Ph. D Thesis

Development and Evaluation of Blind Identification Techniques for Nonlinear Systems

M.V RAJESH

##############

Division of Electronics Engineering
School of Engineering
Cochin University of Science & Technology
Kochi, Kerala, India-682 022

December 2010

Development and Evaluation of Blind Identification Techniques for Nonlinear Systems

a thesis

submitted in partial fulfillment of the degree of

DOCTOR OF PHILOSOPHY

by

M.V RAJESH

under the guidance of
Dr. R Gopikakumari
&
Dr. A Unnikrishnan

##############

Division of Electronics Engineering
School of Engineering
Cochin University of Science & Technology
Kochi, Kerala, India-682 022

December 2010

Title

Ph. D thesis in the field of Intelligent Signal Processing

"Development and Evaluation of Blind Identification Techniques for Nonlinear Systems"

Author

M V Rajesh
Research Scholar
Division of Electronics Engineering
School of Engineering, CUSAT
Registration No.: 3135

Research Advisors

Dr. R Gopikakaumari
Head, Division of Electronics Engineering
School of Engineering, CUSAT
&
Dr. A Unnikrishnan
Associate Director
Naval Physical & Oceanographic Laboratory (NPOL)
DRDO, Thrikkakara, Kochi

December, 2010

Declaration

I hereby declare that the work presented in this thesis entitled "*Development and Evaluation of Blind Identification Techniques for Nonlinear Systems",* is based on the original work done by me under the supervision and guidance of Dr. R Gopikakumari, Head, Division of Electronics, School of Engineering, CUSAT and Dr. A Unnikrishnan, Associate Director, Naval Physical and Oceanographic Laboratory (NPOL), under the Defense Research & Development Organization (DRDO), Thrikkakara, Cochin-22. No part of this thesis has been presented for any other degree from any other institution.

Thrikkakara

December 19, 2010 M.V Rajesh

Certificate

This is to certify that the thesis entitled, "*Development and Evaluation of Blind Identification Techniques for Nonlinear Systems",* is a report of the original work done by Mr. M.V Rajesh, under our supervision and guidance in the School of Engineering, CUSAT. No part of this thesis has been presented for any other degree from any other institution.

Dr. R Gopikakumari
Supervising Guide,
Head, Division of Electronics,
School of Engineering, CUSAT.

Dr. A Unnikrishnan
Co guide,
Associate Director,
Naval Physical and
Oceanographic Laboratory (NPOL)

Thrikkakara
May 29, 2010

Acknowledgements

This work could not be realized without the help and support of many people.

First, I wish to express my sincere gratitude to the Director, Institute of Human Resources Development, Govt. of Kerala, Thiruvananthapuram for providing me no objection certificate for my Ph. D program.

I would like to thank the principal, School of Engineering, Cochin University of Science & Technology, Kerala, India for providing me the resources and facilities to carry out this thesis work.

I am extremely grateful to Dr. R Gopikakumari and Dr. A Unnikrishnan, my supervising guide and co-guide, for providing me with the opportunity to work in the field of nonlinear system analysis. They have encouraged and inspired me with new ideas and fruitful discussions. I am indebted to them for allowing me the chance to pursue my Ph. D under their guidance in the university and sharing their valuable time with me throughout the work.

I am thankful to the members of research committee and faculty of the School of Engineering, for their kind suggestions at various stages of this work.

I would like to gratefully acknowledge Archana. R, Asst. Professor at Federal Institute of Science & Technology (FISAT), Angamali, Cochin, Kerala for her cooperation throughout the completion of the work.

I would like to thank the Technical Education Quality Improvement Program (TEQIP) at Model Engineering College, Thrikkakara, Cochin-21, for the financial assistance for foreign travel to attend and present one of my research papers.

I am thankful to my colleagues at Model Engineering College including Dr. Rajesh V.G, Dr. K P Jose, Mr. Rejikumar M.K and my students for their moral support and encouragement in carrying out the research work.

Thanks are also due to Mr. Jyothish & Mr. Shahul of division of Electronics Engineering and Mr. Babu Varghese of the machine shop, SOE, CUSAT for their assistances during the presentation of the thesis work.

My special thanks go to my wife Dr. Bindhu for her endless and manifold support and encouragement. I am truly grateful to my son Aravindan for his patients with me.

Finally I would like to dedicate this thesis to my mother and the memories of my great father Mr. M K Vasu, who have worked hard to provide me with the chances for engineering education and progress.

Abstract

Identification and Control of Non-linear dynamical systems are challenging problems to the control engineers. The topic is equally relevant in communication, weather prediction, bio medical systems and even in social systems, where nonlinearity is an integral part of the system behavior. Most of the real world systems are nonlinear in nature and wide applications are there for nonlinear system identification/modeling. The basic approach in analyzing the nonlinear systems is to build a model from known behavior manifest in the form of system output. The problem of modeling boils down to computing a suitably parameterized model, representing the process. The parameters of the model are adjusted to optimize a performance function, based on error between the given process output and identified process/model output. While the linear system identification is well established with many classical approaches, most of those methods cannot be directly applied for nonlinear system identification.

The problem becomes more complex if the system is completely unknown but only the output time series is available. Blind recognition problem is the direct consequence of such a situation. The thesis concentrates on such problems. Capability of Artificial Neural Networks to approximate many nonlinear input-output maps makes it predominantly suitable for building a function for the identification of nonlinear systems, where only the time series is available. The literature is rich with a variety of algorithms to train the Neural Network model. A comprehensive study of the computation of the model parameters, using the different algorithms and the comparison among them to choose the best technique is still a demanding requirement from practical system designers, which is not available in a concise form in the literature.

The thesis is thus an attempt to develop and evaluate some of the well known algorithms and propose some new techniques, in the context of Blind recognition of nonlinear systems. It also attempts to establish the relative merits and demerits of the different approaches. Comprehensiveness is achieved in utilizing the benefits of well known evaluation techniques from statistics. The study concludes by providing the results of implementation of the currently available and modified versions and newly introduced techniques for nonlinear blind system modeling followed by a comparison of their performance.

It is expected that, such comprehensive study and the comparison process can be of great relevance in many fields including chemical, electrical, biological, financial and weather data analysis. Further the results reported would be of immense help for practical system designers and analysts in selecting the most appropriate method based on the goodness of the model for the particular context.

Table of Contents

List of Tables

List of Figures

Abbreviations

ANN	Artificial Neural Networks
SLP	Single Layer Perceptron
MLP	Multi Layer Perceptron
FF	Feed Forward
BPA	Back Propagation Algorithm
MA	Moving Average
NNSSIF	Neural network State Space Innovation Function
AR	Auto Regressive
ARMA	Auto Regressive Moving Average
ARMAX	Auto Regressive Moving Average with exogenous input
NAR	Nonlinear Auto Regressive
NARNAX	Nonlinear Auto Regressive Moving Average with exogenous input
SISO	Single Input Single Output
MIMO	Multiple Input Multiple Output
RBF	Radial Basis Function

RNN	Recurrent Neural Networks
MLFFN	Multi Layered Feed Forward Network
KF	Kalman Filter
EKF	Extended Kalman Filter
EM	Expectation Maximization
MLE	Maximum Likelihood Estimation
MSE	Mean Square Error
CRLB	Cramer Rao Lower Bound
PF	Particle Filter
DSS	Discrete State Space
SMC	Sequential Monte Carlo

Chapter 1

INTRODUCTION

Chapter 1 introduces the basic concepts of nonlinear system identification/modeling, the current status of the issue, motivation for the current work, objectives and methodologies adopted organization and outline of the thesis etc.

1.1 System Identification

Identification and Control of Non-linear dynamical systems are challenging problems to the control engineers. The problem of system identification and modeling consists of computing a suitably parameterized model, representing a process [1, 2, 3]. The parameters of the model are adjusted to optimize a performance function, based on error between the given process output and identified process/model output. Most of the real world systems are nonlinear in nature and wide applications are there for nonlinear system identification/modeling. The linear system identification field is well established with many classical approaches whereas most of those methods cannot be applied for nonlinear system identification [4, 5]. The problem becomes more complex if the system is completely unknown but only the output time series is available. The thesis concentrates on such problems. Capability of Artificial Neural Networks to approximate all linear and nonlinear input-output maps makes it predominantly suitable for the identification of nonlinear systems, where only the time series is available [7-13]. Different algorithms are available to train the Neural Network model. A comprehensive study of the models using different algorithms and the comparison among them to choose the best technique is not yet available in any of the published books or technical papers. This thesis is an attempt to develop and implement few of the well known and newly proposed algorithms, in the context of stochastic (where only time series is known) modeling of nonlinear systems, and to make a comparison to establish the relative merits and demerits. When the output time series alone is available, the process is also termed blind identification/modeling [33-36].

Two basic types of modeling problems arise. In the first type, one can associate with each physical phenomenon, a small number of measurable causes (inputs) and a small number of measurable effects (outputs). The outputs and the inputs can generally be related through a set of mathematical equations, in most cases nonlinear partial differential equations. The determination of these equations is the problem of modeling in such cases. These can be obtained either by writing a set of equilibrium equations based on mass and energy balance and other physical laws, or one may use the black box approach which may consists of determining the equations from the past records of the inputs and outputs. Modeling problems of this type appear quite often in engineering practice. Some typical problems are modeling of (i) a stirred – tank chemical reactor, (ii) a multi machine electrical power system, (iii) a synchronous orbit communications satellite and (iv) the control mechanism of a nuclear power reactor [62-64]. In each of these examples one can easily identify certain input and output quantities, and then obtain mathematical model relating them.

Another type of modeling problem arises in those situations where although it is possible to identify a certain quantity as a definite measurable output or effect, the causes are not so well defined. Some typical examples are (i) the annual population of a country, (ii) the annual rainfall in a certain country, (iii) the average annual flow of a river, and (iv) the daily value of a certain stock in the stock market. In all these cases, one have a sequence of outputs, which will be called a time series, but the inputs or causes are numerous and not quite known in addition to often being unobservable. The models in such cases are called stochastic models, due to a certain amount of uncertainty which is unavoidable [32, 33].

1.1.1 System description

A system can be described by one of the following.

- A transfer function
- A linear differential equation with constant coefficient that relates the input and output of the system.
- An impulse response.
- A set of state equations.

By knowing the input of the system, one can determine the response of the system. But in many cases one may not be having the system description .The system transfer function, impulse response, differential equation; state equation etc has to be derived from a sample of input and output [13-14].

Another type of modeling problem arise in those situation where one can identify a certain quantity as a definite measurable output or effect, the causes are not well defined. This is called time **series modeling**, where inputs or causes are numerous and not quite known in addition to often being unobservable. This type of modeling is also called stochastic modeling. System identification is concerned with the determination of the system models from records of system operation. The problem can be represented diagrammatically as below.

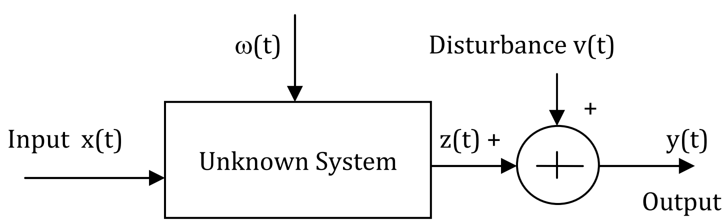


Fig.1.1. A general system configuration

where x(t) is the known input vector of dimension 'm'

z(t) is the output vector of dimension 'p'

ω(t) is the input disturbance vector

v(t) is the measured output vector of dimension 'p'

Thus the problem of system identification is the determination of the system model from records of x(t) and y(t).

1.1.2 System identification using neural networks

For linear systems System identification and control are well developed. For non-linear systems the theory is not well defined significantly. Properties such as controllability, observability and stability are well defined for linear system model, but it is not straight forward in the case of non-linear systems.

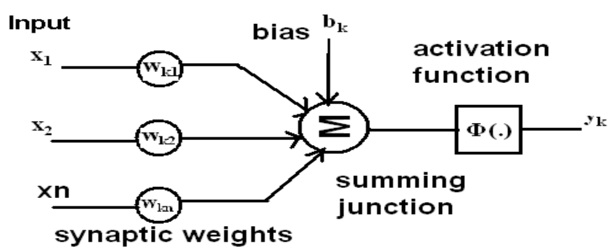


Fig.1.2. Nonlinear model of a neuron

Artificial Neural networks are a powerful tool for many complex applications such as function approximation, optimization, nonlinear system identification and pattern recognition. This is because of its attributes like massive parallelism, adaptability, robustness and the inherent capability to handle nonlinear system. It can extract information from heavy noisy corrupted signals. Fig. 1.2 shows the model of a nonlinear neuron. System identification can be either state space model or input-output model.

1.1.3 The Input-Output modeling

An I/O model can be expressed as $y(t) = g(\phi(t,\theta)) + e(t)$, where,θ is the vector containing adjustable parameters which in the case of neural network are known as weights, g is the function realized by neural network and ϕ is the regression vector. Depends on the choice of regression vector different model structures emerge.

Using the same regressors as for the linear models, a corresponding family of nonlinear models was obtained which are named NARX, NARMAX as in equations 1.1 and 1.2 below. Different model structures in each model family can be obtained by making a different assumption about noise.

$$\text{NARX,}\ \phi(t,\theta) = \left[y(t-1), y(t-2),\ldots..y(t-n), u(t-1),\ldots..u(t-m)\right]^T \qquad (1.1)$$

$$\text{NARMAX}\ \phi(t,\theta) = \left[y(t-1),\ldots..y(t-n), u(t-1),\ldots..u(t-m), e(t-1),\ldots..e(t-k)\right]^T \qquad (1.2)$$

Where y(t) is the output, u(t),the input and e(t) is the error. For the implementation of the above system, Feed forward neural networks can be used [19-21].

1.1.4 State Space modeling

Suppose that the given plant is described by state space model.

$$x(n+1) = f(x(n), u(n)) \quad (1.3)$$

$$y(n) = h(x(n)) \quad (1.4)$$

where *f(.)* and *h(.)* are vector valued nonlinear functions both of which are unknown. *x(n)* and *y(n)* are the models estimate of the plant state and output at time step n. For the implementation of the above state space equations , recurrent neural networks are used .i.e. a single RNN is used to model both process nonlinearity 'f' and measurement function 'g'. Also the model incorporates the past residual in the regression [12, 79-82]. This structure is called Neural network State Space Innovation Function(NNSSIF).State space analysis characterizes dynamics of a system in terms of attractors, geometric description of recurrent trajectories and Lyapunov exponents [130].

1.2. Current status

Many researchers have addressed the problem for dynamic nonlinear black box modeling. Different approaches can be used for solving the problem. Among them Artificial Neural Networks is a powerful tool. The system identification then goes down to estimation of the model parameters. Neural network is best suited where unknown dynamics can be constructively approximated. During the past few years, several authors have suggested neural network implementation for nonlinear dynamical black box modeling [19, 20, 78]. When the mathematical model of the process cannot be derived with an analytical method, the only way for modeling is by deriving the model function using the relationship between input and output of the process. In modeling, a neural network that emulates the behavior of the plant is trained based on the known nonlinear models [9, 11, 14]. Thus dynamical system information is stored in the neural network function. During modeling simulations, the input-output behavior of the neural network is compared to that of the nonlinear plant under study.

Neural network Black Box modeling can be performed using non linear Feed Forward (FF) and Recurrent structures. Recurrent Neural Networks (RNN) is fundamentally different from the feed forward architecture, in the sense that they not only operate in the input space but also in the internal state space. Because of the dynamical structure exhibited by them, these networks have been successfully applied to system characterization problems [19, 80, 82].

The classical approach of training neural network is by using the Back Propagation algorithm. Back propagation was created by generalizing the Windrow-Hoff learning rule to multiplayer networks [61] and has been widely used to train neural networks in many applications. Standard back propagation is a gradient descent algorithm. However the convergence could be slow and appropriate learning parameters need to be chosen; their tuning is not trivial.

Since the development of well-known Kalman filter (KF) [92, 93, 94], the method of linear stochastic state estimation has been widely studied in the literature and applied to many problems in tracking. The Kalman Filter has been extended to the nonlinear systems, which linearises the nonlinear function around the point of interest. The resultant filter is called Extended Kalman Filtering (EKF), which can be implemented in estimating the network parameters in both FF and RNN. The estimation algorithm converges faster than the back propagation algorithms [95, 96]. Also the predictor – corrector approach helps to reduce the computational requirements. Many alternative approaches have been proposed for realizing the Kalman estimation like Decoupled EKF and Unscented Kalman Filter [101]. Computational complexity is quite low when the Decoupled EKF [112] is used.

Expectation Maximization Algorithm (EM) is a *method to calculate the initial states and covariance* avoiding the difficulty in setting proper values for these by trial and error [113]. Maximum Likelihood Estimation (MLE) is a well established procedure for statistical estimation. In this procedure first formulate a log likelihood function and then optimize it with respect to the parameter vector of the probabilistic model under consideration [114-117].

In classical approaches the search for the optimal approximation model is carried out within a parameterized identification family such as Moving average(MA), Auto Regressive(AR) and their combination (ARMA) or ARMAX (X for exogenous) [21, 68] and it is chosen to optimize a given cost function(e.g. Mean square error). Because of its simplicity linear models does not always approximate a nonlinear system throughout its working environment. Therefore to improve approximation accuracy various solutions have been envisaged which generally encompass system linearization around the working environment. Obviously, difficulties increases when the system is completely unknown, is considered to be the black box models.

In fact, the nonlinear parametric family obtainable with neural structures extends the linear ones by nonlinear models, among them are NAR, NARX, NARMAX subfamilies. Neural networks of the multi layer feed forward and recurrent types are employed for system identification. There are different structures and several algorithms for training neural networks for achieving global minima and the selection of these depends upon the problem one have to analyze. There is a wide gap between applications of these methods in real time and simulation. Issues such as stability, processor speed, learning time, type of algorithm etc arise when it comes to real time implementations. Adaptive designs of neural network are capable of optimization over time under conditions of noises and uncertainty.

A large number of literatures and published papers are available for the different techniques of system identification discussed so far. But a cumulative study of all the techniques together and comparative analysis is yet to come. Here in this Thesis, few important techniques are implemented and compared for system identification especially for stochastic modeling of nonlinear systems.

Recently several new approaches to recursive nonlinear filtering have appeared in literature. Particle filters (PF) are suboptimal filters belonging to this category of methods. They perform Sequential Monte Carlo (SMC) estimation based on point mass (or "particle") representation of probability densities [131-137]. The SMC ideas in the form of sequential importance sampling had been introduced in statistic back in the 1950s. Although these ideas continued to be explored sporadically during the 1960s and 1970s, they were largely overlooked and ignored. Most likely the reason for this was the modest computational power available at that time. In addition, all these early implementations were based on plain sequential importance sampling, which as we shall describe later, degenerates over time. The major contribution to the development of the SMC method was the inclusion of the re-sampling step, which, coupled with the faster computers, made the particle filters useful in practice for the first time. Since then research activity in the field has dramatically increased, resulting in many improvements of particle filters and their numerous applications especially for nonlinear system modeling [77].

1.3 Motivation

The problem of system modeling and identification has attracted considerable attention during the past few years mostly because of a large number of applications in diverse fields like chemical processes, biomedical systems, transportation, ecology, electric power systems, hydrology, aeronautics and astronautics. An accurate on-line estimate of critical system states and parameters are needed in a variety of engineering applications like in automatic control, signal processing, echo cancellation, SONAR, fault detection, tracking etc. They are used in many commercial products such as modems, image processing, speech recognition, front end signal processors and biomedical instrumentation [62-65].

The amazing challenges in statistical estimation along with an opportunity to learn different techniques in solving the well known problem motivated to take up the study of system identification technique. The rich literature available on the subject offered an opportunity to dig out solutions in situations that are difficult. Since a comprehensive study of the well known techniques and the comparison of their performance is necessary to choose an efficient technique for particular applications. It is attempted to develop some new approaches and their evaluations based on various criterions for blind identification of nonlinear systems. It is expected that, such comprehensive study and the comparison process can be of great relevance in many fields including control, chemical, electrical, biological, financial and weather data analysis. More specifically the aim of the thesis is to:

- Implement various identification/ modeling techniques for nonlinear systems.
- Develop and suggest certain new approaches for the blind identification of nonlinear system and improve some of the currently available techniques.
- Provide a comprehensive evaluation report of these methods based on a number of evaluation criterion/performance measures.

1.4 Objectives and the methodologies

The system identification process using neural network can be represented by the block diagram shown in Fig 1.3. The objective is to implement the following algorithms for nonlinear system identification and compare the performance of the models in order to evaluate the relative merits and demerits of the algorithms.

- Back Propagation (gradient – descent)
- Radial Basis Function networks (gradient – descent)

- Extended Kalman Filter
- Extended Kalman Filter with Expectation Maximization.
- Decoupled Extended Kalman Filter
- Maximum Likelihood Estimation
  Gauss Newton
  Conjugate Gradient

- Identification with particle filter approach
- State space modeling

Given below in Fig. 1.3 is an illustration of system identification.

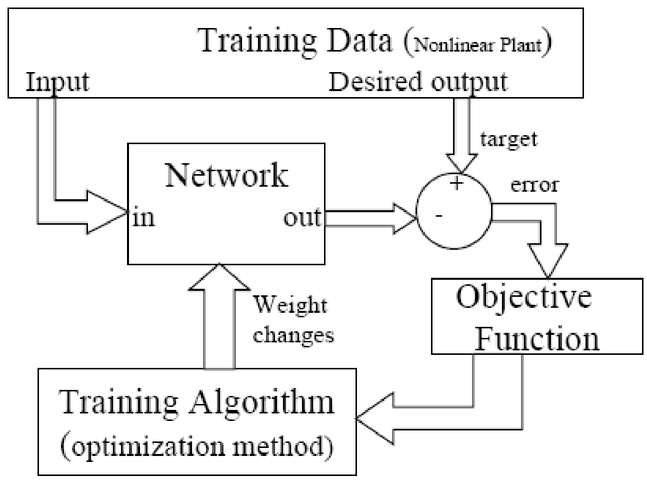


Fig.1.3 Block diagram of system identification using neural network

The state space modeling is done to extract the dynamics of the system which is very helpful in the error detection and control of the plant or process. The model behavior and performance are evaluated in terms of Mean Square Error and also in terms of two well known methods (i) Lyapunov exponents (for stability check) and (ii) Cramer Rao Lower Bound (CRLB) (for efficiency check). The statistical parameter estimation insists that the estimate should be well within the CRLB [121-124].

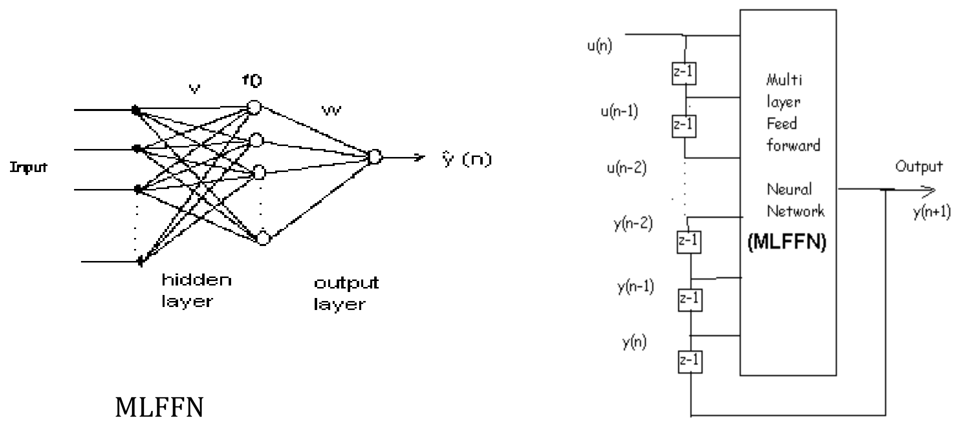


Fig 1.4 NARX modeling for system identification

NARX model is well suited for Input-Output modeling of stochastic nonlinear systems [39]. So in this work, NARX model is chosen as the system model in which the model structure is a Multi Layer Feed Forward Neural Network (MLFFN) as shown in Fig. 1.4 for all the nonlinear systems (using different algorithms).

Many nonlinear systems are modeled using each of the algorithms. Four entirely different systems are selected in order to check the consistency in performance of the algorithms. If the model performs equally well for all the four systems it is assumed to perform well for any other nonlinear systems. The selected nonlinear systems are.

$$y = \sin(t^2+t) \tag{1.5}$$

Real world systems: Ambient noise in the sea
Acoustic source 'A'
Acoustic source 'B'

1.5. Organization of the thesis

An introductory review of the available literature is given in chapter2. Chapter 3 introduces the Neural Network approach using Back Propagation algorithm to estimate the parameters. Due to the local minima problem of BPA, an alternate approach based on Kalman Estimation is explored in chapter 4. Though Kalman Estimation is found good for estimation, the optimality depends on the apriori statistics of states and covariance. To eliminate this problem, the method based on Expectation Maximization is used which is also discussed in chapter 4. The stochastic method based on Maximum Likelihood Estimation is often described as a very standard approach in parameter estimation. Chapter 5 discusses about MLE. In chapter 6 a novel approach for the identification problem with nonlinear filtering method, namely particle filter, has been presented. In order to make the study of system identification problem comprehensive, the state space modeling approach has also been taken up to assess the dynamic behavior of the systems as discussed in chapter 7. The efficacy of the model is demonstrated by plotting the phase plane plots for the systems identified. The Lyapunov exponents are calculated for the models in order to evaluate the convergence nature of the systems which is also included in chapter 7. Since the recommended procedure in the statistical parameter estimation insists that the estimate should be well within the CRLB, it is evaluated in chapter 8 for all the systems modeled in previous chapters. Chapter 9 includes the comparison of performance of different approaches along with their relative merits of implementation and it also summarizes the thesis with discussions, conclusion and the scope for future work.

Chapter 2

BACKGROUND LITERATURE REVIEW

Chapter 2 provides a detailed review of literature on the topic of interest. It explores the state of the art situation in the field of research as well as the topics which provided motivation for the developments of outcomes of the thesis.

2.1 Introduction

This chapter sets the scene for the upcoming sections of the thesis. It basically is an assessment of the present state-of-the-art of the wide area of nonlinear system modeling/identification, blind system identification and system analysis and design techniques.

N.K Sinha et al. [1] provides a basic concept of modeling and identification of dynamic systems from the records of input output data. This provides a detailed theory of the process of identification to start with.

Daniel Grapue [2] also provides the basic theories on modeling of systems. They start with linear system analysis using conventional methods like transfer function, linear differential equation with constant coefficient that relates the input and output of the system, impulse response and a set of state equations etc.

By knowing the input of the system, we can determine the response of the system. But in many cases we may not be having the system description .The system transfer function, impulse response, differential equation; state equation etc has to be derived in this case [13-14]. However, nonlinear systems require a different approach, mentioned in the section to follow.

2.2 Nonlinear system identification using neural networks

Identification and Control of Non-linear dynamical systems are challenging problems to the control engineers. The problem of system identification and modeling consists of computing a suitably parameterized model, representing a process [1, 2, 3]. The parameters of the model are adjusted to optimize a performance function, based on error between the given process output and identified process/model output.

Artificial Neural networks are a powerful tool for many complex applications such as function approximation, optimization, nonlinear system identification and pattern recognition. This is because of its attributes like massive parallelism, adaptability, robustness and the inherent capability to handle nonlinear system. It can extract information from heavy noisy corrupted signals. System identification can be either state space model or input-output model.

S. Chen, S.A Billings [4] provides an outlook into the capabilities of neural networks for modeling non linear systems. The paper presents an approach to system identification of input/output mappings of non-linear stochastic systems in accordance to an information-theoretic criterion. At that, a parameterized description of the system under study is utilized combined with a corresponding technique of estimation of the mutual information (in the Shannon sense), leading, finally, to a problem of the finite dimensional optimization. Solving the latter is based on applying ideas of papers on using neural networks within problems of optimization of continuous functions.

Kumpati .S. Narendra and Kannan Parthasarathy [6-8, 11-13] have done a major contribution to the identification problem. They provide the use of feed forward type of neural networks in modeling of nonlinear systems and an extensive study of the learning approaches. On-line identification algorithm via dynamical neural networks with different time-scales followed by controller design is proposed for the dynamic systems with nonlinearity and uncertainty in these papers. The main contribution of the papers is that the analysis of the modeling error and disturbance. Simulations are given to demonstrate the effectiveness of the theoretical results. They also provide a general performance assessment in the mean square error sense (MSE).

Hava T Siegelman [19], Wen-Xiao Zhao, Han-Fu Chen, Wei Xing Zheng [20], Jinglu Hu, Kousuke Kumamaru and Katsuhiro Inoue [21] provides the modeling with recursive approach to identification for systems like ARMAX, nonlinear ARX, and others. They propose various learning strategies too for these methods. An I/O model can be expressed as $y(t) = g(\phi(t,\theta)) + e(t)$, where,θ is the vector containing adjustable parameters which in the case of neural network are known as weights, g is the function realized by neural network and ϕ is the regression vector. Depending on the choice of regression vector, different model structures emerge. Using the same regressors as for the linear models, a corresponding family of nonlinear models was obtained which are named NARX, NARMAX, etc. Different model structures in each model family can be obtained by making a different assumption about noise. Their study help us to model the next output sample as a function of current and past input and output samples in a recursive fashion. Fa-Long Luo and Rolf Unbehauen [15] give an extensive theory of these approaches.

K. Hornik, M. Stinchcombe [25], Jiancheng Liu Xuping [65] etc in their works proposes the application of multi layer networks in identification. A Multi Layer Perceptron (MLP) network can approximate an arbitrary nonlinear map and is completely determined by the network parameters such as the connection weights and thresholds. This suggest that the MLP networks can be used to construct the nonlinear maps related to the system identification operator say *F[.]* where *F[x(n)]* is the network output and t h e aim is to minimize the norm of the error vector *F[x(n)]-Y(n)* with *Y(n)* as the actual system output. They provides the details of modeling single input single output (SISO) as well as multiple input multiple output (MIMO) systems.

S. Chen, S.A Billings, C.F.N Cowan and P.M Grant [65], C. Wiegand, C. Hedayat, W. John, Lj. Radi´c-Weissenfeld and U. Hilleringmann [70], Thomas F**.** Junge and Heinz Unbehauen [71] et al. introduces the use of radial basis function (RBF) networks in modeling applications. They propose a recursive identification technique for nonlinear discrete-time multivariable dynamical systems. Extending an early result to multivariable systems the technique approaches a nonlinear system identification problem in two stages: One is to build up recursively a RBF neural net model structure including the size of the neural net and the parameters in the RBF neurons; the other is to design a stable recursive weight updating algorithm to obtain the weights of the net in an efficient way.

Simon Haykin [90] and Fa-Long Luo and Rolf Unbehauen [15] in their books give the complete theory of RBF networks and various strategies for its training.

Nenad Todorovic, Petr Klan et al. [77] provides a general evaluation and state-of-the-art technique about dynamical nonlinear system modeling using neural networks.

S.Amari, A.Cichocki, and H.Yang [79], Chao-Chee Ku and Kwang Y. Lee [80], Han-Fu Chen [81] provides the use of recurrent neural networks (RNN) in nonlinear control and modeling. The basic concepts of conbined state and parameter estimation also appear in these references.

2.3 Nonlinear system identification –the Kalman approaches

The Kalman Filter is one of the most widely used methods for estimation and tracking due to its simplicity, optimality, tractability and robustness [91-94]. The Kalman filter gives a linear, unbiased and minimum error variance recursive algorithm to optimally estimate the unknown state of a dynamic system from noisy data taken at discrete real-time. To apply the discrete Kalman filter, the system under study should be represented by a set of linear, finite dimensional state space equation.

If the model turns out to be non-linear, a linearization procedure is usually performed in deriving the filter equations. i.e. the system is linearised about a trajectory that is continuously updated with the state estimates resulting from the measurements. The new filter obtained is called Extended Kalman Filter (EKF).

Yaakov Bar-Shalaom and Xiao-Rong Li et al. [91] introduces the principles of estimation and tracking. The basic concept of nonlinear filters leading to approaches like Kalman filtering is mentioned in this. Simon Haykin [97] provides the comprehensive theory in this.

M.S. Grewal and A.P Andrews [93], A.V. Balakrishnan [94] et al. gives the theory of Kalman filtering which could be later used in RNN training algorithms with Kalman approach.

R.N.Lobbia, S.C.Stubberud, and M.W.Owen, [95], Y. Linguni, H. Sakai, H. Tokumaru [96] gives the extended Kalman filter theory (EKF). They provide the merits and improvements from the simple Kalman algorithm when dealing with nonlinear system analysis problems.

Y. Linguni, H. Sakai, H. Tokumaru [96], Simon Haykin [97] et al. provides a detailed discussion on various real time learning algorithms for multilayered neural network based on the Extended Kalman Filter. These algorithms and their modified versions are used for the nonlinear system identification problem in this thesis. [100-107] also provide a list of application of EKF and some of its variations in control applications.

Ben James, Brian D.O, Anderson, and Robert. C. Williamson [99], Radhakrishnan.K, Unnikrishnan.A, and Balakrishnan K.G [100], A.P.Dempster,N.M.Laird, and D.B.Rubin [116] et al. provide the concepts of EKF with Expectation Maximization approaches for performance improvement.

Joost H. de Vlieger and Robert H.J. Gmelig Meyling [115], K.Abed Meraimand, E.Moulines [117], M.GhoshandC.L.Weber [118], Yonina C. Eldar [123] et al. in various papers introduce and demonstrate the theory for EKF with Maximum Likelihood Estimation, as an improvement to the EKF. Later in the thesis, MLE estimation using Gauss-Newton and Conjugate Gradient methods are developed.

2.4 State space modeling using recurrent neural networks

The state of a dynamical system is formally defined as a set of quantities that summarizes all the information about the past behavior of the system that is needed to uniquely describe its future behavior, except for the purely external effects arising from the applied input. In many control problem the objective is to feedback the states of the system in order to modify its behavior. Hence it is necessary to estimate the states of the system from the measurements which are contaminated with noise. The problem of combined parameter and state estimation is a nonlinear estimation problem by augmenting the state vector with the parameter vector.

Gordon, N.J., D.J. Salmond and A.F.M. Smith [126], Christophe Andrieu Arnaud Doucet Dpts Vladislav B. Tadi´c [127], Radu Dogaru, A.T. Murgan, S. Ortmann, M. Glesne [128] provide some basic concepts of state space analysis, which could be extended for the current problem. [133-139] provide the possibilities of state space estimation using various nonlinear filtering methods including particle filter.

Dmitry Malyuk 'I, Georgy Boyarintsev [131]. give the concept of Lyapunov exponents. The same is utilized here to study the system dynamics and were able to provide an assessment of the system behavior on whether it is chaotic or not.

2.5 Evaluation of the model performance in the MSE and CRLB senses

Once the various models have been developed, it is possible to have a test on its goodness/efficiency in some senses. The Mean Square Error (MSE) and the Cramer Rao Lower Bound (CRLB) for the various estimators can very well be used for this comparison. According to it, the mean square error corresponding to the estimator of a parameter cannot be smaller than a certain quantity related to the likelihood function. If an estimator's variance is equal to the CRLB, then such estimator is called efficient. [5].

Shuhi Li [101] introduces a very basic mode of comparison of back propagation and Extended Kalman filter in Pattern and Batch forms for training Neural Networks. From these basic approaches, mathematically efficient methods can further be developed.

Er-Wei Bai [119] gives the MSE concept, which is extended for the modeling problem in this thesis.

Zhiping Lin, Qiyue Zou, E. Sally Ward, and Raimund J. Ober [112], Yonina C. Eldar[123], R. Niu, P. Willet and Y. Bar Shalom [124], J.H. Taylor [125] et al. provide the computational algorithms for the CRLB and suggest its suitability as a figure of merit for performance evaluation.

2.6 Nonlinear system modeling using Particle Filter

Recently several new approaches to recursive nonlinear filtering have appeared in literature. Particle filters (PF) are suboptimal filters belonging to this category of methods. They perform sequential Monte Carlo (SMC) estimation based on point mass (or "particle") representation of probability densities [143-144]. The major contribution to the development of the SMC method was the inclusion of the re-sampling step, which, coupled with ever faster computers, made the particle filters useful in practice for the first time. Since then research activity in the field has dramatically increased, resulting in many improvements of particle filters and their numerous applications [136-140].

R. van der Merwe, J. F. G. de Freitas, A. Doucet, and E. A. Wan [132], et al.

introduced the basic technical paper on unscented particle filter.

Doucet, A., N. Gordon and V. Krishnamurthy [133], describe particle filters for state estimation which could be extended to the current problem as well.

Jansson, R. Karlsson and Nordlund P-J [134], M.S Arulampalm, B.Ristic,N.Gordon and T.Mansell [135], TIAN-Zengshan, LUO Lei [139] Yaakov Bar-Shalom and Xiao_Rong Li [145] discusses the application of particle filters in tracking problems. They also provide the basic equations for the filter implementation.

V. Kadirkamanathant, M. H. Jawaxdt, S. G. Fabri and M. Kadirkamanathan [136], Marcos del Toro Peral, Fernando Gomez Bravo Alberto Martinho Vale [137], TIAN-Zengshan, LUO Lei [138], Jayesh H. Kotecha and Petar M. Djuri C [139], Z. Zhu and H. Leung [140] discuss the suitability of particle filters in state space analysis, neural network training, dynamical model selection.

Bergman, N [142], Doucet, A., de Freitas, N. and Gordon, N [143], Nordlund, P.J [144], Yaakov Bar-Shalom and Xiao_Rong Li [145] provide the computational theory including Sequential Monte Carlo Methods which is further developed and refined for the modeling/identification problem in the thesis.

Chuan Li, Yun Bai, Xianming Zhang, Hongjun Xia and Jing Chen [146], Katsumi Konishi, Hiroaki Kato [147], Cao Wen-Mmgl Lu Feil Faig Hao' [148], Gustavo Camps-Valls, Manel Martínez-Ramón, José Luis Rojo-Álvarez, Member, IEEE, and Jordi Muñoz-Marí [150], Byung-hwa Lee, Sang-un Kim, Jin-wook Seok and Sangchul Won [152], Ali, M. Ashfaq and Chr. Schmid [153], Xiaodong Wang, Weifeng Liang, Xiushan Cai, Ganyun Lv, Changjiang Zhang and Haoran Zhang et al.[154] suggest some new approaches in system identification using vector machines, RLLM networks, DBF networks, various nonlinear programming approaches etc.

With this background and recent developments, the thesis has implemented various identification/ modeling techniques for nonlinear systems. It also developed and suggested certain new approaches for the blind identification of nonlinear system and improvements in some of the currently available techniques and provided a comprehensive evaluation report of these methods based on a number of evaluation criterion/performance measures.

It is expected that the outcome of this will enable the development as well as evaluation of models/ parameter identification methods for various classes of nonlinear systems from control engineering to financial data analysis and forecasting.

Chapter 3

NONLINEAR SYSTEM MODELING USING NEURAL NETWORKS

Chapter 3 discusses the use of neural network models like SLP, MLP and RBF in blind identification of nonlinear systems with SISO and MIMO cases. Results of the modeling using the well known Back Propagation algorithm are presented. The chapter also introduces the RNN for modeling the combined state and parameter estimation.

3.1 Introduction

The function approximation capability of artificial neural networks can be very effective in designing efficient system identification models and controllers for non-linear systems [4, 6, 8, 9]. The recent emergence of the neural network paradigm as a powerful tool for learning complex input-output mappings has stimulated many studies in using models based on neural network for identification of dynamical systems with unknown non linearity. For neural based identification, there are two main issues that stand out: one is the choice of the model architecture to be adopted for system identification, and the other is the choice of the learning algorithm.

It is well known that a wide class of discrete-time non-linear systems can be represented by the non-linear auto regressive moving average with exogenous inputs (NARMAX) model [21, 68]. The NARMAX model provides a description of the system in terms of a non-linear functional expansion of lagged inputs, outputs, and prediction errors. As given in section 1.1.3, the parameterized form gives,

$$y(t+1) = g\left(y(t-1),\ldots..y(t-n), x(t-1),\ldots..x(t-m), e(t-1),\ldots..e(t-k), \phi\right)^T \quad (3.1)$$

where, *y(t), x(t)* and *e(t)* are the output, input and the exogenous noise components. ϕ is the parameter set viz. the weights.

The mathematical function describing the exact physical behavior of a real-world system can be very complex and its exact form is usually unknown. So in practice, the modeling of a real world system must be based on a chosen model of known functions. A desirable property for this model set is the capability of approximating a system to a prescribed accuracy. Mathematically, it also requires that the set be dense in the space of continuous functions. Polynomial functions offer one obvious choice having the above property. On the other hand, since the derivation of the NARMAX model was independent of the form of the non-linear function, other choices of functional approximation also stand as eligible candidates to be investigated within this framework. Neural networks thus form an obvious alternative, since they can be viewed just as another class of functional representations [19-21]. When used for blind recognition, the model avoids the input sequence x(k) given in equation 3.1.

$$y(t+1) = g\big(y(t-1),\ldots.y(t-n),(t-1),\ldots.e(t-k),\phi\big)^{T} \qquad (3.2)$$

where, *y(t),)* and *e(t)* are the output and the exogenous noise components. ϕ is the parameter set viz. the weights. Here again *g(.)*.

3.2 Nonlinear data sets (Systems) used for analysis

To study the performance of the learning algorithms in evolving the most stable parameterized model relies on typical data sets generated from systems which are representative of the real world situations. Many nonlinear systems are modeled using each of the algorithms. Four entirely different systems are selected in order to check the consistency in performance of the algorithms. If the model performs equally well for all the four systems it should perform well for any of the real world nonlinear systems. The selected nonlinear systems are,

$$y = \sin(t^2 + t)$$

Real world systems: Ambient noise in the sea

Acoustic source 'A'

Acoustic source 'B'

Which are described and plotted in Fig 3.1.

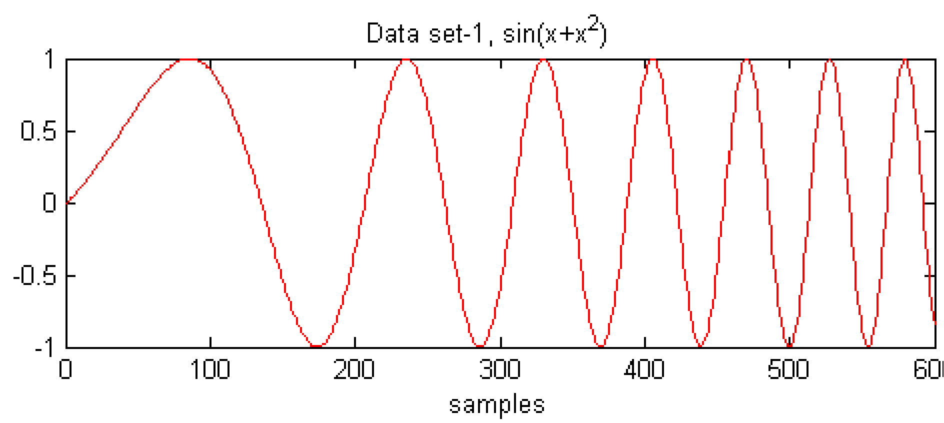
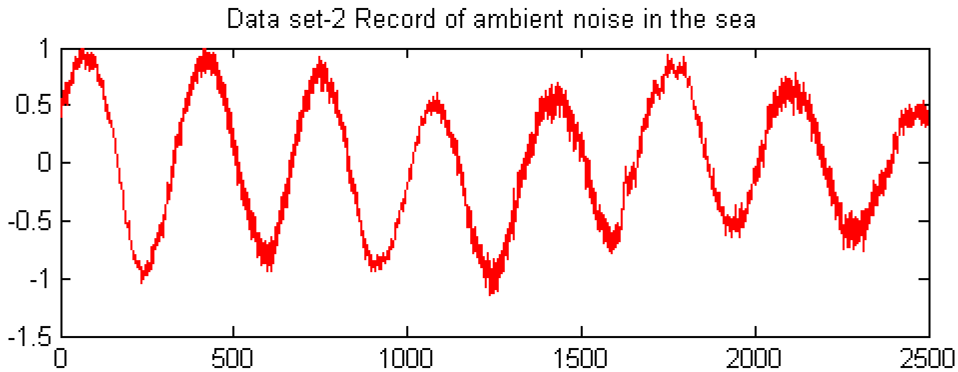
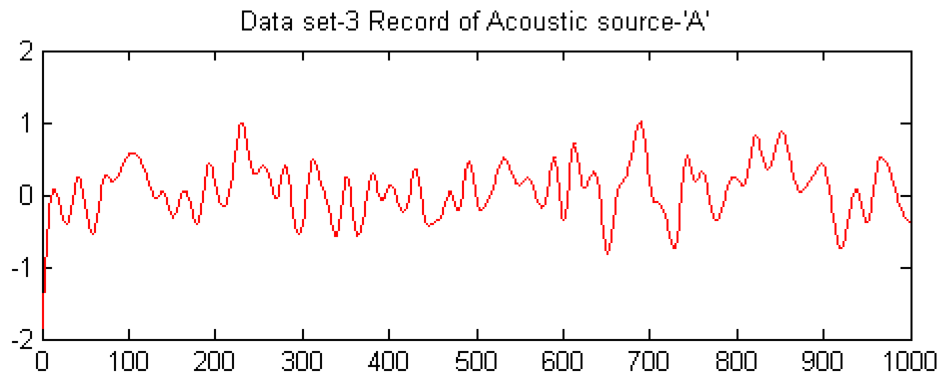
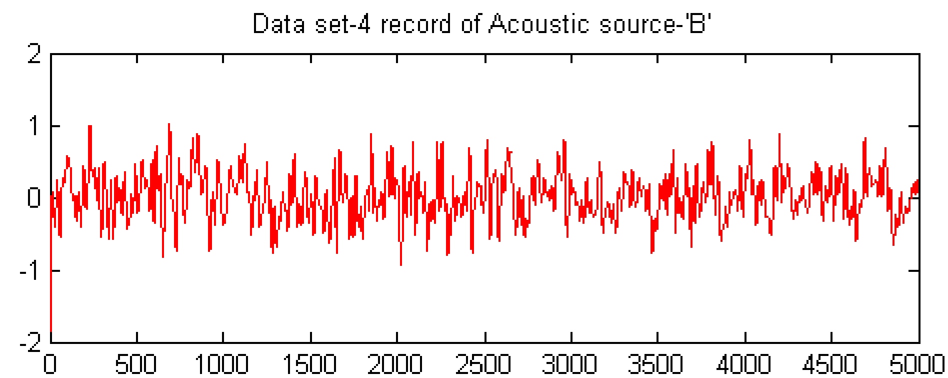


Fig 3.1 Plots of the four nonlinear data sets used for modeling


Data set-2 Record of ambient noise in the sea
1
0.5
0
-0.5
-1
-1.5
0
500
1000
1500
2000
2500

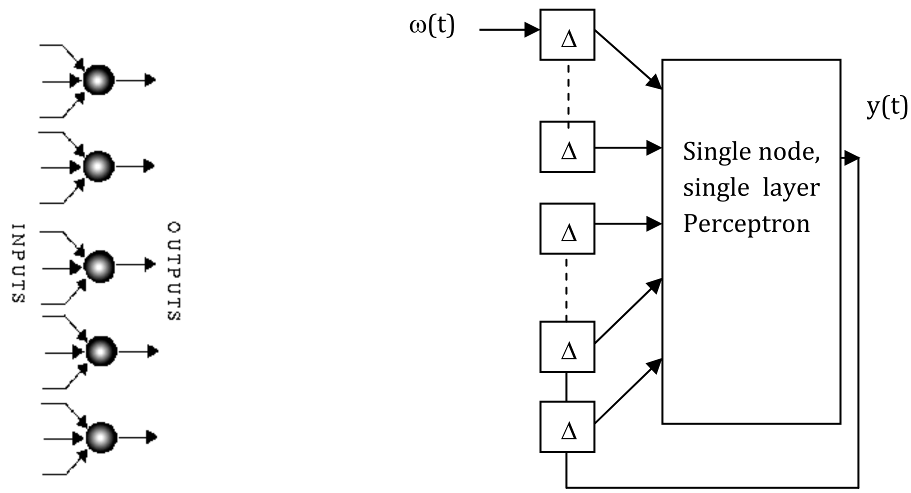

Data set-3 Record of Acoustic source-'A'
2
1
0
-1
-2
0
100
200
300
400
500
600
700
800
900
1000


Data set-4 record of Acoustic source-'B'
2
1
0
-1
-2
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000

Data set -1 is generated from a well known mathematical equation viz. *sin* $(t+t^2)$, with no noise components present in it. Data set -2 consists of data from the ambient noise recorded from sea. The ambient noise represents a typical nonlinear process to be used in the demonstration of the algorithm. The data sets 3 and 4 are from two acoustic sources, which also fall into the category of non linear systems.

