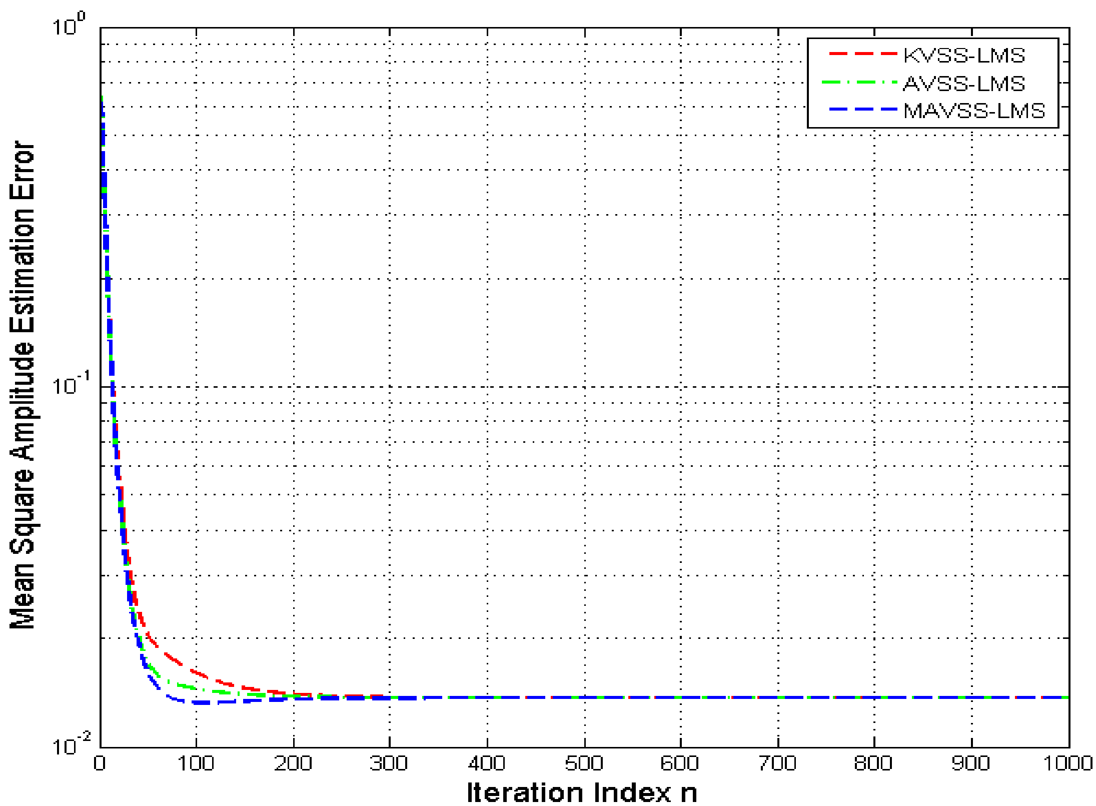

Fig. 6.2. Mean square amplitude estimation error vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

Fig. 6.2 indicates the mean square amplitude estimation error for these algorithms. The mean square amplitude estimate at steady state is 0.01366 and number of iterations required are 290, 230, 150 for KVSS-LMS, AVSS-LMS and MAVSS-LMS respectively.

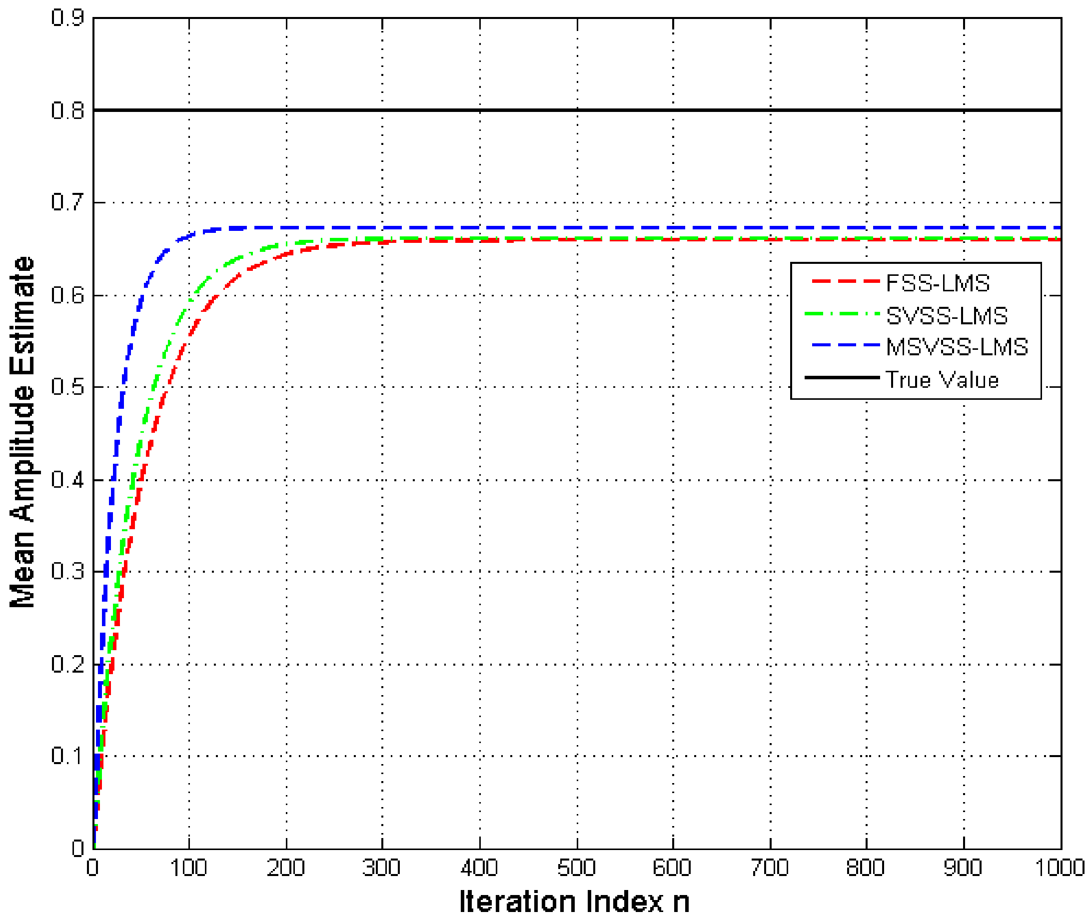

Fig. 6.3. Mean amplitude estimate vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

Fig. 6.3 illustrates the simulation results for FSS-LMS, SVSS-LMS and MSVSS-LMS algorithm of mean amplitude estimates in which amplitude A is held constant equal to 0.8. Step-size assumed is $\mu_{FSS} = \mu_{SVSS} = \mu_{MSVSS} = 0.0165$ for each algorithm. $\beta_{SVSS} = \beta_{MSVSS} = 0.2$ and $\gamma_{SVSS} = \gamma_{MSVSS} = 0.4$ for SVSS-LMS and MSVSS-LMS. The mean amplitude estimate at steady state is 0.6733 and number of iterations required are 450, 360 and 200 for FSS-LMS, SVSS-LMS and MSVSS-LMS respectively.

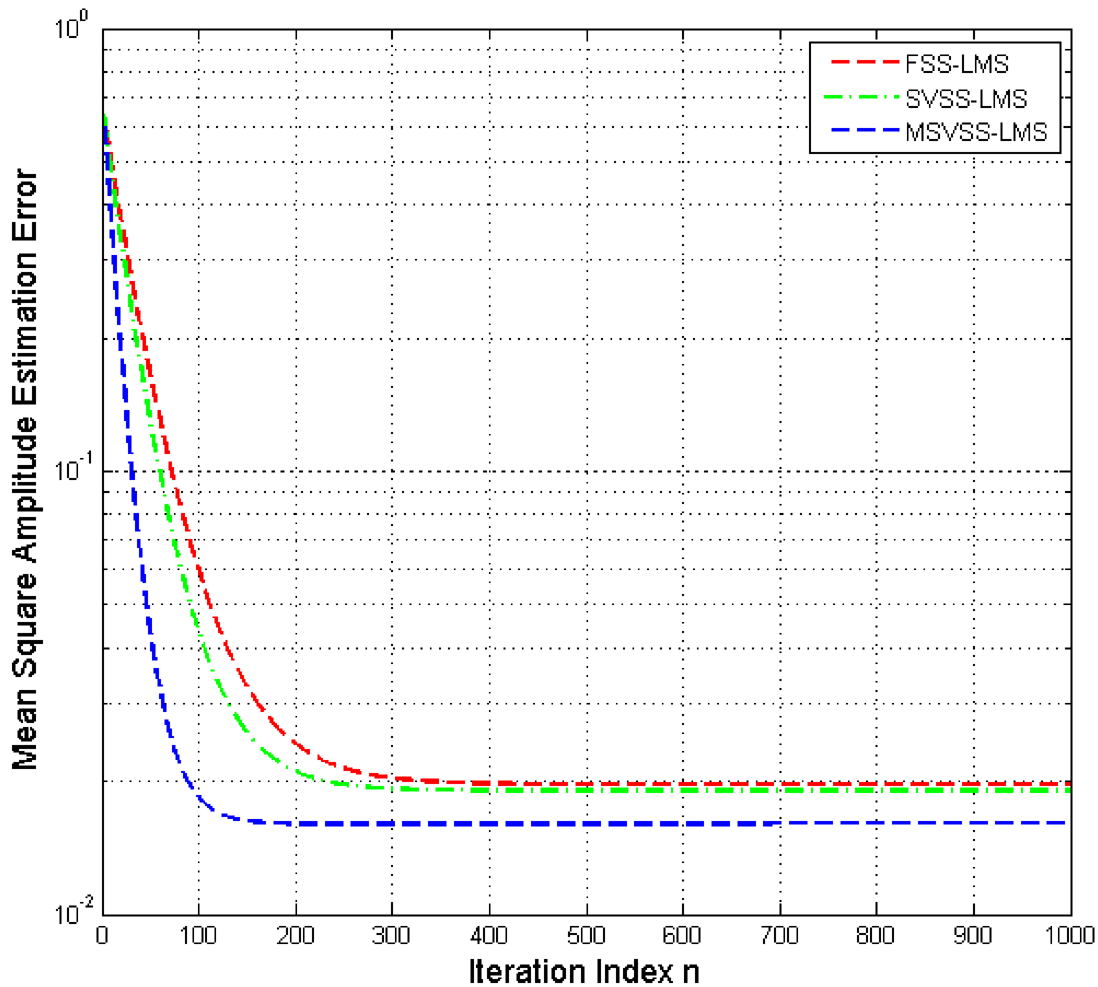

Fig. 6.4. Mean square amplitude estimation error vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

In Fig. 6.4, the mean square amplitude estimation error at steady state is 0.01950 approx. and number of iterations required is 450 for FSS-LMS, 365 for SVSS-LMS and 220 for MSVSS-LMS. Fig. 6.5 shows the trajectories of KVSS-LMS, AVSS-LMS, MAVSS-LMS algorithm for mean delay estimates in which time-delay is held constant equal to 0.2. Step-size assumed is $\mu_{KVSSMIN} = \mu_{MAVSSMIN} = 0.0165$ and $\mu_{KVSSMAX} = \mu_{MAVSSMAX} = 0.1$ for KVSS-LMS and MAVSS-LMS, while for AVSS-LMS step-size μ_{AVSS} is 0.1. $\alpha_{KVSS} = \alpha_{AVSS} = \alpha_{MAVSS}$ and $\gamma_{KVSS} = \gamma_{AVSS} = \gamma_{MAVSS}$ is equated as 0.97 and 0.00048 respectively. β_{AVSS} is 0.99 in AVSS-LMS. η_{MAVSS} and λ_{MAVSS} is 0.97 and 0.0005 respectively in MAVSS-LMS.

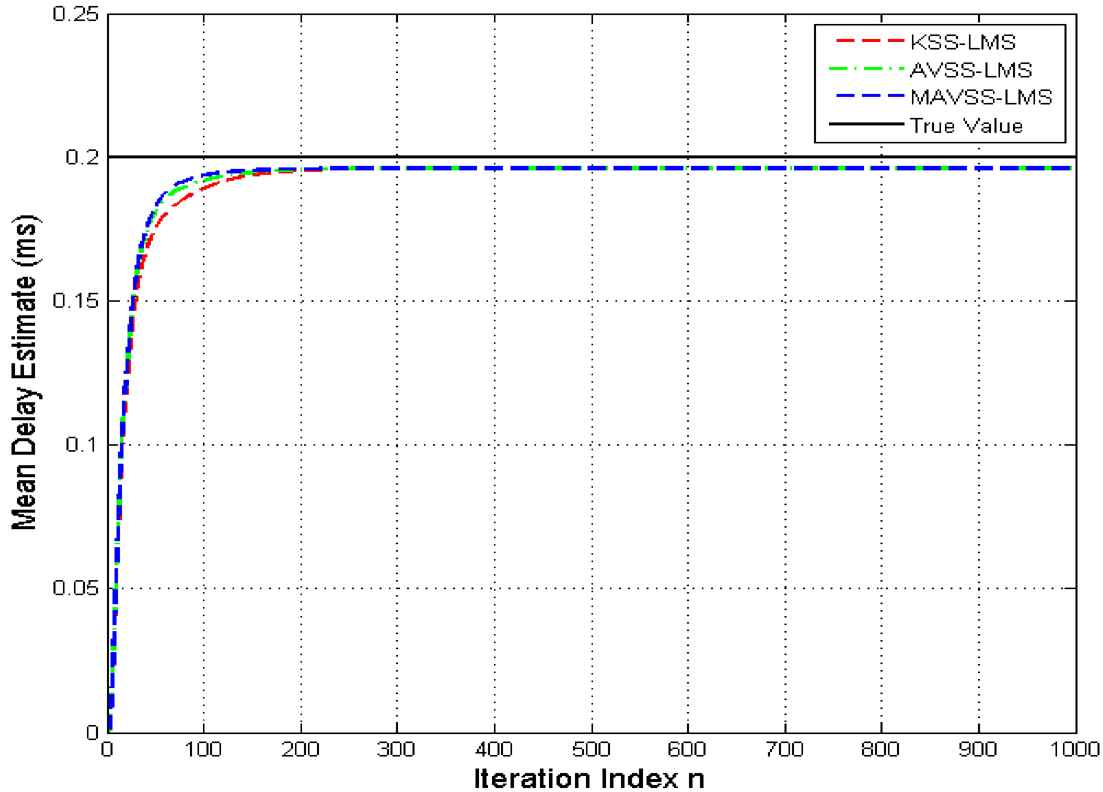

Fig. 6.5. Mean delay estimate vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

The mean delay estimate at steady state is 0.1960 approx. and number of iterations required is 248, 181, 135 for KVSS-LMS, AVSS-LMS and MAVSS-LMS respectively.

Fig. 6.6 indicates the mean square delay estimation error for these algorithms. The mean square delay estimation error at steady state is 0.000015 and number of iterations required is 360, 340, 280 for KVSS-LMS, AVSS-LMS and MAVSS-LMS respectively.

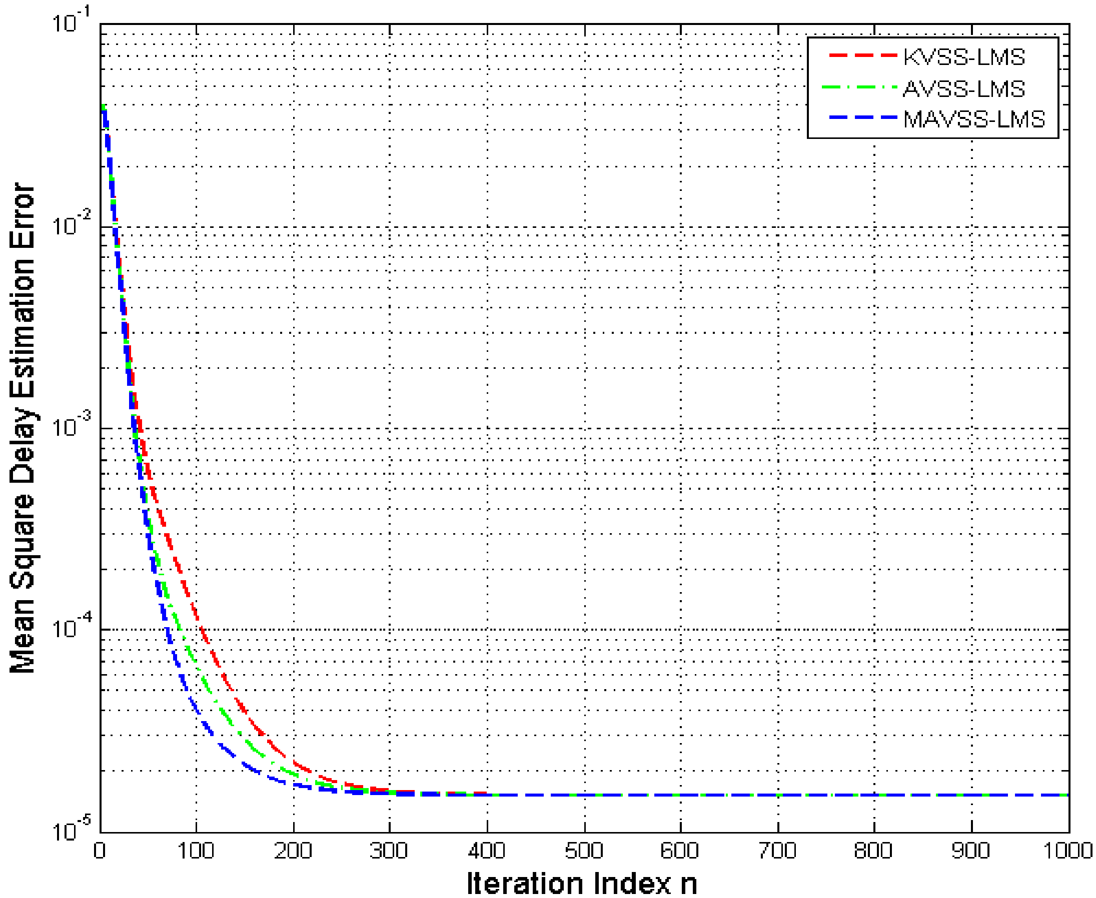

Fig. 6.6. Mean square delay estimation error vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