3.3 System Identification Using SLP Networks

As the first step in understanding the function approximation capability of the neural network systems, a Single Layer Perceptron network with a prescribed nonlinearity, is trained with the lagged output data of the system to be modeled. In the SLP structure, illustrated in Fig 1.4 and reproduced as Fig 3.2 for convenience, there is only one layer of neurons with the activation function and a single output neuron Fig. 3.2 b which provides the output sample at an instant, as a function of a number of past output and noise fed to the network [16, 17, 22, 24]. The activation functions used for the analysis are log sigmoid and tan sigmoid with delta rule as the learning method. In the case of nonlinear blind identification the lagged inputs in Eqn. 3.1 are replaced with lagged noise vector. I.e. lagged outputs and noise components alone will be presented to the network input.



(a) Structure of SLP (b) SLP for modeling the time series of Equation 3.2

Fig. 3.2 Single layer perceptron in modeling time series

3.3.1 Delta Rule for weight update in SLP

The performance of the network with various numbers of neurons in the layer were analyzed and compared. The model performance with 5, 10, 15 and 20 neurons were observed and the optimal structure is taken as the one which has given the least mean square error (MSE). It is observed that the SLP with 15 neurons and tan sigmoid activation function has given the least MSE and further increase in the number of neurons does not give any obvious improvements in the MSE. The delta learning rule is used here for updating the network weights, which is described below.

The delta rule is derived to minimize the error in the output of the perceptron through gradient descent. The error for a perceptron with j outputs can be measured as,

$$E = \sum_j \frac{1}{2}(t_j - y_j)^2 \qquad (3.3)$$

where t_j is the desired or target value and y_j is the value generated by the model described in Eq. 3.2. In this case, it is desired to move through "weight space" of the neuron (the space of all possible values of all of the neuron's weights) in proportion to the gradient of the error function with respect to each weight. In order to do that, the partial derivative of the error is calculated with respect to each weight. For the i^{th} weight, this derivative is only concerned with the j^{th} neuron, and one can substitute the error formula above while omitting the summation[90]:

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial\left(\frac{1}{2}(t_j - y_j)^2\right)}{\partial w_{ji}} \qquad (3.4)$$

The final equation for the gradient is arrived as:

$$\frac{\partial E}{\partial w_{ji}} = -\left(t_j - y_j\right) g^{'}(h_j) x_i \qquad (3.5)$$

As noted above, gradient descent tells that the change for each weight should be proportional to the gradient. Choosing proportionality constant α and eliminating the minus sign enables to move the weight in the negative direction of the gradient to minimize error and gives the target equation for the j^{th} neuron: $\Delta w_{ji} = \alpha(t_j - y_j) g_j^{'}(h_j) x_i$ (3.6)

3.3.2 Single Input Single Output (SISO) System Modeling Using SLP Network

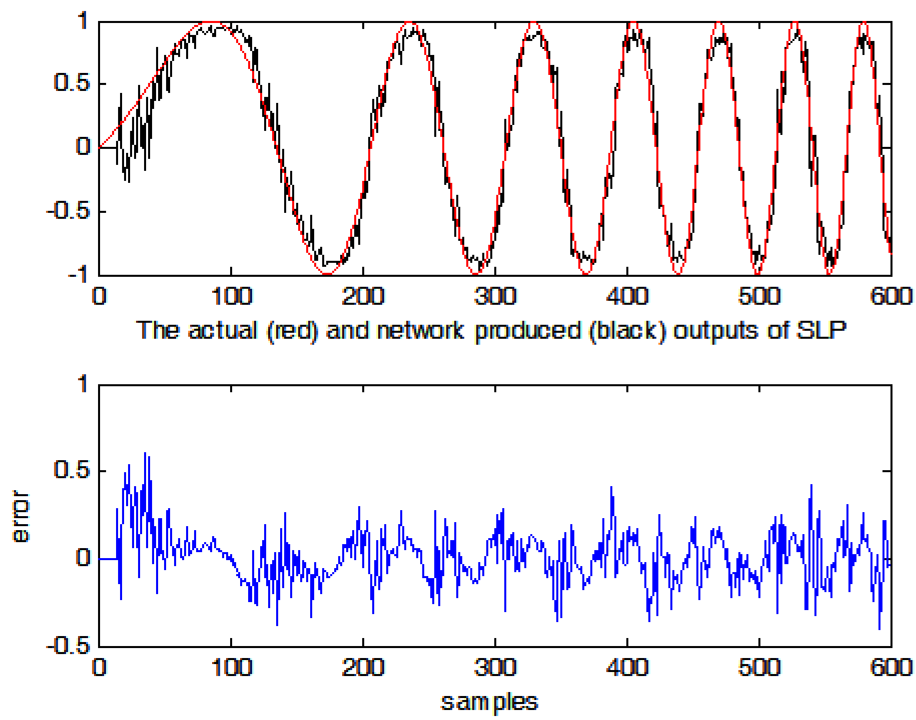


Fig.3.3 Actual and network output and the error vector (below) of the SLP network

A neural network with 15 neurons and tan sigmoid as the activation function was chosen to take care of the system behavior. The results obtained with nonlinear system, data set-1, $y=sin(t+t^2)$ are summarized in Fig.3.3 and 3.4.

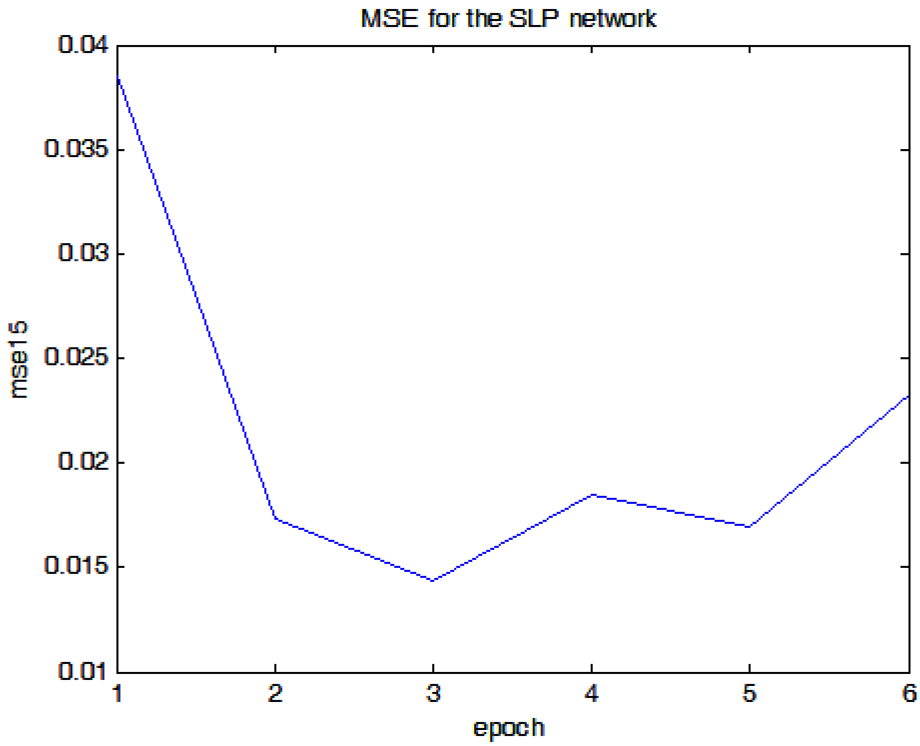


Fig.3.4 Norm of the error vector over the epochs in SLP network

The overall performances of the SLP network cannot be considered superior, but is capable of system mapping to certain extend with satisfactory MSE performance. It can be noticed that the overall MSE could come down to the order of 0.0236. As SLP shows a divergent behavior as shown in Fig 3.4, better structures with more approximation capabilities must be adopted for consistent model performances. Such options are discussed in the following sections.

3.4 System Identification Using MLP Network

A Multi Layer Perceptron (MLP) network can approximate an arbitrary nonlinear map and is completely determined by the network parameters such as the connection weights and thresholds. This suggest that the MLP networks can be used to construct the nonlinear maps related to the system identification operator say *F(.)* where *F(u(n))* is the network output and our aim is to minimize the norm of the error vector *F(u(n))-Y(n)* with *Y(n)* as the actual system output. Without loss of generality, the system represents the following model: [8-10, 25, 28].

$$y(t+1) = g\left(y(t-1),\ldots..y(t-n), x(t-1),\ldots..x(t-m), e(t-1),\ldots..e(t-k), \phi\right)^T \quad (3.7)$$

This is an NARMAX model. The MLP network constructing the system mapping is presented in Fig. 3.5. A multiple input multiple output (MIMO) system has also been trained using the same structure. In blind identification the lagged inputs are avoided, keeping noise and output vectors retained.

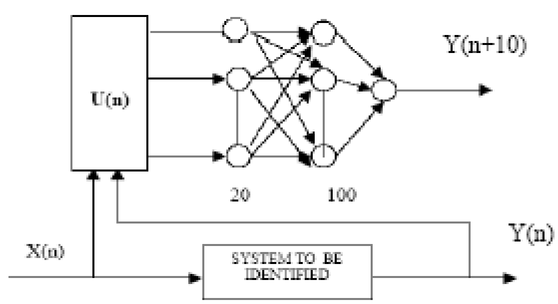


Fig. 3.5 MLP Neural network for nonlinear System Identification

Multilayer perceptrons have been applied successfully to solve some difficult and even linearly no-separable problems, by training them in a supervised manner with a highly popular algorithm known as the *error back-propagation algorithm* [90]. This algorithm is based on the *error – correction learning rule* based on the gradient descent. As such it may be viewed as a generalization of an equally popular adaptive filtering algorithm: the Least Mean Square (LMS) algorithm. The development of the back – propagation algorithm was a landmark in neural networks, in that it provides a computationally efficient method for the training of multilayer perceptrons.

3.4.1 Back Propagation Algorithm

The back propagation is a gradient descent method for training the weights **w** in a multilayer artificial neural network.. The neural network then builds the functional map,

$$y = f(x, w) \tag{3.8}$$

The back propagation process consists of two passes through the different layers of the network, a forward pass and a backward pass. In the forward pass, an input vector (lagged outputs and noise vectors) is applied to the input node of the network and its effect propagates through the network, layer by layer. Finally a set of output ***y(n)*** is produced as the actual response of the network. During the forward pass the synaptic weights **w** of the network are all fixed. During the backward pass, the synaptic weights are all adjusted in accordance with error correction rule [90]. Specifically, actual response of the network is subtracted from the desired response to produce an error signal. This error signal is then propagated backward through the network, against the direction of synaptic connections – hence the name "error back propagation". According to Back propagation Algorithm, outer layer weights are adjusted first and the hidden layer weights next; considering the updated outer layer weights.

Actual response of the network is subtracted from a desired output to produce an error signal defined as,

$$e(n) = y(n) - \hat{y}(n) \tag{3.9}$$

The instantaneous value of the error energy is defined as,

$$E(n) = \frac{1}{2}(e(n))^2 \tag{3.10}$$

The average squared error energy is obtained by summing *E(n)* over all n and then normalizing with respect to the set size N, as shown by

$$Eav = \frac{1}{N}\sum_{n=1}^{N} E(n) \tag{3.11}$$

E_{av} represents the cost function as a measure of learning performance. The objective of the learning process is to adjust the free parameters **w** of the network to minimize E_{av}. It is a measure of how to choose the parameter (synaptic weight) vector **w** of an adaptive filtering algorithm so that it behaves in an optimum manner. It is required to find an optimal solution **w*** that satisfies the condition,

$$E(w^*) \leq E(w) \qquad (3.12)$$

The problem can be solved as an unconstrained optimization problem as: "Minimize the cost function *E(w)* with respect to the weight vector *w***".**

The necessary condition for optimality is

$$\nabla E(w^*) = 0 \qquad (3.13)$$

where ∇ is the gradient operator.

Starting with an initial guess denoted by **w(0)**, generate a sequence of weight vectors **w(1), w(2),...** such that the cost function *E(w)* is reduced at each iteration of the algorithm, as shown by,

$$E(w(n+1)) < E(w(n)) \qquad (3.14)$$

where **w(n)** is the old value of weight vector and **w(n+1)** is its updated value. The algorithm will eventually converge to the optimal solution **w***. Following the method of steepest descent, the successive adjustments applied to the weight vector **w** are in the direction of steepest descent, that is in a direction opposite to the gradient vector ∇ *E(w)*.

$$\begin{aligned} w(n+1) &= w(n) + \Delta w(n) \\ \Delta w(n) &= -\eta \nabla E(w) \end{aligned} \qquad (3.15)$$

To update the hidden layer weight vector **v**;

$$\begin{aligned} v(n+1) &= v(n) + \Delta v(n) \\ \Delta v(n) &= -\eta \nabla E(v) \end{aligned} \qquad (3.16)$$

where η is the learning coefficient, a small positive number.

3.4.2 SISO System Modeling Using MLP Network

The weights in the neural network were adjusted at every instants of time using static back propagation learning algorithm [7, 12, 16, 61]. The gradient method employed a step size of 0.08.The chosen nonlinear systems were modeled using the above described structure. The outputs of the plant, model and the norm of the error vector over epochs are presented in Fig 3.6. The technique shows somewhat good approximation capabilities, even though the process is somewhat time consuming. Here the samples of the output vector is derived, each of which is function of the past samples of the input and output. This means an auto regressive moving average with exogenous input (NARMAX) model. The activation function used is the bipolar sigmoidal function.

$$y(t+1) = g\left(y(t-1),\ldots.y(t-10), u(t-1),\ldots.u(t-10), e(t-1),\ldots.e(t-10), \phi\right)^T \quad (3.17)$$

The results obtained for the data set-2, ambient noise in the sea is presented below to demonstrate the SISO case. The norm of the error vector is plotted and is observed to be decreasing in magnitude so that the desired output and the network output becomes almost the same.

The details of the analysis are presented in Fig. 3.6 and 3.7.

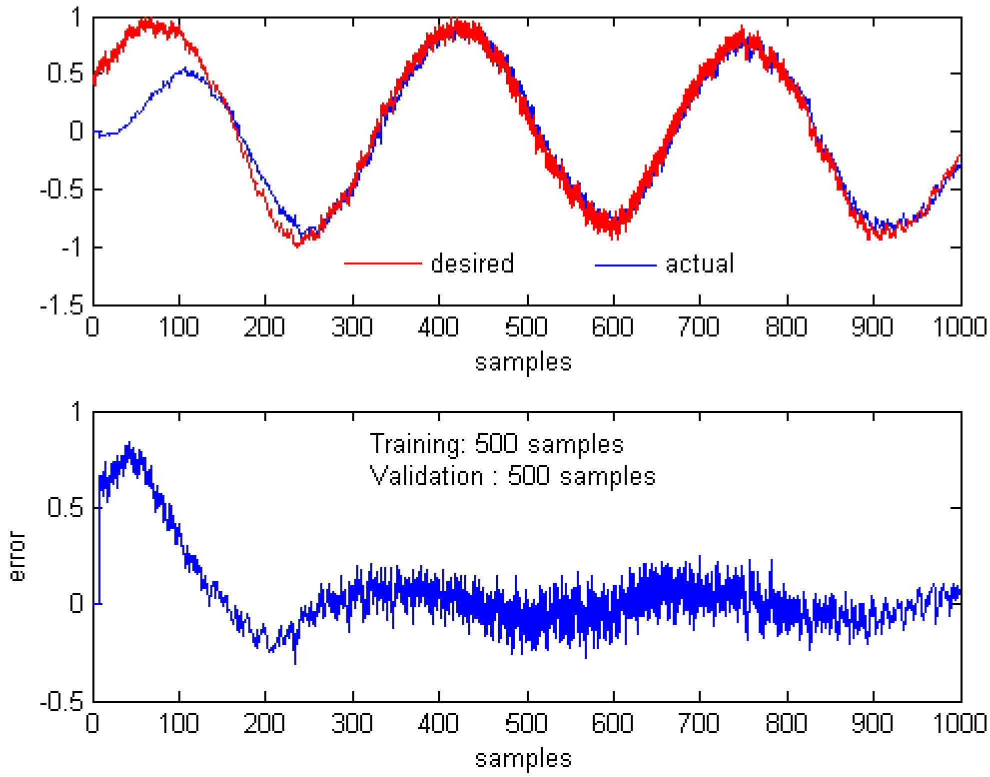


Fig. 3.6 Network and desired output (SISO-upper), the error over the samples (lower)

Fig 3.6 shows the capability of SLP in modeling nonlinear systems along with the error vector over the samples.

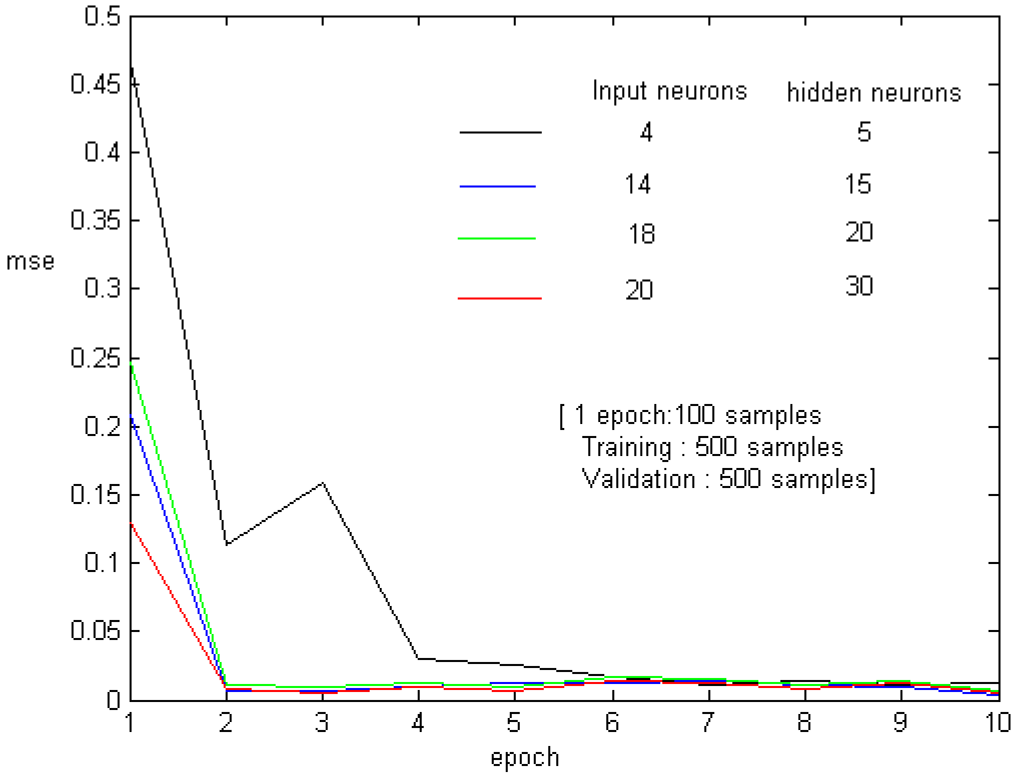


Fig. 3.7 The norm of the error vector over 500 samples (SISO) for different model sizes

To design a neural network model with lowest modeling error, the training and validation has been conducted for different model length p. The hidden layer neurons are having the activation function "bipolar sigmoid", while the output neuron has linear activation function. The number of neurons in hidden layer and number of input nodes are varied (defining the parameter

p) to check the performance in terms of Mean Square Error to obtain optimal network design.

From the above results it is seen that there is no much improvement in MSE even if the length p is increased, beyond 14. But the model 1 with the parameter p = 4 is having large amount of error. So model 2 therefore is selected as the optimal one. For the following implementations Model 2 (14 input neurons and 15 hidden neurons) is used as the model and different algorithms are used to train the model to investigate the comparative merits and demerits. The overall MSE is coming down to a value of around 0.02198 with the training as shown in table 3.1 below.

Table 3.1 Comparison between models of different sizes

Model	Number of input neurons	Number of hidden neurons	Mean Square Error (MSE)
1	4	5	0.10267
2	14	15	0.02905
3	18	20	0.02912
4	20	30	0.02198

3.4.3 MIMO System Modeling Using MLP Network

The same model was used to identify and train the network for multiple inputs multiple output cases [85]. The selected four nonlinear systems itself were used here also for the analysis and parameter identification problem. In this case also the network is trained for 100 times using the same network parameters as in the SISO case. When the termination criterion for the training is selected as a very small value for the error norm (10^{-15}), then also the training required was less than 100 epochs. The results for a general case of two input two output system modeling with data set-2 viz. the ambient noise in the sea and data set-3 viz. the acoustic source-A are given in Fig 3.8 to Fig 3.11.

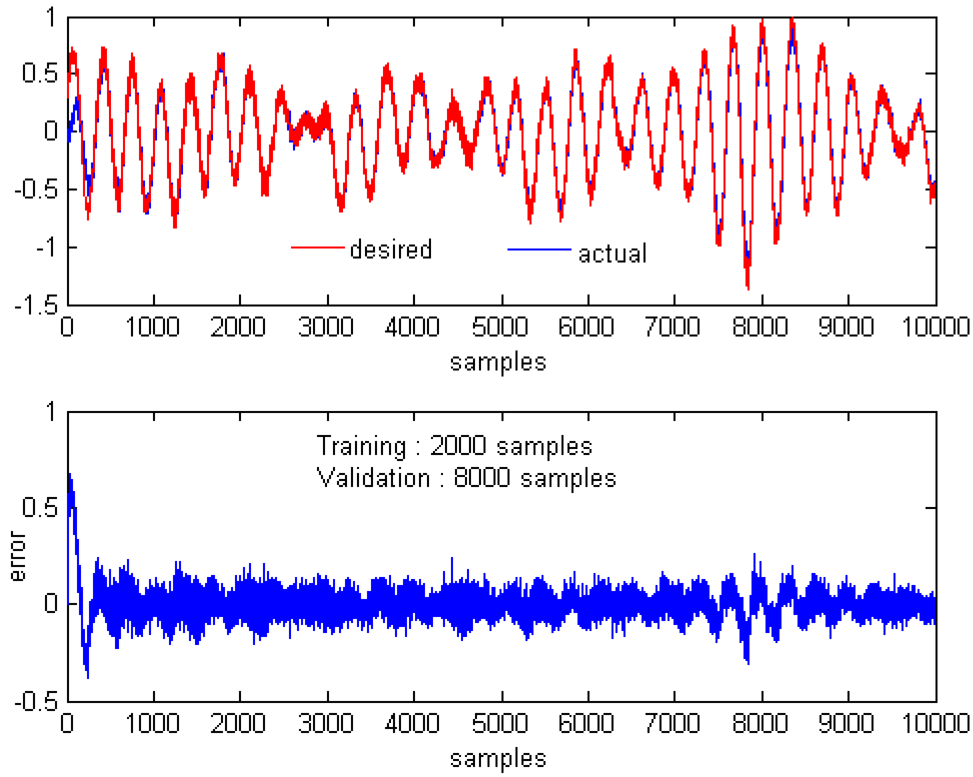


Fig. 3.8 First output of the MIMO system (data set-2 ambient noise in the sea) and the error

It has been observed that, as the number of samples used for training is increased, there is a slight improvement in the MSE performance as demonstrated in Fig 3.9 and table 3.2

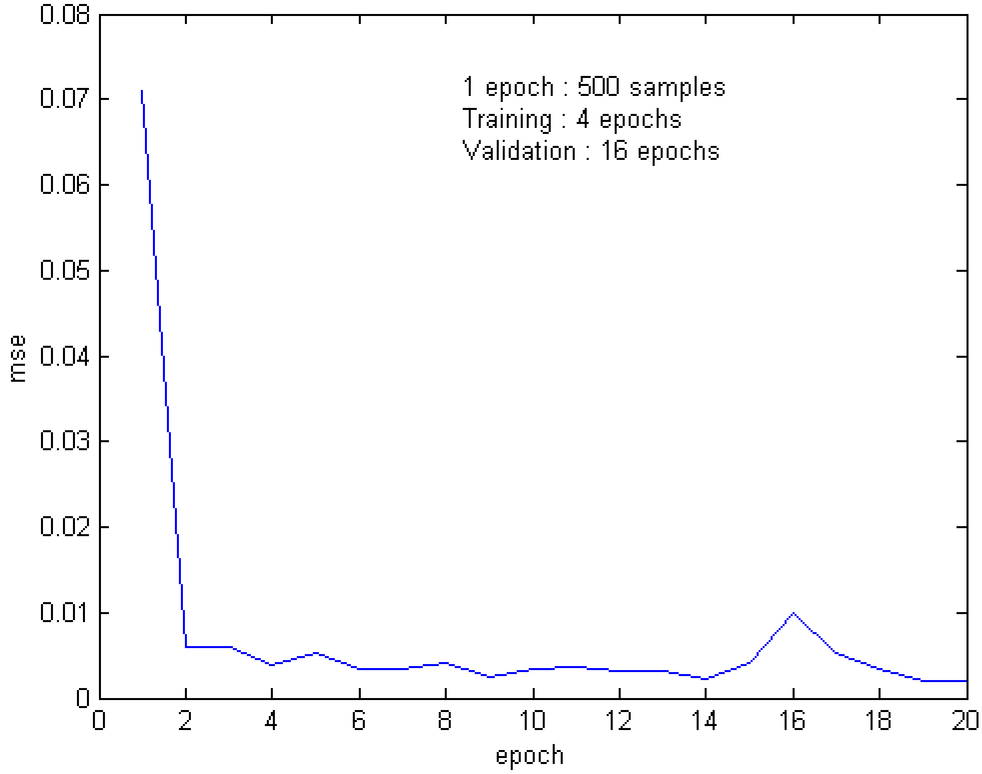


Fig. 3.9 Norm of the collective error vector for the output in Fig 3.8

Table 3.2 MSE for different training and validation set size

Number of training samples	Number of validation samples	Mean Square Error (MSE)
500	500	0.012
500	2000	0.0115
2000	8000	0.0038

Presented next is the results correspond to the data set-2 for the same network structure.

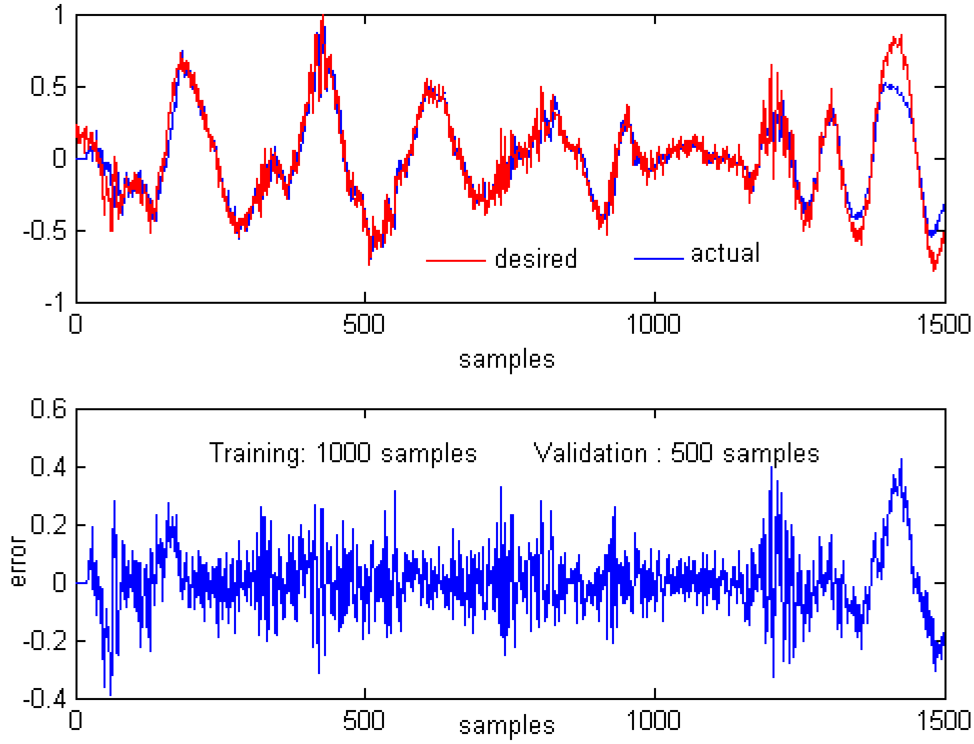


Fig 3.10 Second output of the MIMO system (data set-3 acoustic source-A) and the error

As part of the validation, here also various numbers of training and validation data size have been used and the MSE in each case is recorded. The results obtained and the MSE are presented in Fig 3.11 and Table 3.3.

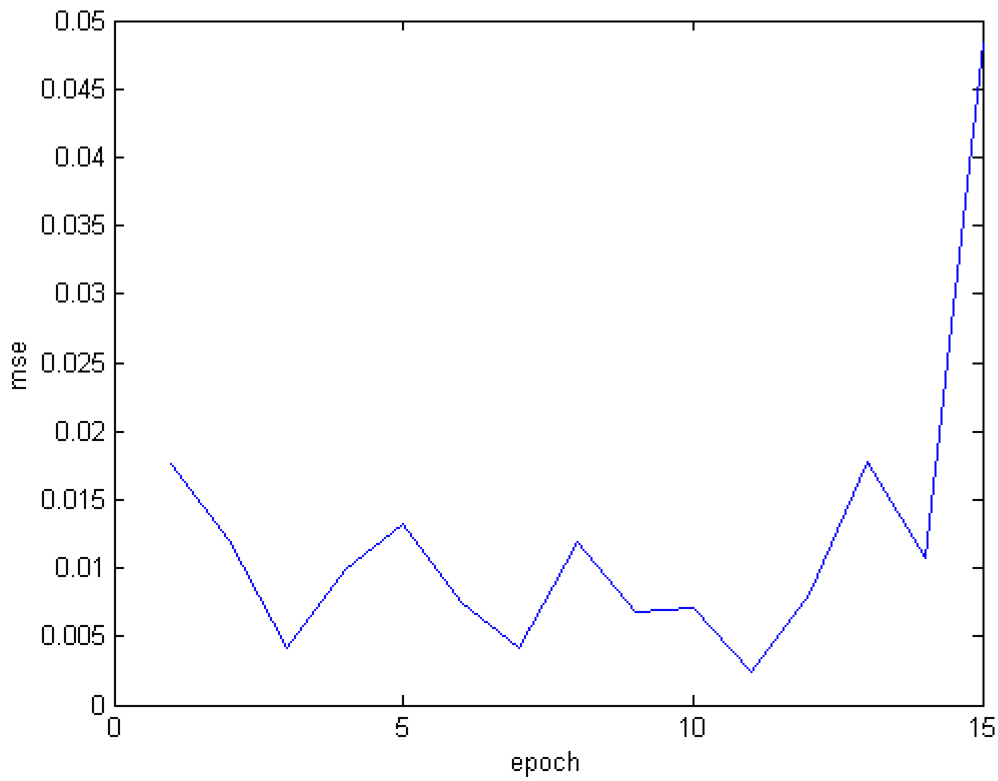


Fig. 3.11 Norm of the collective error vector for the output in Fig 3.10

Table 3.3 MSE for different training and validation set size

Number of training samples	Number of validation samples	Mean Square Error (MSE)
1000	500	0.0175
1000	7000	0.0191
2500	2500	0.0085

3.5 System Identification using RBF network

Radial Basis Function (RBF) network is alternative of the MLP network for performing a non linear mapping. As a result the RBF network can immediately be employed to find the blind system identification operator F[.]. This network consists of three layers (Fig 3.12) [70-76]. The input layer has neurons with a linear function that simply feed the input signals to the hidden layer. The input vector *u(n)* used here also is the zero-mean Gaussian white noise with lagged output and error, as to have the modeling issue described as,

$$y(t+1) = g\left(y(t-1),.....y(t-n), x(t-1),.....x(t-m), e(t-1),.....e(t-k), \phi\right)^T \quad (3.18)$$

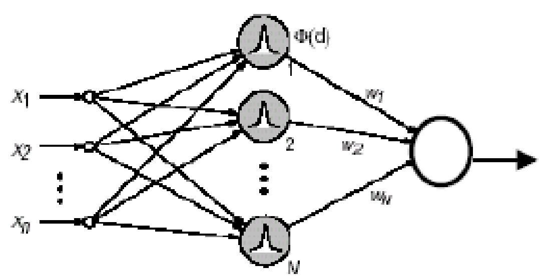


Fig: 3.12 Structure of the RBF network

The RBF network structure explained in Fig. 3.12 was used for the system identification process. It was observed that RBF network also performs in a satisfactory manner in this contest. For the determination of the output layer weight two methods are tried and the results obtained are summarized below.

3.5.1 RBF Network with Pseudo inverse Matrix Method

In this approach , the center vector is updated using the clustering algorithm which is described below [15-16, 90].The training is carried out in two phases for the RBF network.

(1) Training Phase-I

- Initialize the centre vectors C_j s as a random subset of the input vector space X_i
- One cluster center C_j is updated every time an input vector X_i is applied to the network.
- The cluster nearest to X_i has its position updated using

$$C_j(new) = C_j(old) + \alpha[X_i - C_j(old)] \qquad (3.19)$$

where α is the learning rate parameter. It is taken as 0.025 in this problem.

- Notice that the cluster center C_j is moved closer to X_i because this above equation minimizes the error vector $(X_i - C_j)$.

- After this adaptation, the output vector of the hidden layer is calculated to be

$$h_j(n) = \varphi_j\left(\left\|X_j - C_j\right\|\right) \qquad (3.20)$$

for all the hidden layer neurons. Here a multivariate Gaussian function has been selected as the activation function and the output of which is given by,

$$h_j = \exp\left(\frac{\left\|Xi - Cj\right\|^2}{2\sigma_j^{\,2}}\right) \qquad (3.21)$$

The output element of a hidden neuron, h_j has a significant value if the Euclidean distance is the minimum and thus at a time only one hidden neuron output is significant.

(2) Training Phase-II

If H represents the hidden layer output matrix / vector , d the desired o/p vector, and W_o the output layer weight vector, then it is possible to write W_o as ,

$$W_o = (H^T H)^{-1} H^T d \qquad (3.22)$$

Here the output layer weight is obtained in a single step and there is no iterative process to obtain them. Even though the weight vector calculation is very easy, the result was not very encouraging. The error is larger compared to MLP network. The pseudo inverse technique is useful when the pseudo inverse exists and the H matrix is not an exact square matrix to take its inverse. The result corresponds to data set-1 is summarized in Fig 3.13and

3.14. The network produced output is deviated from the desired or plant output as error involved is larger.

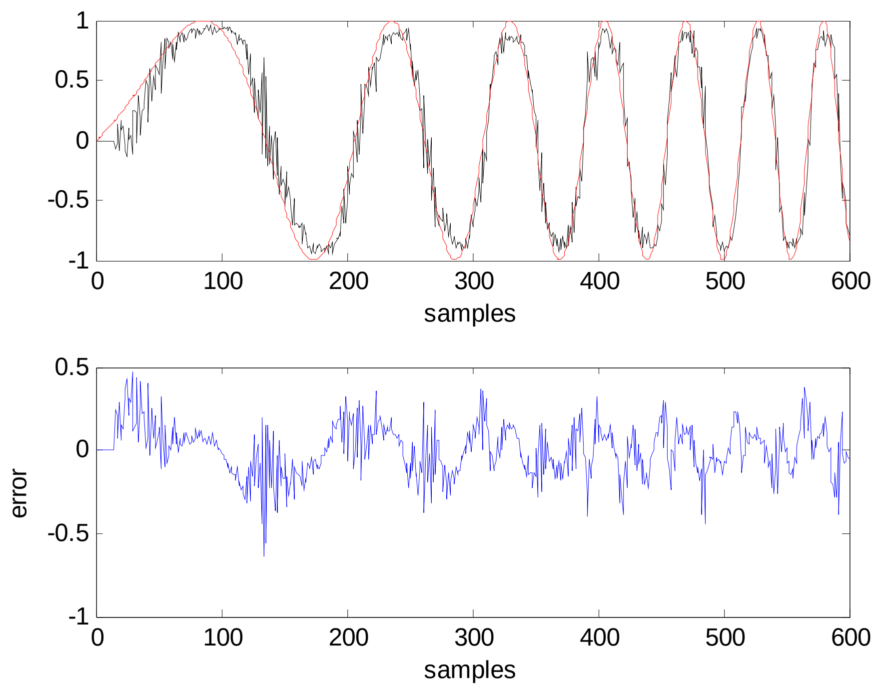


Fig 3.13 Superposition of model and nonlinear system outputs for data set-1 (RBF)

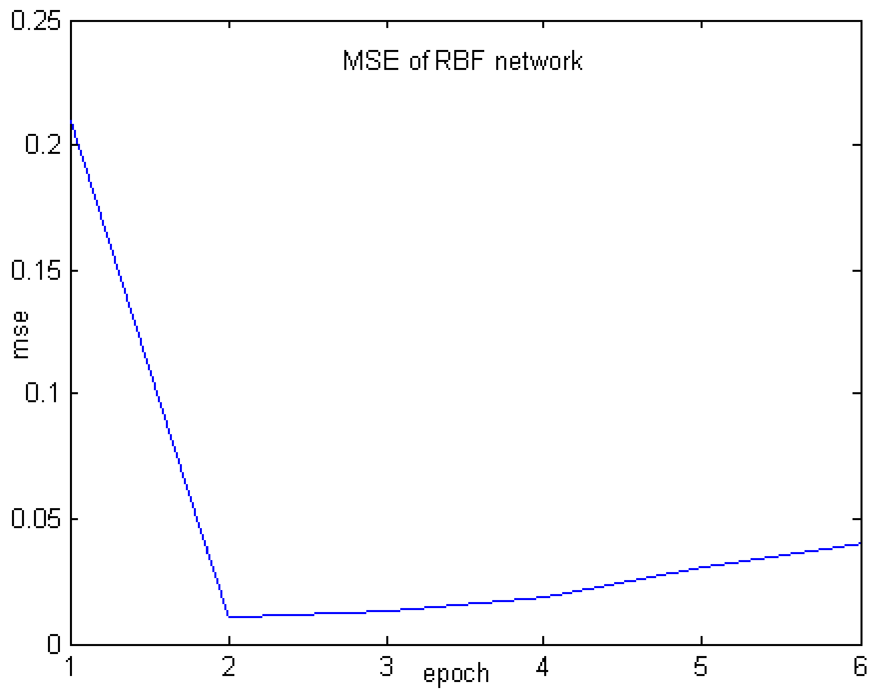


Fig 3.14 Norm of the error vector for the output in Fig 3.13

3.5.2 RBF Network with supervised weight updating

In this approach the center vectors are updated using the same clustering algorithm is used in the previous method The output weight matrix W_o is now obtained using the standard back propagation algorithm. This step is required if the pseudo inverse mentioned in the previous section does not exist. The results were more encouraging in this as the weights are also adapted with an intention to reduce overall error (Fig 3.15 and 3.16). The steps involved in this method are,

- In the first phase update the center vectors using the clustering algorithm of section 3.4.1 until all the C_js stabilize.
- In the second phase update the output layer weight vector W_o using the error back propagation algorithm.

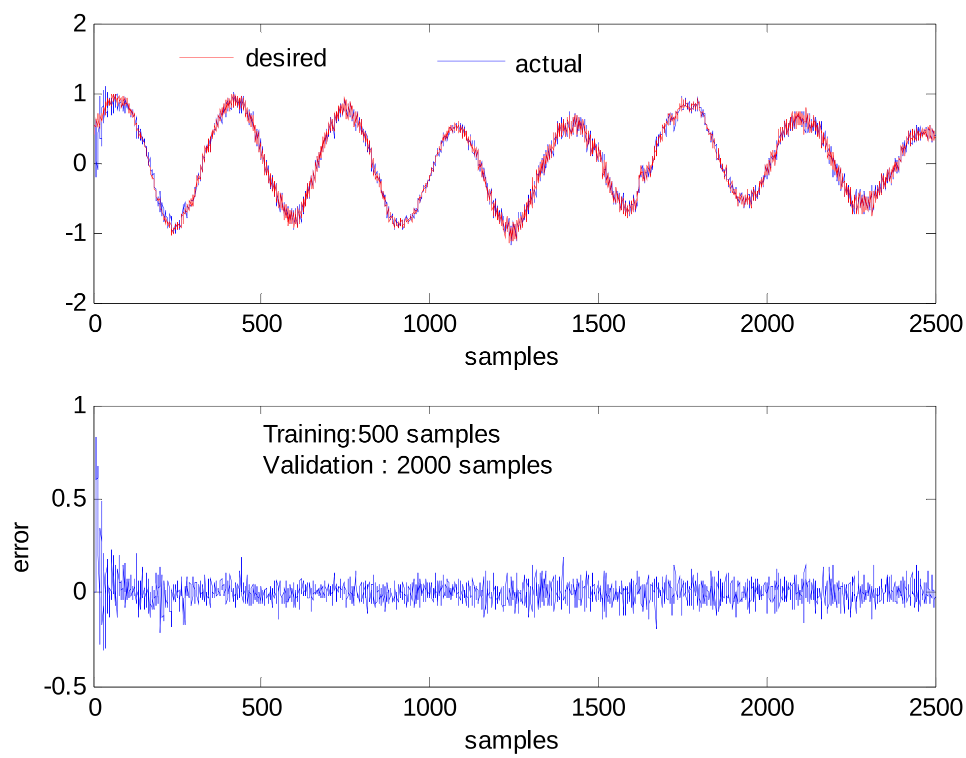


Fig 3.15 Superposition of model and nonlinear system outputs for data set-2 (RBF)

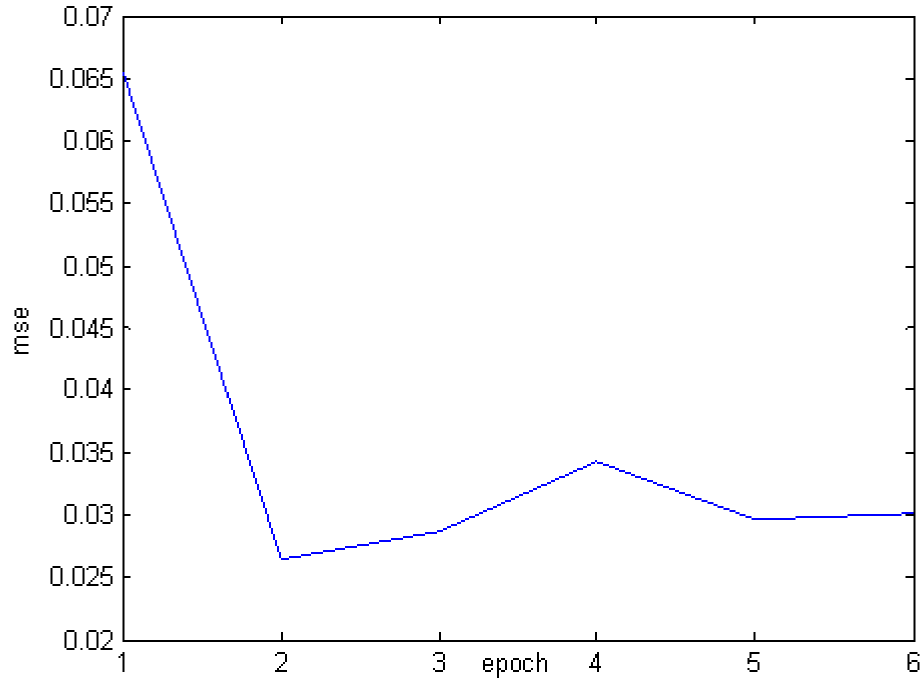


Fig. 3.16 Norm of the error vector for the output in Fig 3.15

It has been noticed that, with supervised weight updating the error has reduced to better small values compared to the first approach. The overall performance of RBF network is almost comparably to that of MLP networks and not much superior on MSE performance. In both the cases, the learning/training algorithm used is the error back propagation method. These examples generally demonstrate the non linear mapping capability of neural networks.