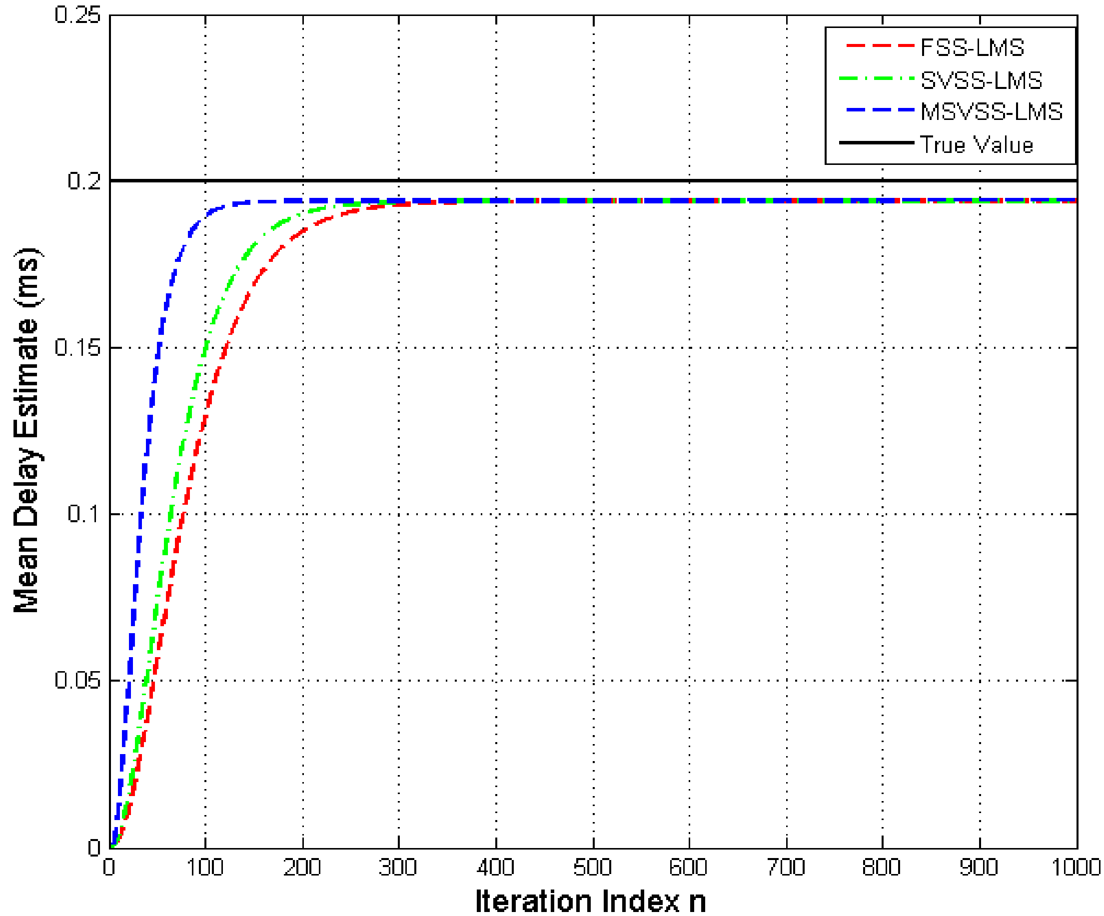

Fig. 6.7. Mean delay estimate vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

Fig. 6.7 and Fig. 6.8 projects the trajectories of FSS-LMS, SVSS-LMS and MSVSS-LMS algorithm for mean delay estimates in which delay D is held constant equal to 0.2. Step-size assumed is $\mu_{FSS} = \mu_{SVSS} = \mu_{MSVSS} = 0.0165$ for different algorithm. $\beta_{SVSS} = \beta_{MSVSS} = 0.2$ and $\gamma_{SVSS} = \gamma_{MSVSS} = 0.4$ for SVSS-LMS and MSVSS-LMS respectively. The mean amplitude estimate in Fig. 6.7 at steady state is 0.194 and number of iterations required are 305, 276 and 165 for FSS-LMS, SVSS-LMS and MSVSS-LMS respectively.

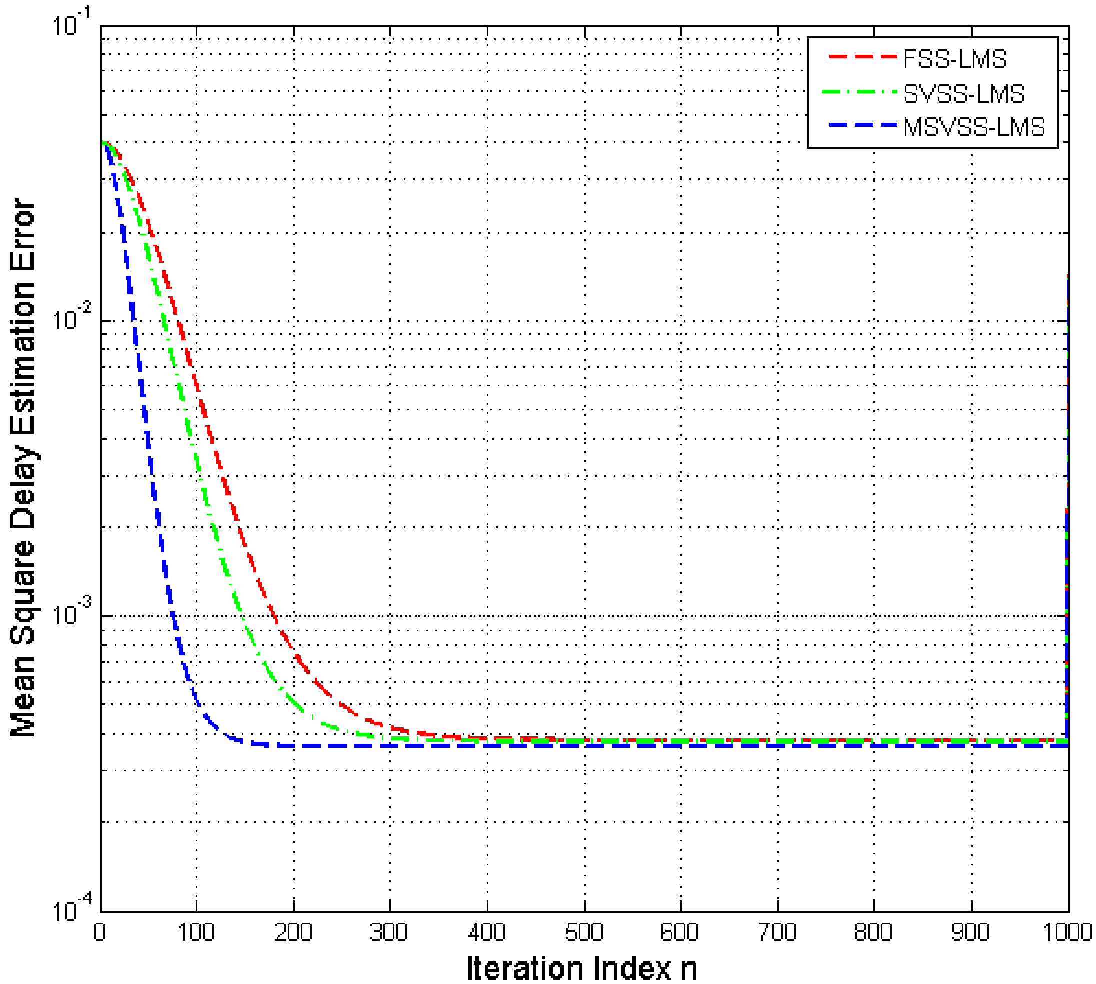

Fig. 6.8. Mean square delay estimation error vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

In Fig. 6.8, the mean square delay estimation error at steady state is 0.00039 approx. and number of iterations required is 450, 380 and 250 for FSS-LMS, SVSS-LMS and MSVSS-LMS respectively.

Following simulation results are carried out under stationary SOE with amplitude A held constant equal to 0.8 and delay D is changed from 0.2 to 0.3 at iteration number 1000. Step-size (μ) assumed is $\mu_{KVSSMIN} = \mu_{MAVSSMIN} = 0.0165$ and $\mu_{KVSSMAX} = \mu_{MAVSSMAX} = 0.1$ for KVSS-LMS and MAVSS-LMS, while for AVSS-LMS step-size μ_{AVSS} is 0.1. $\alpha_{KVSS} = \alpha_{AVSS} = \alpha_{MAVSS}$ and $\gamma_{KVSS} = \gamma_{AVSS} = \gamma_{MAVSS}$ is equated as 0.97 and 0.00048 respectively.

β_{AVSS} is 0.99 in AVSS-LMS. η_{MAVSS} and λ_{MAVSS} is 0.97 and 0.0005 respectively in MAVSS-LMS.