3.6 Conclusions

The chapter explored the capabilities of SLP, MLP and RBF neural network structures for nonlinear system modeling for SISO as well as MIMO cases in detail. Being a method on gradient descent, one has every reason to suspect the optimality of the model defined by the parameter **w** (on account of convergence to local minima). In the chapter to follow, therefore the problem of estimating the system parameters i.e. **w** using some better approaches including the Kalman estimation technique are addressed. Kalman approaches for neural network training generally offers improvement over the back propagation and its obvious demerits, to a great extend.

Chapter 4

ESTIMATION OF NETWORK PARAMETERS USING THE KALMAN FILTER APPROACH

Chapter 4 discusses the use of the Kalman filter to estimate the weights of the neural network. It is demonstrated that the predictor-corrector approach of the Kalman filter ensures an improvement in the MSE during training and the MSE remains within limit during validation also. In order to alleviate the well known problem of initial assumptions in the Kalman filter, the EKF with EM is investigated. A comparison of the performance of the basic EKF and its variations in the MSE sense is provided at the end.

4.1 Introduction

The Kalman Filter is one of the most widely used methods for estimation and tracking due to its simplicity, optimality, tractability and robustness [91-94]. The Kalman filter gives a linear, unbiased and minimum error variance recursive algorithm to optimally estimate the unknown state of a dynamic system from noisy data taken at discrete real-time. To apply the discrete Kalman filter, the system under study should be represented by a set of linear, finite dimensional state space equation. The Kalman filter uses a complete description of the probability distribution of its estimation error, in determining the optimal filtering gains. This probability distribution may be used in assessing its performance as a function of the design parameters of an estimation system. The theory is formulated in terms of state space concepts, providing efficient utilization of the information contained in the input data. Estimation of the state is computed recursively, i.e. each update of the state is computed from the previous estimate and the data currently available, so only the previous estimate requires storage [94].

4.2 Extended Kalman Filter

Kalman filtering is a prediction-correction algorithm and is derived based on the optimality criterion of least squares unbiased estimation of the state vector with the optimal weight, using all available data information. The Kalman Filtering process has been designed to estimate the state vector in a linear stochastic difference model [95-96]. However, some of the most interesting and successful applications of Kalman Filters are non-linear, i.e., the process and measurement models are given by equations of the form,

$$x_{k+1} = f(x_k, u_k, \omega_k) \quad (4.1)$$

$$z_k = h(x_k, v_k) \quad (4.2)$$

where $\boldsymbol{f}$ and $\boldsymbol{h}$ are non-linear functions on matrices, $\boldsymbol{u}_k$ is a deterministic forcing function (regard it as an input), and the random vectors ω_k and v_k again represent the process and the measurement noise and satisfy the same conditions as for the simple Kalman Filter. If the model turns out to be non-linear, a linearization procedure is usually performed in deriving the filter equations. i.e. the system is linearized about a trajectory that is continuously updated with the state estimates resulting from the measurements. The new filter (Fig. 4.1) obtained is called Extended Kalman Filter (EKF).

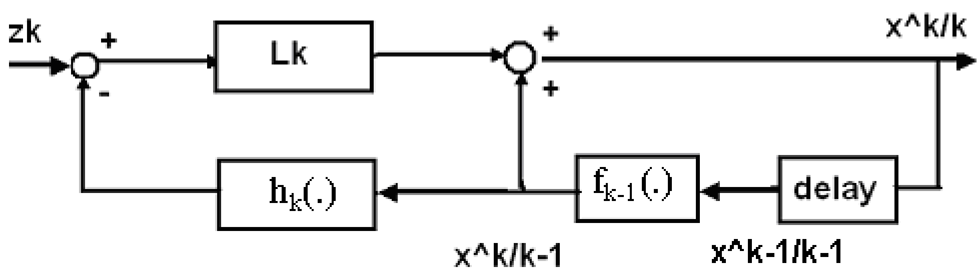


Fig. 4.1 Block diagram of Extended Kalman Filter

In order to be consistent with the linear model, the initial estimate $x\hat{}0$ = E(xo), $x\hat{}_{1|0}$ = f ($x\hat{}0$). Expanding the functions f and h about $\mathbf{x}\hat{}_{k|k+1,}$ along the Taylor series, one gets the following equations for the Extended Kalman Filter (derived from the linear filter)[95-97].

For k = 1,2,…………

$F_k = [\partial f_{k-1} (x_{k-1}\hat{})/\partial x_{k-1}]$; $H_k = [\partial h_k (x_{k|k-1}\hat{}) / \partial x_k]$ (4.3)

State update $x_{k|k-1} = f(x_{k-1|k-1})+ \omega_k$ (4.4)

State co-variance $P_{k|k-1} = F_{k-1} P_{k-1|k-1}F^T_{k-1} + Q_{k-1}$ (4.5)

Gain computation $L_k=P_{k|k-1} H^T_k (H_k P_{k|k-1} H^T_k + R_k)^{-1}$ (4.6)

Co-variance update $P_{k|k} =(I - L_kH_k)P_{k|k-1}$ (4.7)

State estimate $x_{k|k} = x_{k|k-1} + L_k [z_k - h(x_{k|k-1})]$ (4.8)

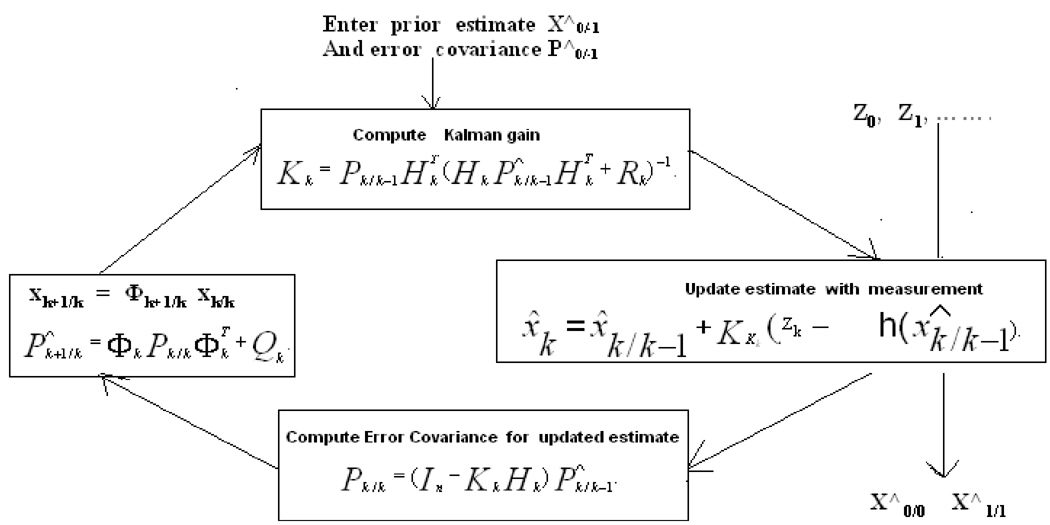


Fig.4.2. Block schematic of Extended Kalman Filter Algorithm

4.3 Formulation of the EKF algorithm for system identification

The investigation of extended Kalman filter (EKF) as the basis for an RNN training algorithm has shown very good results, in terms of number of training data and the total training time [114]. The training algorithm based on EKF requires only smaller training data than pure gradient descent algorithm. [102]. Using the same information as the gradient descent algorithm , the EKF algorithm with appropriate simplifications have modest computational needs for training a FF and an RNN. The RNN training can be viewed as a parameter estimation problem. The only problem lies in the computation of derivative of network output with respect to trainable weights. The training is formulated as a weighted least square minimization problem, where the error vector is the difference between functions of network output nodes and the desired values of these functions. The desired vector at time k is given by $d(k) = [d_1(k) \ldots\ldots d_N(k)]^T$. Let $h(k)$ denote a vector of functions of the network's output $y(k)$. But $d(k)$ and $h(k)$ are of length N.

Consider the pair of equation that serve as the basis for the derivation of the EKF family of neural network training algorithm. A neural network behavior can be described by the following nonlinear discrete time system.

$$w(k+1) = w(k) + \omega(k) \qquad (4.9)$$

$$y(k) = h\,(w(k),u(k),v(k-1)\,) + v(k) \qquad (4.10)$$

The first equation states that the state of the neural network is characterized as a stationary process corrupted by process noise 'ω', where the state of the system is given by network weight parameter 'ω'. The second equation which is the measurement equation represents the network's desired response vector 'y' as a nonlinear function of the input vector 'u', the weight parameter vector 'w'. The measurement is characterized as a zero mean, white noise with covariance given by $E[v(n)v(n)^T] = R(n)$. Similarly process noise is characterized as zero-mean , white noise with covariance given by $E(\omega(n)\,\omega(n)^T) = Q(n)$ [90].

The training problem using Kalman filter theory is described as finding the minimum mean squared estimate of the state w using all observed data. The solution is given by the following recursion.

System dynamics matrix $F_k = I$ [identity matrix] (4.11)

Measurement matrix $H_k = \frac{\partial h_k}{\partial w} \Big|_{w = w_{k|k-1}}$ (4.12)

$$A_k = [R_k + H_k\, P_{k|k-1}\, H_k^T\,]^{-1} \qquad (4.13)$$

$$L_k = P_{k|k-1}\, H_k^T\, A_k \qquad (4.14)$$

$$w_{k|k} = w_{k|k-1} + L_k\,(y(k) - \hat{y}(k)) \qquad (4.15)$$

$$w_{k+1|k} = w_{k|k} + \omega_k\,,\;\; P_{k|k} = P_{k|k-1} - L_k\, H_k\, P_{k|k-1} + Q_k \qquad (4.16)$$

The vector $\hat{w}_k$ represents the estimate of the state (i.e. weighs) of the system at arbitrary time step k. The estimate is a function of Kalman gain matrix K_k and the error vector $e_k = y_k - \hat{y}_k$, where y_k is the target vector and $\hat{y}_k$ is the network's output vector. The Kalman gain matrix is a function of the appropriate error covariance matrix P_k , a matrix of derivative of network's output with respect to all the weight parameter H_k and a scaling matrix S_k. The scaling matrix S_k is a function of measurement noise and covariance matrix R_k, as well as the matrices H_k and P_k. The error covariance matrix evolves recursively with the weight vector estimate and this matrix encodes second derivative information about the training problem and is seen to be augmented by the covariance matrix of the process noise Q_k.

The training process has higher computational and storage cost as compared to the conventional back propagation. However the algorithm converges very fast. In Kalman algorithm, the update procedure for weight vectors depends on all the information available from the start up to the current training sample.

4.4 Performance analysis of models with EKF

The same nonlinear systems as used in BPA and RLS are modeled using EKF. For all these systems the model size and structure used is same, to make the comparison of performance easy. I.e. MLFFN with one hidden layer, 14 input neurons and 15 hidden neurons. The activation function used in hidden neurons is "bipolar sigmoid" and output neuron is linear. The results obtained with EKF modeling is presented in the following sub sections.

4.4.1 Nonlinear system with output $y = \sin(t^2 + t)$

The overlapping of the actual and model outputs along with the error vector over the samples is presented in Fig 4.3. The MSE appears in Fig 4.4. Table 4.1 gives the MSE obtained for the same model with different training and validation data sizes.

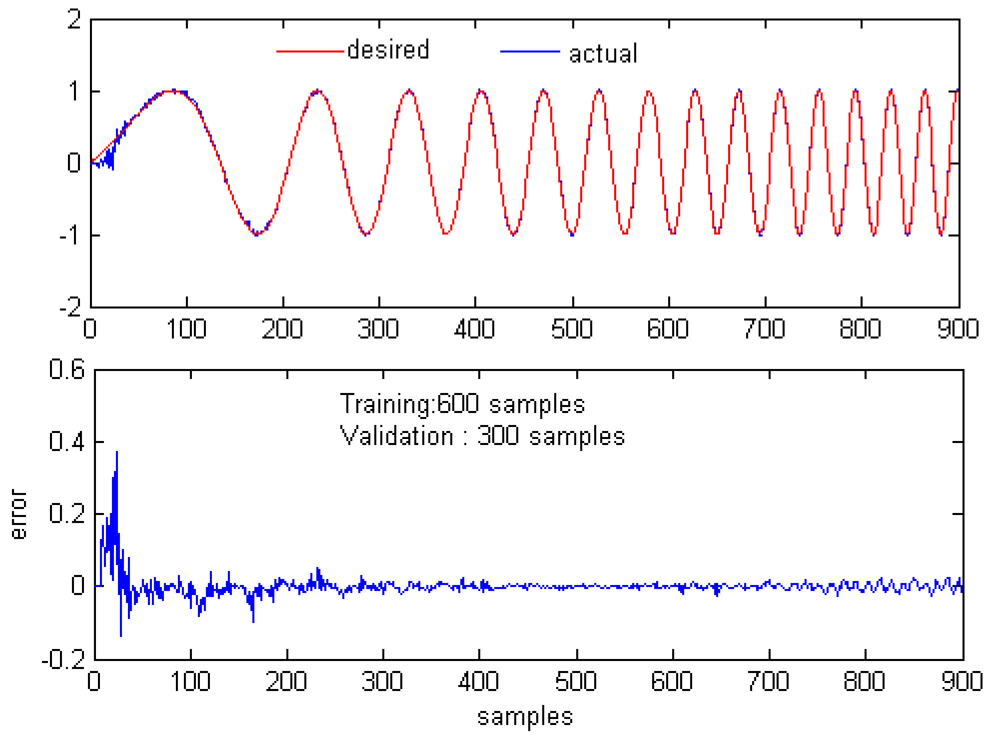


Fig. 4.3 Superposition of model output and desired output

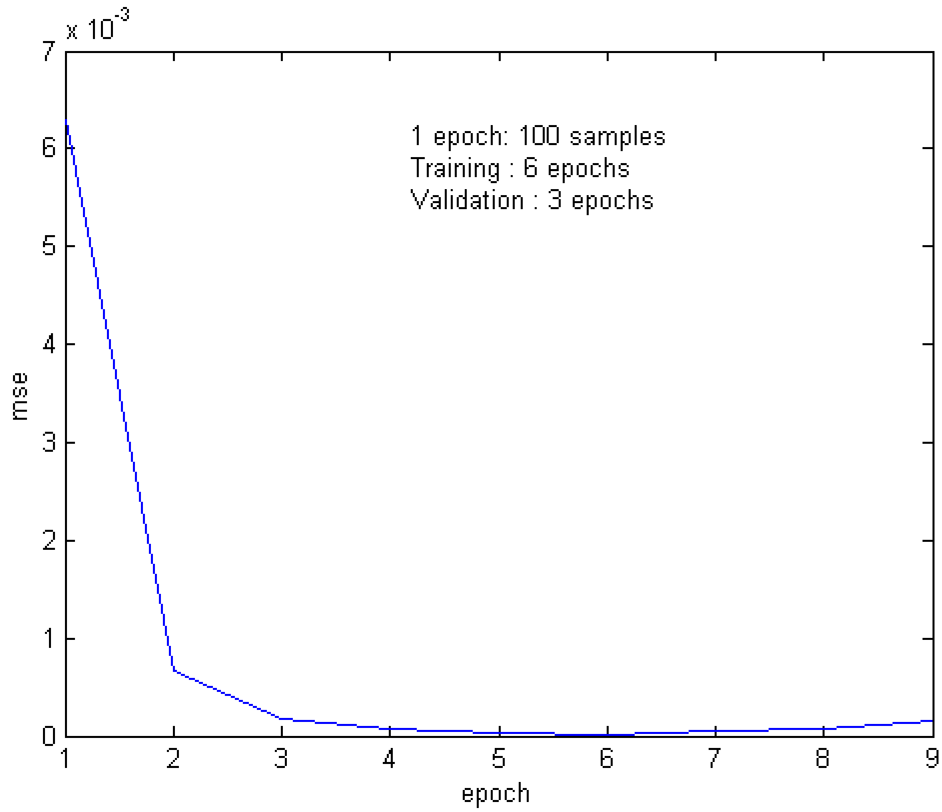


Fig 4.4 The MSE Vs data samples

Table 4.1 MSE for different training and validation set size

Training Samples	Validation Samples	Mean Square Error
300	300	4.5611×10^{-4}
600	300	9.185×10^{-5}
1000	1000	4.2984×10^{-4}

4.4.2 Selection of *P(0/-1)* and R_k

Initial value of state covariance *P(0|-1)* and measurement noise covariance R_k has a role in the accuracy of the model. So the Mean square error for the model at different values of *P(0|-1)* and R_k are evaluated. From the results it is seen that for minimum error *P(0|-1)* should be high (from 1 to 10,000) and the value of R_k should be small. So the value of R_k is chosen as 0.01 and *P(0|-1)* as 100 in the modeling of above system.

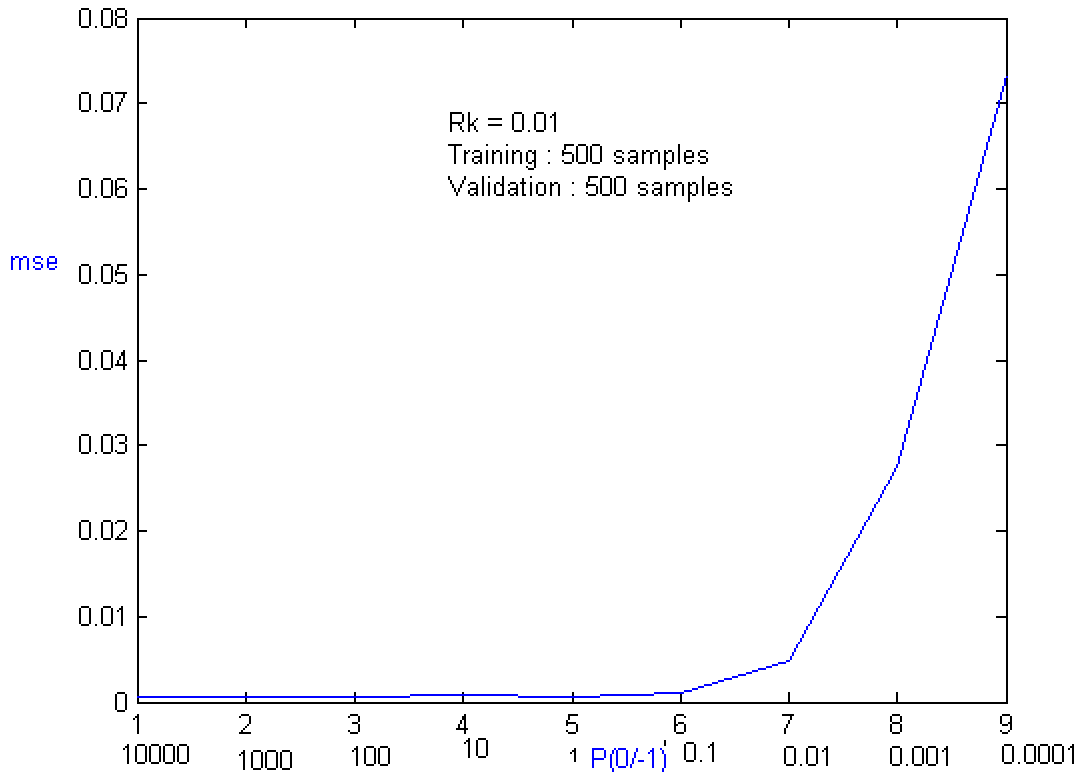


Fig. 4.5 Mean Square Error Vs *P(0|-1)*

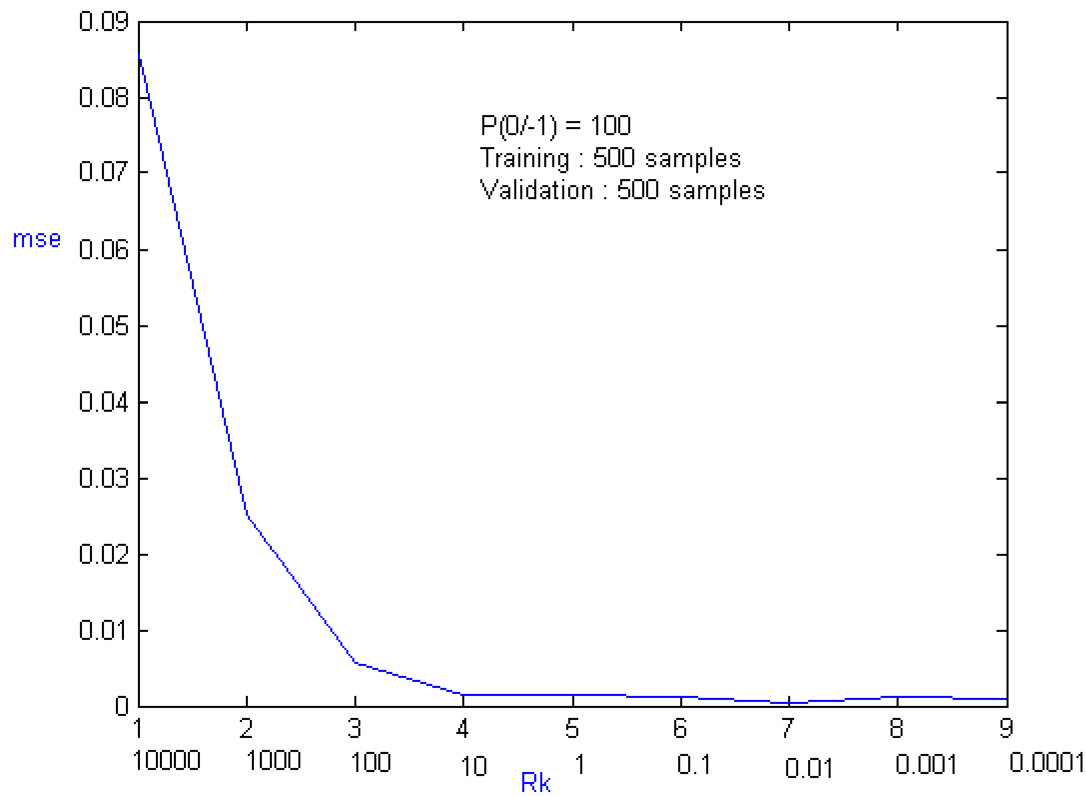


Fig 4.6 The Mean Square Error Vs R_k

Fig 4.5 and 4.6 show the dependency of EKF algorithm on the initial values of state covariance *P(0|-1)* and measurement noise covariance R_k on the overall performance of the same. This in fact is the major challenge in the training process and mechanisms to minimize such dependencies should be identified to improve the techniques.

4.4.3 Ambient noise in the sea

The overlapping of the actual and model outputs along with the error vector over the samples is presented for the second data set in Fig 4.7. The MSE appears in Fig 4.8. Table 4.2 gives the MSE obtained for the same model with different training and validation data sizes.

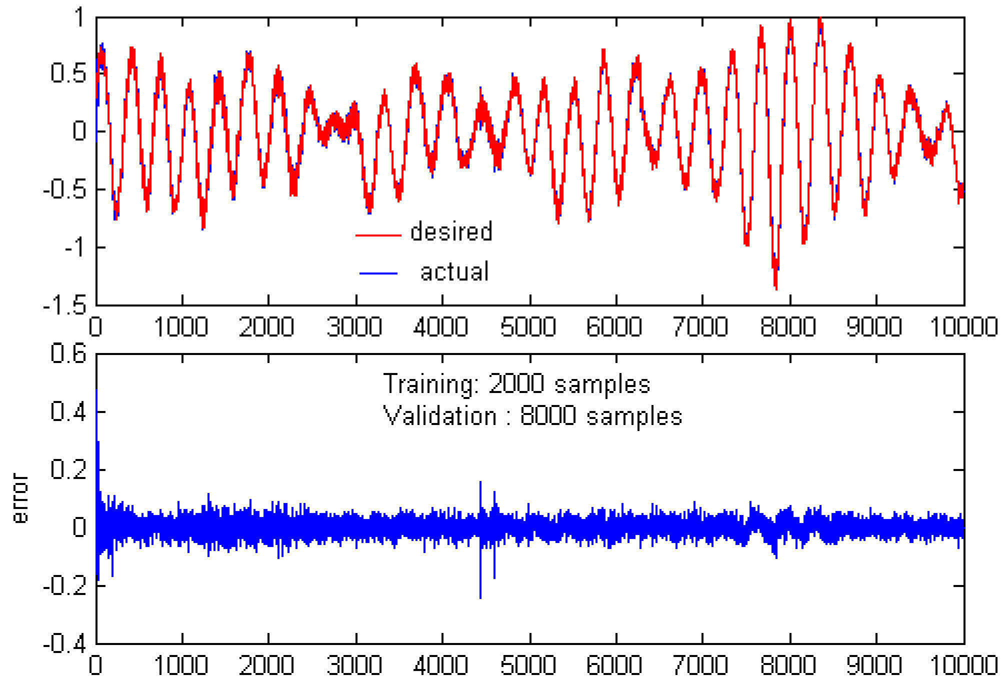


Fig. 4.7 Superposition of model output and desired output, error vector

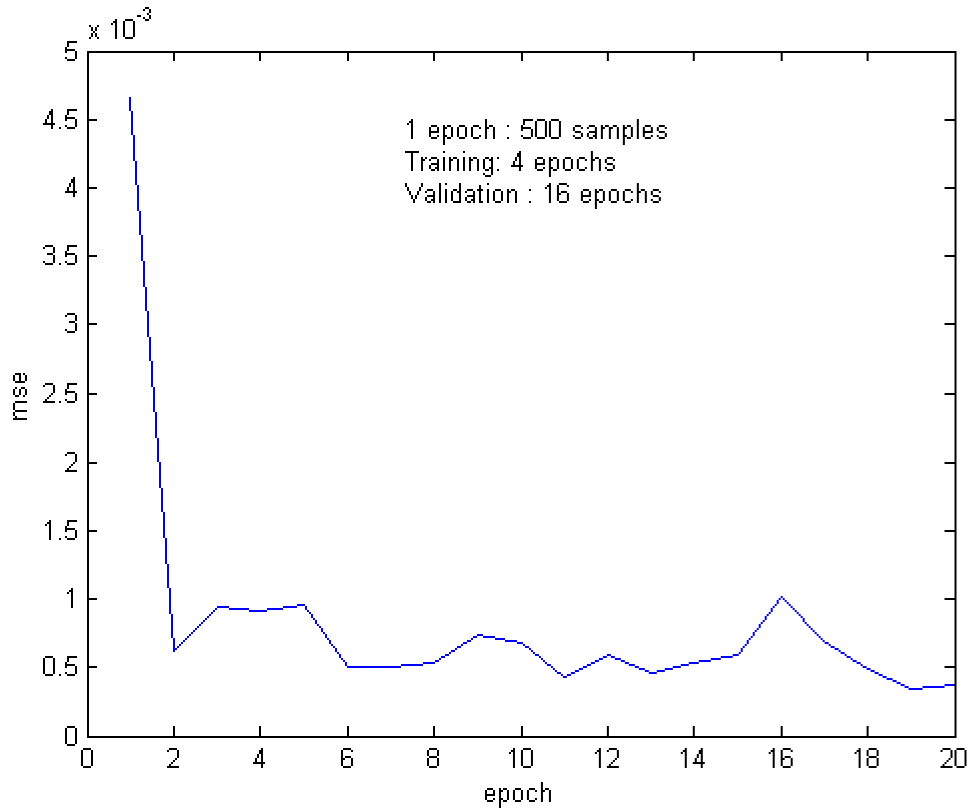


Fig 4.8 MSE Vs data samples

Table 4.2 MSE for different training and validation set size (Ambient Noise)

Training Samples	Validation Samples	Mean Square Error
500	500	1.5×10^{-3}
500	2000	2.2×10^{-3}
2000	8000	5.9013×10^{-4}

4.4.4 Acoustic source-' A'

For the third data set, the overlapping of the actual and model outputs along with the error vector over the samples is presented in Fig 4.9. The MSE appears in Fig 4.10. Table 4.3 gives the MSE obtained for the same model with different training and validation data sizes.

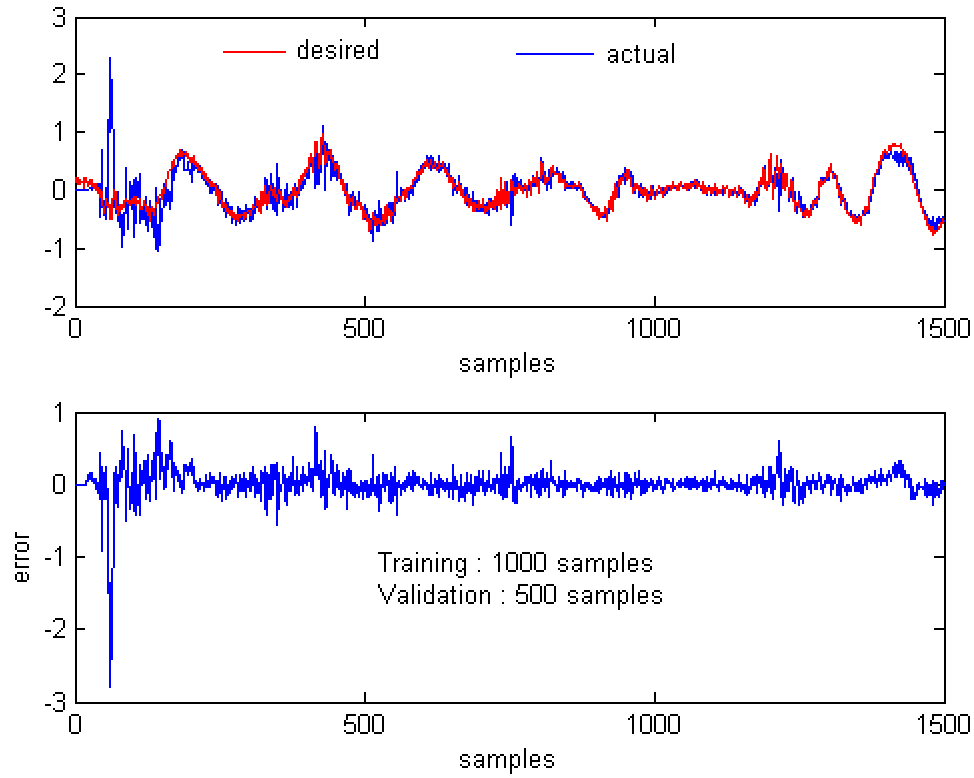


Fig.4.9 Superposition of model output and desired output, error vector

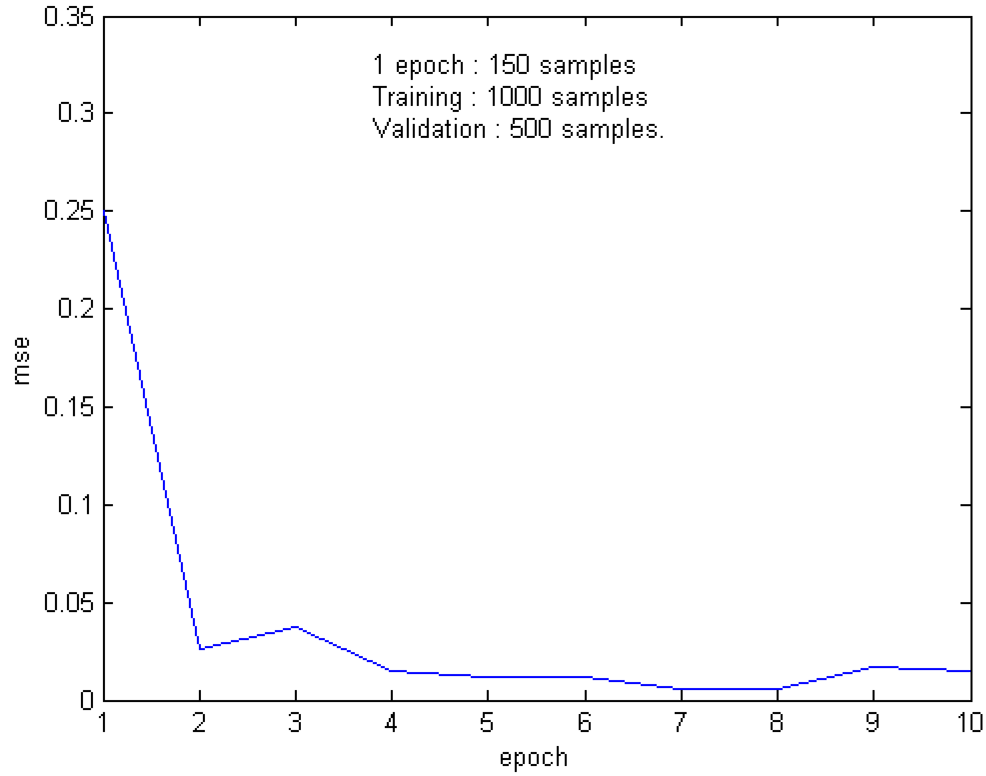


Fig 4.10 MSE Vs data samples

Table 4.3 MSE for different training and validation set size (Acoustic source-A)

Training Samples	Validation Samples	Mean Square Error
1000	500	0.0122
1000	7000	0.0139
2500	2500	0.0054

It can be observed that if sufficiently large data set is used for training, the overall MSE can be brought down to acceptable levels in the network models.

4.4.5 Acoustic source-' B'

For the third data set, the overlapping of the actual and model outputs along with the error vector over the samples is presented in Fig 4.9. The MSE appears in Fig 4.10. Table 4.3 gives the MSE obtained for the same model with different training and validation data sizes.

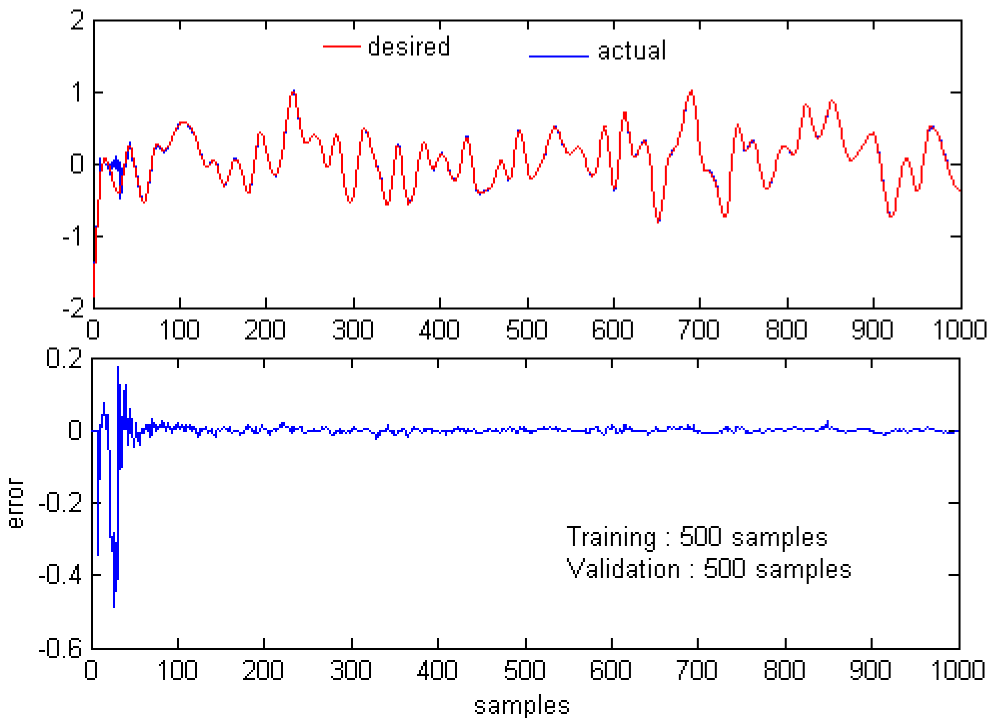


Fig.4.11 Superposition of model output and desired output, error vector

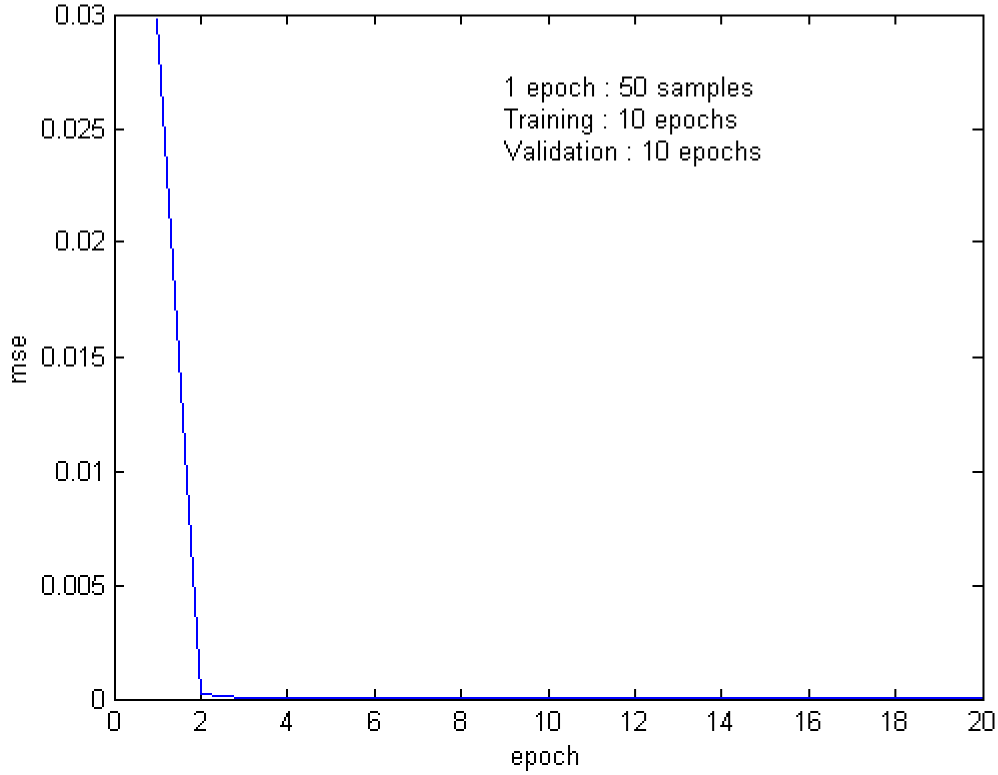


Fig 4.12 MSE Vs data samples

Table 4.4 MSE for different training and validation set size (Acoustic source-B)

Training Samples	Validation Samples	Mean Square Error
1000	500	0.0123
1000	7000	0.0141
2500	2500	0.0014

While doing the implementations, it is noted that the initialization of *P(0|-1)* is having an important role in convergence. The mean square error is minimum when *P(0|-1)* is having values in the range 1 to 10,000 and MSE is high if *P(0|-1)* is initialized with value less than 1. By conducting experiment for different values of *P(0|-1)*, the effect of this in Kalman Gain is evaluated and plotted as in Fig 4.13.

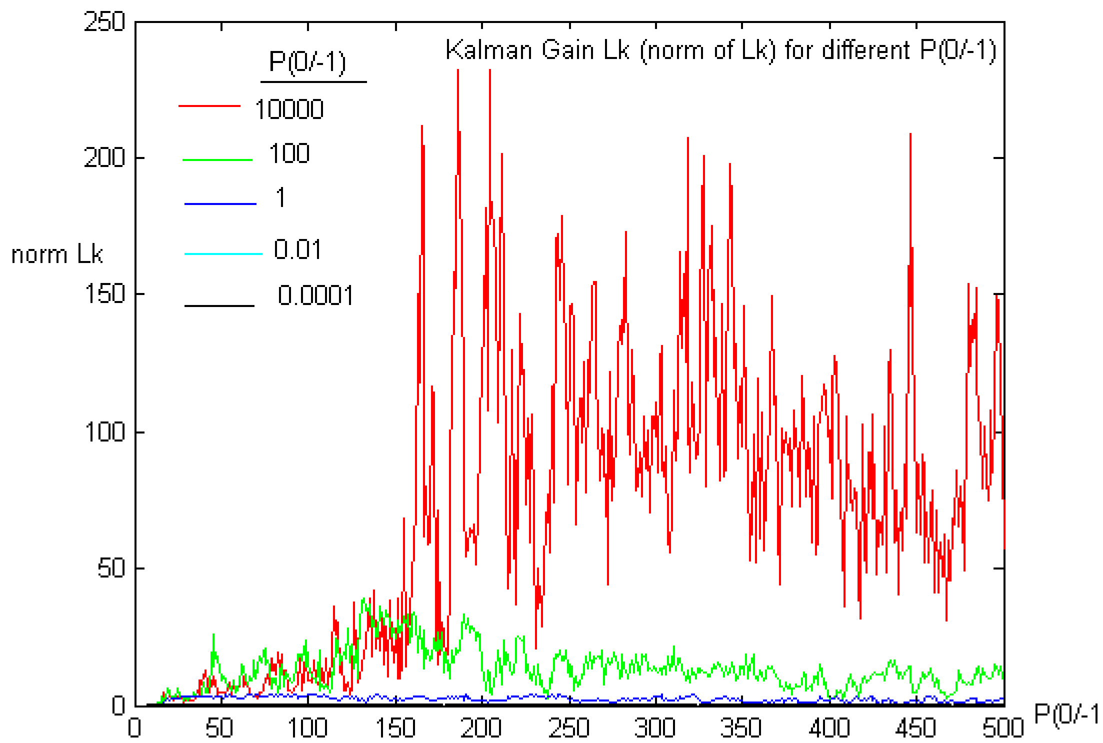


Fig 4.13 Kalman Gain for different values of *P(0|-1)*

Here the influence of initial value of state covariance *P(0|-1)* in Kalman gain (norm of L_k) is analyzed. Kalman gain is high for higher values of *P(0|-1)* and low for lower values. The updating of weights is not proper if the Kalman gain is less. This also supports the result where the MSE is minimum for higher values of *P(0|-1)*.

4.5 EKF algorithm with Expectation Maximization

A well known limitation of Kalman estimators is the assumption of known a priori statistics and initial states and covariance to describe the measurement and process noise. In many applications, it is not straightforward to choose the right noise covariance matrices. More over the matrices of parameters governing the linear transformations in the measurement and process equations are typically unknown. Unfortunately the optimality of the Kalman Filter often hinges to the designer's ability to formulate these matrices a priori. To circumvent this limitation and ensure optimality, it is important to design algorithms for estimating the noise co variances and parameter matrices without leading to a degradation in the performance of the Kalman estimator. Expectation Maximization (EM) algorithm is a *method to calculate the initial states and covariance* avoiding the difficulty in setting proper values for these by trial and error [107-108].

4.5.1 EM Algorithm

A well known limitation of Kalman estimators is the assumption of known a priori statistics and initial states and covariance to describe the measurement and process noise [92]. In many applications, it is not straightforward to choose the right noise covariance matrices. More over the matrices of parameters governing the linear transformations in the measurement and process equations are typically unknown. Unfortunately the optimality of the Kalman Filter often hinges to the designer's ability to formulate these matrices a priori. To circumvent this limitation and ensure optimality, it is important to design algorithms for estimating the noise covariances and parameter matrices without leading to a degradation in the performance of the Kalman estimator. An algorithm based on the Expectation Maximization [108] is a proposed to calculate the initial states and covariance, instead of trial and error approach. To outline the EM algorithm to state space learning, the following nonlinear state space representation is focused onto [102-103].

$$w_{k+1} = Aw_k + d_k \tag{4.17}$$

$$y_k = g(w_k, x_k) + v_k \tag{4.18}$$

The EKF algorithm for training MLPs suffers from serious shortcomings, namely choosing the initial states and covariance$(\mu,\prod)$, the noise covariance matrices R and Q and the state dynamics matrix A. EM algorithm shall be employed to optimize initial values of $\Theta = \{ \mu, \prod, R, Q, \text{ and } A\}$[91].

After computing the forward estimates in EKF with N samples, the "Rauch-Tung-Striebel smoother" is employed to perform the following backward recursions [95]

$$J_{k-1} = P_{k-1}A^T P^{-1}_{k|k-1} \quad (4.19)$$

$$W_{k-1|N} = w_{k-1}J_{k-1}(w_{k|N}-Aw_{k-1}) \quad (4.20)$$

$$P_{k-1|N} = P_{k-1}+J_{k-1}(P_{k|N}-P_{k|k-1})J^T_{k-1} \quad (4.21)$$

$$P_{k,k-1|N} = P_k J^T_{k-1}+J_k(P_{k+1,k|N}-AP_k)J^T_{k-1} \quad (4.22)$$

where K=Kalman gain, G_k=Jacobean of the measurement function g of equation 4.18. The

backward recursion as above is initialized as given below: $W_{N/N} = W_N$, $P_{N/N}=P_N$ and $P_{N,N-1|N}=(I-K_N G^T_N)AP_{N-1}$. (4.23)

The backward recursion results in $W_{1/N}$ and $P_{1/N}$, which can be used to compute $\Theta = \{ \mu, \prod, R, Q, \text{ and } A\}$, as given below:[91].

$$\mu = W_{1/N}\,,\ \Pi = P_{1|N} \text{ and} \quad (4.24)$$

$$R = \frac{1}{N}\sum_{k=1}^{N} G_k^T P_{k/N} G_k + (y_k - g(w_{k/N}, x_k))(g(w_{k/N}, x_k))^T \quad (4.25)$$

$$A = \gamma\nabla^{-1}; Q = \frac{1}{(N-1)}(\Gamma - \gamma\nabla^{-1}\gamma^T); \mu = w_{1/N}; \Pi = P_{1/N} \quad (4.26)$$

where, $$\Gamma = \sum_{k=2}^{N} w_{k/N} w_{k/N}^{T} + P_{k/N} \tag{4.27}$$

$$\Delta = \sum_{k=2}^{N} w_{k-1/N} w_{k-1/N}^{T} + P_{k-1/N} \tag{4.28}$$

and $$\gamma = \sum_{k=2}^{N} w_{k/N} w_{k-1/N}^{T} + P_{k-1/N} \tag{4.29}$$

The simulation results in Fig. 4.14 to 4.21 show the dramatic improvement in the performance of the EKF estimation algorithm, using the Expectation Maximization. The performance of EM is analyzed for the same four nonlinear systems.

4.6 Performance analysis of models using EKF withEM

The EM Algorithm is computationally intensive and it needs inversion of matrices. When tried to implement for the same model structure as in EKF, the covariance matrices become singular and inversion was not valid. So EM is implemented for smaller model structure. Simple EKF and EKF with EM are implemented and compared.

4.6.1Nonlinear system with output y = sin (t^2 + t)

Due to the requirement to invert of large matrices for large model sizes, thesis chose to implement EM algorithm for smaller model structure, (6 inputs, and 5 hidden neurons) [9].