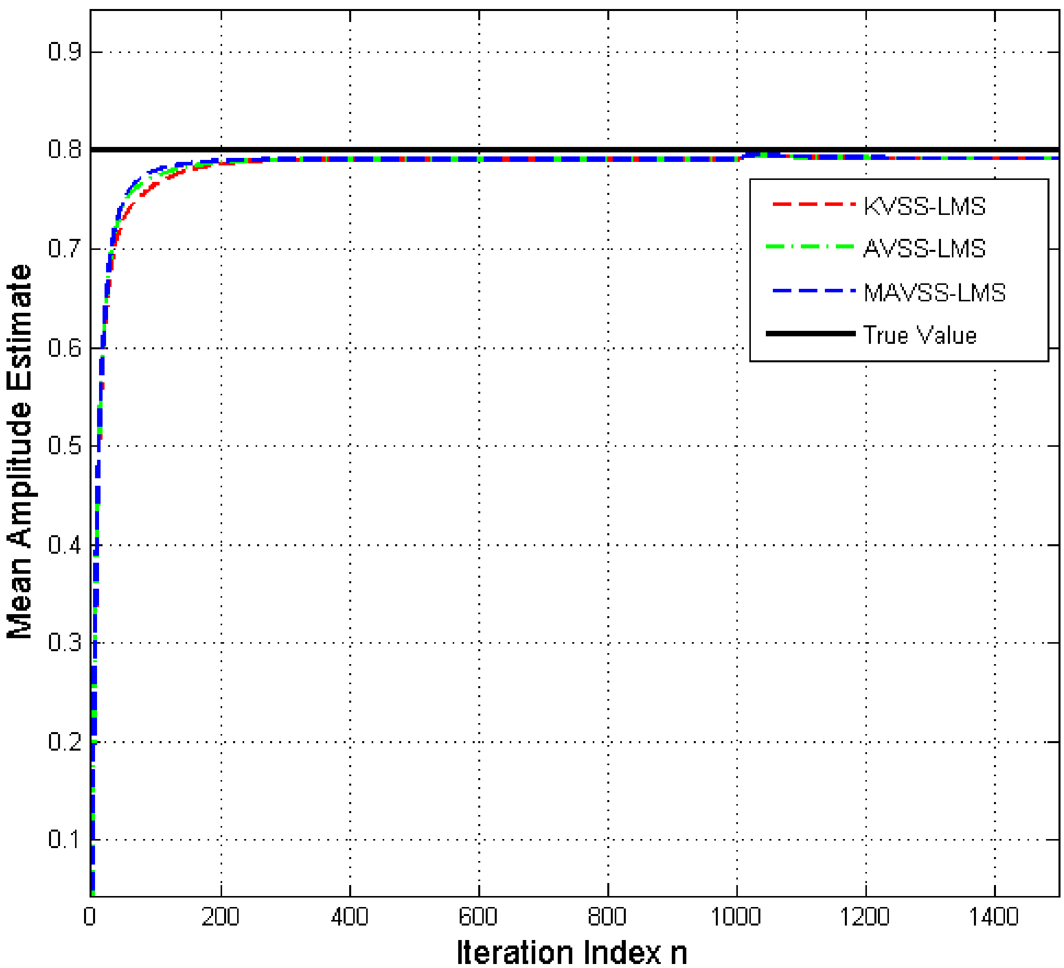

Fig. 6.9. Mean amplitude estimate vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

Fig. 6.9 and Fig. 6.10 addresses tracking ability of KVSS-LMS, AVSS-LMS and MAVSS-LMS in terms of amplitude. When the delay constant value is changed from 0.2 to 0.3 at iteration number 1000, then the mean is followed at different speed indicating the tracking ability. There is not much difference in the convergence rate of various algorithms for mean amplitude as shown in Fig. 6.9, but the steady state value of mean square error after the change requires number of iterations for MAVSS-LMS, AVSS-LMS and KVSS-LMS are 200, 230 and 230 respectively as shown in Fig. 6.10.

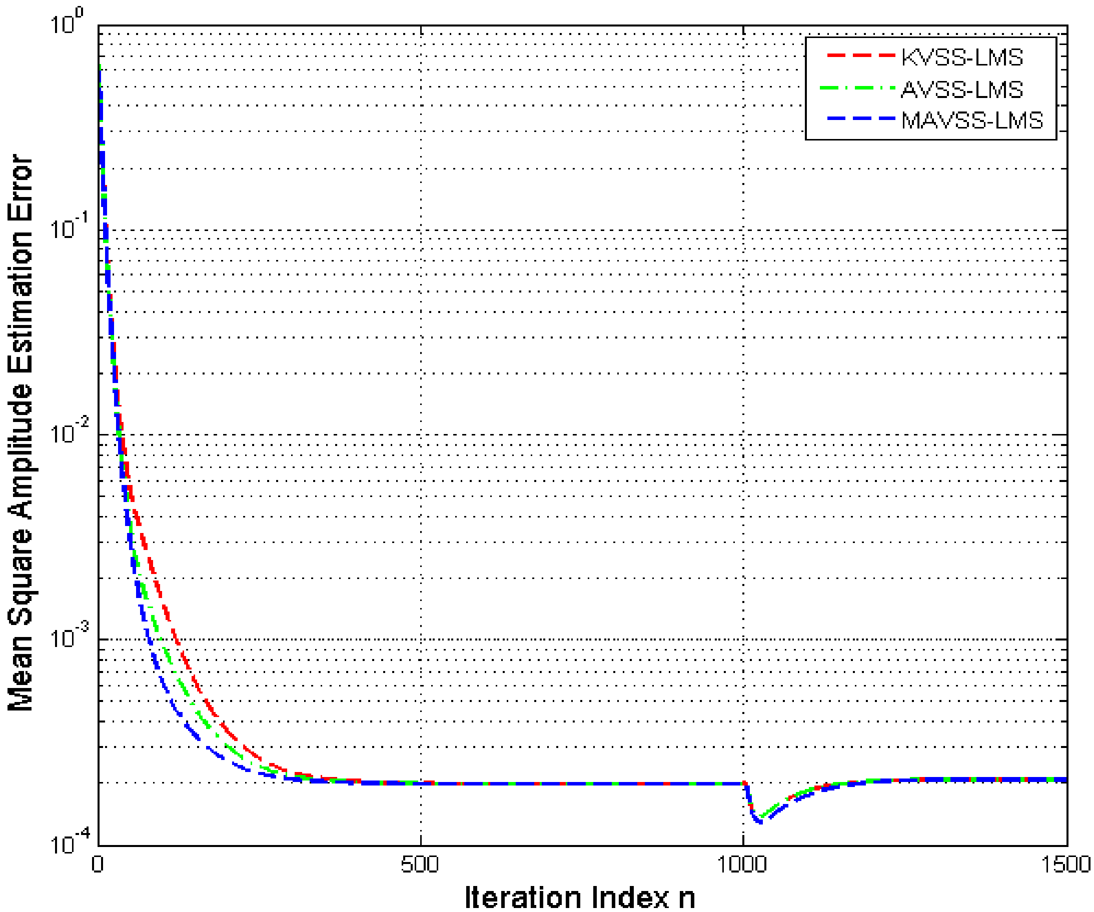

Fig. 6.10. Mean square amplitude estimation error vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

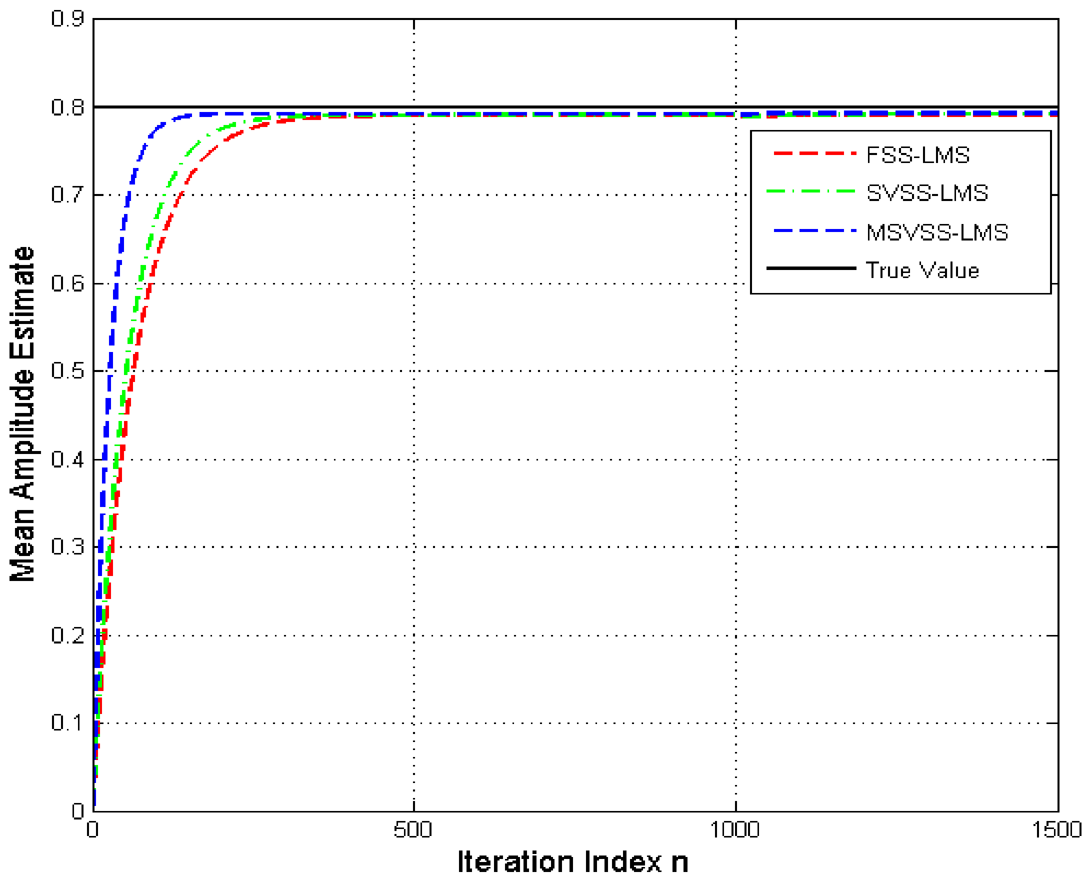

Fig. 6.11. Mean amplitude estimate vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