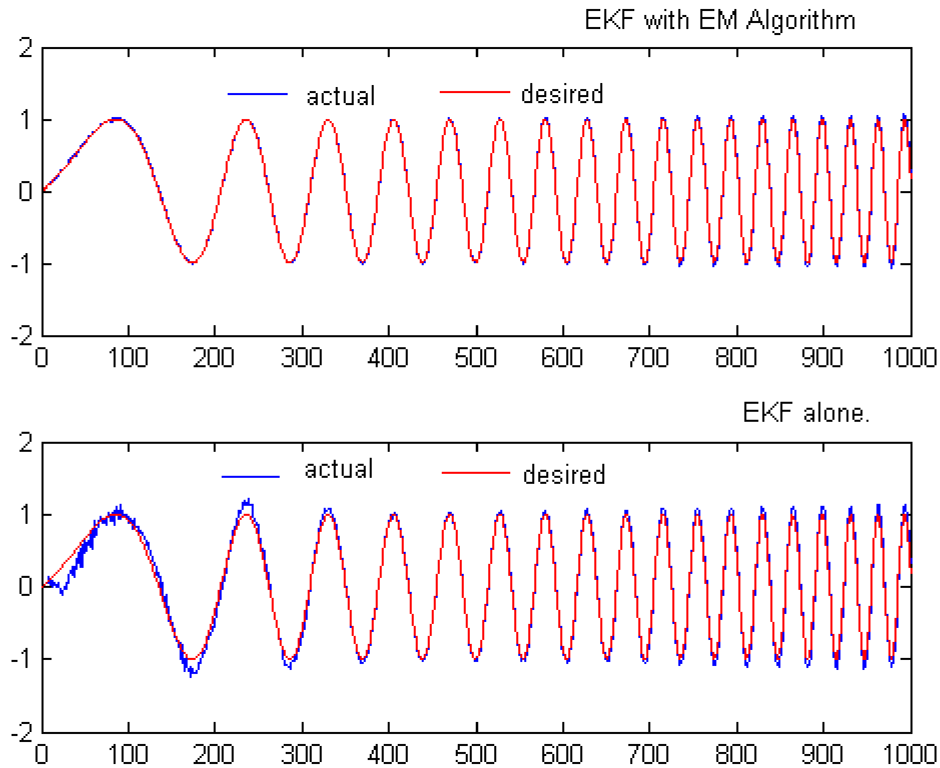


Fig.4.14 Superposition of model and desired output in the EKF and EKF with EM algorithms

But the improvement in the performance of the order of 10^{-4} as shown in Fig 4.14 is quite impressive even with the reduced dimension. This itself shows the improvement over simple EKF algorithm when EM is also incorporated.

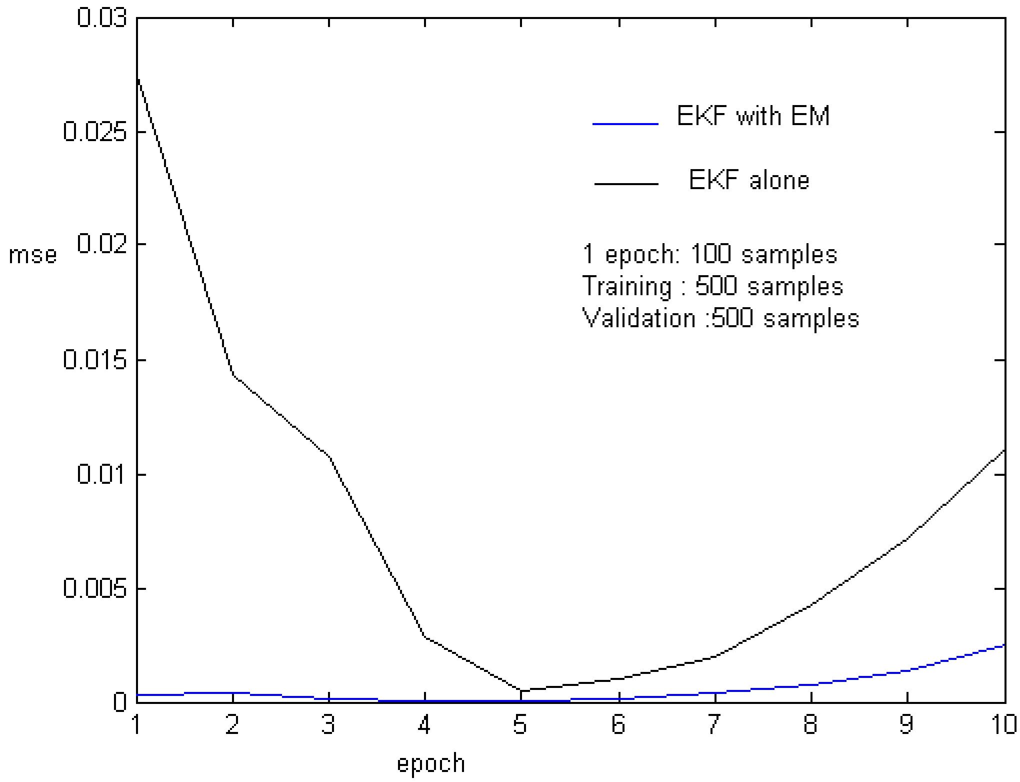


Fig 4.15 The MSE for EKF and EKF with EM algorithms

In the case of the first nonlinear system $y=sin(t^2+t)$, the average MSE is 4.13×10^{-4} for EKF with EM and 0.0906 without EM as demonstrated in figures 4.3 and 4.4.

4.6.2Results of ambient noise in the sea

For the second data set, the overlapping of the actual and model outputs with simple EKF and EKF with EM algorithms is presented in Fig 4.16. The MSE appears in Fig 4.17.

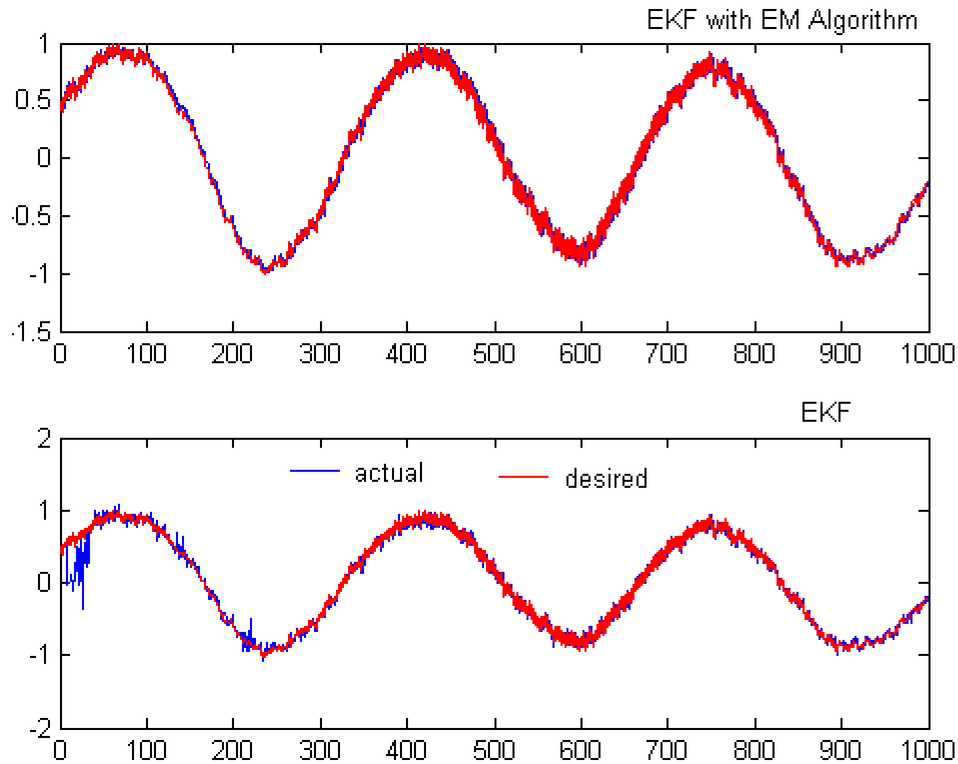


Fig.4.16 Superposition of model and desired output in the EKF and EKF with EM algorithms

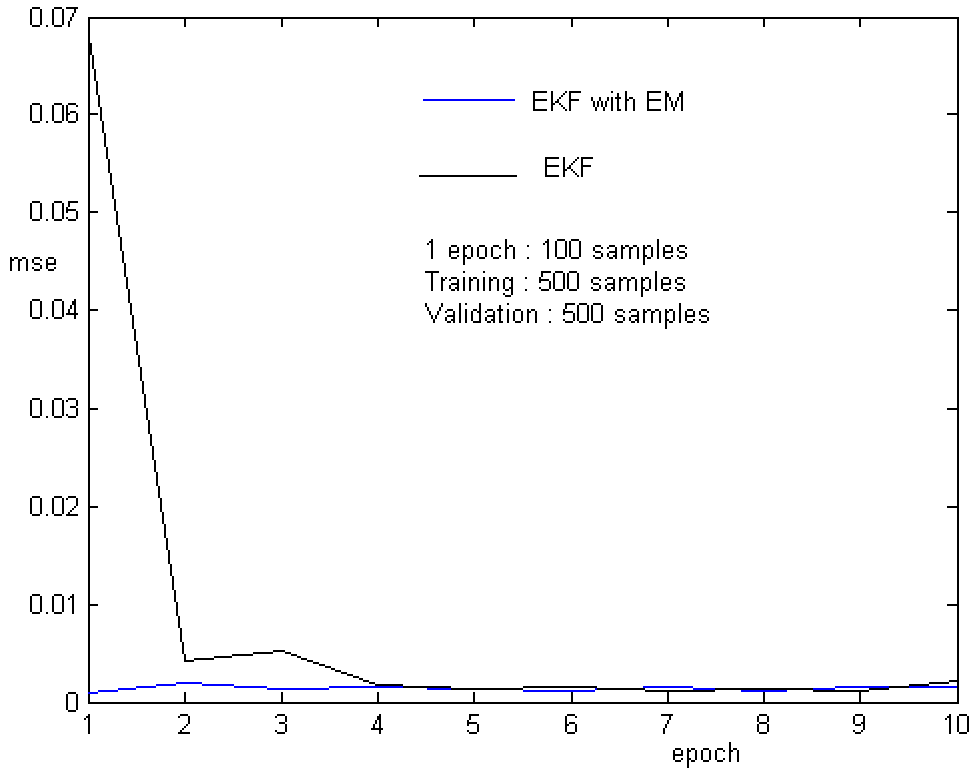


Fig 4.17 The MSE EKF and EKF with EM algorithms

For ambient noise in the sea, the MSE is 8.06×10^{-4} for EKF with EM and 0.0045 without EM as shown in Fig.4.10, which shows the betterment in performance to a good extend.

4.6.3 Acoustic source-' A'

The overlapping of the actual and model outputs with simple EKF and EKF with EM algorithms for the third data set is presented next in Fig 4.18. The MSE in this case appears in Fig 4.19.

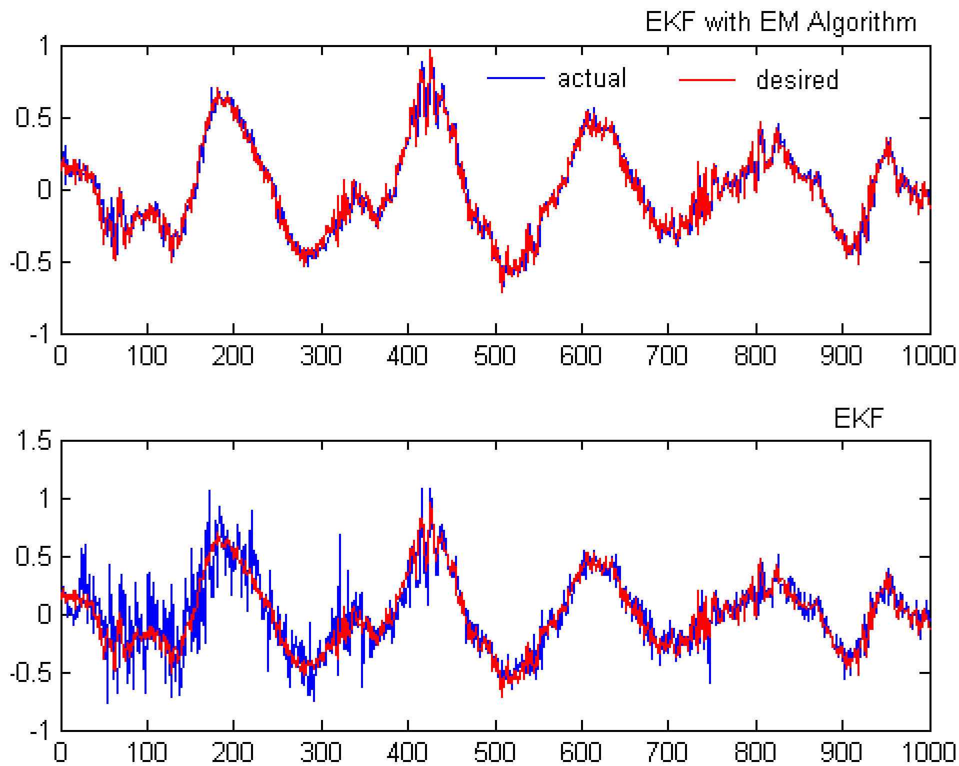


Fig.4.18 Superposition of model and desired output in the EKF and EKF with EM algorithms

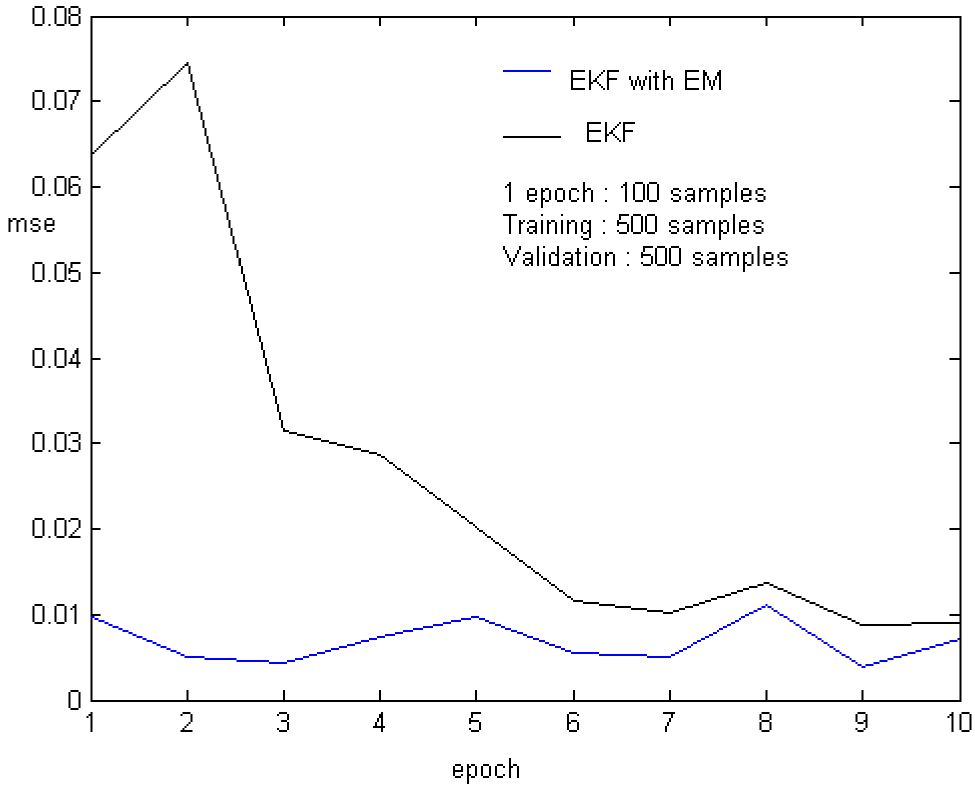


Fig 4.19 The MSE for data set-3 with EKF and EKF with EM algorithms

For acoustic source- A, the average MSE that could be achieved is around 0.0065 for EKF with EM and 0.0106 without EM. There is a very evident improvement with EM as depicted in Fig 4.19.

4.6.4 Acoustic source-' B'

The overlapping of the actual and model outputs with simple EKF and EKF with EM algorithms for the fourth data set is presented next in Fig 4.20. The MSE in this case appears in Fig 4.21.

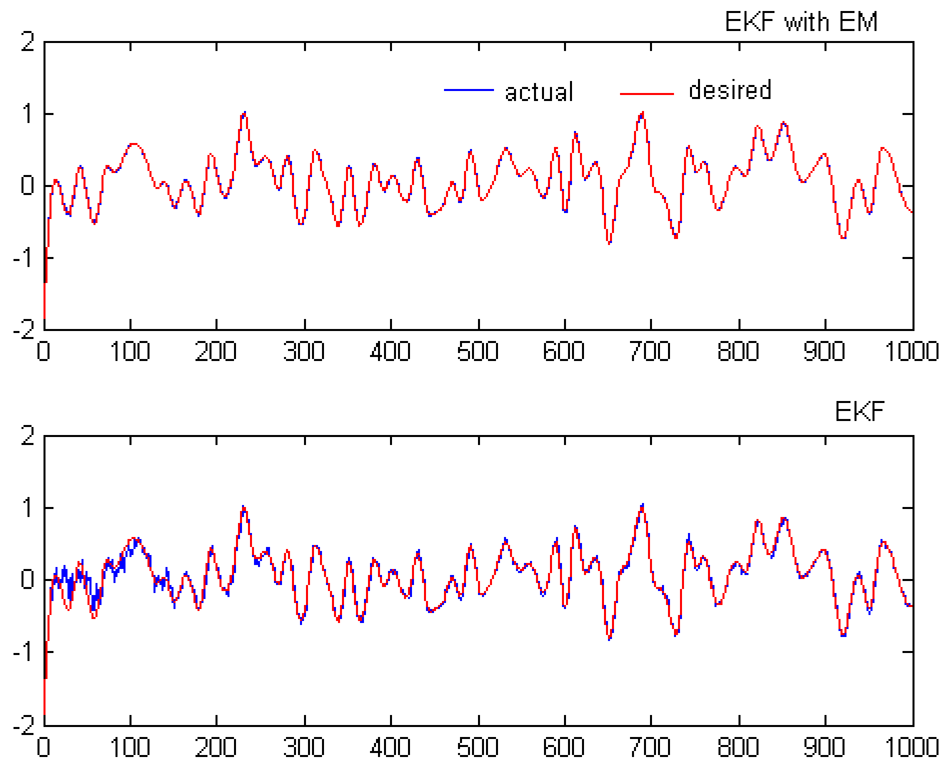


Fig.4.20 Superposition of model and desired output in the EKF and EKF with EM algorithms

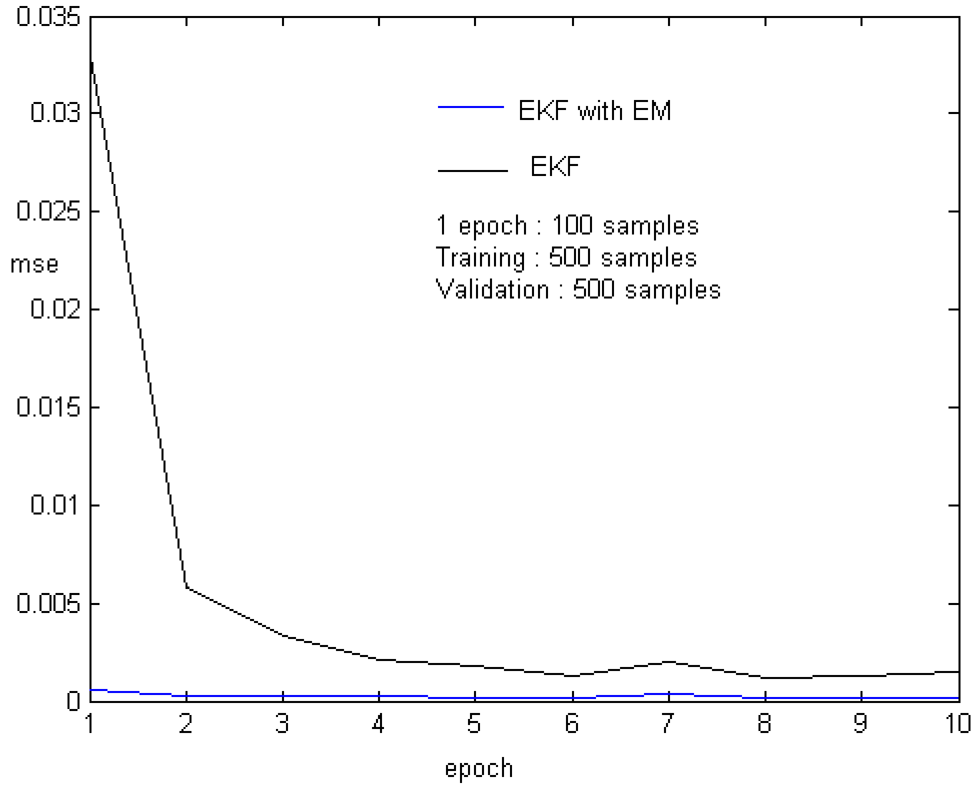


Fig 4.21 The MSE with EKF and EKF with EM algorithms

For acoustic source-B, the average MSE has come down to the range of 0.000174 for EKF with EM and 0.0014 without EM. A summary of the results obtained for the two algorithms with the four data sets under consideration is now consolidated in Table 4.5.

4.6.5 Comparison between EKF based Methods for modeling

Table 4.5 Performance comparison of simple EKF and EKF with EM in the MSE sense

System	Mean Square Error (MSE) of	
	Simple EKF	EKF with EM
$y=sin(t+t^2)$	0.0906	4.132×10^{-4}
Ambient noise	0.0045	8.06×10^{-4}
Acoustic source 'A'	0.0054	0.0065
Acoustic source 'B'	0.0038	0.000174

From the above results it is seen that EKF algorithm converges faster and has marginally good performance compared to the other algorithms. It is also consistent for all the nonlinear systems modeled. The performance of EKF can be increased further by EM algorithm. The algorithms in general give good results and are computationally efficient and in problems where faster convergence is required, as in adaptive filters and real world problems, Kalman estimation has to be used.

4.7 Conclusion

This chapter discussed about Kalman Filter and implementation in the Feed forward neural network. Four different nonlinear time series are modeled using neural network and trained with Extended Kalman Filter algorithm. All the models converged with very small error. The performance of the models is superior to that of back propagation algorithm. The performance is consistent for the four different time series where different types of nonlinearity involved, which indicates that the EKF algorithm is well suited for nonlinear system identification in general. The dependence of mean square error on initialization of states, process and measurement covariance are also evaluated and the suitable values are found out by running the simulations at different values (trial and error method). Expectation Maximization technique is applied overcome this difficulty.

The results show that the performance of EKF is improved with EM algorithm. But the computational cost is more. The EM algorithm constitutes a good estimator of the noise covariance in stationary environments and, hence, is well suited for the initialization of filtering techniques.

Different algorithms can be used to train the neural network model for nonlinear system Identification. Such an algorithm 'Maximum Likelihood Estimation (MLE)' is introduced in chapter 5.

The study of system identification is not comprehensive without phase plane analysis. A mathematical model describing the dynamics of the system is the state space model and can be implemented efficiently in a Recurrent Neural Network. These concepts are well discussed in the chapters to follow.

Chapter 5

NONLINEAR SYSTEM MODELING WITH MAXIMUM LIKELYHOOD ESTIMATION

Chapter 5 introduces the Maximum Likelihood Estimation, a well established statistical tool, of network parameters. The theory is extended for the training of ANN for nonlinear modeling. Comparison of the performance is also made with the EKF methods developed in chapter 4.

5.1 Introduction

Many approaches are available to estimate the weights for training the neural network model in nonlinear system identification. So far the Back Propagation algorithm and Extended Kalman Filter algorithm have been examined in details. Maximum likelihood is a well established procedure for statistical estimation [99-100] and is implemented for modeling nonlinear syatems and the performance is evaluated. In this procedure first formulate a log likelihood function and then optimize it with respect to the parameter vector of the probabilistic model under consideration [116-119]. The same four nonlinear systems are used for modeling.

5.2 Maximum Likelihood Estimation

The term "maximum likelihood estimate" with the desired asymptotic properties usually refers to a log of the likelihood equation that globally maximizes the likelihood function $L(x)$ [90-91]. In other words the ML estimate x_{ML} is that value of the parameter vector x for which the conditional probability density function $P(z/x)$ is maximum[115].

The maximum likelihood estimate $\mathbf{x}_{ML}$ of the target parameters $\mathbf{x}$ from N independent measurements is the mode of the conditional probability density function (likelihood function):

$$p\left(z/x\right) = \frac{1}{(2\pi)^{\frac{N}{2}}} \prod_{k=1}^{N} e^{\left(-\frac{(r_k)^2}{2\sigma_k^2}\right)} \qquad (5.1)$$

In the log likelihood function, $\log(P(z/\mathbf{x})) = -\frac{1}{2}(r_k^2)$, r_k is the residual, $r_k = (d_k - z_k) / \sigma_k$, σ_k is the standard deviation, d_k-desired value and z_k - estimated value. Maximizing log likelihood function $\log(P(z|\mathbf{x}))$ is equivalent to minimizing the negative log likelihood function $L(z, \mathbf{x})$.

Switching over to the parameter w in place of x, and by using the negative log-likelihood function $L(z, \mathbf{w})$, the ML problem is reformulated as a nonlinear least square problem:

$$Minimize(L(x,z)) where L(x,z) = \sum_{k=1}^{N} \frac{1}{2} r_k^2 \qquad (5.2)$$

The ML estimate must satisfy the following optimality condition:

$$\Delta_x L(z, w_{ML}) = J(w_{ML})^T R(w_{ML}) = 0 \qquad (5.3)$$

R(w) is given by $R(w) = [\, r_1(w) \ldots\ldots r_N(w)]^T$ (5.4)

where $r_k = (d_k - z_k)/\sigma_k$,

and $J(w)$ the $N \times n$ Jacobian matrix, and $J(w)^T = \Delta_w r(w)^T$ (5.5)

The operator Δ_w is defined as, $\Delta_w = [\, \partial/\delta w_1 \;\; \partial/\delta w_2 \;\; \ldots\ldots \;\; \partial/\delta w_N]^T$ (5.6)

One could employ many optimization methods to find the ML Estimate [12].

Two well known optimization techniques chosen are:

1. Gauss-Newton method [90]
2. Conjugate-Gradient method [90]

5.3 System modeling using Gauss-Newton Method

A feed forward neural network model similar to earlier cases is designed for the identification of the same nonlinear systems and trained using Gauss – Newton method. The Gauss-Newton method is applicable to a cost function that is expressed as the sum of error squares.

$$E(w) = \sum_{k=1}^{N} \frac{1}{2} r_k^2 \qquad (5.7)$$

The error signal *r(k)* is a function of adjustable parameter vector **w.** Given an operating point *w(n),* one could linearise the dependence of *r(k)* on **w** by writing,

$$r^l(k,\mathbf{w}) = r(k,\mathbf{w}(n)) + [\partial r(k)/\partial \mathbf{w}]^T_{\mathbf{w}=\mathbf{w}(n)} (\mathbf{w} - \mathbf{w}(n)), \; k=1,2,\ldots n \qquad (5.8)$$

Equivalently, by using matrix notation one may write

$$r^l(k,\mathbf{w}) = r(k, \mathbf{w}(n)) + J(n)^T (\mathbf{w}-\mathbf{w}(n)) \qquad (5.9)$$

The updated parameter vector **w(**n+1) is then defined by

$$\mathbf{w}(n+1) = \arg\min_{(\mathbf{w})} \{ \tfrac{1}{2} r^l(k,\mathbf{w})^2 \} \qquad (5.10)$$

The squared Euclidean norm of $r^l(n,\mathbf{w})$ is,

$$\tfrac{1}{2}\, r^l(n,\mathbf{w})^2 = 1/2\, r(n)^2 + r(n)^T J(n)\,(\mathbf{w} - \mathbf{w}(n)) + \tfrac{1}{2}\,(\mathbf{w} - \mathbf{w}(n))^T J(n)^T J(n)\,(\mathbf{w} - \mathbf{w}(n)) \quad (5.11)$$

Hence differentiating this expression with respect to **w** and setting the result equal to zero, it is possible to obtain,

$$J(n)^T r(n) + J(n)^T J(n)\,(\mathbf{w} - w(\mathbf{n})) = 0 \qquad (5.12)$$

Solving this equation for **w**,

$$\mathbf{w}(n+1) = \mathbf{w}(n) - [J(n)^T J(n)]^{-1} J(n)^T r(n) \qquad (5.13)$$

which describes the pure form of the Gauss-Newton method.

However, for the Gauss-Newton iteration to be computable, the matrix product $J(n)^T J(n)$ must be nonsingular. To guard against the possibility that $J(n)$ being rank deficient, the usual practice is to add the diagonal matrix δI to the matrix $J(n)^T J(n)$. The parameter δ is a small positive constant chosen to ensure that, $J(n)^T J(n) + \delta I$ is positive definite for all n. The update equation accordingly becomes,

$$\mathbf{w}(n+1) = \mathbf{w}(n) - (J(n)^T J(n) + \delta I)^{-1} J(n)^T r(n) \qquad (5.14)$$

where $J(n)$ is the Jacobian matrix equal to $\Delta_{\mathbf{w}}\, r(n)$

5.4 Performance Analysis of MLE (Gauss-Newton)

The same set of four nonlinear systems, as in Chapter 4 is modeled with MLE also. The performance analysis is done by plotting the mean square error in each case. Among the many systems modeled, the outputs of two nonlinear systems are presented in fig. 5.1 to fig.5.8. The results are further used to get a conclusion on the model accuracy, its consistency and generalization capability.

5.4.1 Nonlinear system $y = \sin(t^2 + t)$

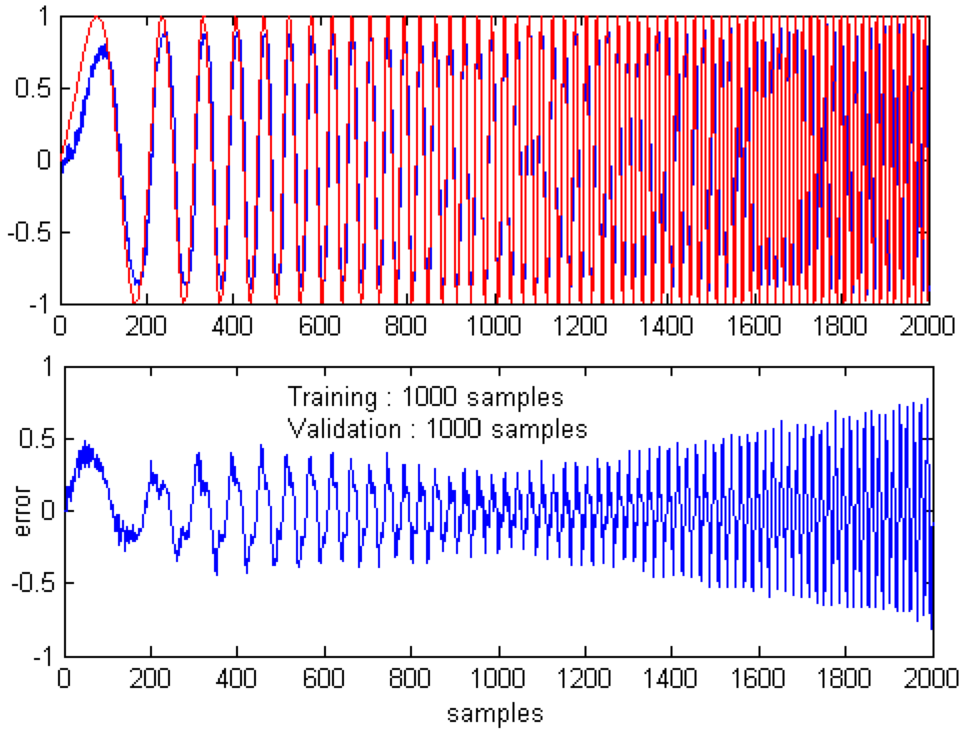


Fig.5.1 Superposition of model and desired output with the MLE algorithm and the error vector (data set-1)

Fig 5.1 shows the overlapped network and the model output for easy comparison of the performance. The second plot is the error vector over the training as well as validation samples. The overall MSE is plotted in Fig 5.2 below.

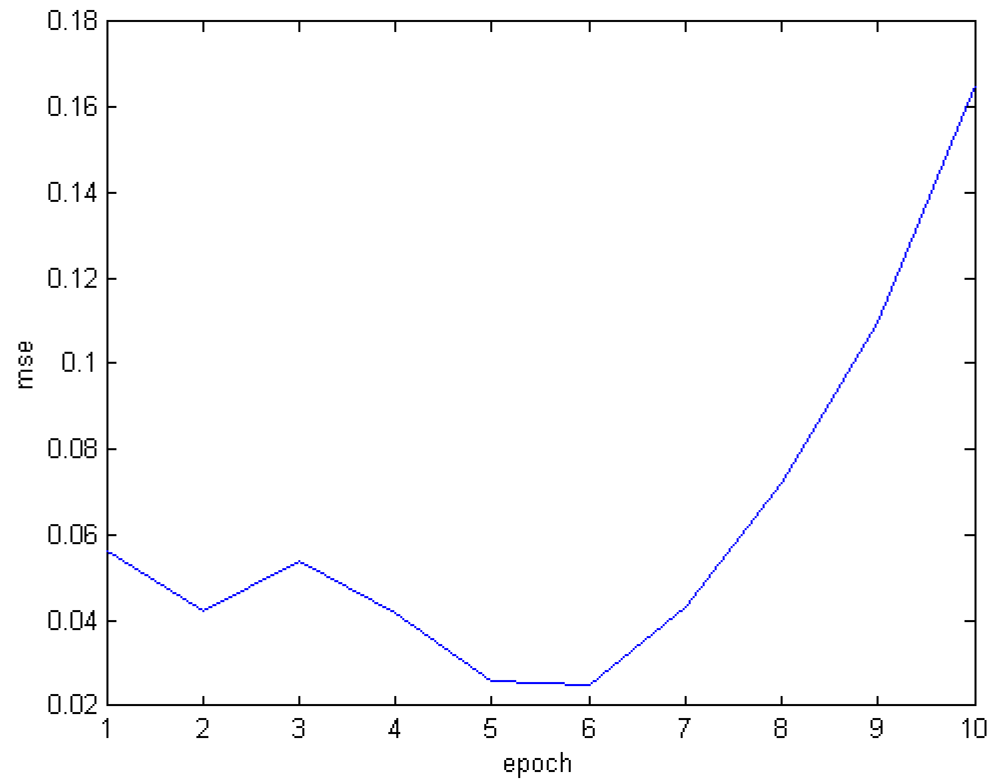


Fig 5.2 MSE for the result described in Fig 5.1

The overall MSE in this case is coming down to around 0.06356 only, compared to 0.0906 in EKF for the same nonlinear system. At the same time, the consistency of the model over a number of the systems is found to be generally good. The results obtained with the remaining data sets are described in the following figures.

5.4.2 Ambient Noise in the sea

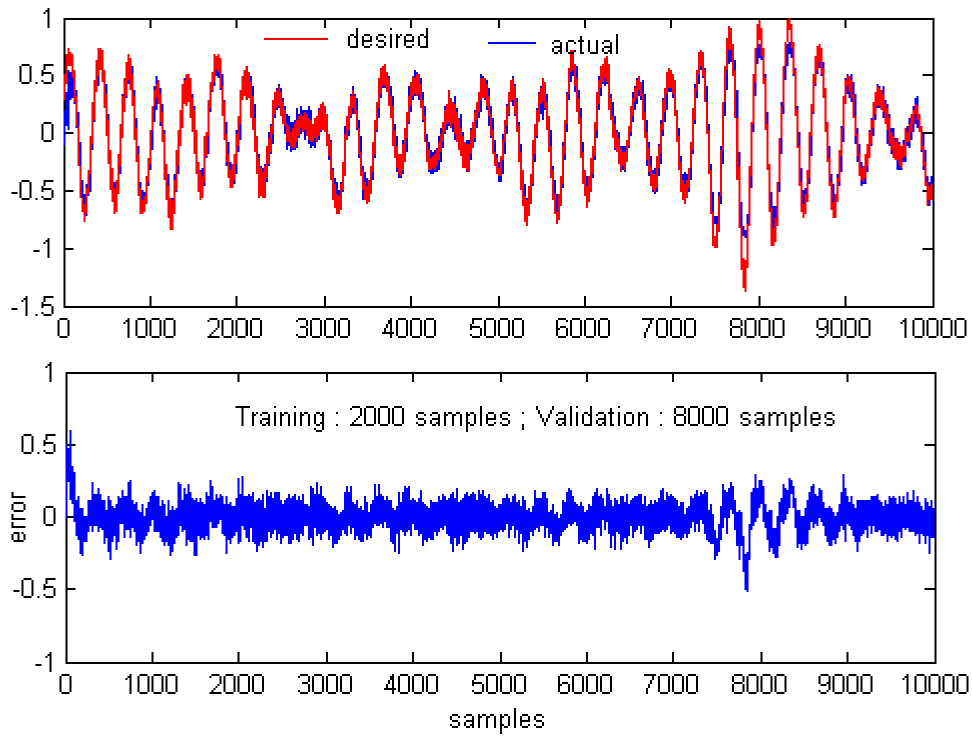


Fig.5.3 Superposition of model and desired output with the MLE algorithm and the error vector (data set-2)

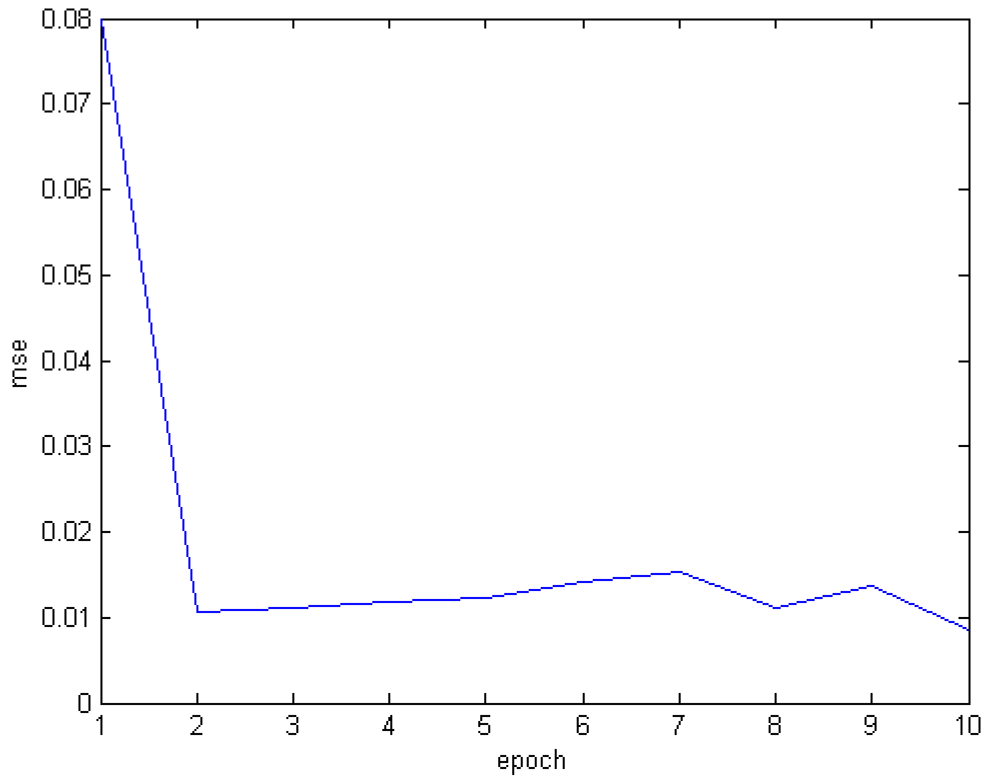


Fig 5.4 MSE for the result described in Fig 5.3

The average MSE for the first nonlinear system $y=sin(t+t^2)$, is 0.0635 for MLE (G-N) and 0.0906 in EKF as demonstrated in figures 5.1 and 5.2. The MSE for ambient noise in the sea is 0.0083 for MLE and 0.0045 for EKF algorithm.

5.4.3 Acoustic source- 'A'

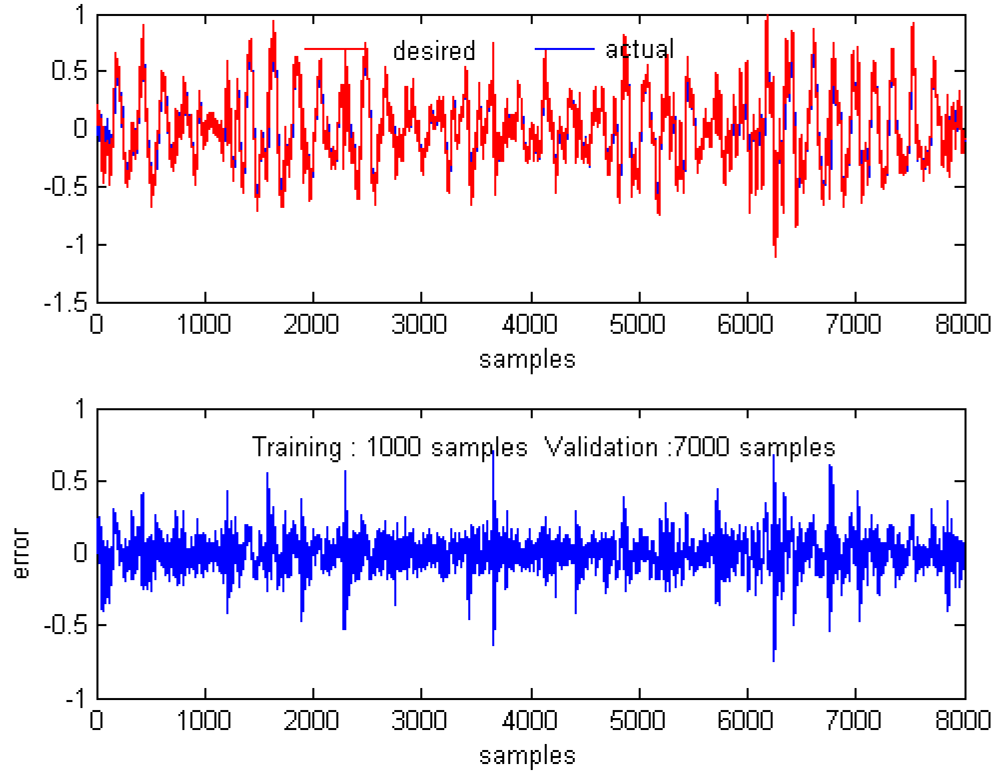


Fig.5.5 Superposition of model and desired output with the MLE algorithm and the error vector (data set-3)

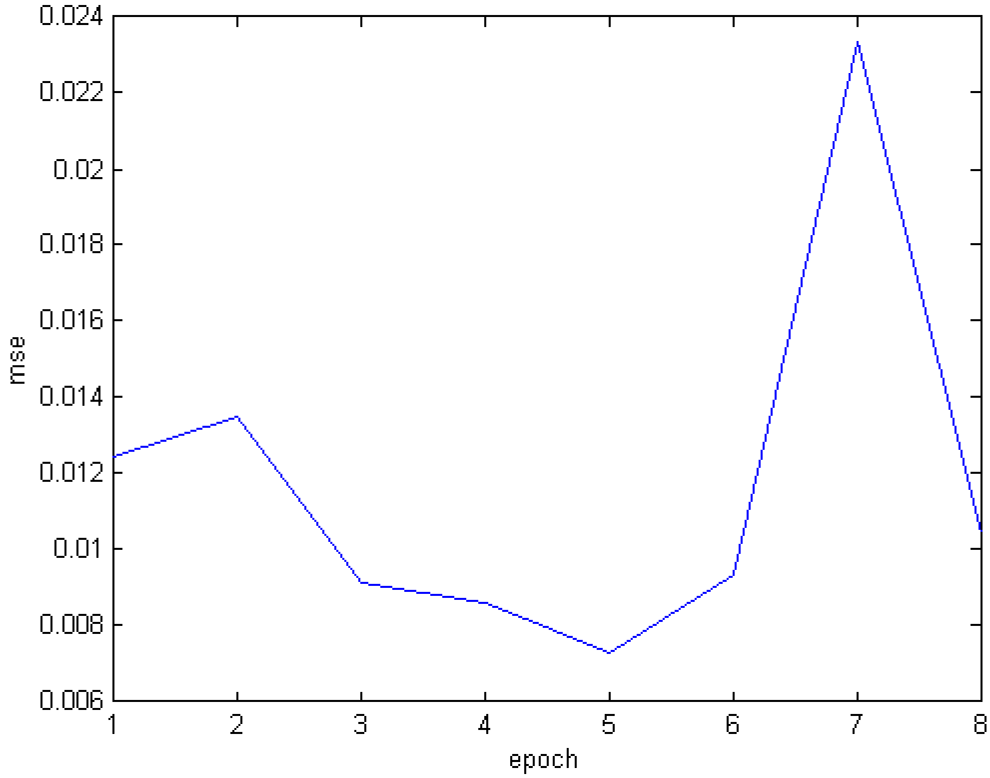


Fig 5.6 MSE for the result described in Fig 5.5

The overall MSE for the acoustic source-A is around 0.0118

5.4.4 Acoustic source- 'B'

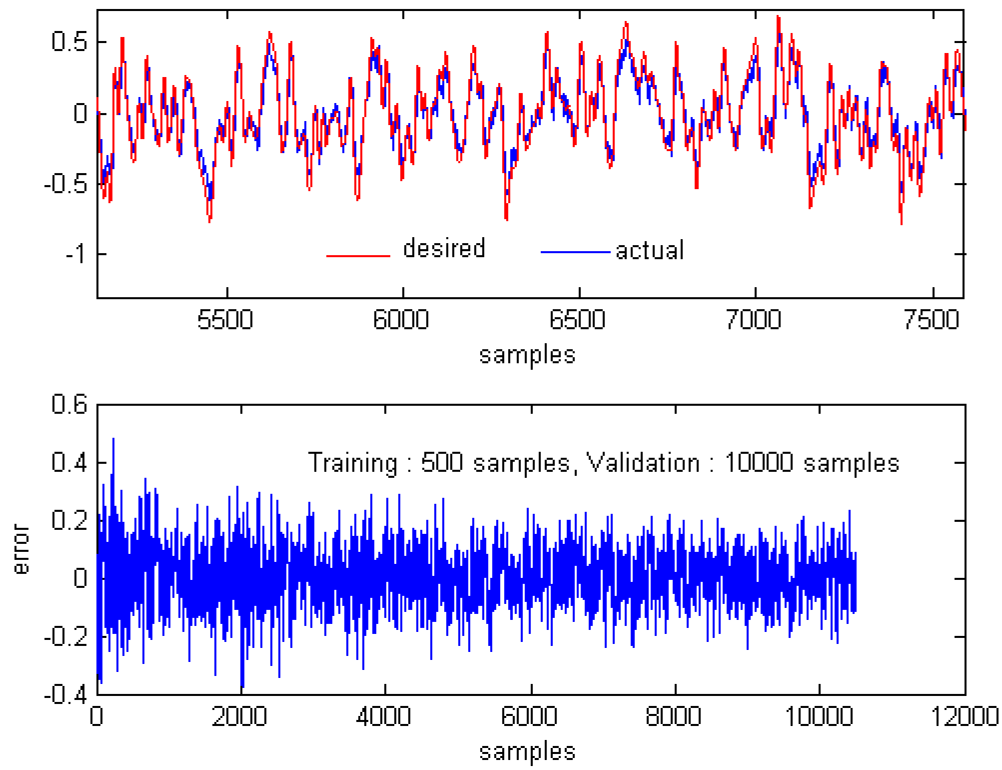


Fig.5.7 Superposition of model and desired output with the MLE algorithm and the error vector (data set-4)

In Fig 5.8, the MSE corresponding to this result is presented. It can be noticed that the MSE is coming down to around 0.005.

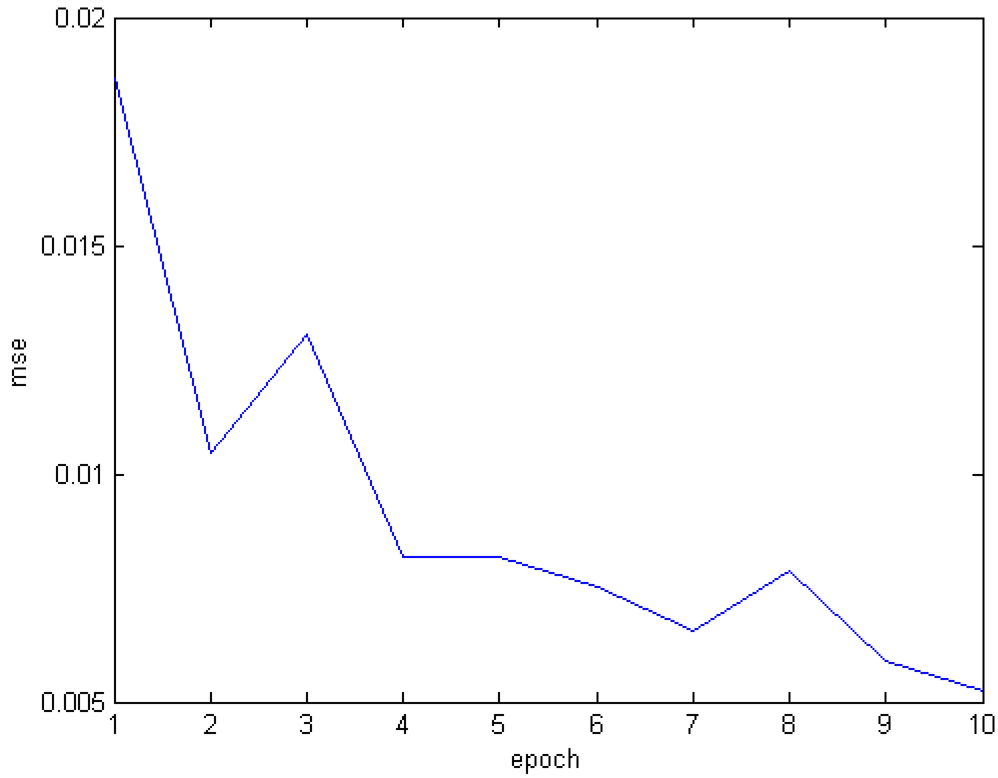


Fig 5.8 MSE for the result described in Fig 5.7

The overall MSE for the acoustic source-B is around 0.005.

5.5 System Identification using Conjugate – Gradient method

The conjugate gradient method belongs to a class of second order optimization methods known collectively as Conjugate direction methods [90].

Let $E_{av}(\mathbf{w})$ denote the cost function averaged over the training sample. Using Taylor series,

it is possible to expand $E_{av}(\mathbf{w})$ about the current point $\mathbf{w}(n)$ on the error surface,

considering the second order terms.

$$E_{av}(\mathbf{w}(n)+\Delta\,\mathbf{w}(n)) = E_{av}((\mathbf{w}(n)) + g(n)^T\,\Delta\,\mathbf{w}(n)$$

$$+1/2\,\Delta\,\mathbf{w}(n)^T H(n)\,\Delta\,\mathbf{w}(n) + \text{ third and higher order terms} \quad (5.15)$$

$$g(n) = \partial E_{av}(\mathbf{w}) / \partial \mathbf{w}\,|_{\mathbf{w}=\mathbf{w}(n)} \quad (5.16)$$

and $H(n)$ is the local Hessian matrix defined by

$$H(n) = \partial^2 E_{av}(\mathbf{w}) / \partial \mathbf{w}^2\,|_{\mathbf{w}=\mathbf{w}(n)} \quad (5.17)$$

For the Minimization of equation (4.42), third and higher order terms are neglected, differentiate w.r.t $\Delta\,\mathbf{w}(n)$ and equate to zero, to obtain

$$\Delta\,\mathbf{w}(n) = H(n)^{-1} g(n) \quad (5.18)$$

However the computation of $H(n)$ at every point $\mathbf{w}(n)$ is difficult. On the other hand, the unconstrained minimization of the quadratic error function E_{av} can be done using a set of A-conjugate vectors $(S_0, S_1, S_2 \ldots\ldots S_{(W-1)})$ is defined by

$$S_i^T A\, S_j = 0 \text{ for } i \neq j \tag{5.19}$$

The update equation is given by

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \zeta(n)\, S(n) \quad n=0,1,2\ldots\ldots W-1 \tag{5.20}$$

where $\mathbf{w}(0)$ is an arbitrary starting vector and $\zeta(n)$ is a scalar defined by

$$E_{av}\,(\mathbf{w}(n)+ \zeta(n)\, S(n)\,) = \min\ E_{av}\,(\mathbf{w}(n) + \zeta\, S(n)) \tag{5.21}$$

where ζ is obtained from a one dimensional minimization problem.

$$\zeta(n) = -\, S(n)^T A\, \epsilon(n) \,/\, S(n)^T A\, S(n): \qquad n=0,1,\ldots\ldots W-1 \tag{5.22}$$

where $\epsilon(n)$ is the error vector $w(n) - w^*$ (5.23)

But for the conjugate direction method to work, it requires the availability of a set of A conjugate vectors $(S(0), S(1) \ldots \ldots S(W-1))$, and the final position w^*, which is not available. So the following procedure is adopted

It is a special form of conjugate direction method. Here the successive direction vectors are generated as A-conjugate versions of the successive gradient vectors of the quadratic function E_{av} as the method progresses. Except for n=0, the set of direction vectors $\{S(n)\}$ is not specified beforehand but rather it is determined in a sequential manner at successive steps of the method.

Define residual as the steepest descent direction:

$$r(n) = g(n) - H(n)\, \Delta w(n) \tag{5.24}$$

Then to proceed $S(n)$ is taken as a linear combination of $r(n)$ and $S(n-1)$ as shown by,

$$S(n) = r(n) + \beta(n)\, S(n-1)\ ,\ n=1,2,\ldots w-1. \qquad (5.25)$$

where $\beta(n)$ is a scaling factor.

Similar to gradient direction method,

$$\beta(n) = -\, S(n-1)^T A\, r(n)\ /\ S(n-1)^T A\, S(n-1) \qquad (5.26)$$

Using equations for $S(n)$ and $\beta(n)$ it is possible to define vectors $S(0), S(1),\ldots S(n-1)$. But equations need knowledge of matrix A. So it is required to evaluate $\beta(n)$ without explicit knowledge of A.

The formula defining $\beta(n)$,

$$\beta(n) = r(n)^T\, (r(n) - r(n-1))/r(n-1)^T\, r(n-1) \qquad (5.27)$$

This is known as Polak-Rebiere formula. A summary of the algorithm is now presented next [90-92]

Initialization

Unless prior knowledge on the weight vector w is available, choose the initial value $w(0)$ as random.

Computation

1. For $w(0)$, compute the gradient vector $g(0)$.

2. Set $S(0) = r(0) = -g(0)$

3. At time step n, use a line search to find $\zeta(n)$ that minimizes $E_{av}(\zeta)$ sufficiently, representing the cost function E_{av} expressed as a function of ζ

for fixed values of w and S.

4. Test to determine if the Euclidean norm of the residual $r(n)$ has fallen below a specified value, that is a small fraction of the initial value $r(0)$.

5. Update the weight vector $w(n+1) = w(n) + \zeta(n)\, S(n)$.

6. For $w(n+1)$, use back propagation to compute the updated gradient vector $g(n+1)$.

7. Set $r(n+1) = - g(n+1)$

8. Use Polak-Ribiere method to calculate:

   $$\beta(n+1) = max\ \{\ [r(n+1)(\ r(n+1) - r(n)\ \ /\ \ r(n)^T\, r(n)],0\ \}$$

9. Update the direction vector $S(n+1) = r(n+1) + \beta(n+1)\, S(n)$

10. Set $n=n+1$ and go back to step 3.

<u>Stopping Criterion</u>

Terminate the algorithm when the following condition is satisfied.

$$|r(n)| <= \epsilon\, |r(0)|$$

where ϵ is a prescribed small number.

5.6 Performance analysis of MLE (Conjugate-Gradient)

The same nonlinear systems are modeled with MLE (Conjugate – gradient algorithm). The results are further used to get a conclusion on the model accuracy, its consistency and generalization capability.

5.6.1 Nonlinear system with output y = sin (t^2 + t)

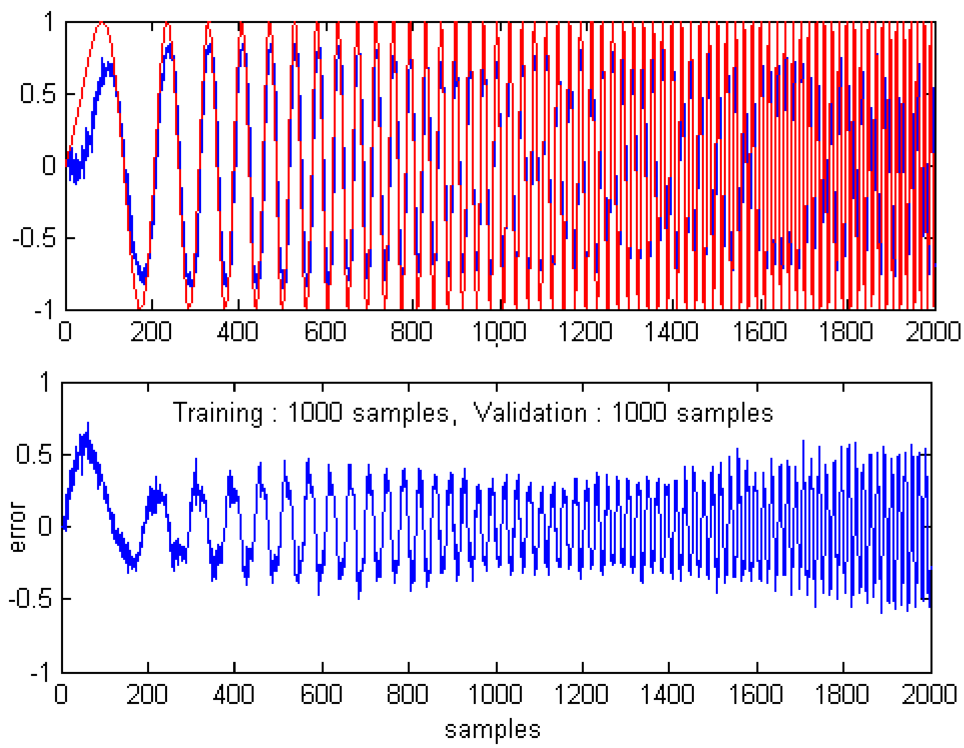


Fig.5.9 Superposition of model and desired output with MLE-CG algorithm and the error vector (data set-1)

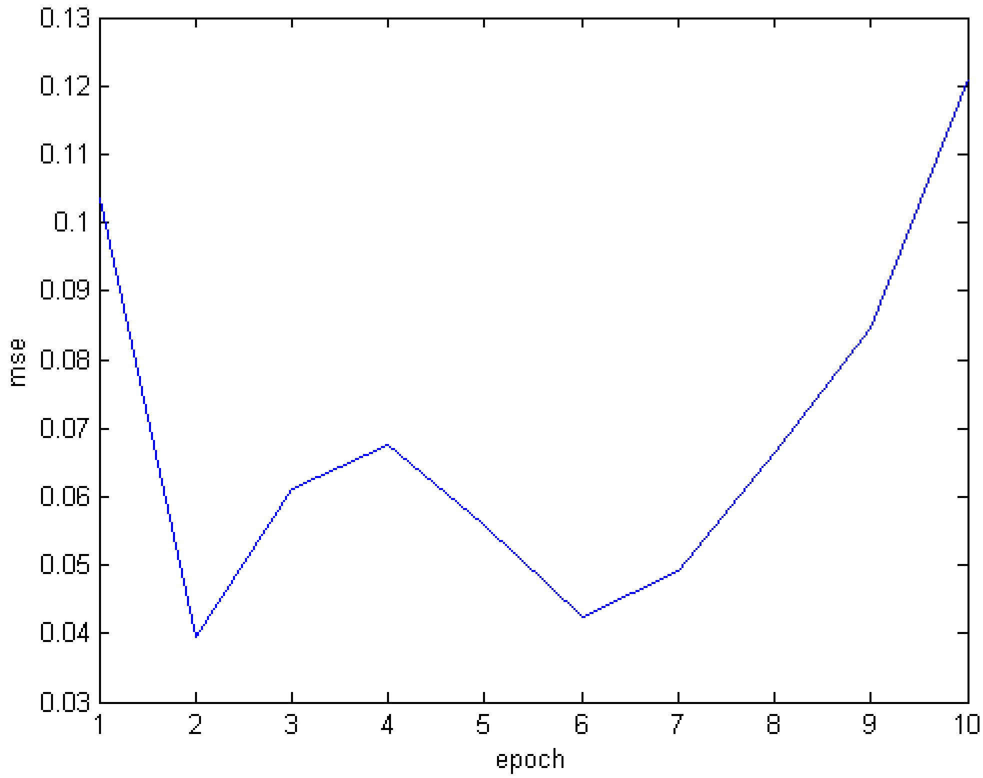


Fig 5.10 MSE for the result described in Fig 5.9

The error in MLE algorithm, MSE over the number of epochs is around 0.0825

5.6.2 Ambient noise in the sea

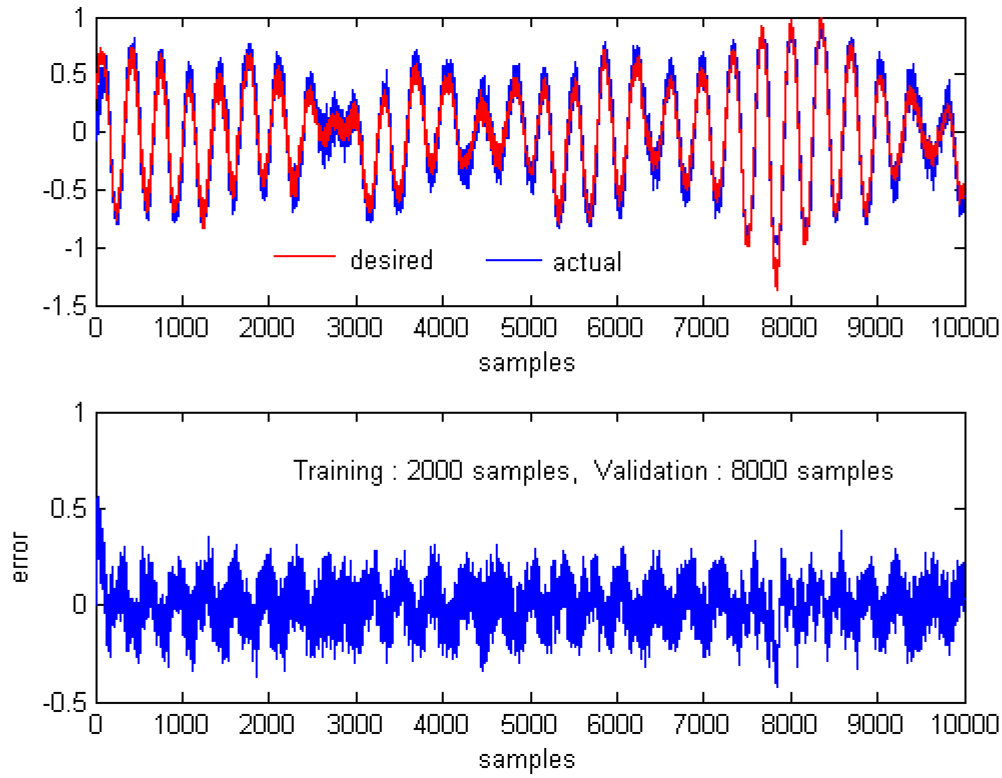


Fig.5.11Superposition of model and desired output with MLE-CG algorithm and the error vector (data set-2)

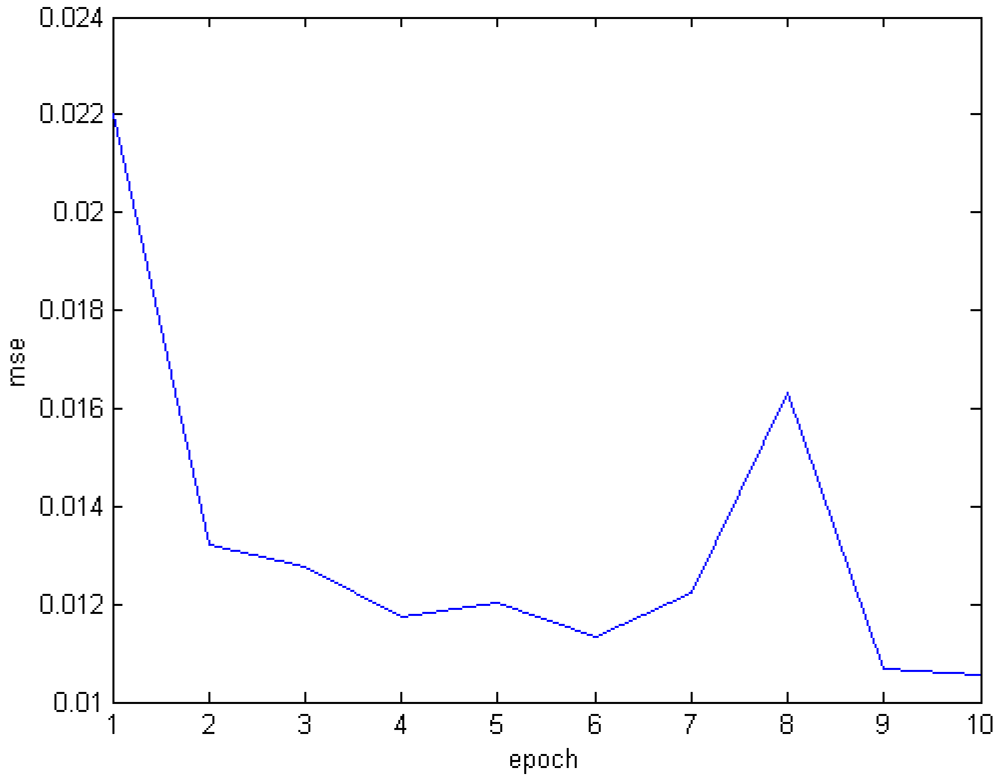


Fig 5.12 MSE for the result described in Fig 5.10

It can be observed that the validation error (MSE) is around 0.0122

5.6.3 Acoustic source- 'A'

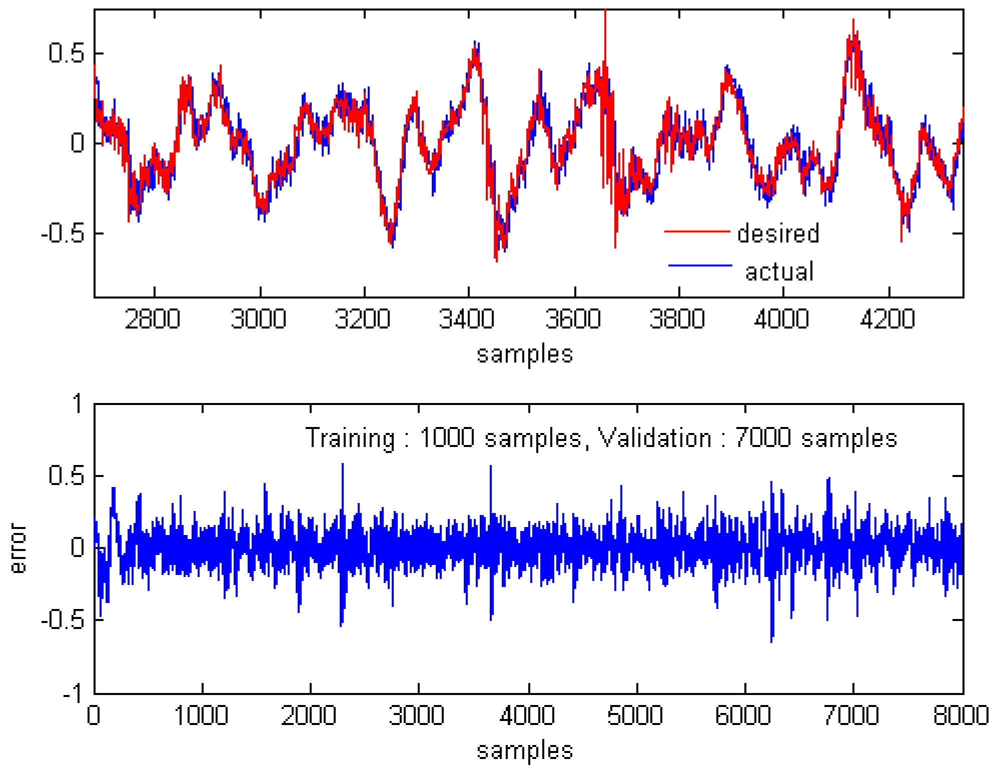


Fig.5.13 Superposition of model and desired output with MLE-CG algorithm and the error vector (data set-3)

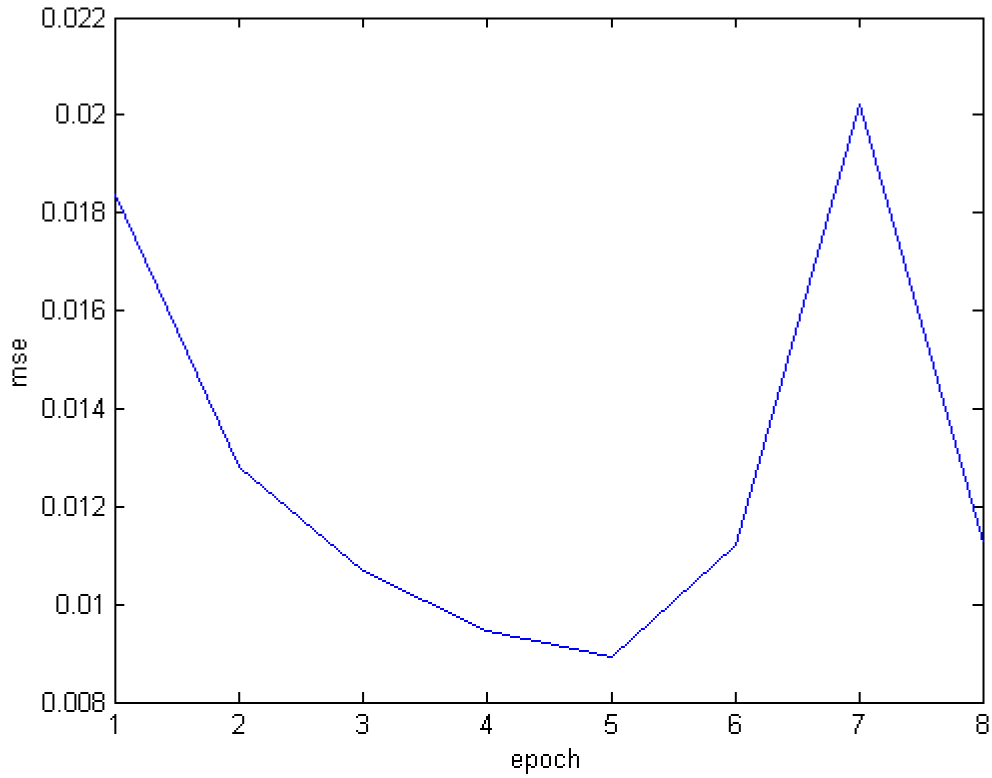


Fig 5.14 MSE for the result described in Fig 5.13

It can be observed that the validation error (MSE) is around 0.0122

5.6.4 Acoustic source- 'B'

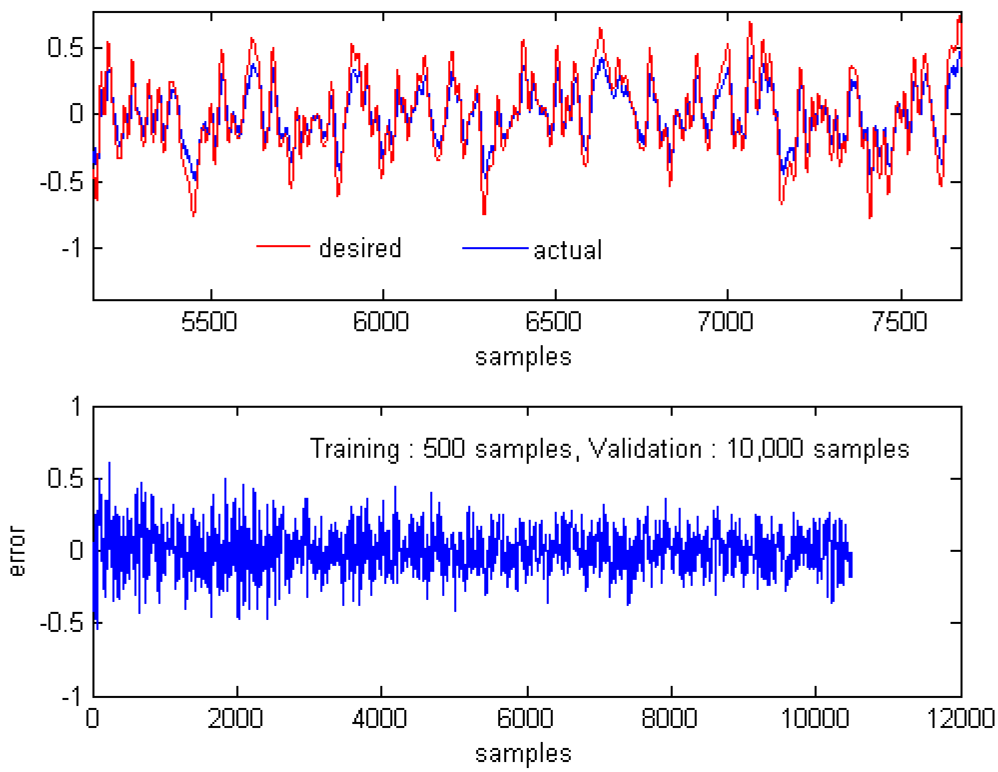


Fig.5.15 Superposition of model and desired output with MLE-CG algorithm and the error vector (data set-4)

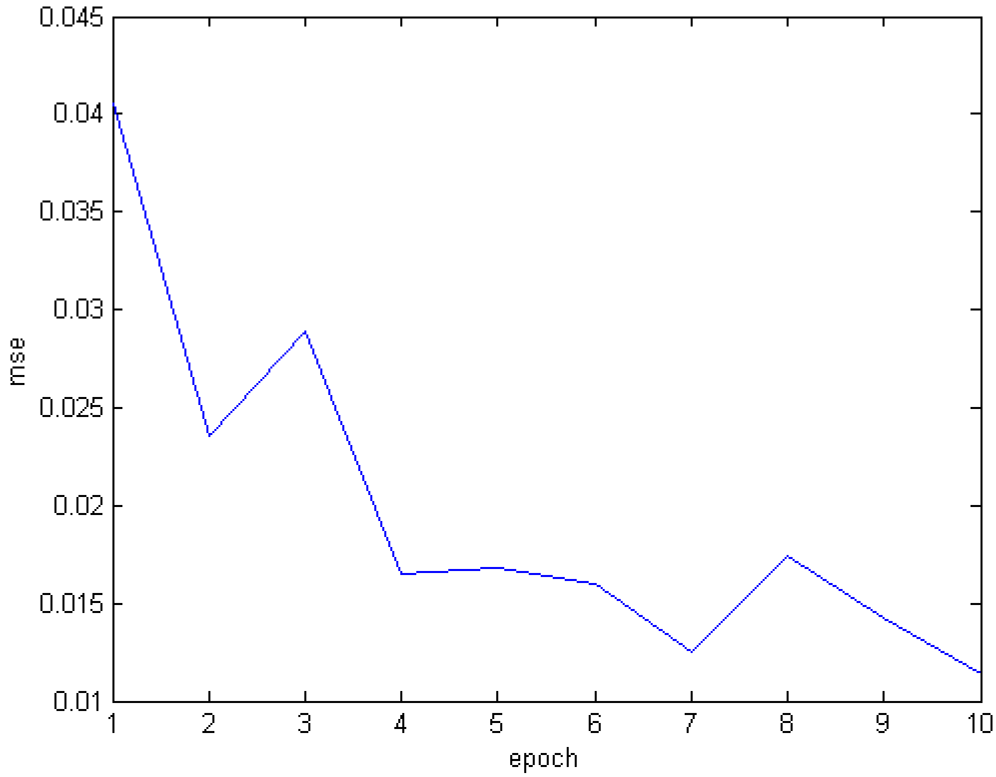


Fig 5.16 MSE for the result described in Fig 5.15

Fig 5.16 shows that the validation error (MSE) is around 0.0122

Thus all the systems have been modeled with Gauss-Newton and Conjugate gradient methods. A detailed performance measure has been provided with the plots of mean square error (MSE) in each case. The error trajectory and final values of MSE can be used for the performance assessment. A detailed comparison of the EKF and its variances with the current method of MLE is presented in the next section.

5.7 Comparison between the various MLE methods for modeling

Table 5.1 Performance comparison of EKF, EKF with EM and MLE in the MSE sense

System	Mean Square Error (MSE)			
	Simple EKF	EKF with EM	MLE: Gauss-Newton	MLE: Conjugate Gradient
$y=sin(t+t^2)$	0.0906	4.132×10^{-4}	0.0635	0.0825
Ambient noise	0.0045	8.06×10^{-4}	0.0083	0.0122
Acoustic source 'A'	0.0054	0.0065	0.0118	0.0121
Acoustic source 'B'	0.0038	0.000174	0.0092	0.0102

Table 5.1 shows the performance comparison of EKF and MLE. The EKF algorithm converges faster and has marginally good performance compared to BP algorithm and MLE. It is also consistent for all the nonlinear systems modeled. The performance of EKF can be again improved by EM algorithm as shown in the table. The MLE algorithm also gives good results and

computationally efficient but in problems where faster convergence is required, as in adaptive filters and real world problems; Kalman Estimation has to be used.

5.8 Conclusion

In an attempt to compare the performance of the EKF based algorithms for estimating the network parameters, Maximum Likelihood Estimation (MLE) for modeling nonlinear systems has been implemented and the performance results are compared. The same Feed forward neural network is used as the model structure. Four nonlinear systems are modeled using two different techniques of implementing the MLE viz. the Gauss- Newton method and Conjugate gradient methods. The results show good performance of the estimation technique in respect of MSE. A comparison is also made among all the different methods tried out viz. the EKF algorithm, the EKF algorithm with EM, the Gauss-Newton method and the Conjugate gradient method. It is seen that the performance of MLE is good but inferior to that of EKF in terms of mean square error. It could be due to the variances in the choice of the parameters like β in the Conjugate gradient descent algorithm. On the other hand, the EKF algorithm and its variants seem to be performing very well with minimum dependence on the choice of parameters.

Chapter 6

NONLINEAR SYSTEM MODELING USING PARTICLE FILTER

Chapter 6 introduces the application of Particle Filter as a new approach for nonlinear system modeling/identification. The results of applying the concept to nonlinear modeling and the state space analysis of the systems so modeled are also presented.

6.1 Introduction

The system identification/modeling problem looks for a suitably parameterized model, representing a given process. The parameters of the model are adjusted to optimize a performance function based on error between the given process output and identified process output. The capability of the Artificial Neural Networks in modeling non linear systems was demonstrated in the previous chapters, using the Extended Kalman Filter, as an effective tool in the estimation of the weights of the neural network. Nonlinear filtering process can be generally viewed as recursively estimating, based on a set of noisy observations, at least the first two moments of the state vector governed by a dynamic nonlinear non-Gaussian state space model (DSS) [34]. A discrete time DSS consists of a stochastic propagation (prediction or dynamics) equation which links the current state vector to the prior state vector and a stochastic observation equation that links the observation to the current state vector. However the capability of EKF in handling the non Gaussianity , resulting from the non linear transformations was always a matter of concern for the research community. Recent results reported on the Particle Filters (PF) [132-140] appear to be offering an effective solution in alleviating the problems due to the non Gaussianity. The present chapter therefore addresses the performance of modeling approach using the particle filters in the context of non linear system identification.

Recently several new approaches to recursive nonlinear filtering have appeared in literature. Particle filters are suboptimal filters belonging to this category of methods. They perform sequential Monte Carlo (SMC) estimation based on point mass (or "particle") representation of probability densities [143-144]. The SMC ideas in the form of sequential importance sampling had been introduced in Statistics back in the 1950s. Although these ideas continued to be explored sporadically during the 1960s and 1970s, they were largely overlooked and ignored. The most likely reason for this was the modest computational power available at the time. Since then research activity in the field has dramatically increased, resulting in many improvements of particle filters and their numerous applications [136-140]. In addition, all these early implementations were based on plain sequential importance sampling, which as shall be described later, degenerates over time. The major contribution to the improvement of the SMC method with the inclusion of the re-sampling step, coupled with ever faster computers, made the particle filters useful in nonlinear modeling for the first time.

6.2 Nonlinear Estimation using Particle Filter

A nonlinear stochastic system can be defined by a stochastic discrete-time state space transition (dynamic) equation [139]

$$x_n = f_n(x_{n-1}, w_{n-1}) \qquad (6.1)$$

and the stochastic observation (measurement) process

$$y_n = h_n(x_n, v_n) \qquad (6.2)$$

where at time t_n, x_n is the (usually hidden or not observable) system state vector, w_n is the dynamic noise vector, y_n is the real observation vector and v_n is the observation noise vector. The deterministic functions f_n and h_n link the prior state to the current state and the current state to the observation vector, respectively.

In a Bayesian context, the problem is to quantify the posterior density, $p(x_n \mid y_{1:n-1})$ where the observations are specified by $y_{1:n} = \{y_1, y_2, ...y_n\}$. The above nonlinear non-Gaussian state space model, Eq.6.1, specifies the predictive conditional transition density, $p(x_n \mid x_{n-1}, y_{1:n-1})$, of the current state given the previous state and all previous observations. Also observation process equation, Ep.6.2, specifies the likelihood function of the current observation given the current state, $p(y_n \mid x_n)$. The prior probability $p(x_n \mid y_{1:n-1})$ is defined by Bayes' rule as,

$$p(x_n \mid y_{1:n-1}) = \int p(x_n \mid x_{n-1}, y_{1:n-1})\, p(x_{n-1} \mid y_{1:n-1})\, dx_{n-1} \qquad (6.3)$$

Here, the previous posterior density is identified as $p(x_{n-1} \mid y_{1:n-1})$. The correction step generates the posterior probability density function from

$$p(x_n \mid y_{1:n}) = c\,p(y_n \mid x_n)\, p(x_n \mid y_{1:n-1}), \qquad (6.4)$$

where c is normalization constant. The filtering problem is to estimate, in a recursive manner, the first two moments of x_n given $y_{1:n.}$ For general distribution *p(x),* this consists of the recursive estimation of the expected value of any function of *x*, says $\langle g(x) \rangle_{p(x)}$, using Eq.6.3 and 6.4 together requires calculation of integral of the form

$$\langle g(x) \rangle_{p(x)} = \int g(x)\, p(x)\, dx \qquad (6.5)$$

These integrals in many cases will be evaluated using some form of integration approximations like the Monte Carlo method [143-144]. In cases where $p(x_n \mid y_{1:n-1})$ is multivariate and non standard or multimodal, it may be difficult to generate the samples from $p(x_n \mid y_{1:n-1})$. To overcome this difficulty the principle of Importance Sampling is utilized. Suppose $p(x_n \mid y_{1:n-1})$ is a PDF from which it is difficult to draw samples. Also suppose that $q(x_n \mid y_{1:n-1})$ is another PDF from which sample can be easily drawn (referred to as the Importance Density) [22] [23]. One can now write, $p(x_n \mid y_{1:n-1}) \alpha q(x_n \mid y_{1:n-1})$

, where the symbol ∝ means that $p(x_n \mid y_{1:n-1})$ is proportional to $q(x_n \mid y_{1:n-1})$ at every x_n. It becomes necessary to evaluate integrals of the type,

$$\int p(x_n \mid x_{n-1}, y_{1:n}) p(x_{n-1} \mid y_{1:n-1}) dx_{n-1} = \int g(x_n) p(x_n) dx_n$$

$$= \langle g(x_n) \rangle p(x_n) \qquad (6.6)$$

On defining,

$$w(x_n) = \frac{p(x_n \mid y_{1:n})}{q(x_n \mid y_{1:n})} \text{ and } w(x_n^{(i)} = \frac{w(x_n^{(i)})}{\sum_{j=1}^{N_s} x_n^{(i)}} \qquad (6.7)$$

$\langle g(x_n) \rangle p(x_n) \mid y_{1:n-1}$ becomes, $\langle g(x_n) \rangle p(x_n \mid y_{1:n-1}) = \sum_{i=1}^{N_s} g(x_n^{(i)}) w(x_n^{(i)})$, (6.8)

where the N_s samples { $x_n^{(i)}$, $i = 1 \ldots N_s$ } are generated.

Using Eq.6.4 above, it is possible to re-cast $w(x_n)$ as,

$$w(x_n) = \frac{c p(y_n \mid x_n) p(x_n \mid y_{1:n-1})}{q(x_n \mid y_{1:n-1})} \qquad (6.9)$$

$$= \frac{c p(y_n \mid x_n) \int p(x_n \mid x_{n-1}, y_{1:n-1}) dx_{n-1}}{\int q(x_n \mid x_{n-1}, y_{1:n}) q(x_{n-1} \mid y_{1:n-1}) dx_{n-1}} \qquad (6.10)$$

$$w(x_n) = \frac{cp(x_n \mid y_n)\int p(x_n \mid x_{n-1}, y_{1:n-1})\, p(x_{n-1} \mid y_{1:n-1}) dx_n}{\int q(x_n \mid x_{n-1}, y_{1:n-1}) q(x_{n-1} \mid y_{1:n-1}) dx_{n-1}} \tag{6.11}$$

When Monte Carlo samples are drawn from the importance density, this leads to a recursive formulation for the importance weights.

Importance sampling is a general MC integration method that is now applied to perform nonlinear filtering specified by the conceptual solution. The resulting sequential importance sampling (SIS) algorithm is a Monte Carlo method that forms the basis for most sequential MC filters developed over the past decades; this sequential Monte Carlo approach is known variously as bootstrap filtering, the condensation algorithm, particle filtering, interacting particle approximation, and survival of the fittest. It is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these examples and weights. As the number of samples becomes very large, this Monte Carlo characterization becomes an equivalent representation to the usual functional description of the posterior PDF, and the SIS filter approaches the optimal Bayesian estimator [126, 142].

6.3 The Particle Filter algorithm

In the case of sequential importance sampling, each iteration will have the random measure $\{x_{n-1}^{(i)}|_{n-1}, w_{n-1}^{(i)}\}_{i=1}^{N_s}$, drawn from $q(x_{n-1} \mid y_{1:n-1})$, but constituting an approximation to $p(x_{n-1} \mid y_{1:n-1})$. The aim is to then find out a set of new samples and weights to approximate $p(x_n \mid y_n)$.

$$w_n^{(i)} = \frac{p(y_n \mid x_{n|n-1}^{(i)})\, p(x_{n|n-1}^{(i)} \mid x_{n-1|n-1}^{(i)})}{q(x_{n|n-1}^{(i)} \mid x_{n-1|n-1}^{(i)})} \times \frac{p(x_{n-1|n-1}^{(i)})}{q(x_{n-1|n-1}^{(i)})} \qquad (6.12)$$

$$w_n^{(i)} = \frac{p(y_n \mid x_{n|n-1}^{(i)})\, p(x_{n|n-1}^{(i)} \mid x_{n-1|n-1}^{(i)})}{q(x_{n|n-1}^{(i)} \mid x_{n-1|n-1}^{(i)})}\; w_{n-1}^{(i)} \qquad (6.13)$$

The boot strap approximation leads to the assumption that p = q, when

$w_n^{(i)} = p(y_n \mid x^{(i)}_{n|n-1})\ w^{(i)}_{n-1}$, which also works as a SIS.

Assuming that $p(y_n \mid x^{(i)}_{n|n-1})$ can be approximated by $N(y_n, f(x^{(i)}_{n|n-1})$, the SIS algorithms for modeling can be executed as shown in the block diagram in Fig. 6.1below (***f***: the model (or measurement) function):

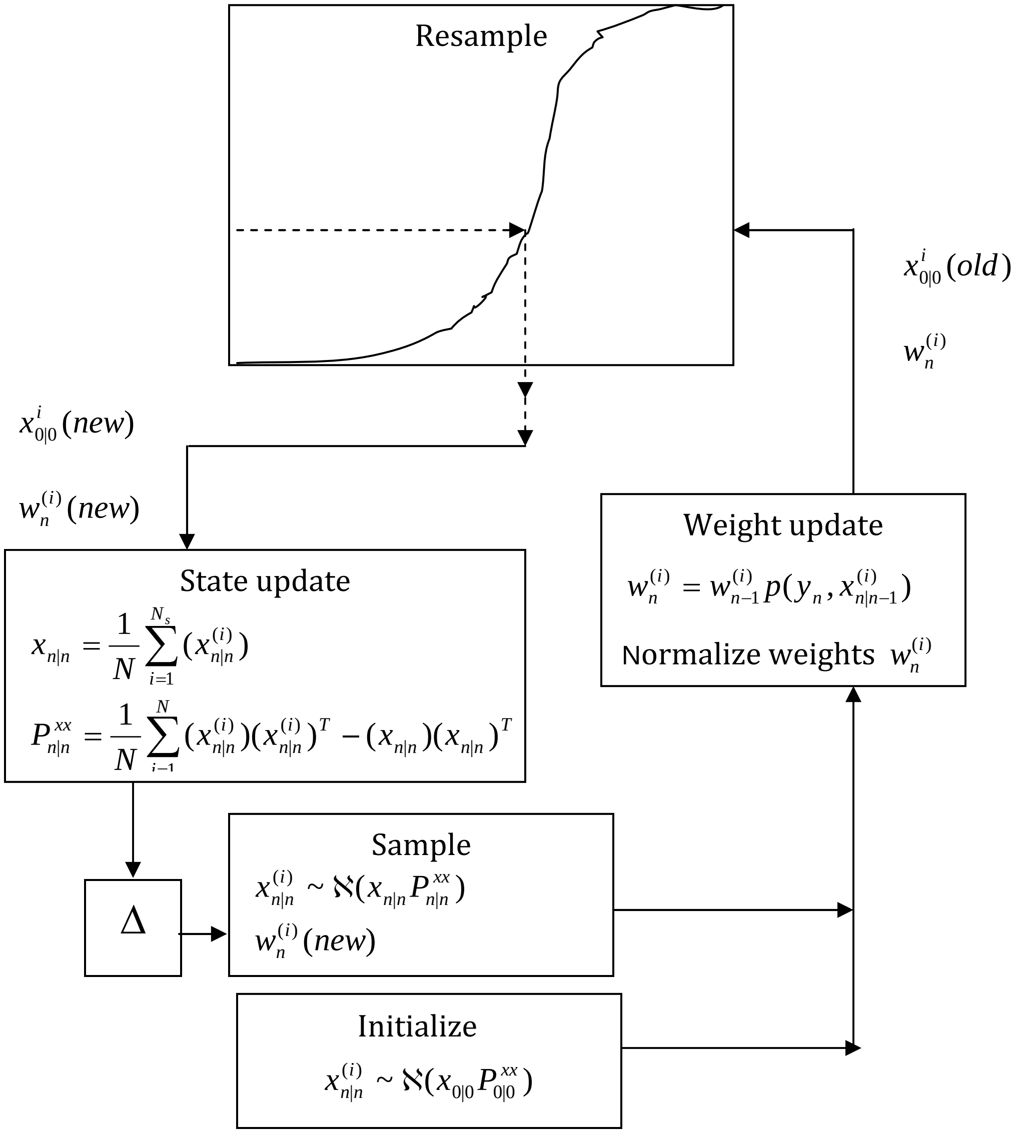


Fig 6.1 Block diagram of Sequential Importance Sampling (SIS)

6.4 Performance analysis using particle filter

The PF algorithm, with SIR version was applied to the neural network model for the same systems used in EKF and the results are presented below. As has been done in EKF, the plant & model outputs and the error vectors are presented. It can be seen that the algorithm converges very fast.