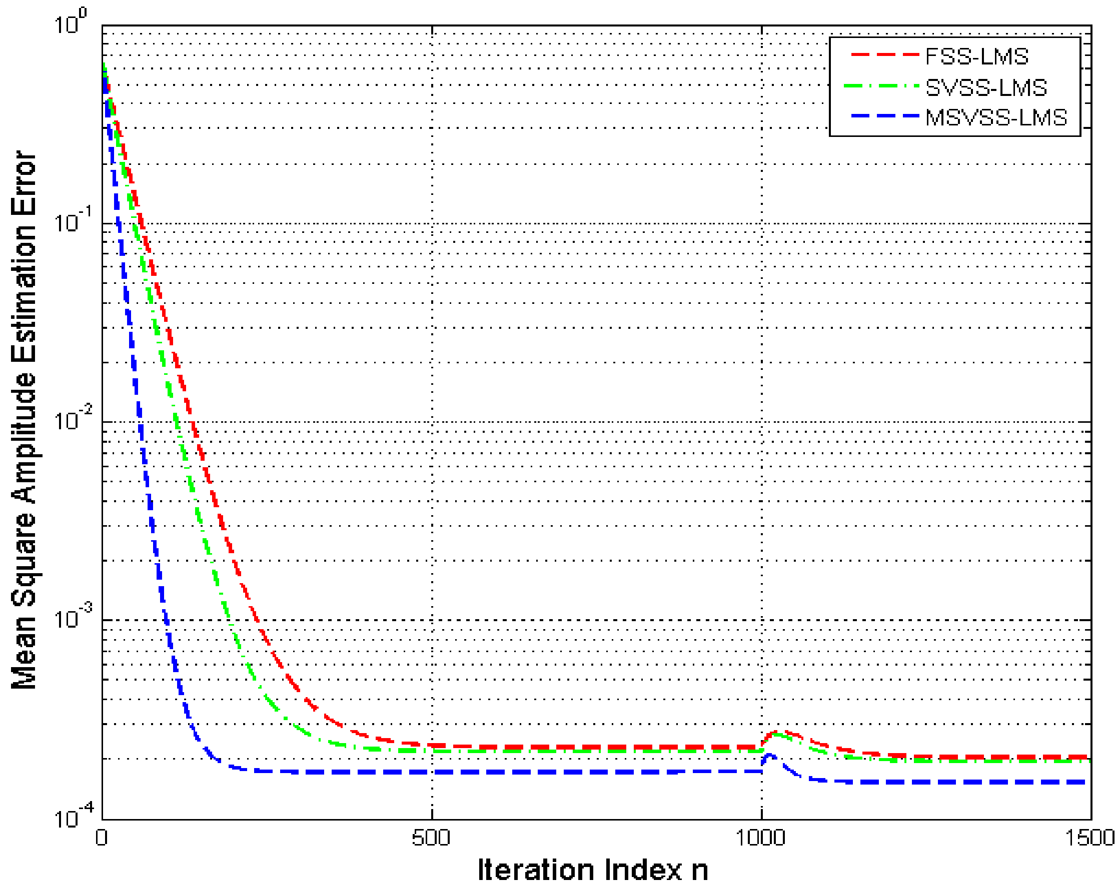

Fig. 6.12. Mean square amplitude estimation error vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

Fig. 6.11 and Fig. 6.12 depicts the tracking ability of FSS-LMS, SVSS-LMS and MSVSS-LMS in terms of amplitude. When the delay constant value is changed from 0.2 to 0.3 at iteration number 1000, then the mean is followed at different speed indicating the tracking ability. There is not much difference in terms of mean amplitude estimate. For steady state value of mean square error amplitude, number of iterations required for MSVSS-LMS, SVSS-LMS and FSS-LMS is 220, 300 and 400 respectively as shown in Fig. 6.12.

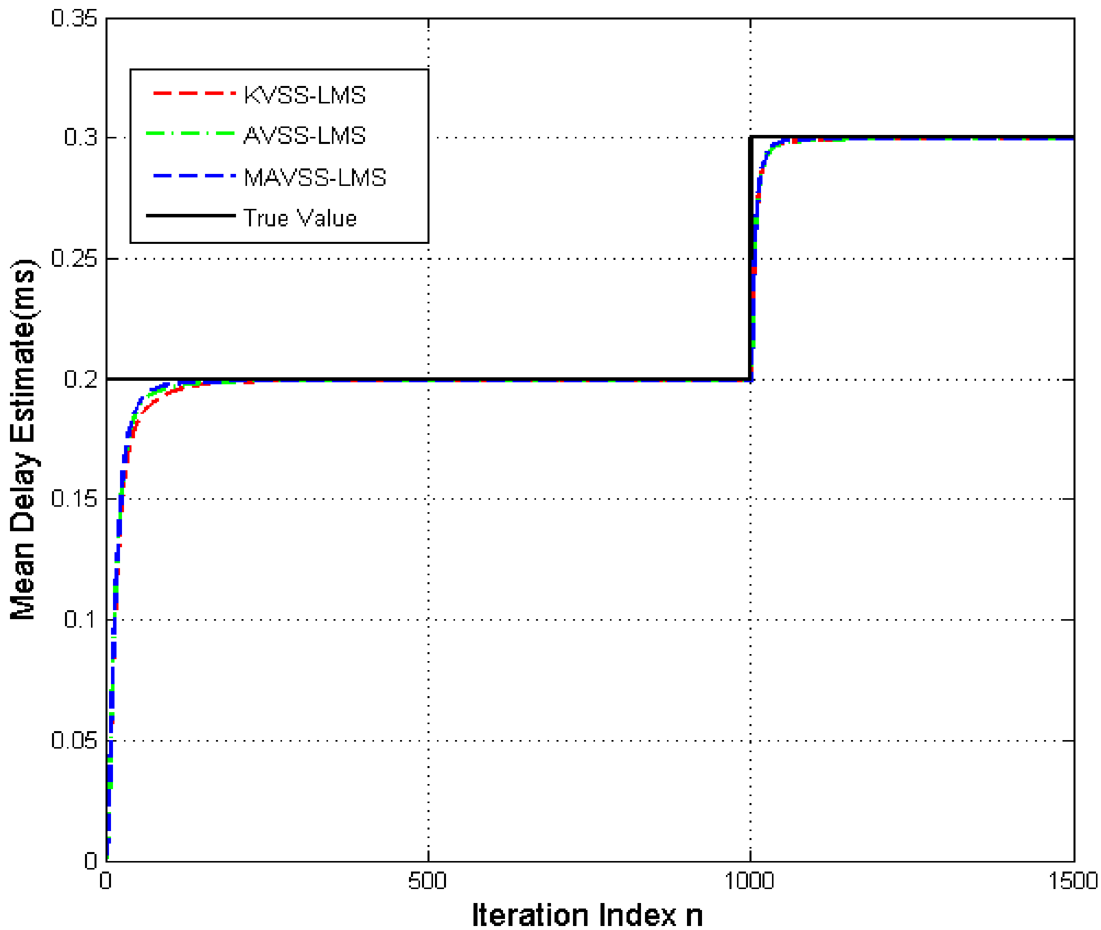

Fig. 6.13. Mean delay estimate vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

Fig. 6.13 and Fig. 6.14 projects the steady state value of mean delay with SNR=20dB. From Fig. 6.13, we can conclude that as the SNR increases, the accuracy of system increases. The mean estimation of mean delay changes to 0.2. Moreover, Fig. 6.14 shows the effect of parameter change with tracking response. Iterations required for MAVSS-LMS, AVSS-LMS and KVSS-LMS are 70, 110, and 120 respectively. Iterations needed for steady-state mean square delay estimation error are 221 for MAVSS-LMS, 300 for AVSS-LMS and 400 for KVSS-LMS.