6.4.1 Results of nonlinear system, $y = \sin(t^2 + t)$

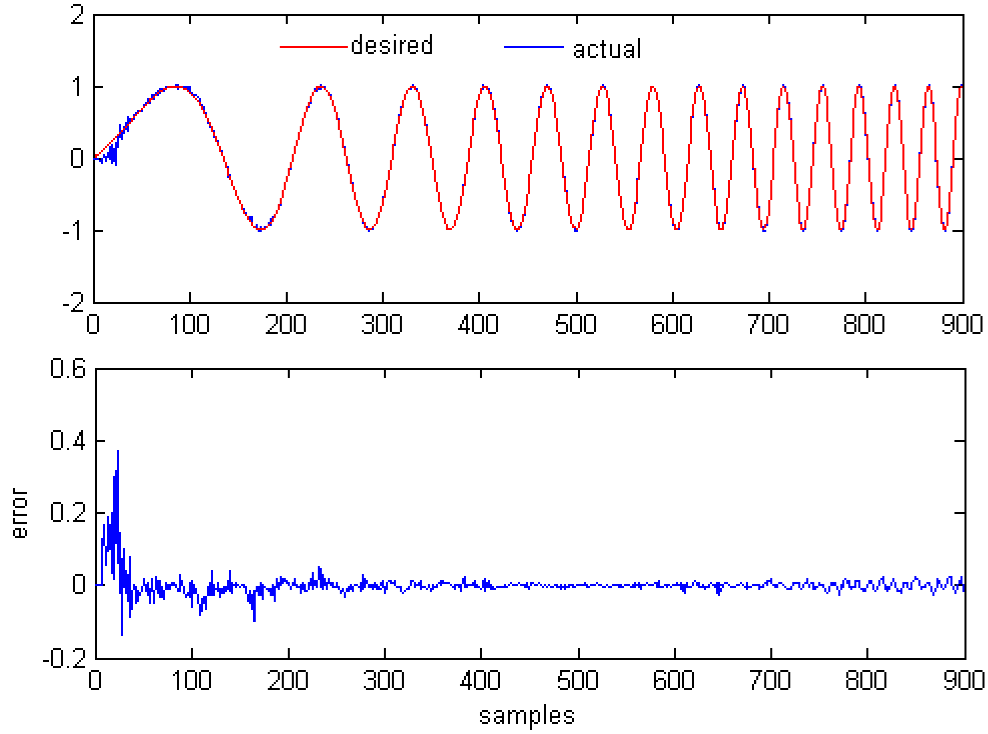


Fig. 6.2 Output Vs data samples for data set-1

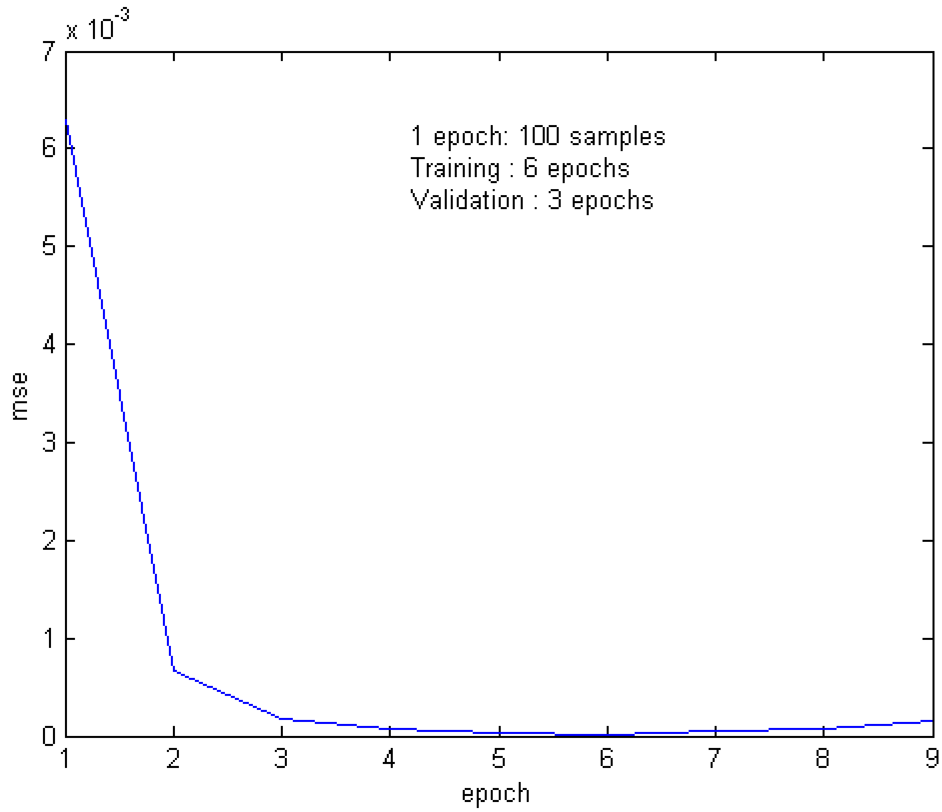


Fig 6.3 MSE Versus number of epochs (MSE =3.789×10^{-4})

It can be noticed that for the data set-1, sin $(t+t^2)$ the MSE on convergence is less than 3.789×10^{-4} with a set of 300 training samples, 300 validation with one epoch of 50 samples.

6.4.2 Results of ambient noise in the sea

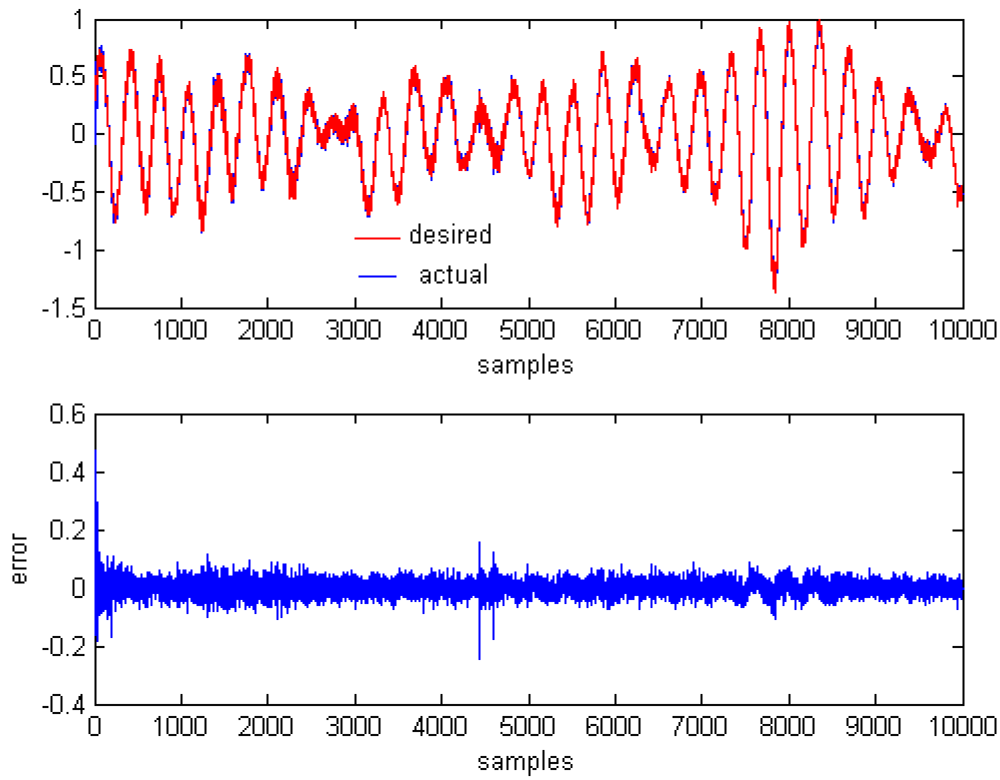


Fig 6.4 Output Vs data samples for data set-2

Here also the performance of PF model is comparable with techniques like EKF and the convergence is also noticed to be faster and excellent.

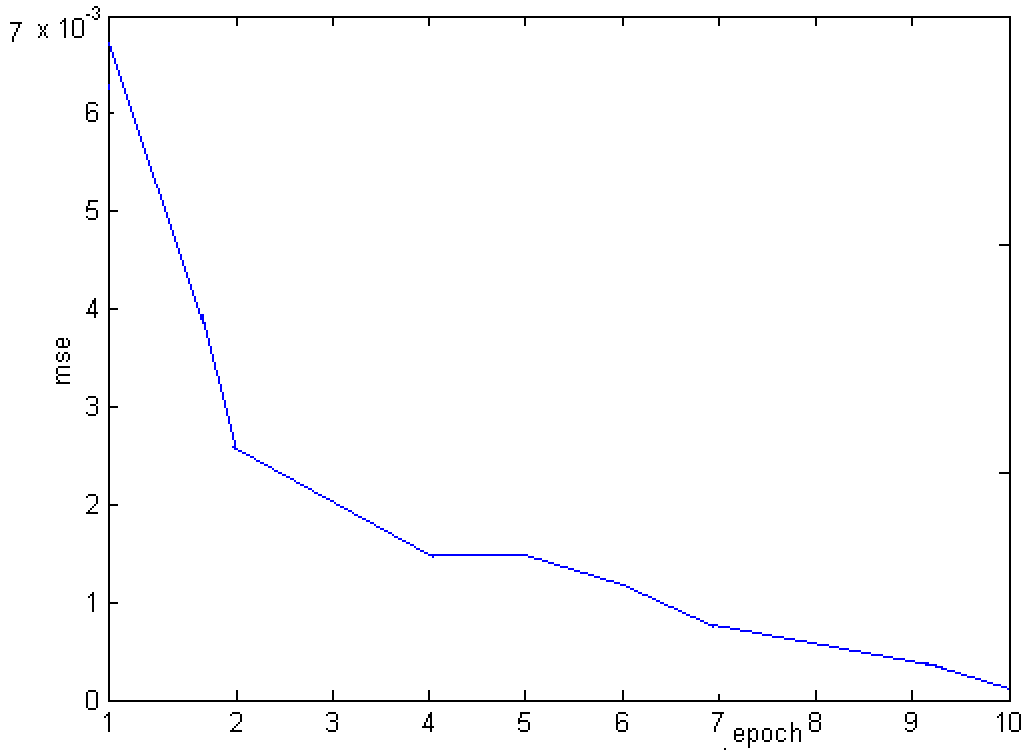


Fig 6.5 MSE Versus number of epochs (MSE=5.321× 10^{-4})

The mean square error obtained with the PF approach is within an acceptable level and is comparable with the MSE performances obtained with EKF. It could be notice that error is coming down to 3.789 × 10^{-4}, which may be contrasted with the MSE of 5.808×10^{-4}, obtained for the same model while using the EKF. In order to fully assess the capability of the particle filter, the state space modeling is also carried out and the results so obtained are also been presented, in chapter 7 in detail.

6.5 Conclusions

A given higher order system can be analyzed by modeling it with minimum of 2 or 3 states. Here the systems are modeled with three states. Both the system states $\boldsymbol{x_k}$ and the set of model parameters $\boldsymbol{w}$ for a dynamic system are simultaneously estimated from the observed noisy signal $\boldsymbol{y_k}$ only.

The approach of the particle filtering is found equally competent with EKF giving almost comparable MSE performance in the RNN models for nonlinear system study.

As an extension of the current work, performance of RNN training algorithms with PF can be compared with variations of EKF such as, EKF with Expectation Maximization, Maximum Likelihood estimation, Unscented Kalman Filtering etc. It is also possible to compare modeling approaches in terms of various measures including the Cramer Rao Lower Bound (CRLB), The Lyapunov exponent's method etc. This will finally enable to select the most efficient and optimum RNN model and training scheme for a given nonlinear data or system. Investigation is also possible to arrive at the optimum number of states to be used for representing a given nonlinear system, and the behavioral study using other available techniques.

Chapter 7

STATE SPACE MODELING USING RECURRENT NEURAL NETWORKS

Chapter 7 addresses the problem of state space modeling with RNN. The combined state and parameter estimation is demonstrated here. The phase plots so developed give an insight on the dynamics of the system under consideration. The Lyapunov exponent is also computed for the systems as a measure of their chaotic behavior. As a practical application the analysis of a set of arrhythmia data is also presented.

7.1 Introduction

Recurrent Neural Networks (RNN) forms a wider class of neural networks, as they allow feedback connection between neurons making them dynamic systems. In order to proceed with the study of neuro dynamics, one needs a mathematical model for describing the dynamics of the system. A model most naturally suits for this purpose is the state space model and it can be implemented efficiently in RNN. They are neural networks with one or more feedbacks. Since this network has the feedback structure, it embodies short term memory and has powerful representation capability for modeling many complex nonlinear systems. Thus they are different from feed forward architecture in the sense that they not only operate on an input space but also on the internal state space. Implementing a feed forward network is just a static mapping of the input vectors. In order to model a high dimensional nonlinear dynamical system, it is essential to create a neural network which is capable of storing internal state and thus implementing complex dynamics [31, 57, 79-91].

7.2 System identification using RNN

In a recurrent network, the state of the system can be encoded in the activity pattern of the units and a wide variety of dynamical behavior can be programmed by the connection weights. A simple recurrent network has activation feedback, in which a state layer is updated not only with the external input of the network but also with the activation from the previous forward propagation.

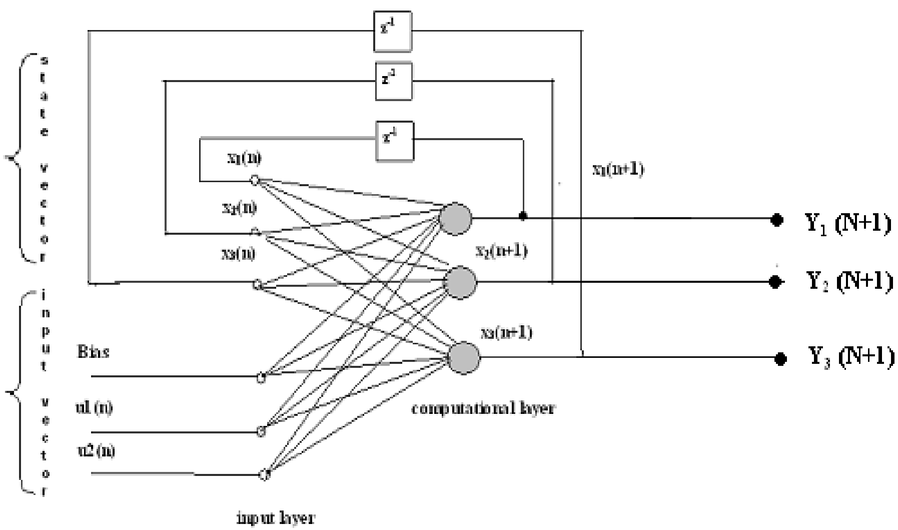


Fig. 7.1 Single layer Recurrent Neural Network

In the neural network, there could be one or more hidden layers, whose computation nodes are correspondingly called hidden neurons or hidden units. The function of hidden neurons is to intervene between the external input and network output in some useful manner. By adding one or more hidden layers the network is enabled to extract higher order statistics.

The major difference in applying parameter based EKF algorithm for the training of weights in feed forward and RNN architecture lies in the computation of ordered derivatives of network output with respect to weight vector w_k. Once the derivatives are computed, the same parameter based EKF algorithm applies to either class of network architecture. The advantage of RNN is that the system states can be obtained by the simulation of these networks, which helps in construction of phase space [98-127].

7.3 Combined State and Parameter Estimation

The state of a dynamical system is formally defined as a set of quantities that summarizes all the information about the past behavior of the system that is needed to uniquely describe its future behavior, except for the purely external effects arising from the applied input [90].

In many control problems, the objective is to feedback the states of the system in order to modify its behavior. Hence it is necessary to estimate the states of the system from the measurements, which are contaminated with noise.

The problem of combined parameter and state estimation is treated as a nonlinear estimation problem, by augmenting the state vector with the parameter vector. Kalman filter based training algorithm for recurrent neural network, has been found to be very efficient in modeling and estimation. But it is not taking into account the optimization of hidden state variables of the recurrent network. Also their formulation requires Jacobean evaluation of the entire network, adding to additional computational complexity. A new algorithm is suggested which reduces computational complexity of Jacobean evaluation by decoupling the gradients of each layer, at the same time calculating the internal hidden states.

7.4 RNN Training using EKF Algorithm

In the modified Kalman algorithm the state and measurement equations are described as follows.

$$w_{k+1} = w_k \qquad (7.1)$$

$$x_{k+1} = f(x_k, w_k, u_k) \qquad (7.2)$$

$$y_k = h(x_k, w_k) + v_k \qquad (7.3)$$

Considering the parameter optimization as a state estimation, as described above, allows us to use the extended Kalman filter to update the weight estimates as well as the optimal hidden states[79-81].

The algorithm is formally explained as below:

The state of the systems is augmented to contain the n parameters w and m states.

$$\mathbf{X} = [w_1, w_2, \ldots w_n, x_1, x_2, \ldots x_m]^T \qquad (7.4)$$

Initialize all the weights and states to small random values. Initialize state covariance matrix ***P*** to diagonal with relatively small values. Let the covariance matrix for measurement noise is ***R*** and that of process noise is ***Q***. As usual compute the output at each node of the recurrent network. Find the Jacobean matrix with respect to the state of the process ***f(.)*** and output ***h(.)*** equation at the current estimate of internal state and weights of the RNN. These matrices are given by ***A*** and ***C***.

$$A = \begin{bmatrix} I \cdots & 0 \\ 0 \cdots & \partial f(.)/\partial x \end{bmatrix} \qquad (7.5)$$

$$C = \begin{bmatrix} \partial h/\partial w & \ldots\ldots\ \ldots & \partial h/\partial x \end{bmatrix} \qquad (7.6)$$

The error e_k for the new training sample is evaluated, and the Kalman gain matrix is computed using Eqn. (7.9)

$$e_k = y_k - h(x_k, w_k, u_k) \qquad (7.7)$$

$$P_{k/k} = APA^T \qquad (7.8)$$

$$K = P_{k/k-1} C^T (C P_{k/k-1} C^T + R)^{-1} \qquad (7.9)$$

$$X_{k/k} = X_{k/k-1} + K e_k \qquad (7.10)$$

$$P_{k+1/k} = (1 - KC) P_{k/k-1} + R \qquad (7.11)$$

The estimates for the optimal weights and the internal state of the RNN given by X are updated using the Kalman gain matrix and the current output error (Eqn. 7.10). The process is repeated with the subsequent training sample. Incorporating the hidden state variable of the recurrent network into the state vector of Kalman filter allows the decomposition of the network into layers. Therefore Jacobian calculations can be carried out in each layer independently from all other layers. The recursive nature of Kalman filter process equation takes care of time recurrent nature of the gradients eliminating the requirement to back propagate the gradient through time. Therefore this new approach provides an unfolding of the recurrent network in time as well as layers [121].

7.5 Performance analysis

Recurrent Neural Network (RNN) is used as the model structure and algorithm used for the training is Extended Kalman Filter (EKF). The same set of nonlinear systems modeled using feed forward network is used here and the performance is analyzed in terms of mean square error. The dynamics of the states of the systems also evaluated using recurrent network. The state space analysis is done for the same four nonlinear systems.

7.5.1 Nonlinear system with output y = sin (t^2 + t)

The phase plots generally provide an insight into the dynamical behavior of the system. They are extensively used in understanding the system behavior including chaotic behaviors and stability characteristics. The capability of the phase plot, generated out of the model built around the neural networks, in presenting the evolutionary dynamics of the systems is also very well brought out in this chapter.

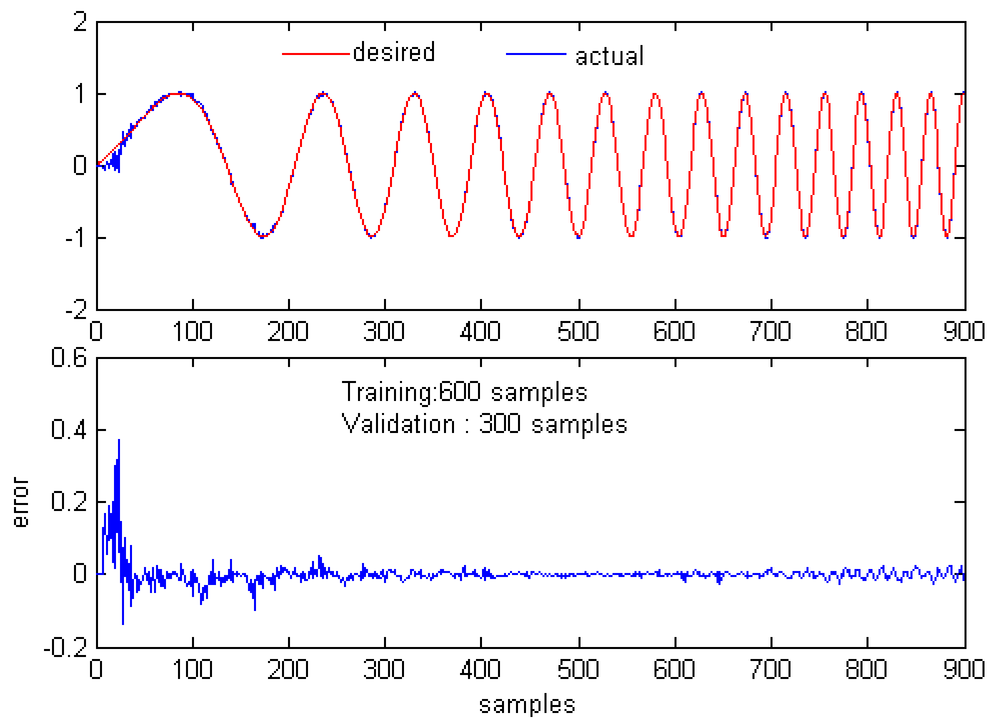


Fig 7.2 Superposition of model output and the actual data (data set-1) and the error

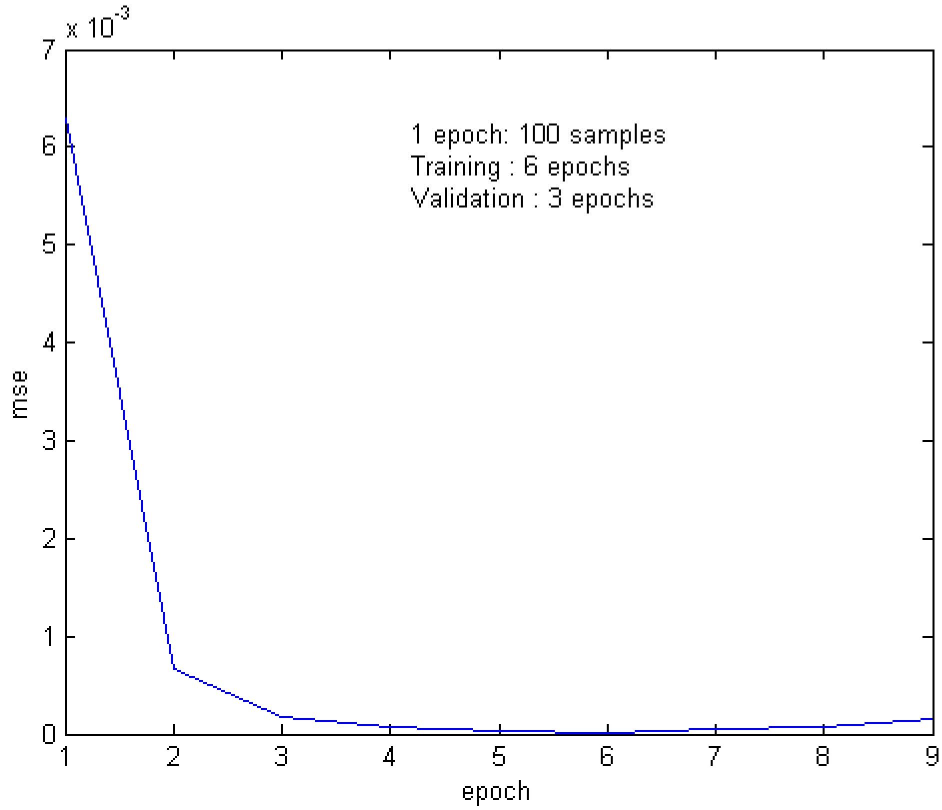


Fig 7.3 MSE Vs data samples

These results are actually taken from chapter-4, where the modeling with EKF has been discussed in detail. With this in hand, the detailed phase plot analysis is carried out and the result is presented in Fig 7.4.

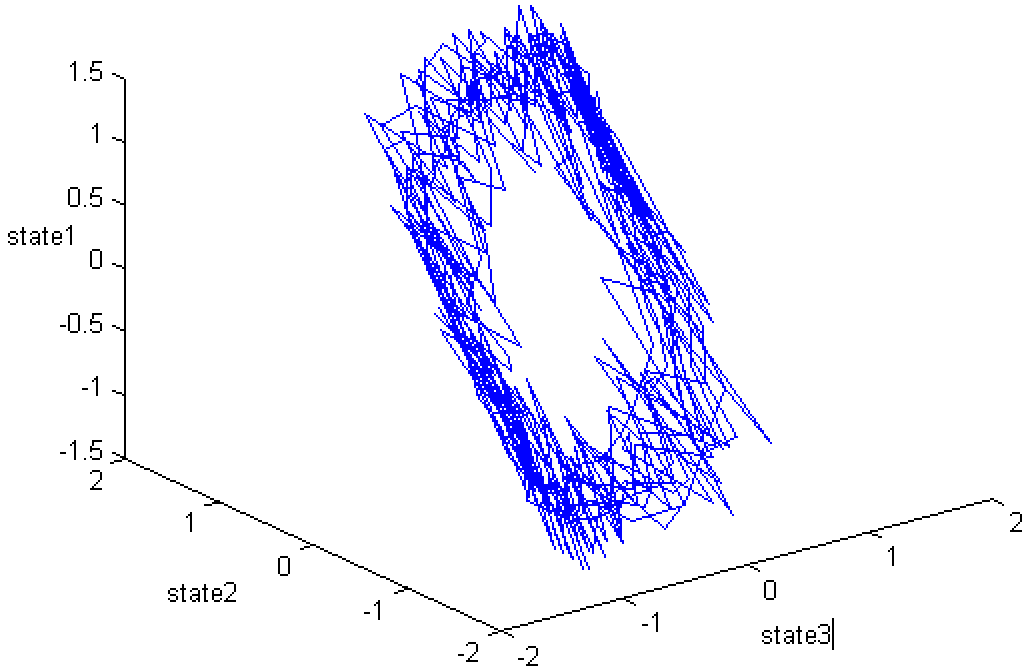


Fig 7.4 The phase plot corresponds to $y = sin\ (t^2 + t)$

It can be seen that the system phase plot is constrained into the shape described above and has not seemed to be highly fluctuating/chaotic system in its behavior. Later in this chapter, the Lyapunov exponent approach is further utilized to get better insight on the system dynamics.

7.5.2 Ambient noise in the sea

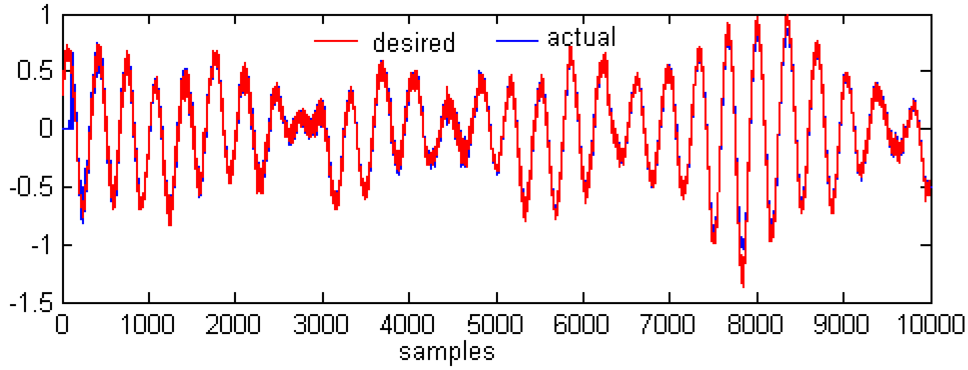


Fig.7.5 Superposition of model output and the actual data (data set-1)

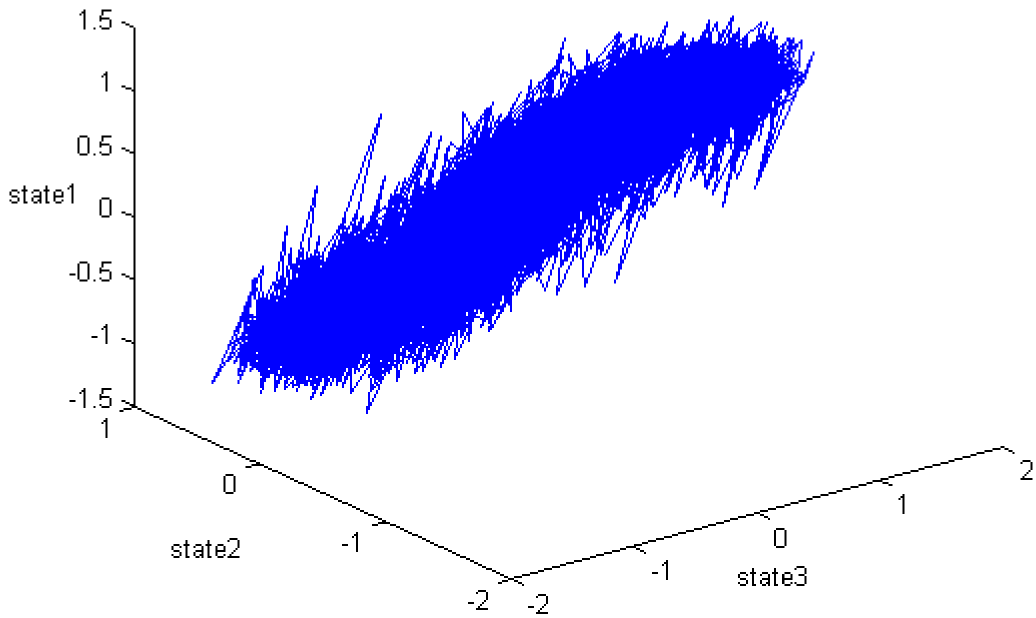


Fig.7.6 Phase plot corresponds to ambient noise in the sea

7.5.3. Acoustic source- 'A'

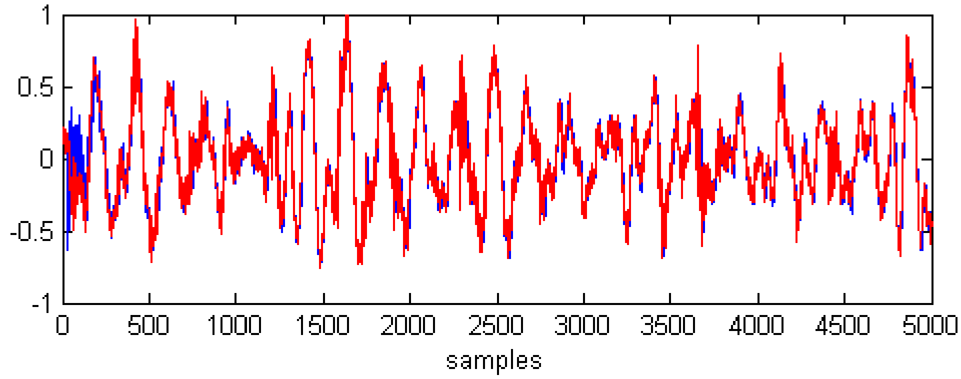


Fig.7.7 Superposition of model output and the actual data (data set-3)

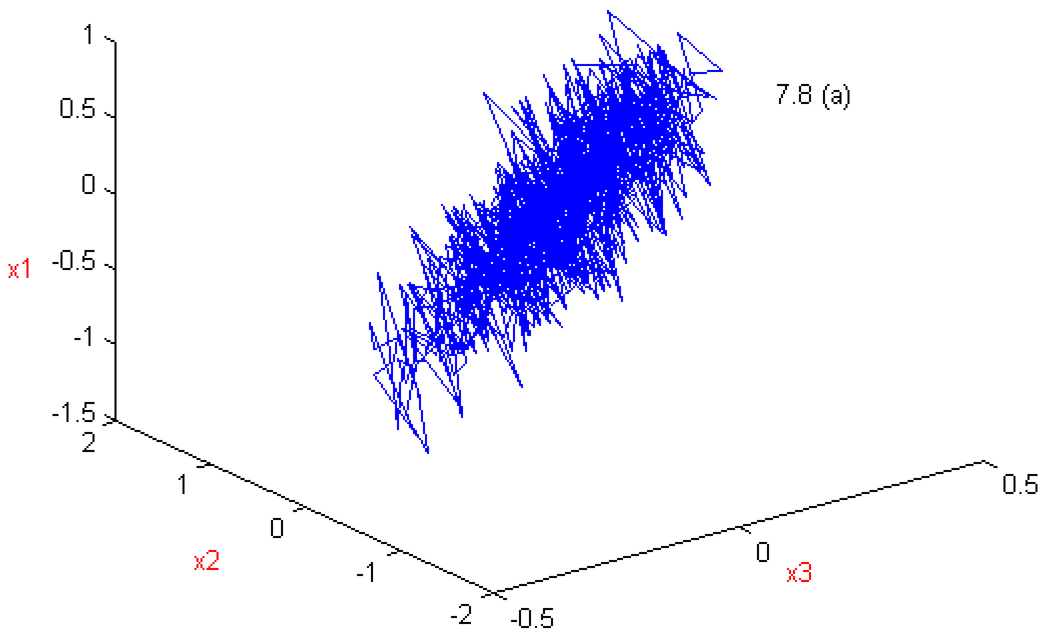
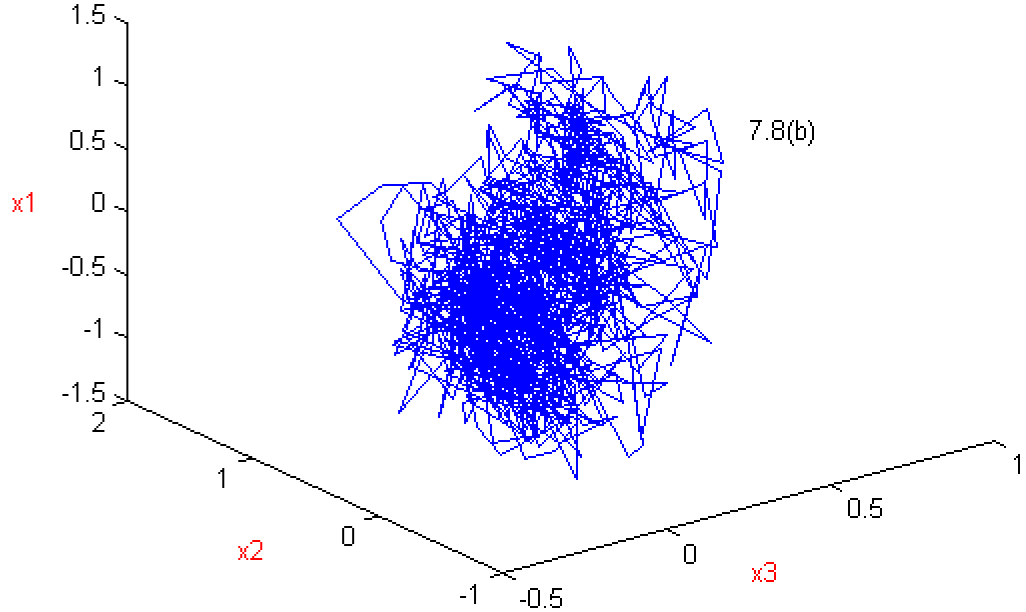
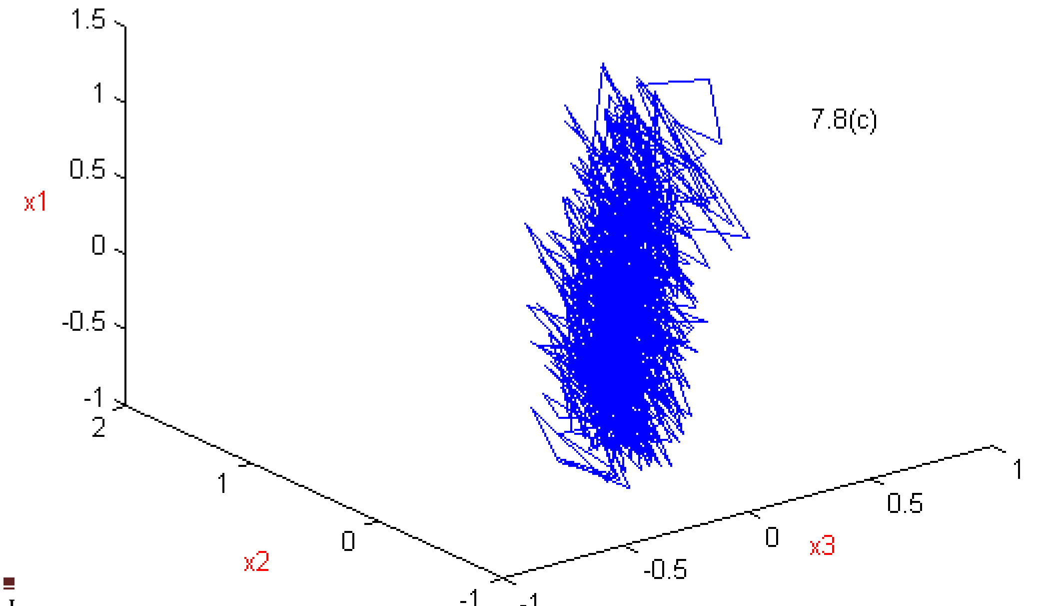


Fig.7.8 Phase plot at different intervals showing the change in dynamics of the system (7.8(a), 7.8 (b) and 7.8(c))


7.8(b)
x1
1.5
1
0.5
0
-0.5
-1
-1.5
x2
2
1
0
-1
x3
-0.5
0
0.5
1


7.8(c)
x1
1.5
1
0.5
0
-0.5
-1
x2
2
1
0
-1
x3
-1
-0.5
0
0.5
1

7.5.4. Acoustic source- 'B'

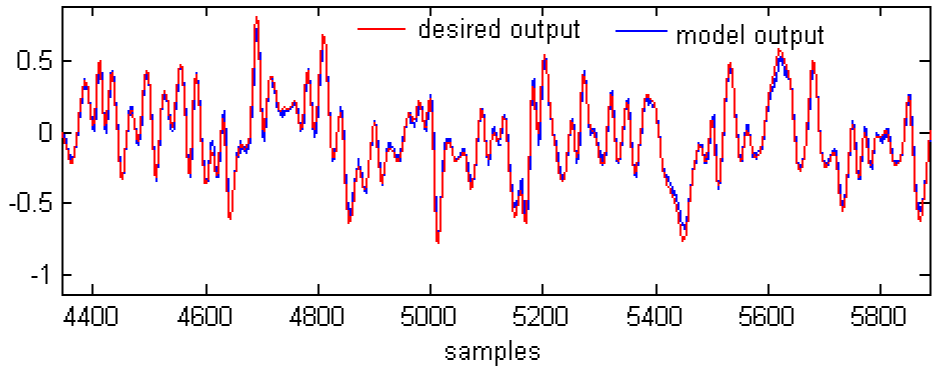


Fig.7.9 Superposition of desired and model output (data set-4)

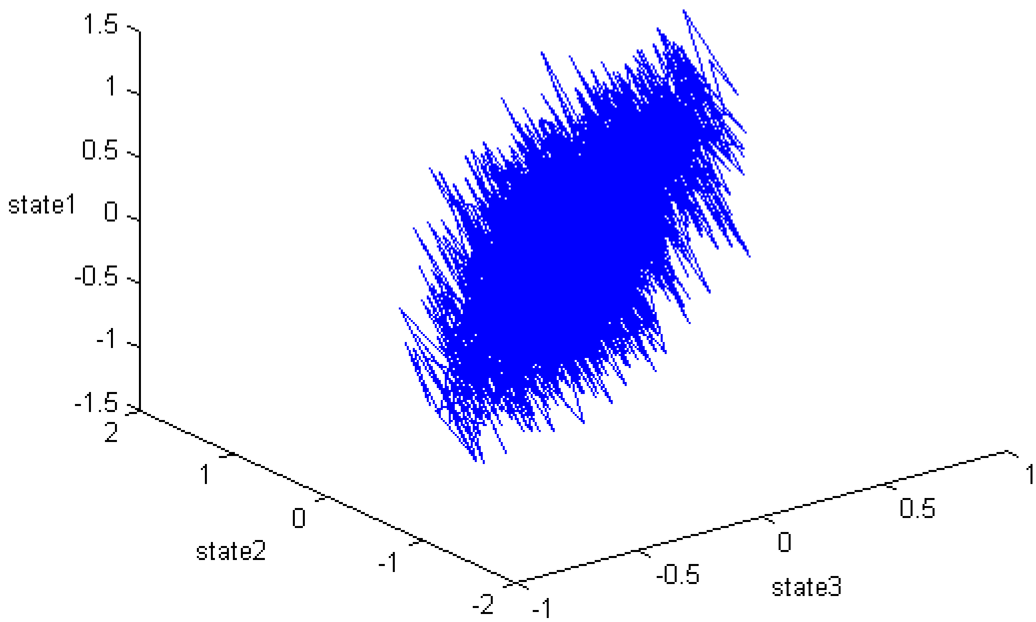


Fig.7.10 Phase plot corresponds to data set-4

7.6 State space analysis of particle filter based models

Phase plots for two of the nonlinear systems modeled using EKF and PF are presented for the completion of the evaluation. It is observed that the system dynamics could be well assessed from the phase plots as depicted in Fig 7.11 and Fig 7.12.

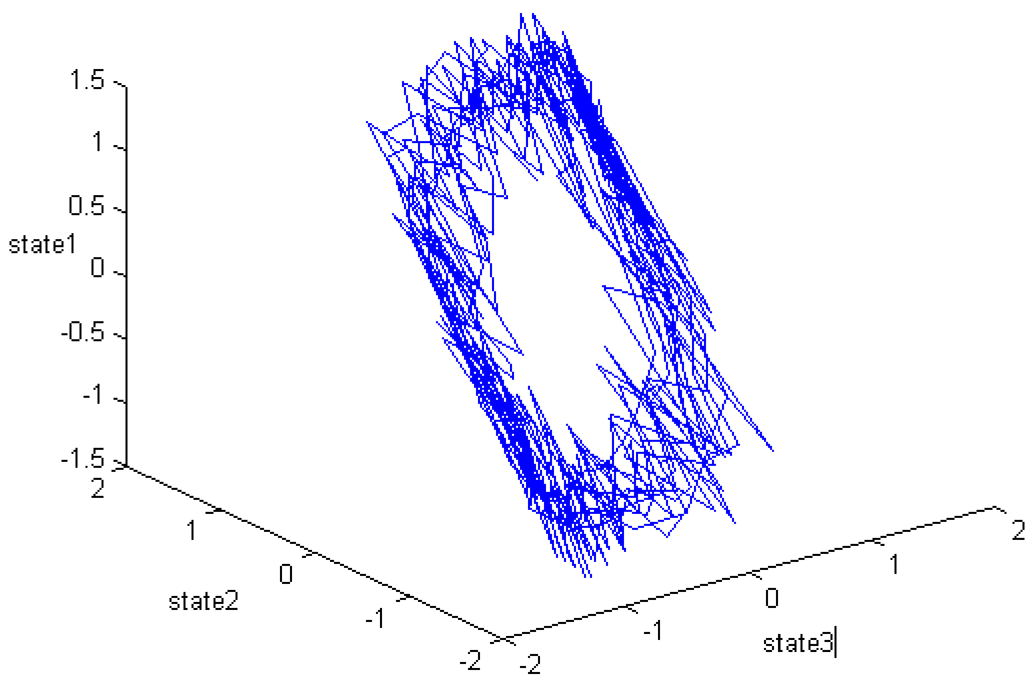


Fig 7.11 Phase plot for the system $y= sin(t+t^2)$ (PF model)

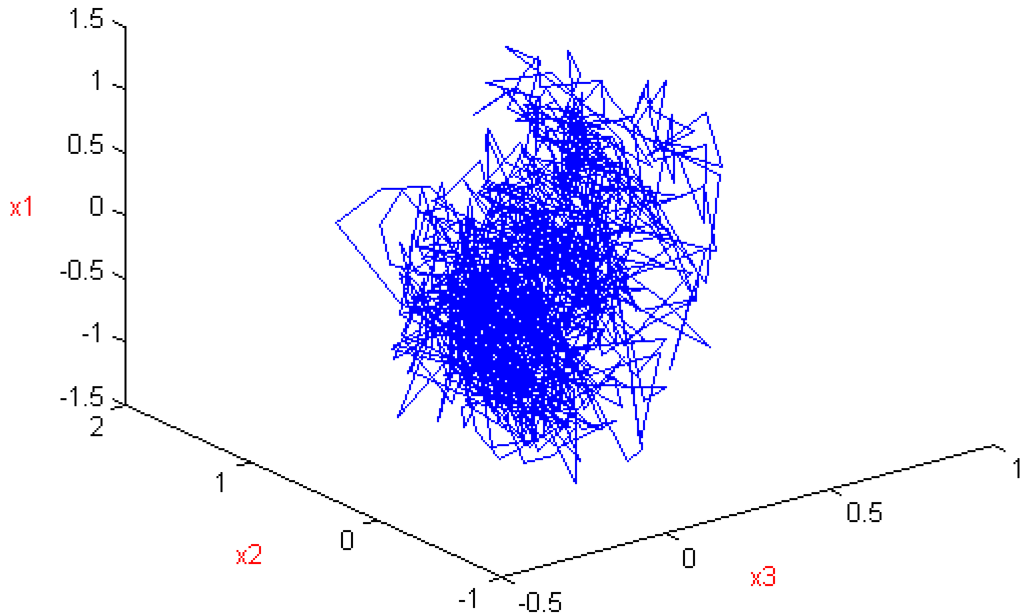


Fig 7.12 Phase plot for ambient noise in the sea (PF model)

7.7 Analysis of the Arrhythmia data

A number of recordings from the standard data base and other available sources were used for the model development and analysis with the inspiration from the control system paradigm.

As a first step in the analysis processes, the RNN is trained for various EEG recordings. The training is carried out in such a manner that the mean square error (MSE) is reduced to a tolerable value with successive training epochs. The output of a sufficiently trained network/model for one of the data is plotted along with the actual biosignal data and the MSE in Fig 7.13and Fig 7.14.