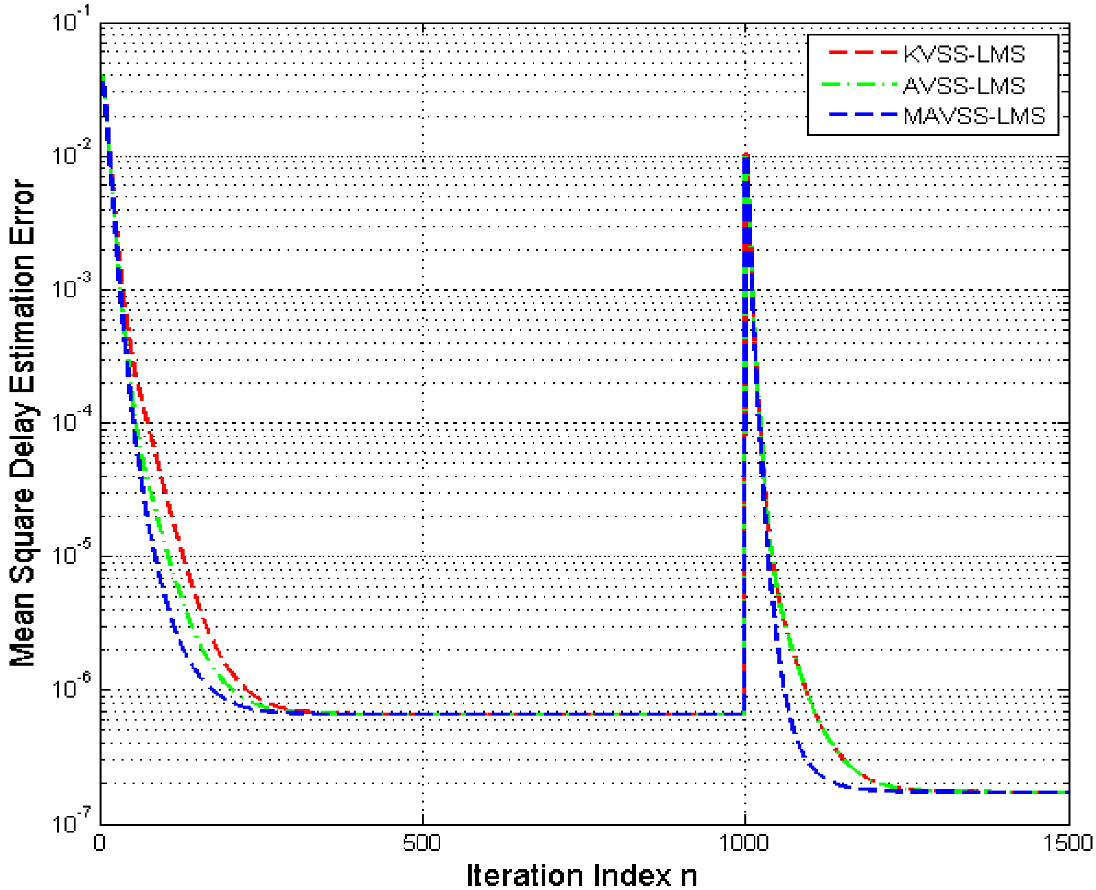

Fig. 6.14. Mean square delay estimation error vs. iteration number using KVSS-LMS, AVSS-LMS, and MAVSS-LMS.

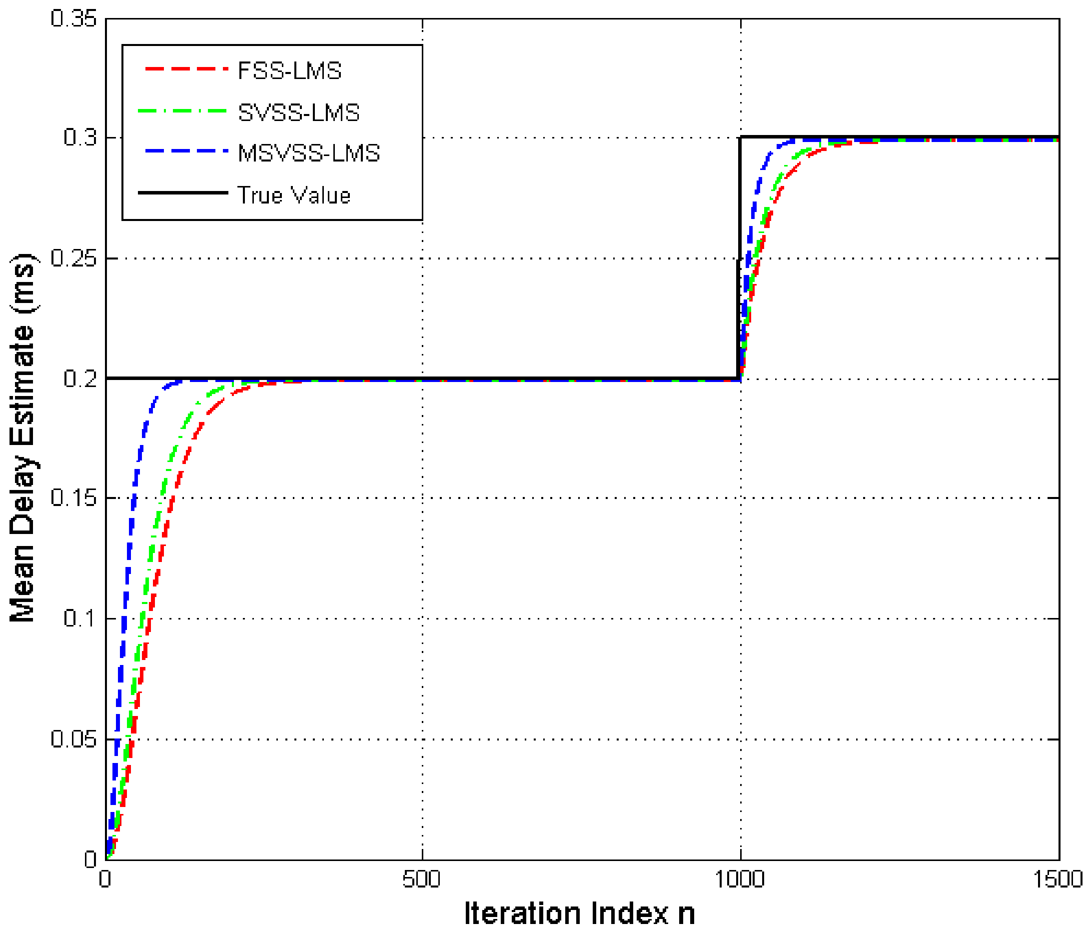

Fig. 6.15. Mean delay estimate vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

Similarly Fig. 6.15 and Fig. 6.16 represents the steady state value of mean delay with tracking, number of iterations required for MSVSS-LMS, SVSS-LMS and FSS-LMS after delay change from 0.2 to 0.3 are 115, 200 and 330 respectively. Iterations required for MSVSS-LMS, SVSS-LMS and FSS-LMS for steady-state mean square delay estimation error are 170, 270 and 360 respectively.

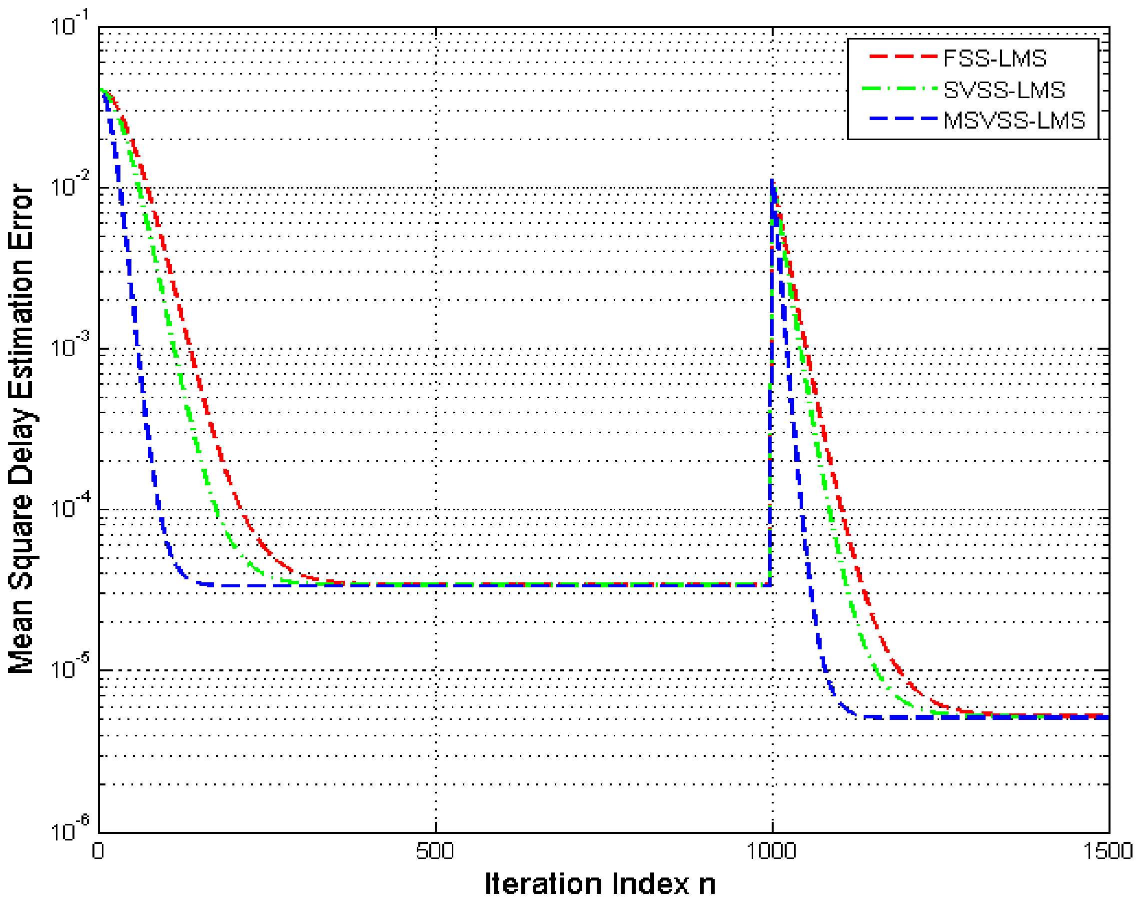

Fig. 6.16. Mean square delay estimation error vs. iteration number using FSS-LMS, SVSS-LMS, and MSVSS-LMS.