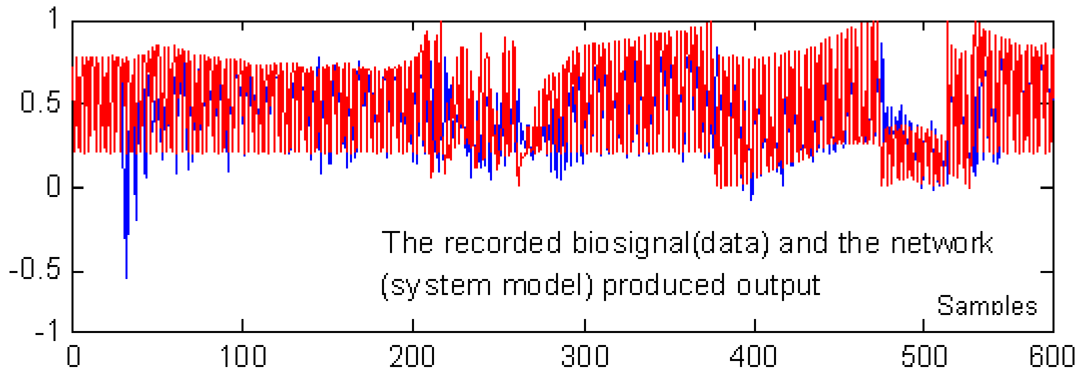


Fig.7.13 Super position of the model output along with the actual bio signal data

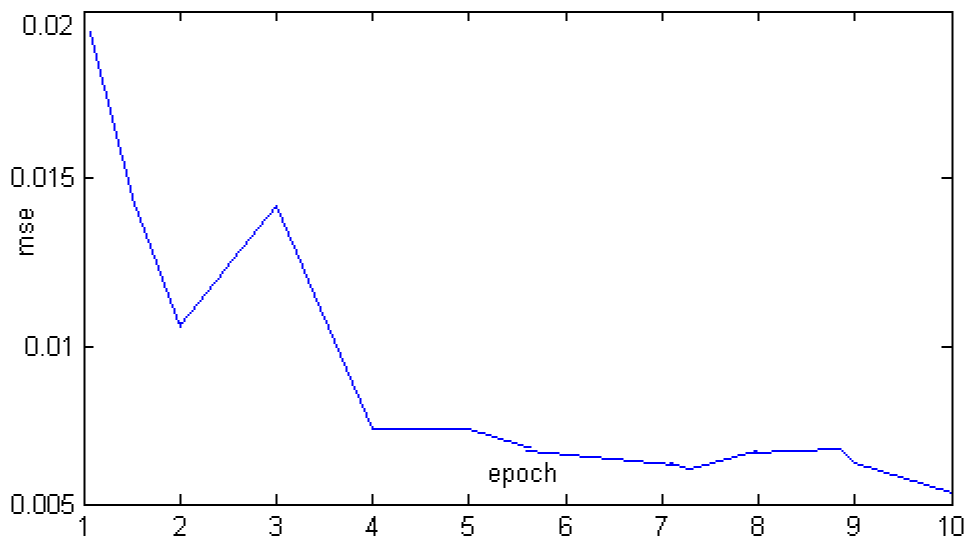


Fig 7.14 MSE verses the number of epochs for the RNN (bio signal data)

As a performance evaluation and validity test for the model, the mean square error criterion is being considered. It has been observed that the Extended Kalman filter (EKF) algorithm is an efficient approach for the training of RNN in nonlinear system modeling problems. A plot of the MSE over 10 epochs for the above data shows that the error has been reduced down to the order of 0.0056 during the training and validation.

The state space analysis is now carried out for the bio signal data and the phase trajectories are also been presented. The phase plots of two different data with somewhat similar medical interpretations are shown in Fig. 7.15.

The plot clearly shows an overlap which indicates a close similarity of the two records (or outputs of two nonlinear systems, i.e. is the human body with similar dynamic behaviors). They are almost indistinguishable from each other because of their similarities in the dynamics.

In successive sections of this chapter bio signal data with slightly similar and different medical interpretations. The phase plot analysis has also been performed for all such data to demonstrate the suitability of the approach in practical nonlinear system analysis. The study also gives an insight to the system dynamics with phase plane plots.

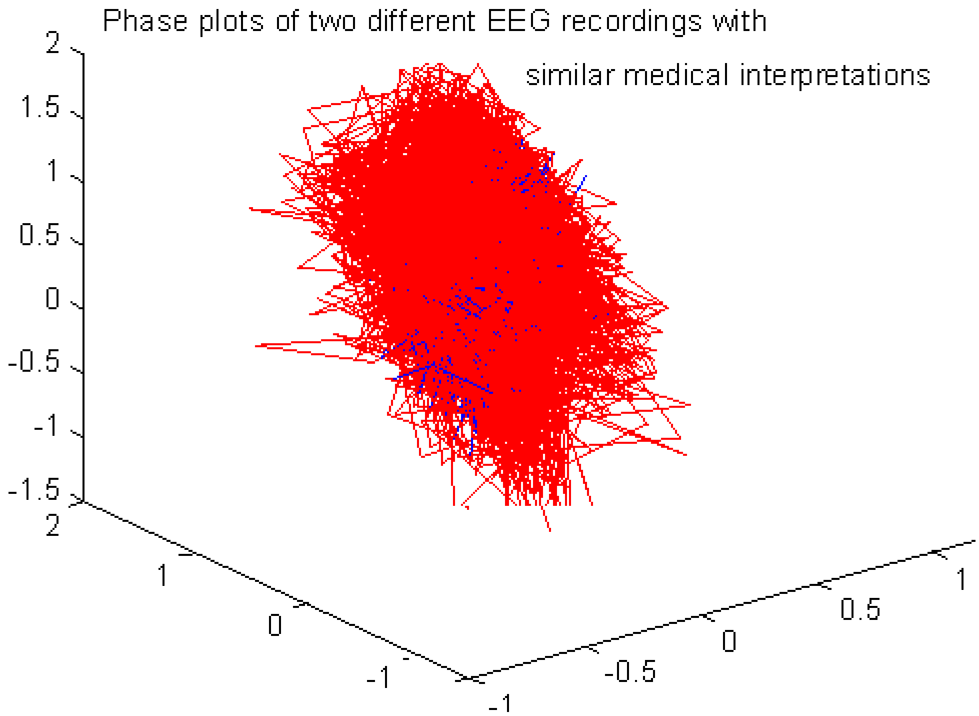


Fig 7.15 Phase plot of two EEG data with similar medical interpretations

In the next stage of the analysis the RNN is trained with the EEG recording of a normal person without any health disorders. Using this neural network model, the phase trajectories of the normal EEG used in training/modeling (blue color) and an EEG recording with an abnormality named supra ventricular ectopy (red color) are plotted and compared (Fig.7.16). The phase trajectories are well distinguished from each other to indicate the difference in dynamics/medical interpretations.

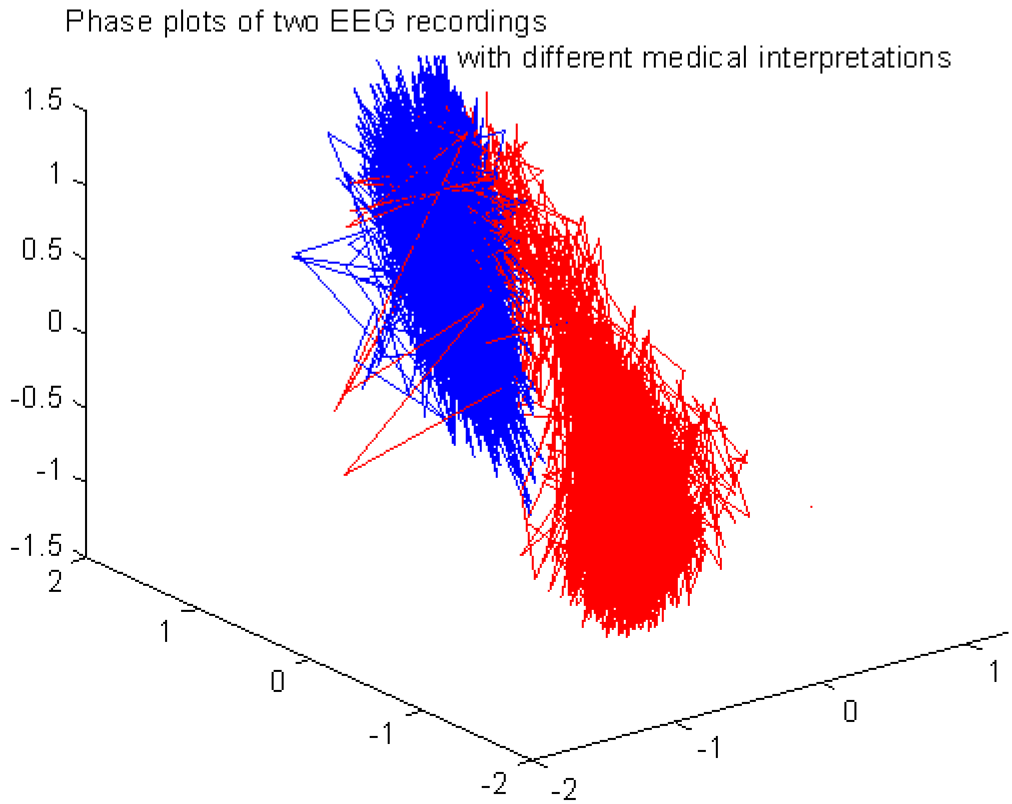


Fig 7.16 Phase plot of two EEG data with different medical interpretations

When the modeling is repeated with two different EEG data having minor similarities and some major differences, the phase trajectories in Fig. 7.17 is obtained. The amount of overlapping could be interpreted as an indication of the minor similarities, even when they are distinguishable because of the major differences.

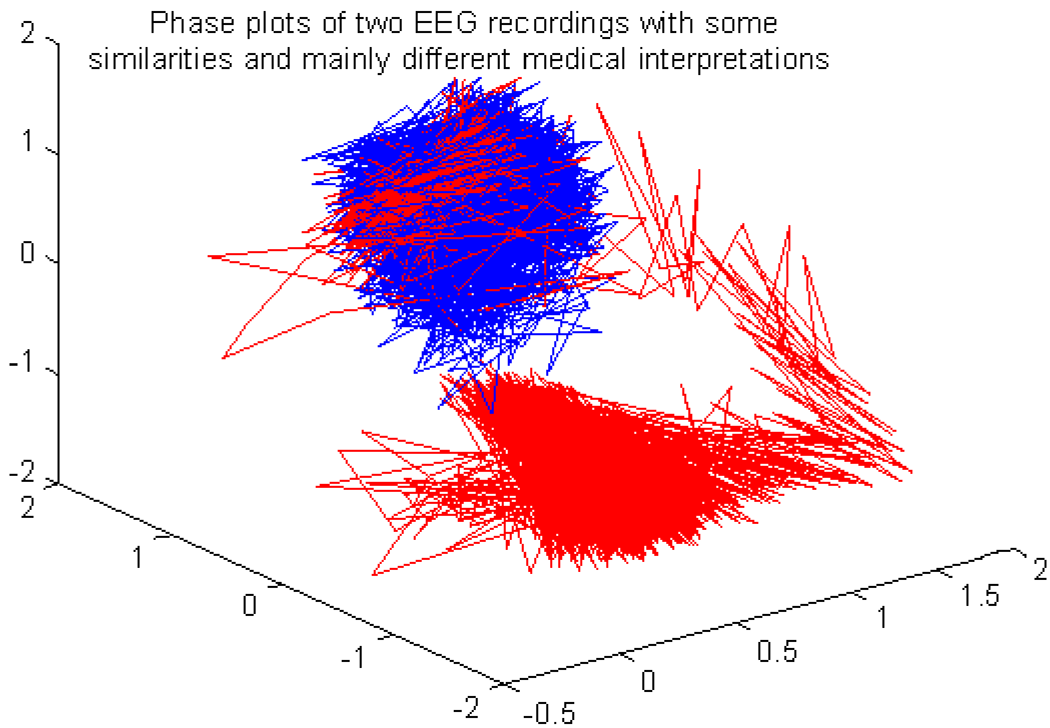


Fig 7.17 Phase plot of two EEG data with minor similar and mainly different medical interpretations

The extension of the modeling approach based on neural networks can thus be successfully extended to combined estimation of state and parameter. By constructing the phase space of the system using the model, the evolutionary nature of the system can be studied. It is worthwhile to note that the entire exercise was done only using the output of the system in a totally blind mode. The phase plane generated using the model changes, in response to the systems changes, demonstrates the success of the modeling approach based on neural networks.

7.8 The Lyapunov Exponent

The Lyapunov exponents of a system are a set of invariant geometric measures that describe the dynamical content of the system. In particular, they serve as a measure of how easy it is to perform prediction on the system under consideration. Lyapunov exponents quantify the average rate of convergence or divergence of nearby trajectories in a global sense. A positive exponent implies divergence, and a negative one implies convergence. Consequently, a system with positive exponents has positive entropy in that trajectories that are initially close together move apart over time. The more positive the exponent, the faster they move apart. Similarly, for negative exponents, the trajectories move together. A system with both positive and negative Lyapunov exponents is said to be chaotic [131].

Mathematically, Lyapunov exponent can be defined by,

$$\lambda_j = \lim_{n \to \infty} \left(\frac{1}{n}\right) \sum_{i=1}^{n-1} \ln\left[\partial f(x_i) / \partial x_j\right] \tag{7.10}$$

where, x_i is the i^{th} state of the system and $f(x_i)$ is the output of the system. [130]

As such, it can be seen that the Lyapunov exponents describe the average rate of exponential change in the distance between trajectories in a set of orthonormal directions within the embedding space. The number of exponents is equal to the number of states of the system.

Positive Lyapunov exponents are responsible for the sensitivity of a chaotic process to initial conditions. Negative Lyapunov exponents on the other hand govern the decay of transients in the orbit. A zero Lyapunov exponent signifies the fact that the underlying dynamics responsible for the generation of chaos are describable by a coupled system of nonlinear differential equations, that is, the chaotic process is a flow. A volume in d-dimensional state space behaves as *exp*(L ($\lambda_1 + \lambda_2 + \lambda_3 + \ldots\ldots \lambda_d$), where L is the number of time steps into the future. It follows therefore that for a dissipative process, the sum of all Lyapunov exponents must be negative. This is a necessary condition for a volume in state space to shrink as time progresses, which is a requirement for physical realizability.

The Lyapunov exponents calculated for the four nonlinear systems modeled using the RNN state space model are summarized in Table 7.1.

Table 7.1 Lyapunov exponents of the systems

System	Lyapunov Exponent
$y=\sin(t^2+t)$	-5.8175 , -3.0505 , -5.24+3.14i
Ambient noise	-3.6654, -3.6119 , - 4.267+3.14i
Nonlinear system 'A'	-3.6654, -3.6119 , - 4.267+3.14i
Nonlinear system 'B'	-1.8682 , -6.53+3.14 i, -3.3792

From the results it is seen that the three nonlinear systems have all negative real Lyapunov exponents. So all the systems converge and they are not chaotic.

7.9 Conclusions

Extending the basic modeling approach discussed in earlier sections to RNNs, the combined state and parameter estimation has been carried out for the four nonlinear systems. For a given higher order system, analysis can be done by modeling it with minimum of 2 or 3 states. The systems are modeled with three states. Both the system states $\boldsymbol{x_k}$ and the set of model parameters $\boldsymbol{w}$ for the dynamic system are simultaneously estimated from the observed noisy output $\boldsymbol{y_k}$ only. The Lyapunov exponents are calculated for the systems modeled, and evaluated the nature of the evolutionary process. Through the analysis of the model developed for bio-signals like EEG, it has been demonstrated that the state space recurrent neural network models could be better explored for the classification and characterization of the behavior of the subject at different time instants. A sufficiently trained neural network model can provide a better perspective on the dynamics of the biological systems compared to the conventional time domain interpretations of the recorded data.

The invariant quantities of the systems like Lyapunov exponents, which can be taken as a tool for model validation, could be calculated for the systems for better assessment of the suitability of the same for this type of signal/system analysis.

Chapter 8

EVALUATION OF THE ANN BASED NONLINEAR SYSTEM MODELS IN THE CRLB SENSE

Chapter 8 discusses the evaluation of the systems modeled and analyzed with a variety of techniques as developed in the previous chapters. The CRLB approach to measure the goodness of the model has also been demonstrated. It also gives a comparative study of model performances.

8.1 Introduction

Objective of this chapter is to evaluate the performance of the algorithms discussed in the previous chapters. While the performance in terms of the Mean Square Error (MSE) was demonstrated in the respective chapters, the Cramer Rao Lower Bound (CRLB) is also established for each of the algorithms. [122-125].

The CRLB is calculated for all the cases using the Nonlinear Auto Regressive with Exogenous input (NARX) model [4] for representing the system. Here again the examples used are the four nonlinear sets of data available viz.

1. Data generated using $y=sin(t^2+t)$
2. Ambient noise
3. Nonlinear Source-A
4. Nonlinear Source-B

The amazing challenges in statistical estimation along with an opportunity to learn different techniques in solving the well known problem motivated the authors to compare the performance of these approaches. The model behavior and performance are evaluated in terms of the Mean Square Error (MSE) and the Cramer Rao Lower Bound (CRLB) [121].

8.2 Comparison based on CRLB

The efficiency of an estimator can be checked by, establishing the Cramer Rao Lower Bound for the various estimators. According to it, the mean square error corresponding to the estimator of a parameter cannot be smaller than a certain quantity related to the likelihood function. If an estimator's variance is equal to the CRLB, then such estimator is called efficient. [5].

The Cramer Rao Lower Bound on the covariance matrix of the target parameter estimate *w* is (assuming this estimate to be unbiased),

$$E\,(w - w^{\wedge})\,(w - w^{\wedge})^{T} \geq FIM^{-1} \qquad (8.1)$$

where FIM is the Fisher Information Matrix. Following [5], the FIM can be written as,

$$r^{-1} \sum_{k} \partial h(k,w)/\partial x \; \partial h(k,w)/\partial w \,\Big|_{w = w^{\wedge}} \qquad (8.2)$$

where *h(.)* is the modeling function and r the variance of the measurement z(k) given by,

$$z(k) = h(w, x) \qquad (8.3)$$

This follows from the assumption that the measurement noises are white, zero mean and with variance r. [122]. A necessary condition for an estimator to be consistent in the mean square sense is that there must be an increasing amount of information (in the sense of Fisher) about the parameter in the measurements. The Fisher information has to tend to infinity as $k \rightarrow \propto$, then the CRLB converges to zero as $k \rightarrow \propto$ and thus the variance in the estimate can also converge to small values.

CRLB calculations are done for the models and thus checked the consistency of the estimation methods. If the method satisfies CRLB, that is an acceptable estimator. Model convergence is checked for 100 different values of initializations of the parameter vector w (keeping mean and variance same) and based on that CRLB is calculated. CRLB checking in effect involves comparison of two matrices; the parameter covariance matrix and the inverse of Fischer Information matrix. In view of the fact that these two matrices are always diagonally dominant, the checking becomes easy by comparing the diagonal elements of the matrices. The comparison is also possible by subtracting one matrix from the other and checking the positive semi definiteness of the resultant. (A-B is positive semi definite, if A>B) [145]. In the illustrations given below, the blue color indicates the Inverse of FIM, while the red color shows the covariance matrix.

8.2.1 Back Propagation Algorithm

In this section, the CRLB computations are carried out for the models developed in the previous chapters which make use of BP algorithm and its variations for training. This includes SLP, MLP and RBF type of network structures in general. One typical result for the general conclusion and discussion is demonstrated in Fig 8.1.

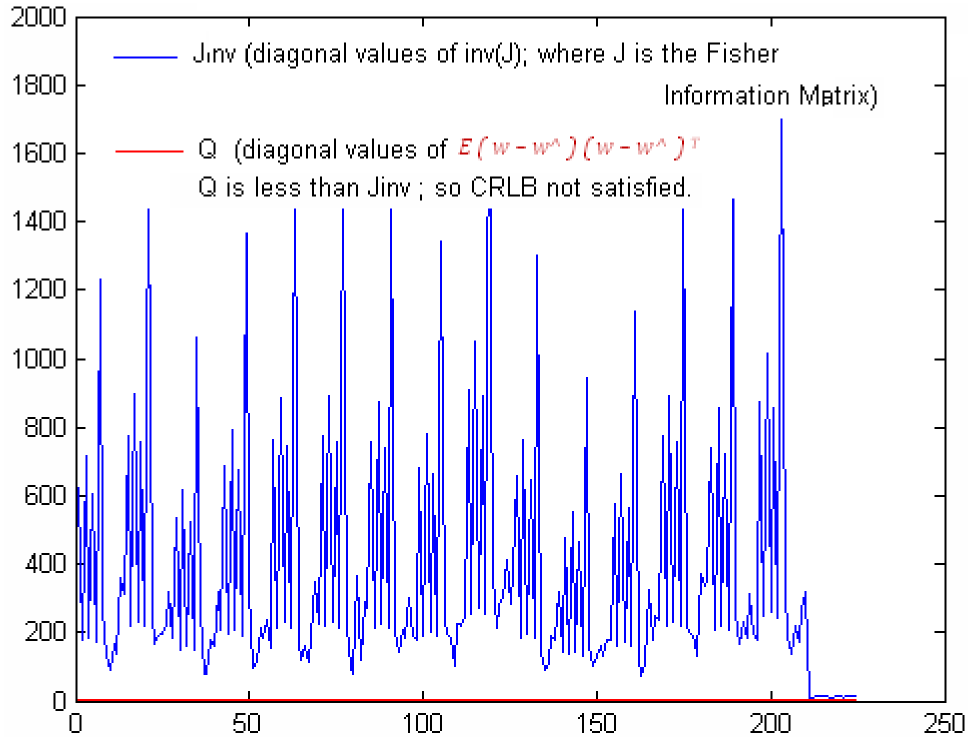


Fig 8.1 CRLB plot for the BPA trained network; here the variance (close to X axis) is much lower than the inverse of the uncertainty matrix. $E\,(w - w^{\wedge})\,(w - w^{\wedge})^{T} \geq FIM^{-1}$

From the simulation results given in Fig 8.1, it is seen that the BPA does not satisfy CRLB. So BPA is not an efficient algorithm for system identification. BPA is a gradient based algorithm so chances are there to settle in local minima. The CRLB is calculated for the four nonlinear systems modeled using BPA, over 100 independent runs and got the similar results.

8.2.2 EKF Algorithm

Here the CRLB computations are done for EKF, EKF with EM and the results are presented in Fig 8.2 and 8.3.

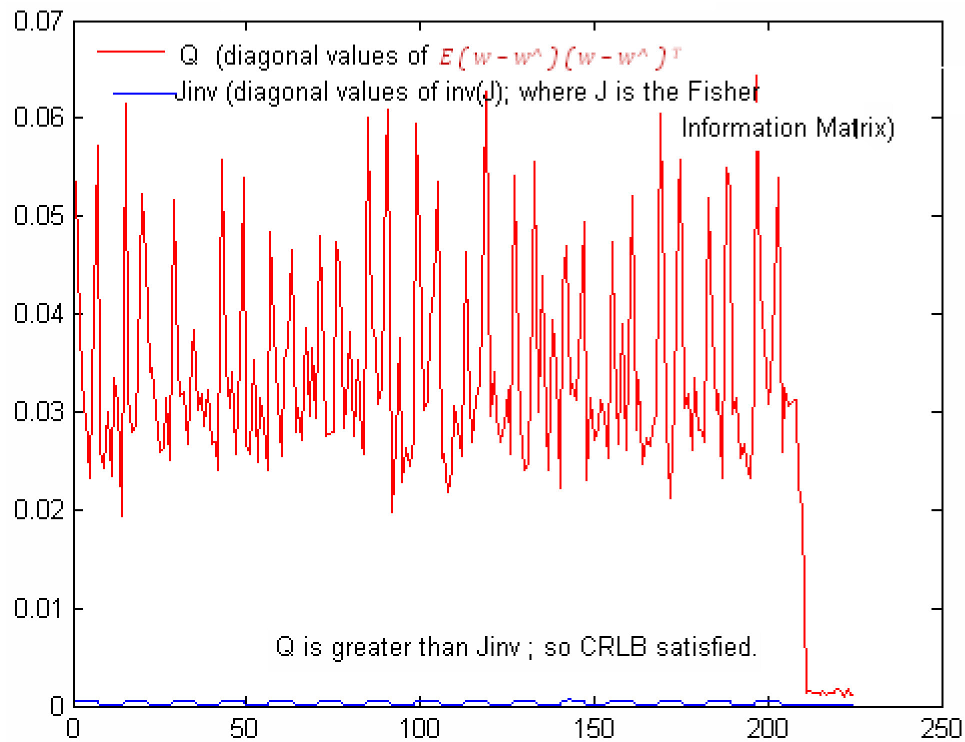


Fig 8.2 CRLB results for EKF training algorithm (ambient noise).
$E\,(w - w^{\wedge})\,(w - w^{\wedge})^{T} \geq FIM^{-1}$

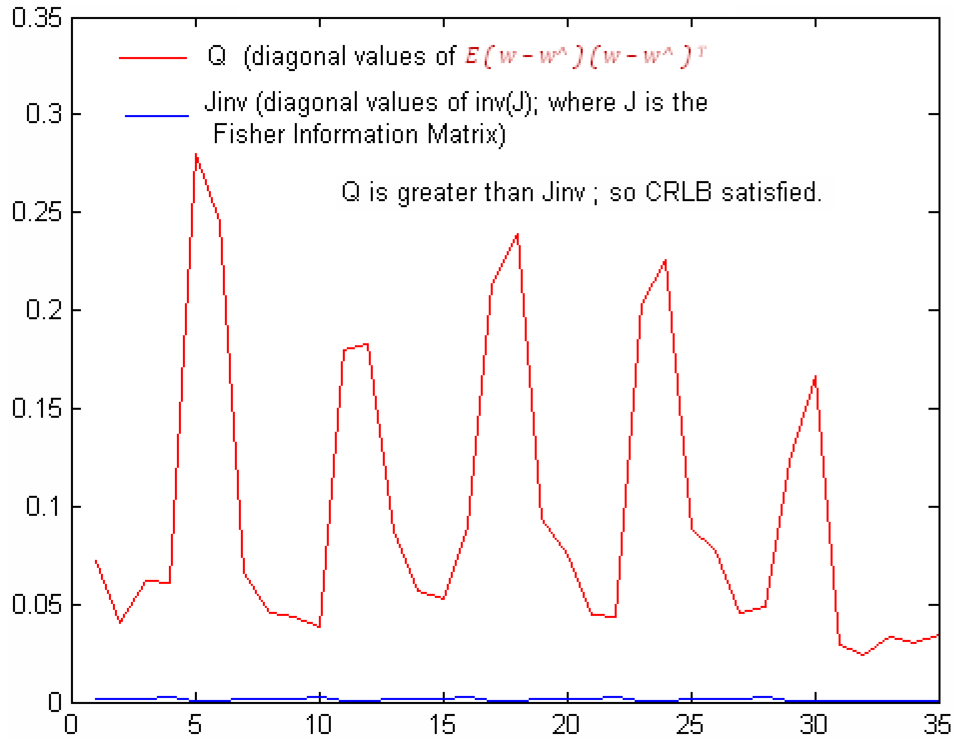


Fig 8.3 CRLB results for EKF with EM (ambient noise). $E(w - w^{\wedge})(w - w^{\wedge})^{T} \geq FIM^{-1}$

Here the both methods viz. the EKF and the EKF with EM primarily satisfy CRLB, which indicates that the methods are very efficient. CRLB is checked for the four nonlinear systems and is satisfied. From these illustrations, it is obvious that the performance of EKF with EM is better than the EKF, considering the proximity of the co-variance (in red) to the FIM (in blue) in

Fig 8.2 and 8.3. In terms of mean square error also EKF has superiority among other algorithms implemented, as is evident from Table 4.5.

8.2.3 Maximum Likelihood Estimation

The CRLB computations pertaining to the statistical method of MLE with two of its variations, viz. conjugate gradient and Gauss-Newton are presented with conclusions on their performances.

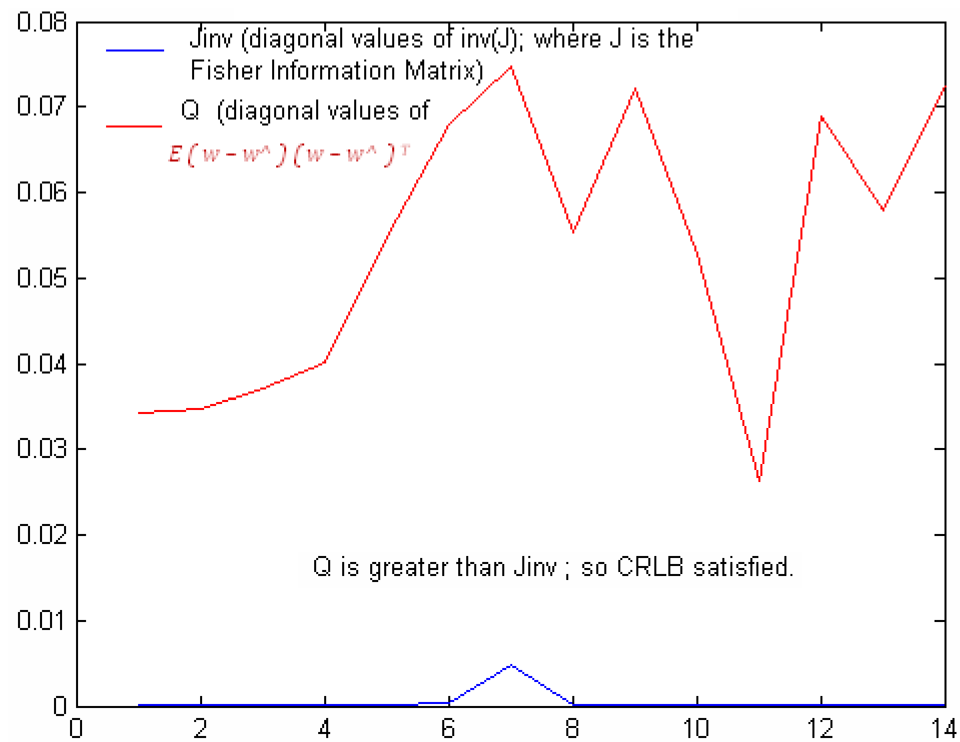


Fig 8.4 CRLB results for MLE (Conj-Gradient) (ambient noise). $E(w - \hat{w})(w - \hat{w})^T \geq FIM^{-1}$

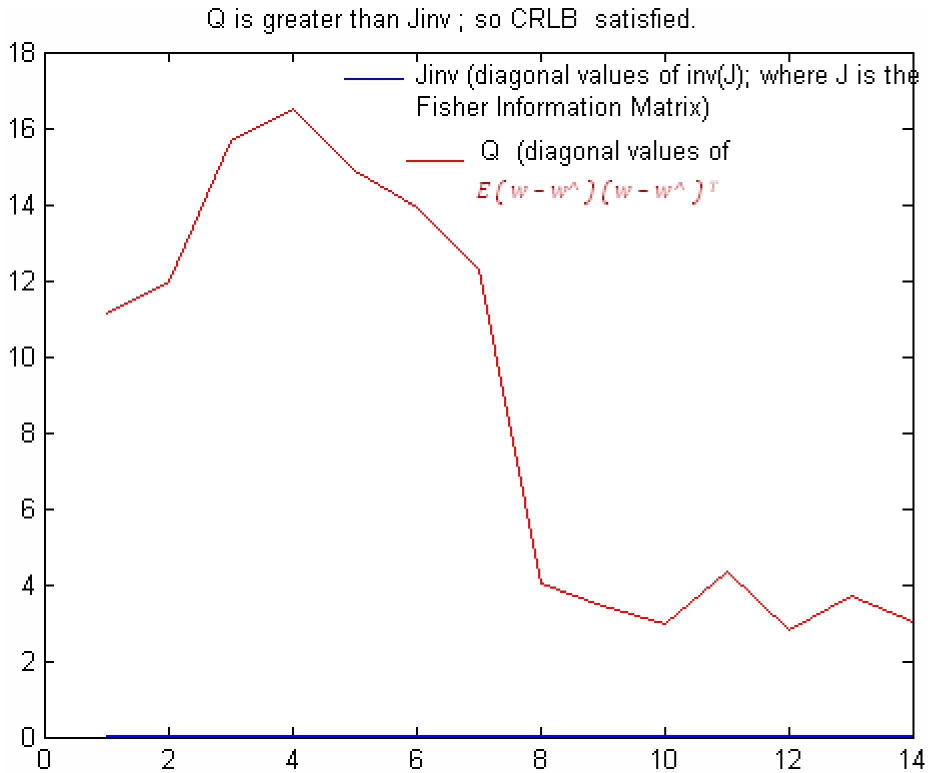


Fig 8.5 CRLB results for MLE (Gauss-Newton) (ambient Noise in the sea).

$E\,(w - w^{\wedge})\,(w - w^{\wedge})^{T} \geq FIM^{-1}$

Maximum Likelihood Estimation (Both Conjugate Gradient and Gauss-Newton methods) satisfies CRLB. This proves the efficiency of MLE for system identification. Even if these two methods are also based on gradient of cost function, the convergence of the model to local minimum is avoided.

8.2.4 Particle filter estimation

The CRLB performance of the particle filter method is depicted in Fig 8.6 below. It could be recollected, from the analysis on MSE sense, in chapter 6, that PF methods are superior in their performance on nonlinear modeling.

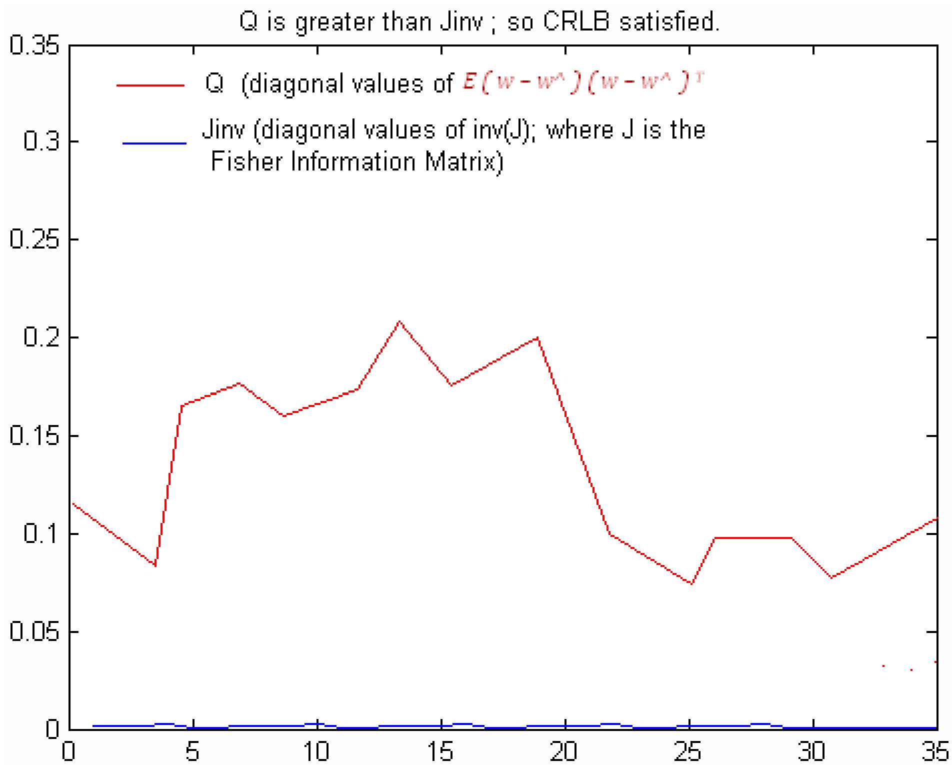


Fig 8.6 CRLB results for particle filter (ambient Noise in the sea).

$E (w - w\hat{}) (w - w\hat{})^T \geq FIM^{-1}$

The PF method has been proved to satisfy the CRLB conditions as it satisfied the MSE analysis. The computations have been done for all the nonlinear systems and similar kind of results were obtained to confirm the quality of PF approaches for nonlinear system modeling with neural networks.

8.3 Conclusions

The different estimation methods tried out have been checked according to Cramer Rao Lower Bound and their relative efficiency is analyzed. From the results it is seen that the Particle Filter, EKF algorithm and Maximum Likelihood Estimation techniques are more efficient for nonlinear system identification compared to Back Propagation Algorithm. The CRLB estimate also brought out the supremacy of the EKF based estimation algorithm, with the EM variant and the Particle filter approaches. It is hoped that the results brought out in this chapter would be helpful for the system engineers to choose proper approach for blind identification, of nonlinear systems.

Chapter 9

SUMMARY, BENEFITS AND FUTURE DIRECTIONS

Chapter 9 provides a collective summary of the results, contributions and major outcomes of the work carried out in this thesis. It also suggests the possible future research in the area of nonlinear system modeling/identification.

9.1 Introduction

The thesis has presented the outcome of the exhaustive analysis of the algorithms for modeling and identification of Non linear systems, and puts forward a few new approaches in the modeling of non linear systems. The modeling approach adopted is largely based on the NARX model, realized using Neural Network with different nonlinear functions like sigmoid and tan sigmoid. In the exhaustive evaluation of the performance of the algorithms, the model resorts to the optimization algorithms for training the weights of the neural networks to make the model behavior equivalent to that of actual system [19-21]. The algorithms considered in the thesis are Back Propagation (BPA), Extended Kalman Filter (EKF) for feed forward and recurrent NN, Expectation Maximization (EM), Maximum Likelihood Estimation (MLE), and Particle Filter (PF). Their performance is compared both in the sense of MSE and CRLB.

The evaluation has been carried out using four data sets viz. (i) the one generated from the nonlinear function $\sin(t^2+t)$, data recorded from the ambient noise in sea and two acoustic sources A and B. The diverse nature of the non linear properties of the data generated along with the noise content, across the non linear function from $\sin(t^2+t)$ to the ambient noise and the acoustic sources, gave sufficient challenge in exhaustively evaluating the algorithms.

9.2 Comparison between BPA, EKF, EKF with EM, MLE and the Particle Filter Models

A comparison summary of the various methods for calculating the weights of the neural network of the model, which are described in the previous chapters, is presented here in the MSE sense. All the methods viz. back propagation, EKF, EKF with EM, MLE algorithm with conjugate gradient and Gauss-Newton approaches and the Particle filter based methods are compared. The selected four nonlinear systems mentioned above were modeled with all these methods during the analysis and the results are demonstrated through Fig 9.1 to 9.4 and the same is summarized in Tables 9.1 to 9.5 below.

The MSE for the data set corresponding to all data set clearly shows that the EKF with the EM and the Particle filter based approach performs best, both during training and validation. The algorithms converge faster and do not show any tendency to diverge during validation. It is interesting to note that the relatively simpler version of the Particle filter implemented has resulted in commendably good performance.

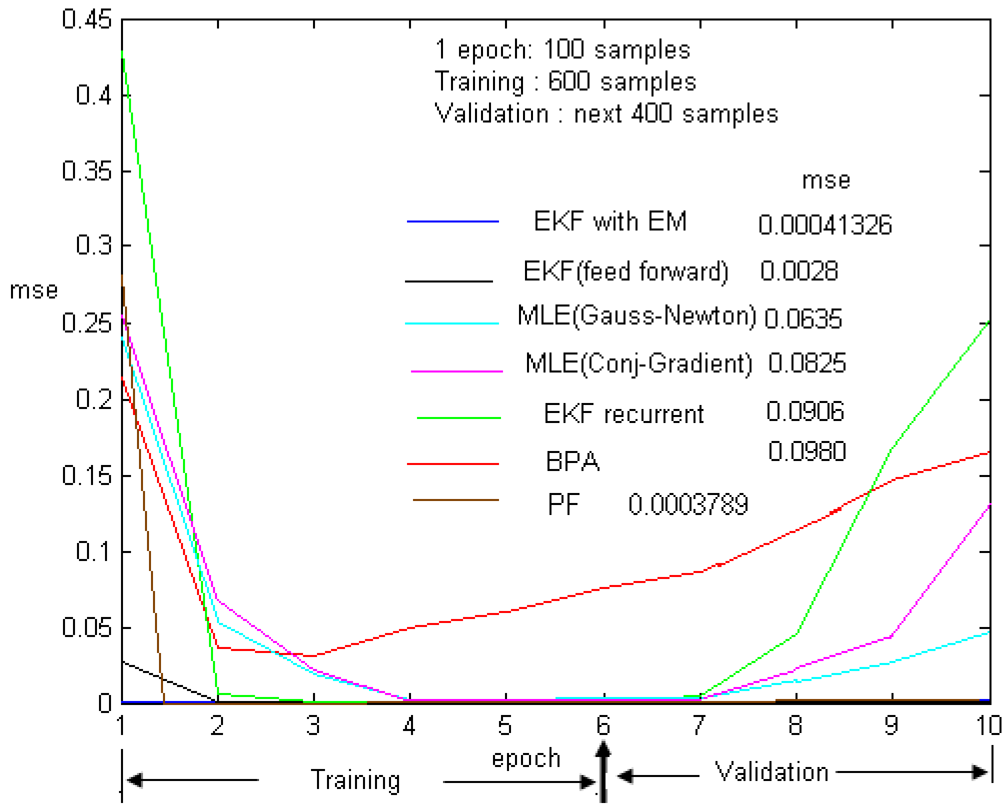


Fig 9.1 MSE for the data set $y = \sin(t^2 + t)$

The results correspond to the first data set is shown in Fig 9.1. The BP algorithm shows a slight divergence behavior, especially during the validation period. It is also observed that as the number of training and validation data samples (the data size) have some effect on the overall model performance. This is summarized in Table 9.1.

Table 9.1 Comparison of Performance with $y=\sin(t^2+t)$

No. of Training Samples	No. of Validation Samples	MSE	
		BPA	EKF
300	300	0.0190	4.56×10^{-3}
600	300	0.0118	9.18×10^{-5}
1000	1000	0.0383	4.29×10^{-4}

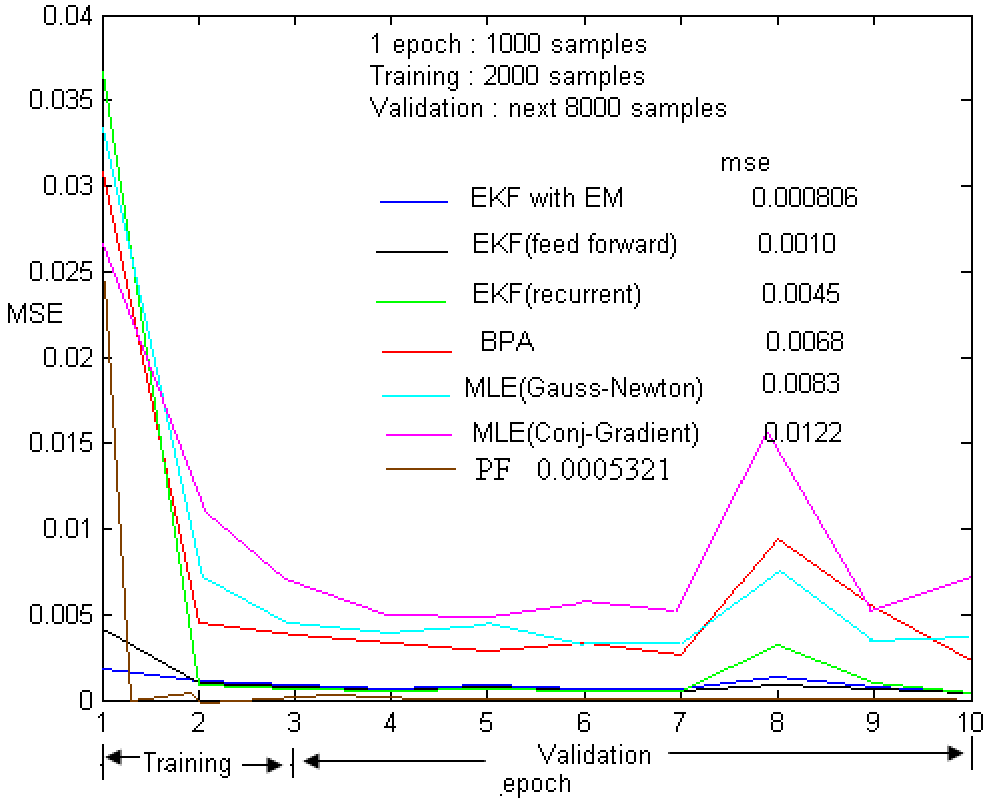


Fig 9.2 MSE for the data set ambient Noise in the sea

Table 9.2 Comparison of Performance with acoustic source a

No. of Training Samples	No. of Validation Samples	MSE	
		BPA	EKF
1000	500	0.0175	0.0122
1000	7000	0.0191	0.0139
2500	2500	0.0085	0.0054

In case of the second data set also EKF with EM and Particle Filter methods shows better performances.

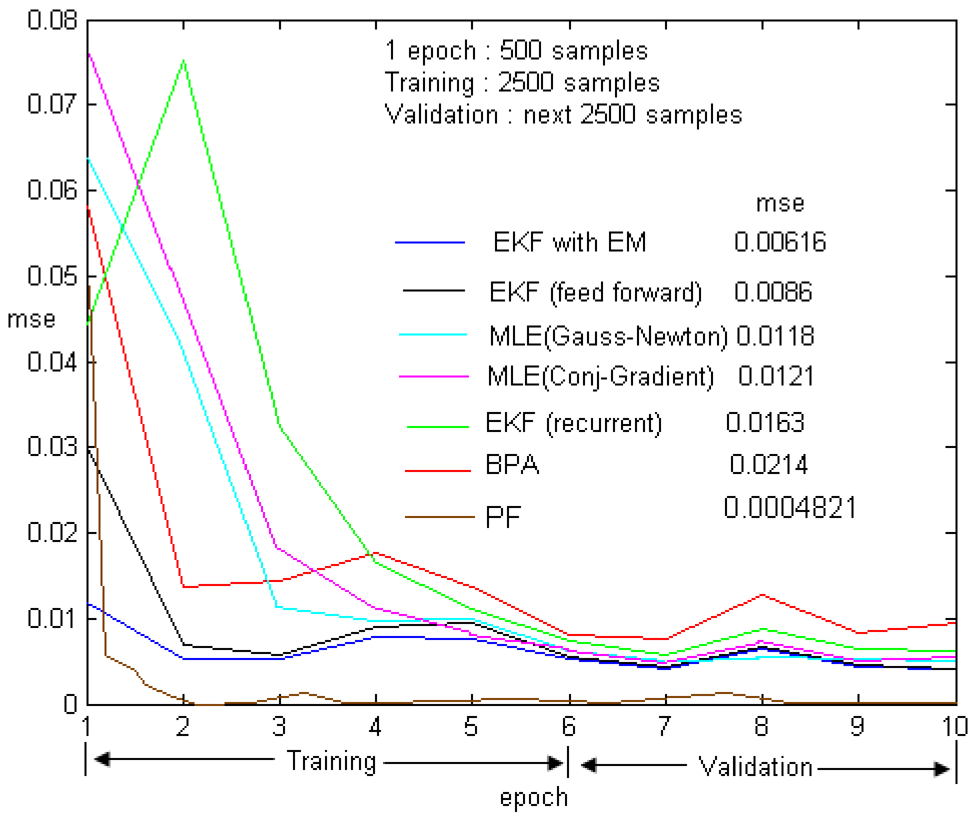


Fig 9.3 MSE for the data set acoustic source 'A'

The non monotonic nature of the convergence for the EKF (recurrent), while modeling the third data set appears to be because of the outliers in the recorded data. This conclusion is made since the convergence is seen to be remaining monotonic in the case of all other data sets, for all algorithms.