Chapter 7

CONCLUDING REMARKS AND FUTURE SCOPE

This is the conclusion of this research work and future scope that can be carried out to uplift this research work.

7.1 Concluding Remarks

The main rational behind the adaptive filtering methodology is to reduce the distance from the optimum solution for a given problem and adapt the step-size of filter accordingly. The performance of system is susceptible to degradation due to the presence of strong noise. In practical scenario, prior knowledge of signal power is not available, thus, it is very difficult to choose the step-size. In my research work, the performance of various VSS with respect to FSS in terms of mean amplitude, mean delay, mean square amplitude error estimation and mean square delay estimation error has been evaluated. As the SNR reduces, the accuracy of system is affected. To demonstrate the effect of SNR, the simulation results are performed with 10dB and 20dB SNR. The performance measures used are convergence rate and tracking ability.

From Fig. 6.1, Fig. 6.2, Fig. 6.5, and Fig. 6.6, it is concluded that the convergence rate of MAVSS-LMS is more, followed by AVSS-LMS and KVSS-LMS under different SNR. In VSS algorithms, the steady state error greatly influences the misadjustment while in the case of FSS, the misadjustment remains constant. When the step-size is varied in accordance with squared autocorrelation of error at adjacent intervals, the noise immunity of the algorithm increases. When the prediction error increases, the step-size is increased to track the input signal and when error decreases, the step-size is reduced to achieve the required minimum misadjustment. KVSS-LMS performs good results in high SNR but has poor anti-noise ability while AVSS-LMS have the poor tracking ability with good anti-noise ability. MAVSS-LMS has better noise ability than AVSS-LMS and KVSS-LMS algorithms.

From Fig. 6.3, Fig. 6.4, Fig. 6.7, and Fig. 6.8, the evaluation of convergence rate of FSS-LMS, SVSS-LMS and MSVSS-LMS is carried out. MSVSS-LMS takes minimum iterations followed by SVSS-LMS and FSS-LMS. Moreover, the mean square steady state estimation error is minimum in MSVSS-LMS while keeping the step-size equal to 0.0165.

Fig. 6.8 illustrates the tracking ability of KVSS-LMS, AVSS-LMS and MAVSS-LMS. The delay constant value is varied at iteration number 1000. There is not much difference in amplitude tracking ability but in the case of delay estimate, MAVSS outstands others. Tracking ability of MSVSS-LMS is more as compared to SVSS-LMS and FSS-LMS. It depends on γ_{SVSS} and γ_{MSVSS}, if its value increases then the convergence rate increases but it increases the steady state error.

In this thesis, the convergence rate and tracking the performance of different VSS-LMS algorithms is investigated and compared, while estimating the time-delay as well as relative amplitude simultaneously for replicas of sinusoidal signals (at same carrier frequency) corrupted by noise, which is a linear system identification problem. It may be inferred from results that MAVSS-LMS exhibits higher convergence rate than AVSS-LMS and KVSS-LMS algorithm based approaches in the time-delay and amplitude estimation. The convergence rate of MSVSS-LMS algorithm is better than SVSS-LMS and FSS-LMS algorithm based approaches. But, the mean squared delay estimation error is less in the case of MAVSS-LMS, AVSS-LMS, and KVSS-LMS than MSVSS-LMS, SVSS-LMS and FSS-LMS algorithms. The presented work gives satisfactory results under the high SNR conditions only.

7.2 Future Scope

The algorithm discussed in this research work performs effectively under high SNR conditions, but as the signal deteriorates, accuracy of the system response is affected. Therefore, if the input signal has low variance non-stationary noisy signal or both “input and noise” are non-stationary at the same time, then the problem can be formulated and implemented at high SNR only. For the above said problem, smoothing operation can be used (taking an average of present and past samples).

The efficiency of FSS-LMS, KVSS-LMS, AVSS-LMS, MAVSS-LMS and SVSS-LMS algorithms depends on apriori knowledge of the step-size, but practically it is not possible, so, some algorithm with an initial guess of step-size should be formulated in order to make the algorithm versatile with the environment and computes tracking ability in time-varying or non-stationary environment.

Higher-order statistics like kurtosis, skewness for updating of step-size to reduce the impact of strong noise. For Gaussian process, the cumulant with the order of higher than two is zero. This reduces the sensitivity of the algorithm to additive noise.

REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, 3rd ed., Englewood Cliffs, NJ: Prentice-Hall, 1985.

[2] B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, and R. C. Goodlin, "Adaptive noise cancelling: Principles and applications," *Proc. IEEE*, vol. 63, no. 12, pp. 1692-1716, December 1975.

[3] M. Chakraborty, "A new adaptive filter for estimating and tracking the delay and the amplitude of a sinusoid," *IEEE Trans. Instrum. Meas.,* vol. 59, no. 11, pp. 3049-3057, November 2010.

[4] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Trans. Signal Process.,* vol. 40, no. 7, pp. 1633-1642, July 1992.

[5] T. Aboulnasr and K. Mayyas, "A robust variable step-size LMS-type algorithm: Analysis and Simulations," *IEEE Trans. Signal Process.,* vol. 45, no. 3, pp. 631-639, March 1997.

[6] Z. Kun and Z. Xiubing, "A new modified robust variable step size LMS algorithm," *Proc. IEEE Conf. Ind. Elect. App.,* Xi'an, China, pp. 2699-2703, May 2009.

[7] Y. Chen, J. Tian, and Y. Liu, "Variable step-size LMS algorithm based on modified sigmoid function," *Proc. IEEE Conf. Aud. Lang., Image Process.,* Shanghai, China, pp. 627-630, July 2014.

[8] G.C. Carter, "*Coherence and Time-delay Estimation: An Applied Tutorial for Research, Development, Test and Evaluation Engineers*," Piscataway, NJ: IEEE Press, February 1993.

[9] C. H. Knapp and G.C Carter, "The generalized correlation method for estimation of time-delay," *IEEE Trans. Acoust., Speech, Signal Process.,* vol. ASSP-24, no. 4, pp. 320-327, August 1976.

[10] D. H. Youn, N. Ahmed, and G. C. Carter, "On the Roth and Scot algorithms: time domain implementations," *Proc. IEEE*, vol. 71, no. 4, pp. 536-538, April 1983.

[11] V. Zetterberg, M.I. Pettersson, and I. Claesson, "Comparison between whitened generalized cross-correlation and adaptive filter for time-delay estimation with scattered

arrays for passive positioning of moving targets in Baltic Sea shallow waters," *Proc. IEEE Oceans*, vol. 3, pp. 2356-2361, 2005.

[12] K. W. Wilson and T. Darrell, "Learning a precedence effect-like weighting function for the generalized cross-correlation framework," *IEEE Trans. Aud., Speech and Lang. Process.* , vol. 14, no. 6, pp. 2156-2164, November 2006.

[13] T. Qiu and H. Wang, "An Eckart-weighted adaptive time-delay estimation method," *IEEE Trans. Signal Process*., vol. 44, no. 9, pp. 2332-2335, September 1996.

[14] S. Haykin*, Adaptive filter theory*, 3rd ed., Englewood Cliffs, NJ: Prentice-Hall, 1986.

[15] Paulo S.R. Diniz, *Adaptive filtering-Algorithms and Practical Implementation*, 2nd ed., LLC 233, NY: Springer, 2008.

[16] H.C. So, "A comparative study of two discrete-time phase delay estimators," *IEEE Trans. Instrum. Meas.,* vol. 54, no. 6, pp. 2501-2504, December 2005.

[17] D. L. Maskell and G. S. Woods, "The discrete time quadrature sub-sample estimation of delay*," IEEE Trans. Instrum. Meas.*, vol. 51, no. 1, pp. 133-137, February 2002.

[18] A. K. Nandi, "On the subsample time-delay estimation of narrowband ultrasonic echoes," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control.,* vol. 42, no. 6, pp. 993-1001, November 1995.

[19] F. A Reed, P. L. Feintuch, and N.J. Bershad, "Time-delay estimation using the LMS adaptive filter-static behavior*," IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 3, pp. 561-571, June 1981.

[20] M. Chakraborty, H.C. So, and J. Zheng, "A new adaptive algorithm for delay estimation of sinusoidal signals," *IEEE Signal Process. Lett.*, vol. 14, no. 12, pp. 984-987, December 2007.

[21] A. K. Kohli and D. K. Mehra, "Tracking of time-varying channels using two-step LMS-type adaptive algorithm," *IEEE Trans. Signal Process.,* vol. 54, no. 7, pp. 2606-2615, July 2006.

[22] B. Widrow, J. M McCool, M. G. Larimore, and C. R. Johnson, "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE,* vol. 64, no. 8, pp. 1151-1162, August 1976.

[23] J. K. Hwong and Y. P. Li, “Variable step-size LMS algorithm with a gradient-based weighted average,” *IEEE Signal Process. Lett.*, vol. 16, no. 12, pp. 1043-1046, December 2009.

[24] K. Gedalyahu and Y. C. Eldar, “Time-delay estimation from low rate samples: a union of subsamples approach,” *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3017-3031, June 2010.

[25] V. H Nascimento, “Analysis of the hierarchical LMS algorithm,” *IEEE Signal Process. Lett.*, vol. 10, no. 3, pp. 78-81, March 2003.

[26] Y. Guo and C. Meng, “Subsample time-delay estimation in impulsive noise environment,” *Proc. IEEE Intl. Conf. Inf. Theory, Security*, Beijing, China, pp. 810-812, December 2010.

[27] A. Sandmair, M. Lietz, J. Stefan, and F. P. Leon, “Time-delay estimation in time-frequency domain based on a line detection approach,” *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Process.*, Prague, Czech Republic, pp. 2716-2719, May 2011.

[28] J. P. Delmas and Y. Meurise, “On the Cramer Rao bound and maximum likelihood in passive time-delay estimation for complex signals,” *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Process.,* Kyoto, pp. 3541-3544, March 2012.

[29] R. L. Dass and M. Chakraborty, “On convergence of proportionate-type normalized least mean square algorithms,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 5, pp. 491-495, May 2015.

[30] H. Lee, S. Kim, J. Lee, and W. Song, “A variable step-size diffusion LMS algorithm for distributed estimation,” *IEEE Trans. Signal Process.*, vol. 63, no. 7, pp. 1808-1820, April 2015.

[31] D. I. Paziatis and A.G. Constantinides, “A novel kurtosis driven variable step-size adaptive algorithm,” *IEEE Trans. Signal Process.*, vol. 47, no. 3, pp. 864-872, March 1999.

[32] R. W. Harris , D. M. Chabries, and F. A. Bishop, “A variable-step adaptive filter logarithm,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 2, pp. 309-316, April 1986.

[33] S. B. Gelfand, Y. Wei, and J. V Krogmeier, “The stability of variable step-size LMS algorithms,” *IEEE Trans. Signal Process.*, vol. 47, no. 12, pp. 3277-3288, December 1999.

[34] J. B Foley and F.M Boland, “A note on the convergence analysis of LMS adaptive filters with gaussian data*,” IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 7, pp. 1087-1089, July 1988.

[35] K. Mayyas, “Performance analysis of the deficient length LMS adaptive algorithm,” *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2727-2734, August 2005.

[36] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step-size,” *IEEE Trans. Signal Process.*, vol. 41, no. 6, pp. 2075-2087, June 1993.

[37] T. J. Shan and T. Kailath, “Adaptive algorithm with an automatic-gain-control feature,” *IEEE Trans. Circuits Syst.*, vol. 35, no. 1, pp. 122-127, January 1988.

[38] N. J. Bershad and L.Z Qu, “On the probability density function of the LMS adaptive filter weights,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 1, pp. 43-56, January 1989.

[39] Y. Huang, A. Hu, W. Liu, T. Li, and S. Xie, “An extension of DTFT based sinusoidal signal time-delay estimation algorithm for linear time varying situation,” *Proc. IEEE Cyber, Physical, Social Comput.,* vol. 32, no. 11, Beijing, China, pp. 3071-3078, August 2013.

[40] H. C So, P. C Ching, and Y. T Chan, “A new algorithm for explicit adaptation of time-delay,” *IEEE Trans. Signal Process.*, vol. 42, no. 7, pp. 1816-1820, July 1994.

[41] S. R. Dooley and A. K. Nandi, “Adaptive subsample time-delay estimation using Lagrange interpolators,” *IEEE Signal Process. Lett.*, vol. 6, no. 3, pp. 65-67, March 1999.

[42] Z. Cheng and T. T. Tjhung, “A new time-delay estimator based on ETDE,” *IEEE Trans. Signal Process.*, vol. 51, no. 7, pp. 1859-1869, July 2003.

[43] D. L. Maskell and G. S. Woods, “Adaptive subsample delay estimation using a modified quadrature phase detector,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 10, pp. 669-674, October 2005.

[44] D. L. Maskell and G. S. Woods, “The estimation of subsample time-delay of arrival in the discrete time measurement of phase delay,” *IEEE Trans. Instrumen. Meas.*, vol. 48, no. 6, pp. 1227-1231, December 1999.

[45] P. L. Feintunch, N. J. Bershad, and F. R. Reed, “Time-delay estimation using the LMS adaptive filter-dynamic behaviour,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 3, pp 571-576, June 1981.

[46] M. Chakraborty, A.S. Dhar, and M. H. Lee, “A trigonometric formulation of the LMS algorithm for realization on pipelined CORDIC,” *IEEE Trans. Circuits Syst. II, Exp. Briefs.*, vol. 54, no. 9, pp. 530-534, September 2005.

[47] S. Marcos and O. Macchi, “Tracking capability of the least mean square algorithm application to an asynchronous echo canceller,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 11, pp. 1570-1578, November 1987.

[48] J. B. Evans, P. Xue, and B. Liu, “Analysis and implementation of variable step-size adaptive algorithm,” *IEEE Trans. Signal Process.,* vol. 41, no. 8, pp. 2517-2535, August 1993.

[49] W. B. Mikhael, F. H. Wu, L. G. Kazovsky, G. S. Kang, and L. J. Fransen , “Adaptive filters with individual adaptation of parameters,” *IEEE Trans. Circuits Syst.*, vol. CAS-33, no. 7, pp. 677-686, July 1986.

[50] A. Feuer and E. Weinstein, “Convergence analysis of LMS filters with uncorrelated gaussian data,” *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, no. 1, pp. 222-230, February 1985.

[51] V. H. Nascimento and A. H. Sayed, “On the learning mechanism of adaptive filters,” *IEEE Trans. Signal Process.*, vol. 48, no. 6, pp. 1609-1625, June 2000.

[52] H. Shin, A. H. Sayid, and W. Song, “Variable step-size NLMS and affine projection algorithms,” *IEEE Signal Process. Lett.*, vol. 11, no. 2 , pp. 132-135, February 2004.

[53] J. E. Greenberg, “Modified LMS algorithm for speech processing with an adaptive noise canceller,” *IEEE Trans. Speech, Audio Process.*, vol. 6, no. 4, pp. 338-351, July 1998.

[54] J. C. H. Peng and N. K. C. Nair, “Adaptive sampling scheme for monitoring oscillations using prony analysis,” *IET Gener. Transm. Distrib.,* vol. 3, no. 12, pp. 1052-1060, December 2009.

[55] Y. Huang, A. Hu, S. Xie, and D. Zhu, “A novel radio interferometric time-delay tracking scheme,” *J. Applied Mechanic, Materials*, vols. 411-414, pp. 1470-1474, September 2013.

[56] Li Xiao-Bo, Fan Yang-Yu, and Peng-Ke, "A variable step-size LMS adaptive filtering algorithm," *Proc. IEEE Wireless Commun., Networking, Mobile Comput.*, Beijing, China, pp. 1-4, September 2009.

[57] S. Zhong, W. Xia, J. Song, and Z. He, "Super-resolution time-delay estimation in multipath environments using normalized cross spectrum," *Proc. IEEE Commun., Circuits, Syst.,* vol. 1, pp. 289-291, November 2013.

[58] S.C. Douglas, *Introduction to Adaptive filters*, 2nd ed., Boca Raton: CRC Press LLC, 1999.

LIST OF PUBLICATIONS

[1] R. Garg and A. K. Kohli, "Parameter estimation and tracking of sinusoidal signal using variable step-size algorithms," *Elsevier, Optik-Int. J. Light, Electron Optics,* Under Review, 2016.

801463021

by Rohit Garg

801463021

ORIGINALITY REPORT

9%	5%	7%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	zone.ni.com Internet Source	1%
2	Rai, Amrita, and Amit Kumar Kohli. "Adaptive Polynomial Filtering using Generalized Variable Step-Size Least Mean pth Power (LMP) Algorithm", Circuits Systems and Signal Processing, 2014. Publication	1%
3	Chakraborty, Mrityunjoy. "A New Adaptive Filter for Estimating and Tracking the Delay and the Amplitude of a Sinusoid", IEEE Transactions on Instrumentation and Measurement, 2010. Publication	1%
4	dspace.thapar.edu:8080 Internet Source	1%
5	www.ijert.org Internet Source	1%
6	Zou kun. "A new modified robust variable step size LMS algorithm", 2009 4th IEEE	1%