Table 9.3 Comparison of Performance with acoustic source b

No. of Training Samples	No. of Validation Samples	MSE	
		BPA	EKF
500	500	0.0054	2.9×10^{-5}
500	1000	0.0042	2.13×10^{-5}
2500	2500	0.0016	8.98×10^{-7}

Here also there is an improvement on the MSE performances, as the data size for training and validation is increased. The results corresponds to data set-4 is described next.

Table 9.4 Comparison of Performance with ambient noise in the sea

No. of Training Samples	No. of Validation Samples	MSE	
		BPA	EKF
500	500	0.012	1.5×10^{-3}
500	2000	0.0115	2.2×10^{-3}
2000	8000	0.0038	5.9×10^{-4}

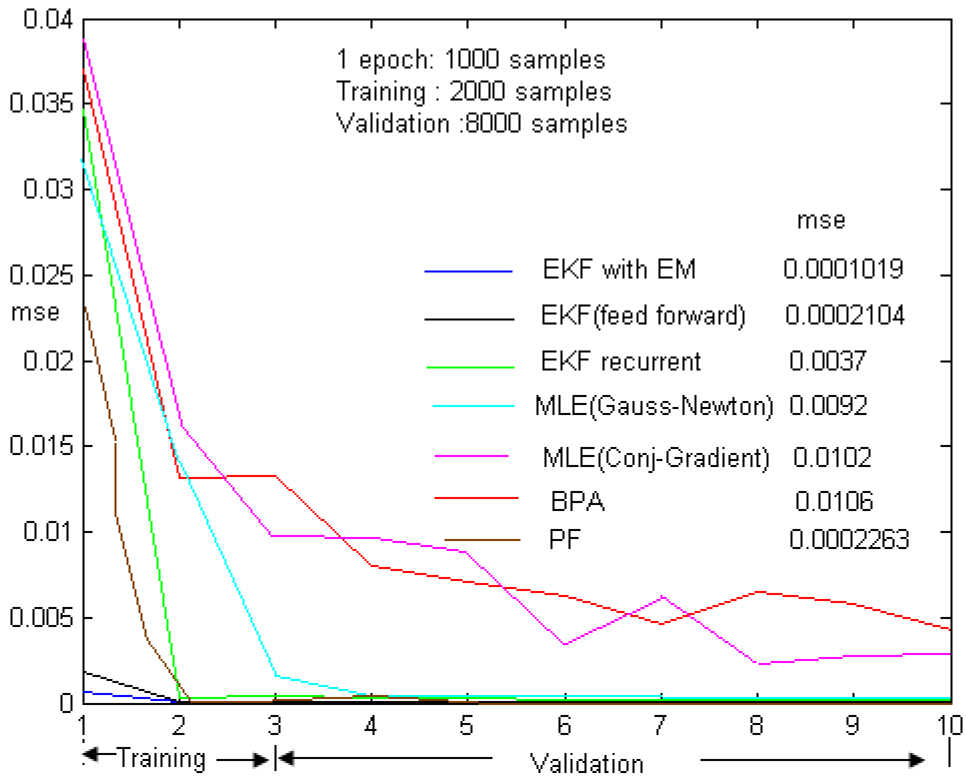


Fig 9.4 MSE for the data set acoustic source 'B'

Same number of samples is used for training and validation for each data set, for each of the different algorithms, (to make the comparison relevant). For $y=sin(t^2+t)$, First 600 samples are used for training and next 400 samples for validation. For Ambient Noise, 2000 samples were used for training and 8000 samples for validation. The Acoustic source A, and B were modeled using 2500 samples for training and 8000 samples for validation. Table-9.5 below gives a total summary of the above results.

TABLE 9.5
COMPARISON SUMMARY

Nonlinear System	Mean Square Error (MSE) Validation						
	Particle Filter	BPA	EKF-Feed forward network	EKF Recurrent network	EKF with EM	MLE	
						Gauss-Newton	Conjugate Gradient
$y=\sin(x^2+x)$	3.789×10^{-4}	0.0980	2.8×10^{-3}	0.0906	4.132×10^{-4}	0.0906	0.0906
Ambient noise in the sea	5.321×10^{-4}	0.0068	0.0010	0.0045	8.06×10^{-4}	0.0045	0.0045
Acoustic source A	4.821×10^{-4}	0.0182	0.0086	0.0163	6.16×10^{-3}	0.0163	0.0163
Acoustic Source B	2.263×10^{-4}	0.0106	2.104×10^{-4}	0.0037	1.019×10^{-4}	0.0037	0.0037

From the above results it is seen that EKF algorithm converges faster and has marginally better performance compared to the other Algorithms. It is also consistent for all the nonlinear systems modeled. The performance of EKF can be again increased by EM algorithm as shown. The other algorithms also give good results and computationally efficient but in problems where faster convergence is required as in real world problems Kalman Estimation is preferably better. The use of particle filtering algorithm for training of the neural network models could reduce the overall mean square error to around 2.263×10^{-4}, which is acceptable and comparable with the achievements in EKF. However, the computational cost in terms of memory for particle filters will have to be weighed against the humble memory requirements of other algorithms.

It is further suggested to evaluate the systems in the CRLB and other figure of merit senses for a better conclusion on the model validity and goodness. Chapter 8 addresses the CRLB technique and theory in detail. The results obtained have successfully proved the merits of EKF and Particle filter algorithms over the other methods.

9.3 Discussions and future directions

In summary, thesis has successfully demonstrated the development and evaluation of the comparative performance of estimating the parameters of a neural network used in the system identification function for non-linear systems. The estimation of parameters (viz. the weight of the Neural Network) has been carried out using an improved technique based on BPA, EKF, EKF with EM, MLE, and Particle filter. The simulation results demonstrated in previous chapters and summarized in Section 9.2 demonstrate that Kalman estimation technique and Particle Filter approaches are efficient tools for system identification and they can be implemented as a powerful algorithm for training Neural Networks. One of the main contributions is the demonstration of the utilization of the EKF with EM for the computation of the state space evolution. This effort supplemented by the usage of Particle Filters has helped to minimize the modeling errors, apart from promising stability in the model performance. However, the experience in deriving the results brings out that one has to take some preliminary measures before implementation for modeling in order to improve the accuracy of estimation. This will help to achieve improved solution for identification problems. These are explained below.

- The input-output data has to be scaled and transformed to reasonable ranges
- Weight values are initialized to small random values drawn from a zero mean uniform or normal distribution
- Appropriate error covariance matrix is initialized so that a priori knowledge was used to initialize the weights
- The covariance matrix R of the measurement noise is set to a scaled identity matrix with scaling factor of the order of unity or more

Arbitrary assumptions of the initial values and parameters for the update equations in different algorithms may cause divergence, when error covariance matrix computed by the filter becomes small compared to the actual error in the estimate. When this matrix becomes small, the gain matrix becomes small and new measurements are given too little weight. The number of arithmetic operation grows as the matrix dimension increases because of model size, which causes large computational errors. Due to this state covariance matrix looses symmetry, which causes numerical instability and divergence. Decoupled EKF is recommended to be used to reduce the computational complexity and to improve the performance of the filter. The problem of improper assumptions of initial values in EKF can be alleviated by using algorithm Expectation Maximization (EM). The thesis has effectively demonstrated this aspect, which yielded very good results.

Unscented Kalman Filter (UKF) is another approach to incorporate non Gaussianity due to nonlinear transformations, in terms of nonlinear approximation of probability density function. The computational complexity of UKF algorithm is order L^3 (L is the number of parameters) where as the complexity of EKF is order L^2. The UKF algorithm also necessitates the computation of the matrix square root at each time step.

It is also proved that Maximum Likelihood Estimation is an efficient estimation technique and well suited for implementations using artificial neural network. A series of experiments has been conducted to study the efficiency of Recurrent Networks for parameter and state estimation. Identification of simulated system is compared to results obtained using

Feed forward networks. The results show that the evolutionary nature of the state is possible through Recurrent Neural Network. By constructing the phase space of the system the evolutionary nature of the system can be studied. When the system dynamics changes, the phase space geometry changes, which is evidenced by the state trajectory. From this the invariant quantities of the system like Lyapunov exponent can be found.

Cramer Rao Lower Bound is a universally accepted tool for defining the efficiency of the estimator. The thesis has practically computed the CRLB for the simulations done for the different algorithms and assessed their performance. The CRLB estimate also brings out the supremacy of the EKF algorithm, with EM and the Particle filter approaches.

Literature survey showed that many other approaches are there for this problem. Among these some of the methods viz. System identification using Neuro-Fuzzy Inference systems(ANFIS), system identification using Genetic Algorithm, Implementation of Extended Kalman Filter Algorithm in Digital Signal Processors, frequency domain approach for system identification, Support vector machines and its variations, modeling and identification of chaotic systems etc can be further explored with detailed study and analysis. The major contributions of the thesis are discussed next.

9.4 Contributions

In this thesis a comprehensive study of different approaches for nonlinear system identification is done and their performance is compared by implementation of different algorithms in a Neural Network NARX model. The adaptive feature revealed by feed forward and recurrent neural network as well as their ability to model nonlinear time varying process, provides a surplus value to the model based predictive control. When applied correctly, a neural or adaptive system may considerably outperform other methods. While working in real time, these algorithms can be suitably coded in Digital Signal Processor for improving computation time. The algorithms implemented are recursive in which the weights are updated recursively, immediately after the presentation of data; they are on-line mode of training. Hence they can work in the continuous fashion in nonstationary environment. It is hoped that this thesis has lit a small candle in the emerging world of blind nonlinear system identification. The significant contributions of the thesis are given below:

1. A number of novel methods for blind identification of nonlinear system are exhaustively evaluated, so as to help to select the right method for a given application.

2. The formulation of the modeling problems using different approaches has been well established.
3. Training approaches of EKF, EKF with EM, MLE (both Gauss-Newton and Conjugate gradient) and Particle filter methods are implemented and compared.
4. An improvement in the EKF algorithm has been shown with EM.
5. The statistical method of MLE has been applied for nonlinear system identification problems.
6. The method of Particle Filtering (PF) for the use of nonlinear system identification/modeling has been suggested and its novelty is established.
7. Validation technique based on CRLB sense, along with the MSE sense has been introduced.
8. State space analysis including phase plane plots and Lyapunov exponent computations have been established for the proper understanding of the system dynamics, which is of great relevance in system study.
9. A comprehensive comparative study of various methods for nonlinear system identification/modeling has been successfully performed.

9.5 Conclusion

Beginning with the available methods for nonlinear blind system identification, the development of certain new approaches have been presented in the thesis. The suggestions for the improvement of some of the existing approaches like EKF is also a benefit of the results presented. The state space analysis, the Lyapunov exponents' methods, validation in the CRLB and MSE senses of the models etc can also be of appreciable use for control and data analysis applications, including bio-signal processing. The major contribution of Particle Filter method is a demonstration of the application of the nonlinear filtering approaches in modeling/identification problems. This enhancement tool can therefore be an efficient approach for the analysis of nonlinear dynamics in general.

List of papers published

International Journal/IEEE/Indexed publications

- **M.V Rajesh** [1], Archana. R [2], R. Gopikakumari[3], A Unnikrishnan[4], "Analysis of Nonlinear System Dynamics in State Space Modeling with Recurrent Neural Networks", International Journal of Applied Engineering Research, Vol.5, Number 5, 2010, pp 861-870. (ISSN 0973-4562). URL: http://www.ripublication.com/ijaer.htm
- **M.V Rajesh[1]**, Archana R[2], A Unnikrishnan[3] , R Gopikakaumari[4], "Particle Filter based Neural Network Modeling of Nonlinear Systems for State Space Estimation", It is indexed by ISTP and included in the IEEE Xplore database, as well as indexed by EI Compendex. http://ieeexplore.ieee.org/xpl/tocresult.jsp?isnumber=5191464&isYear=2009&count=1246&page=11&ResultStart=275
- **M.VRajesh[1]**; R, Archana[2]; A, Unnikrishnan[3]; R, Gopikakaumari[4], "Comparative Study on E KF Training Algorithm with EM and MLE for ANN Modeling of Nonlinear Systems", accepted for presentation in the 29th Chinese Control Conference (CCC-2010), to be held in Beijing, China during July 27 to 29, 2010, which will be appearing in IEE Xplore after the conference. (Communicated) http://ieeexplore.ieee.org/search/srchabstract.jsp?tp=&arnumber=5573550&queryText%3Drajesh%26openedRefinements%3D*%26filter%3DOR(Publication+Number:5562463)%26searchField%3DSearch+All
- **M.V Rajesh[1]**, Archana.R[2], A Unnikrishnan[3], R Gopikakumari[4], "Evaluation of the ANN Based Nonlinear System Models in the MSE

and CRLB Senses", Published in the proceedings of WASET, Volume 48, December, 2008, ISSN 2070-3740. http://www.waset.org/journals/waset/v48/v48-34.pdf.
(Abstracting & Indexing: CiteSeerX, Google Scholar, Serials Solutions, Engineering Index (Compendex), Open Science Directory, Open Science Research, INTUTE, Scientific Commons and Electronic Journals Library)

International Conferences

- M.V, Rajesh*; R, Archana[2]; A, Unnikrishnan[3]; R, Gopikakaumari[4], "Comparative Study on EKF Training Algorithm with EM and MLE for ANN Modeling of Nonlinear Systems", accepted for presentation in the 29th Chinese Control Conference (CCC-2010), to be held in Beijing, China during July 27 to 29, 2010, which is organized by IEEE Control Systems Society, Beijing Institute of Technology, Technical Committee on Control Theory (CCA) and the Society of Instrument and control engineers.
- M.V Rajesh[1], Archana R,[2], A Unnikrishnan[3] , R Gopikakaumari[4], "Particle Filter based Neural Network Modeling of Nonlinear Systems for State Space Estimation", Accepted for presentation in the 21st Chinese Control and Decision Conference (CCDC) which is an annual international conference. (2009 CCDC) held in Guilin, China in June 17 to 19, 2009, which is organized by IEEE Industrial Electronics (IE) Chapter, Singapore , Northeastern University, China, Guilin University of Electronic Technology, China, IEEE Control Systems Society and IEEE Industrial Electronics Society.

- **M.V Rajesh[1]**, Archana R[2], R. Gopikakumari[3], A Unnikrishnan[4] , Jeevamma Jacob[5], "On the State Space Modeling and Identification of Arrhythmia Data Using Recurrent Neural Networks." Selected for oral presentation and published in the proceedings of the " International Conference on Optoelectronics, Information and Communication Technologies 2009 (ICOICT2009) from 26-27 February 2009 at Trivandrum, Kerala, India, jointly organized by SCT College of Engineering, Pappanamcodu, Govt. College of Engineering, Trivandrum and Model Engineering College, Kochi.
- **M.V Rajesh[1]**, Archana R[2], A Unnikrishnan[3], R Gopikakumari[4], "Evaluation of the ANN Based Nonlinear System Models in the MSE and CRLB Senses". Accepted for oral presentation at the "5th International Conference on Artificial Intelligence and Neural Networks AINN\'08", organized by the World Congress on Science, Engineering and Technology (WASET) at Bangkok, Thailand during December 17-19, 2008.
- **M.V Rajesh** [1], Archana. R [2], R. Gopikakumari[3], A Unnikrishnan[4], "Analysis of Nonlinear System Dynamics in State Space Modeling with Recurrent Neural Networks". Selected for oral presentation and published in the proceedings of ICCCS 2008. (The 9th International Conference on Computers, Communications and Systems) at Daegu University, Korea held on November 7, 2008.
- Archana R[1], **M.V Rajesh[2]**, R. Gopikakumari[3],Jeevamma Jacob[4], A Unnikrishnan[5], "Study and Analysis of Nonlinear System Dynamics Using state Space Recurrent neural Network Modeling". Presented and published in the proceedings of the "Inter National Conference on

Advanced Computing & Communication Technologies for High Performance Applications", organized by the Federal Institute of Science & Technology (FISAT), Angamali, Kerala, India, during September, 24 to 26, 2008.

- **M.V Rajesh**[1], A Unnikrishnan[2], R Nandakumar[3], "Neural Network Approaches for System Identification". Published in the proceedings of the "International Conference on Instrumentation (INCON-2004)", organized by the Department of Instrumentation & Control, Pune Institute of Engineering and Technology (Govt. engineering College, Shivajinagar, Pune-5), under the auspices of the Instrument Society of India during December 19 to 21, 2004.

National Conferences

- **M.V Rajesh**[1], Archana R[2], A Unnikrishnan[3] , R Gopikakaumari[4], "A Study on EKF Training Algorithm with MLE for the ANN Modeling of Nonlinear Systems." Accepted for oral presentation and publication in the proceedings of the National Conference on Recent Innovations in Technology (NCRIT 2009) organized by Rajiv Gandhi Institute of Technology, (RIT), Govt. Engineering College, Kottayam, during March 26 to 28, 2009.
- **M.V Rajesh**[1], Archana.R[2], A Unnikrishnan[3], R Gopikakumari[4], "A Study on EKF Training Algorithm with EM for ANN Modeling of Nonlinear Systems". Proceedings of the National Conference on Computational Science and Engineering, NCCSE 2009, organized by the Department of Computer Science, Rajagiri College of Social Sciences, Kalamassey, Kochi, Kerala, India-683 104, during 6-7,

February, 2009. ISBN 978-81-907044-0-3.

- **M.V Rajesh**[1], A Unnikrishnan[2], R Nandakumar[3] "System Identification Using MLP and RBF Neural Networks". Published in the proceedings of the "National Seminar on Emerging Trends in Soft Computing based Artificial Intelligence", organized by the Department of Computer Science & Engineering, MBM Engineering College, Faculty of Engineering, Jai Narain Vyas University, Jodhpur, India-342011 during February 27 to 29, 2004.
- Archana. R [1], **M.V Rajesh** [2] , R. Gopikakumari[3], A Unnikrishnan [4], "On the Analysis of Nonlinear System Modeling with Feed Forward and RBF Neural Networks". Presented and published in the proceedings of the national seminar on "Emerging Trends in Information and Communication Technology", organized by the Institution of Engineers India (IEI) at Cochin, Kerala, India, during 28th to 29th August, 2008.
- **M.V Rajesh**[1], Archana. R[2] , R. Gopikakumari[3],Jeevamma Jacob[4], A Unnikrishnan[5], " Neural Network Approaches for System Identification". Presented and published in the proceedings of the "National Seminar on Information, Communication & Intelligent Systems", organized jointly by the Institution of Electronics & Telecommunication Engineers India (IETE) and Model Engineering College, at Cochin, Kerala, India during 8th to 9th February-2008.

References

1. N.K Sinha and B. Kuszta, "*Modeling and Identification of dynamic systems*", Van Nostrand Reinhold Company, New York, ISBN 0-442-28162-5

2. Daniel Grapue, "*Identification of systems*", Van Nostrand Reinhold Company, New York

3. Derrick H. Nguyen and Bernard Widrow "*Neural networks for self learning control systems",* IEEE control system magazine. Pp 18-23, April 1990

4. S. Chen, S.A Billings, and P.M Grant, "*Non linear system identification using neural networks*", International journal of control, 51(6):1191-1214,1990

5. G. Cybenko, "*Approximation by super positions of a sigmoidal function*,'' Math. Contr., Signals, Syst., vol. 2, pp. 303-314, 1989

6. Kumpati .S. Narendra and Kannan Parthasarathy , "*Identification and control of dynamical systems using neural networks*", IEEE transactions on neural networks,vol.2, no 2,pp 252-262,march 1991

7. Kumpati. S. Narendra and Kannan Parthasarathy "*Gradient methods for the optimization of dynamical systems containing neural networks*", IEEE transactions on neural networks, vol.2, no.2, pp 252-262, March 1991

8. K. S. Narendra and K. Parthasarathy, "*Identification and control of dynamical systems using neural networks,*" IEEE Trans. Neural Networks, vol. 1, pp. 4-27, Mar. 1990

9. S. Chen and S. A. Billings, "*Neural networks for nonlinear dynamic system modeling and identification*," Int. J. Contr., vol. 56, no. 2, pp. 319–346, 1992
10. Ljung, L. (1999), "System *identification, Theory for the user*", System sciences series. 2 ed. Prentice Hall, Upper Saddle River, NJ

11. A. U. Levin, K. S. Narendra and D. E. Elliot. "*Identification of nonlinear dynamical systems using neural networks*," in Neural Networks for Control. Norwood, NJ: Ablex, 1994

12. A. U. Levin and K. S. Narendra, "*Recursive identification using feed forward neural networks*," Int. J. Contr., vol. 61, no. 3, pp. 533-547, 1995

13. Naresh S Sinha, "*Identification of continuous systems from samples of input-output data: An introduction*", Sadhana, vol.25, part 2 ,pp 75-83, April 2000,printed in India

14. S. Lang, Leontaritis, S. A. Billings, "*Input-output parametric models for nonlinear systems, Part I: Deterministic nonlinear system*," Int. J. Contr., vol. 41, pp. 303-328, 1985

15. Fa-Long Luo and Rolf Unbehauen, "*Applied neural networks for signal processing*" Cambrigde University Press, UK,ISBN 0 521 56391 7

16. Jaceck M Zuranda, "*Introduction to artificial neural systems*", 2/e, Jaico Publishing house, Mumbai, India, ISBN 81-7224-650-1

17. S. Bhama and H. Singh, "*Single layer neural networks for linear system identification using gradient descent technique*," IEEE Trans. Neural Networks, vol. 4, pp. 884–888, Sept. 1993

18. N. Sadegh, "*A perceptron based neural network for identification and control of nonlinear systems*," IEEE Trans. Neural Networks, vol. 4, pp. 982–988, Nov. 1993
19. Hava T Siegelman, "*Computational capabilities of recurrent NARX neural networks*", Technical report, UMIACS-TR-95-12 and CDS-TR-3408, University of Maryland

20. Wen-Xiao Zhao, Han-Fu Chen, Wei Xing Zheng, "*Recursive Identification for Nonlinear ARX Systems Based on Stochastic Approximation Algorithm*", IEEE Transactions on Automatic Control, October 2009

21. Jinglu Hu, Kousuke Kumamaru and Katsuhiro Inoue, "*A Hybrid Quasi-ARMAX Modeling Scheme for Identification and Control of Nonlinear Systems*", Proceedings of the 35th Conference on Decision and Control Kobe, Japan December 19961413-18

22. L. K. Jones, "*Constructive approximations for neural networks by sigmoidal functions*," Proc. IEEE, vol. 78, pp. 1586–1589, Oct. 1990

23. M. I. Jordan, ''*Learning inverse mappings using forward models*," in Proc. 6th Yale Wkshp. Adaptive Learning Syst., Yale Univ., New Haven, Aug. 1990, pp. 145-151

24. S. S. Yang and C. S. Tseng, "*An orthonormal neural network for function approximation*," IEEE Trans. Syst., Man, Cybern. B, vol. 26, pp. 779–785, Oct. 1996

25. K. Hornik, M. Stinchcombe, and H. White, "*Multilayer feed forward networks are universal approximators*," Neural Networks, vol. 2, pp. 359-366, 1989

26. Songwu Lu and Tamer Basar, "*Robust nonlinear system identification using neural network models*", IEEE transactions on neural networks, vol.9,no 3,pp 407-429,may 1998

27. A.Stubberudand, H.Wabgaonkar, "*Approximation and estimation techniques for neural networks,"* in Proc. 28th Conf. Decision Control, Honolulu, HI, Dec.1 990, pp.2736–2740

28. B. Castillo-Toledo and A. Hernandez, "*On Output Regulation for SISO Nonlinear Systems with Dynamic Neural Networks*", Proceedings of International Joint Conference on Neural Networks, Montreal, Canada, July 31 - August 4, 2005(372-377)

29. Shaohua Tant, Jianbin Hao and Joos Vandewalle, "*Identification of Nonlinear Discrete-Time Multivariable Dynamical Systems by RBF Neural Networks*", 0-7803-1901-X/94 01994 EEE 3250-55

30. A. U. Levin, "*Neural networks in dynamical systems: A system theoretic approach,*" Ph.D. dissertation, Yale Univ., New Haven, CT, Nov. 1992

31. Tech. Rep. 8902 "*Neural networks and dynamical systems, Part II: Identification*," Center Syst. Sci., Dept. Elect. Eng., Yale Univ., New Haven, CT, Feb. 1989

32. T.G.Stockham,Jr., T.M.Cannon, and R.B.Ingebretsen, "*Blindde convolution through digital signal processing*," Proc.IEEE,vol.63,pp.678–692,Apr.1975

33. J.A.CadzowandX.Li,"*Blinddeconvolution*,"DigitalSignalProcessin g,vol.5,pp.3–20,1995

34. K.S.LiiandM.Rosenblatt, "*Deconvolution and estimation of transfer function phase and coefficients for non Gaussian linear processes*," Ann. Statistics, vol.10, pp.1195–1208,1982

35. O.ShalviandE.Weinstein,"*Universal methods for blind deconvolution*," in Blind Deconvolution, S.Haykin,Ed.EnglewoodCliffs,NJ:Prentice-Hall,1994

36. Karim Abed Meraim, Wanzhi Qiu, "*Blind System Identification*", Proceedings of the IEEE Vol.85, No.8, 1997

37. Mirai Oshiro, Hiroshi Ochi, "*Noise Robust Blind System Identification Using LMS Method*", Roc. 43rd IEEE Wdwest Symp. on Circuits and Systems, Lanslng MI, Aug 8-1 1,2000

38. Chong-Yung Chi*, Senior Member, IEEE,* and Chii-Horng Chen, "*Two Dimensional Frequency-Domain Blind System Identification Using Higher Order Statistics with Application to Texture Synthesis*", IEEE Signal Processing Letters Vol. 49, No. 4, April 2001

39. Chunjian Li*, Member, IEEE*, and Søren Vang Andersen*, Member, IEEE, "Efficient Blind System Identification of Non-Gaussian Autoregressive Models With HMM Modeling of the Excitation",* IEEE Transactions on Signal Processing Vol. 55, No.6, June 2007

40. L. Tong and S. Perreau, "*Multichannel blind identification: From subspace to maximum likelihood methods*," *Proc. IEEE*, vol. 86, pp. 1951–1968, Oct. 1998

41. S.V. Vaseghi, "*Advanced Digital Signal Processing and Noise Reduction*, 2nd ed.", New York: Wiley, 2000

42. Hua, Y., Xiang, Y., and Abed-Meraim, K., "*Blind identification of colored signals distorted by FIR channels*," Proc of IEEE ZCASSPZOOO, Istanbul, Turkey, June 2000

43. Hua, Y., "*Blind identification and equalization of channels driven by colored signals*," Chapter 4, Signal Processing Advances in Wireless Communications, Prentice-Hall, 2000

44. Shun-Ichi Amari, Anderzej Cichocki, "*Adaptive Blind Signal Processing on Neural Network Approaches*", Proceedings of the IEEE Vol.86, No.10, October 1998

45. Andrzej Cichocki, Rolf Unbehauen, "*Robust Neural Networks with On-Line Learning for Blind Identification and Blind Separation of Sources*", IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, Vol.43, No.11, November 1996

46. Jagdish C. Patra, RanendraN.Pal, B.N.Chatterji ,and Ganapati Panda "*Identification of Nonlinear Dynamic Systems Using Functional Link Artificial Neural Networks* ", IEEE transactions on systems, man, and cybernetics, -PartB, Cybernetics, vol.29, No.2, April 1999

47. R. W. Brockett and M. R. Mesarovic, "*The reproducibility of multivariable control systems*", J. Math. Anal. Applicant. vol. 11, pp. 548-563, 1965

48. U. Kotta, "*Right inverse of a discrete-time nonlinear system*," International Journal of Contr., vol. 51, pp. 1-9, 1990

49. D. S. Broomhead and D. H. Lowe, "*Multivariable functional interpolation and adaptive networks*," Complex Syst., vol. 2, pp. 321–355, 1988

50. G.Xu,H.Liu,L.Tong,andT.Kailath,"*A least squares approach to blind channel identification*", IEEE Trans. Signal Processing,vol.43,pp.2982–2993,Dec.1995

51. Y.-H. Pao, S. M. Phillips and D. J. Sobajic, "*Neural-net computing and intelligent control systems,*" Int. J. Contr., vol. 56, no. 2, pp. 263–289,1992

52. J. C. Patra, "*Some studies on artificial neural networks for signal processing applications,*" Ph.D. dissertation, Indian Inst. Technol., Kharagpur, Dec. 1996

53. A. U. Levin and K. S. Narendra, "*Control of nonlinear dynamical system using neural networks, Part E. Observability and identification*," Ctr. Syst. Sci., Yale Univ., New Haven

54. CT, Tech. Rep. 9116, June 1992. -, "*Control of nonlinear dynamical systems using neural net- works-Controllability and stabilization*," IEEE Tram. Neural Networks, Reading, MA: Addison-Wesley, 1983. vol. 4, Mar. 1993

55. Dhrubajyoti Kalita, "*Identification and control of non-linear dynamical systems using neural networks*", January 30, 1998

56. J. C. Patra, R. N. Pal and B. N. Chatterji, "*FLANN-based identification of nonlinear systems*," Proc. 5th Eur. Congr. Intelligent Techniques Soft Computing (EUFIT'97), Sept. 1997, Aachen, Germany, vol. 1, pp.454–458

57. T. Yamada and T. Yabuta, "*Dynamic system identification using neural networks*," IEEE Trans. Syst., Man, Cybern., vol. 23, pp. 204–211, Jan./Feb. 1993

58. Kirill R. Chernyshov *V.A*. Trapeznikov, "*An Information Theoretic Approach to Neural Network Based System Identification*", 100-107 Siberian Conference on Control and Communications SIBCON–2009

59. Ivan Petrovic, Mato Baotic, Nedjeljko Peric "*Model structure selection for nonlinear system identification using feed forward neural networks"* Department of Control and Computer Engineering in Automation, Unska 3, HR – 10 000 Zagreb, Croatia.

60. D. Jiang and J. Wang, "*On-line learning of dynamical systems in the presence of model mismatch and disturbances*,'' IEEE Trans. Neural Networks, vol. 11, pp. 1272-1283, Nov. 2000

61. M.GoriandA.Tesi, "*On the problem of local minima in back propagation*," IEEETrans.PatternAnal.Mach.Intell.,vol.14,no.1,pp.76–86,Jan.1992
62. Zhang Huaguang and Quan Yongbing, "*Modeling, Identification, and Control of a Class of Nonlinear Systems*", IEEE Transactions on fuzzy systems, vol. 9, No. 2, APRIL 2001 349-354

63. Faisal A. Mohamed' Heikki Koivo, "*Modeling of induction motor using identification nonlinear neural network systems*", (*977-82)* PR000U04/0000-0977 Y4CO *0* 2004 SICE

64. N. V. Bhat et al., "*Modeling chemical process systems via neural computation*, IEEE Contr. Syst. Mag., pp. 24–29, Apr. 1990

65. Jiancheng Liu Xuping, "*Identification of Nonlinear Dynamical Systems using A Higher Order Multi-Layer Neural Networks*", ICSP2006 Proceedings0-7803-9737-1/06©2006 IEEE

66. Klaus Robert Muller, Sebastian Milka, Gunnar Ratsch, Koji Tsuda and Bernard Schollkopf, "*An introduction to kernel based learning algorithms*", IEEE transactions on neural networks, vol.12,no 2,pp 181-201,march 2001

67. C. W. Li and Y. K. Feng, "*Functional reproducibility of general multivariable analytic nonlinear systems,*" Int. J. Contr., vol. 45, pp. 255-268, 1987

68. S. Chen, S.A Billings, C.F.N Cowan and P.M Grant, "*Practical identification of NARMAX models using radial basis functions*", International journal of control,vol.52,no.6,pp 1327-1350,1990

69. D. Psaltis, A. Sideris, and A. A. Yamamura, "*A multilayered neural network controller*," IEEE Contr. Syst. Mag., vol. 8, pp. 17-21, 1988

70. C. Wiegand, C. Hedayat, W. John, Lj. Radi´c-Weissenfeld and U. Hilleringmann "*Nonlinear Identification of Complex Systems using Radial Basis Function Networks and Model Order Reduction*", 1-4244-1350-8/07/©2007 IEEE

71. Thomas F**.** Junge and Heinz Unbehauen, "*On-line Identification of Nonlinear Time-Variant SystemsUsing Structurally Adaptive Radial Basis Function Networks*", Proceedings of the American Control Conference Albuquerque, New Mexico June 1997 0-7a03-3a32-41971 I0 9 97 AACC

72. Nan Xie and Henry Leung*, "Blind Equalization Using a Predictive Radial Basis Function Neural Network*" IEEE Transactions on Neural Networks, Vol.16, No.3, May 2005

73. Ji Zhao Wei Chen Wenbo Xu, "*Nonlinear System Identification Based On Radial Basis Function Neural Network Using Improved Particle Swarm Optimization*", 2009 Fifth International Conference on Natural Computation(409-413)

74. S. V. T. Elanayar and Y. C. Shin, "*Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems*," IEEE Trans. Neural Networks, vol. 5, pp. 594–603, July 1994

75. Lou Haichuan, Dai Wenzhan Su Jie, "*Control of Time Varying Nonlinear System Based on RBFNN-DMC Algorithm*", 2008 International Conference on computational Intelligence and

Security978-0-7695-3508-1/08 © 2008 IEEEDOI 0.1109/CIS.2008.110134-138

76. S. Chen, S. A. Billings and P. M. Grant "*Recursive hybrid algorithm for nonlinear system identification using radial basis function networks*," Int. J. Contr., vol. 55, no.5, pp. 1051–1070, 1992

77. Nenad Todorovic, Petr Klan, "*State of the Art in Nonlinear Dynamical System Identification using Artificial Neural Networks*", 8th Seminar on Neural Network Applications in Electrical Engineering, NEUREL-2006 ME_E Faculty of Electrical Engineering, University of Belgrade, Serbia, September 25-27, 2006/1-4244-0433-9/06©2006 IEEE 103-108

78. Won-Kuk Son, Kenneth E. Bollinger and Chang-Goo Lee, *"Black-box Model and Adaptive Control Identification Using Neural Networks for Fast Time-Varying Nonlinear Systems*", 0-7803-3280-6/96/$5.00 '1996 IEEE - 356 -60

79. S.Amari, A.Cichocki, and H.Yang, "*Recurrent neural networks for blind separation of sources*," in Proc. Int. Symp. Nonlinear Theory and Applications, Dec.1995,pp.37–42

80. Chao-Chee Ku and Kwang Y. Lee, "*Nonlinear System Identification Using Diagonal Recurrent Neural Networks*", Department of Electrical and Computer Engineering The Pennsylvania State University, University Park, PA 16802
81. Han-Fu Chen, "*Recursive Approach to System Identification and Control*", ThPP.1 Proceedings of the 7th Asian Control Conference, Hong Kong, China, August 27-29,2009 (1-8)

82. A. González-Olvera and Yu Tang, "*Black-Box Identification of a Class of Nonlinear Systems by a Recurrent Neuro-fuzzy Network Marcos*", IEEE TRANSACTIONS ON NEURAL NETWORKS1045-9227/ © 2010 IEEE

83. Aleksander Malinowski, Damon A. Miller, Jacek M. Zurada, "*Reduction of Neural Network Models for identification and Control of Nonlinear Systems",* 0-7803-3210-5/1996 IEEE 2250-54

84. Ibrahim Haskarat and Umit Ozgunert tVisteon, "*Estimation based discrete-time sliding control of uncertain nonlinear systems in discrete strict feedback form*", Proceedings of the 39" IEEE Conference on Decision and Control Sydney, Australia December, *2000*0-7803-6638-7/*0 2000* IEEE 2599-2604

85. Pia0 Yingguo' Yang Zhenqiang Li Yanquan, "*A fuzzy-neural adaptive control for MIMO nonlinear system with application*", Proceedings of the 3d World Congress on Intelligent Control and Automation June 28-July 2,2000, Hefei, P.R. China(935-939)

86. G. Vanderstleen, Y. Rolain, J. Schoukens and R. Pintelon, "*On the Use of System Identification for Accurate Parametric Modeling of Non-Linear Systems using Noisy Measurements*", 723-728)-- 0-7803-261 5-6/95/ *0* 1995 IEEE

87. Xuan Han and Wen-Fang Xie, "*Nonlinear Systems Identification and Control using Dynamic Multi-Time Scales Neural Networks*", Proceedings of the IEEE International Conference on Automation and Logistics Shenyang, China August 2009 (96-101)

88. D. H. Nguyen and B.Widrow, "*Neural networks for self-learning control systems*," Int. J. Contr., vol. 54, no. 6, pp. 1439–1451, 1991

89. Y.-H. Pao, "*Adaptive Pattern Recognition and Neural Networks*", Reading, MA: Addison-Wesley, 1989

90. Simon Haykin, "*Artificial neural networks: A comprehensive foundation 2/e"*, Pearson education, inc, India, ISBN-81-708-300-0

91. Yaakov Bar-Shalaom and Xiao-Rong Li, *Estimation and Tracking: Principles, Techniques and Software*. Artech House, Boston, London, 1993.

92. G.V. Puskorius and L.A Feldkamp, " *Neuro Control of nonlinear dynamical systems with Kalman Filter trained recurrent networks"* IEEE Transactions on neural networks, Vol 5, No.2, pp 279-297, 1994

93. M.S. Grewal and A.P Andrews, "*Kalman Filtering Theory and Practice"*, Prentice Hall, Englewood Cliffs, 1993

94. A.V. Balakrishnan, "*Kalman Filtering Theory, Optimization Software*", Inc. Publications Division, Newyork,1992

95. R.N.Lobbia, S.C.Stubberud, and M.W.Owen, "*Adaptive extended Kalman filter using artificial neural networks,*" Int. J. Smart Eng. Syst. Des., vol.1, pp.207–221,1998

96. Y. Linguni, H. Sakai, H. Tokumaru, "*A Real Time Learning Algorithm for Multilayered Neural Network based on the*

Extended Kalman Filter", IEEE Transactions on Signal Processing, Vol 40, No.4, pp 959-966,1992.

97. Simon Haykin, "*Adaptive Filter Theory"*, Prentice Hall International editions, 1986.

98. Anders Forsgren and Robert Kling *"Implementation of Recurrent Neural Networks for Prediction and control of Nonlinear Dynamic Systems"*, Lulea University of Technology, Lulea, Sweden.

99. Ben James, Brian D.O, Anderson, and Robert. C. Williamson "*Conditional Mean and Maximum Likelihood approaches to Multi harmonic frequency estimation*.", IEEE Transactions on Signal Processing, Vol 42, No.6, pp 1366-1375, June 1994.

100. Langford B White, "*Robust Approximate likelihood ratio Tests for Nonlinear dynamic systems*", IEEE Transactions on Signal Processing, Vol 43, No.8, pp 2028-2031, August 1995.

101. Shuhi Li, "*Comparative Analysis of back propagation and Extended Kalman filter in Pattern and Batch forms for training Neural Networks*", IEEE Transactions on Neural Network, Vol 2, No.1, pp144-149, 2001

102. Ronghui Zhan and Jianwei Wan, "*Neural Network-Aided Adaptive Unscented Kalman Filter for Nonlinear State Estimation*", IEEE Signal Processing Letters 1-1070-9908,2006 IEEE

103. Chen, R. and J.S. Liu (2000), "*Mixture Kalman filters*", Journal of the Royal Statistical Society 62(3), 493–508

104. R.E. Kalman, "A *new approach to linear filtering and prediction problems*", Trans. ASME, Journal of Basic Engineering, Mar. 1960, pp.35-45

105. R.E. Kalman and R.S. Bucy, "*New results in liner filtering and prediction Theory*," Trans. ASME, Journal of Basic Engineering, Mar, 1961, pp. 95-108

106. C.K.Chui and Chen, "*Kalman Filtering with Real Time Applications*", Springer Verlag, New York, pp.108-118, 1991

107. A.H. Jaswinski, "*Stochastic Processes and Filtering Theory*", Academic Press, 1970

108. Radhakrishnan.K, Unnikrishnan.A, and Balakrishnan K.G, "*EM based extended kalman filter for estimation of rotor time constant of induction motor*," in 2006 Proc. International symposium on industrial electronics, (ISIE 06), Montreal, Que Canada., vol. 3, pp. 2434-2438

109. Radhakrishnan.K, Unnikrishnan.A, and Balakrishnan K.G, "*Model recognition for estimation of direction of arrival using extended Kalman filter*," in 2006 Proc. Third IASTED International con. Signal processing, pattern recognition, and Applications, Innsbruck, Austria, pp. 1-5

110. E. A. Wan and A. T. Nelson, "*Neural dual extended Kalman filtering: applications in speech enhancement and monaural blind signal separation*", In Proc. Neural Networks for Signal Processing Workshop. IEEE, 1997

111. E. A. Wan and R. van der Meme, "*The Unscented Bayes Filter-Technical report*", CSLU, Oregon Graduate Institute of Science and Technology, 2000. In preparation (http://cslu.cse.ogi.edu/nsel)

112. S. J. Julier and J. K. Uhlmann. "*A New Extension of the Kalman Filter to Nonlinear Systems*", the 11th Inr. Symp. on Aerospuce/Dejence Sensing, Simulation and Controls, 1997

113. G. Puskorius and L. Feldkamp, "*Decoupled Extended Kalman Filter Training of Feed forward Layered Networks*", In IJCNN, volume 1, pages 771-777, 1991

114. S. Singhal and L. Wu, "*Training multilayer perceptrons with the extended Kalman filter*", In Advances in Neural Information Processing Systems 1, pages 133-140, San Mateo, CA, 1989. Morgan Kauffman

115. Joost H. de Vlieger and Robert H.J. Gmelig Meyling , "*Maximum Likelihood Estimation for Long Range Target Tracking Using Passive Sonar Measurements*", IEEE transactions on Signal Processing, Vol. 40, No.5, pp 1216-1225,May 1992.

116. A.P.Dempster,N.M.Laird, and D.B.Rubin, "*Maximum likelihood estimation from incomplete data via EM algorithm*,"J.Roy.Statist.Soc.SER.,vol.39,pp.1–38,1977

117. K.Abed Meraimand, E.Moulines, "*A maximum likelihood solution to blind identification of multichannel FIRfilters*," in Proc. EUSIPCO, Edinburgh, Scotland, vol.2,1994,pp.1011–1014

118.M.GhoshandC.L.Weber,"*Maximum likelihood blind equalization*," presented at the1991SPIE Conf., San Diego, CA, July1991

119.Er-Wei Bai, "*Non-Parametric Nonlinear System Identification: An Asymptotic Minimum Mean Squared Error Estimator*", IEEE Transactions on Automatic Control. November 2009

120.R. J. Williams and J. Peng, "*An efficient gradient-based algorithm for on-line training of recurrent network trajectories*," Neural Computation, vol. 2, pp. 490-501, 1990

121.A. F. Atiya and A. G. Parlos, "*New results on recurrent network training: Unifying the algorithms and accelerating convergence*," IEEE Trans. Neural Networks, vol. 11, pp. 697-709, May 2000

122.Zhiping Lin, Qiyue Zou, E. Sally Ward, and Raimund J. Ober, "*Cramer–Rao Lower Bound for Parameter Estimation in Nonlinear Systems*", IEEE Signal Processing Letters Vol. 12, NO. 12, December 2005

123. Yonina C. Eldar*, Member, "Uniformly Improving the Cramér-Rao Bound and Maximum Likelihood Estimation*", IEEE Signal Processing Letters Vol.54, No.8, August, 2006

124. R.Niu, P.Willet and Y. Bar Shalom, "*Matrix CRLB scaling due to measurements of uncertain origin*," IEEE Trans. On Signal Processing, vol. 49, pp. 1432-1335, July 2001

125.J.H. Taylor, "*The Cramer – Rao estimation error lower bound computation for deterministic non-linear systems*," IEEE Trans. Automatic Control, vol.25, pp. 117-119,Feb. 1980

126. Gordon, N.J., D.J. Salmond and A.F.M. Smith, "*A novel approach to nonlinear/non-Gaussian Bayesian state estimation*". In: IEEE Proceedings on Radar and Signal Processing. Vol. 140.pp. 107–113. (1993)

127. Christophe Andrieu Arnaud Doucet Dpts Vladislav B. Tadi´c, "*On-Line Parameter Estimation in General State-Space Models*", Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005 Seville, Spain, December 12-15, 20050-7803-9568-9/05©2005 IEEE 332-37

128. Radu Dogaru, A.T. Murgan, S. Ortmann, M. Glesne, "*Searching for Robust Chaos in Discrete Time Neural Networks using Weight Space Exploration*", 0-7803-321 0-5/96 $4.0001996 IEEE 688-94

129. Xiaojun Yang1,2,3, Kunlin Shi2, and Keyi Xing1, "*Joint parameter and state estimation based on particle filtering and stochastic approximation*", International Conference on Computational Intelligence for Modeling Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-AWTIC'06) 0-7695-2731-0/© 2006

130. J. Humberto Pérez-Cruz and Alexander Poznyak, "*Trajectory Tracking based on Differential Neural Networks for a Class of Nonlinear Systems*", 2009 American Control Conference Hyatt Regency Riverfront, St. Louis, MO, USA June 10-12, 200992940-2945)

131. Dmitry Malyuk 'I, Georgy Boyarintsev, "*Identification and Forecasting of Sharp Changes in Economical Systems by Tracking of Local Lyapunov Exponents*", IEEE lntemational Workshop *00* Intelligent Data Acquisition and Advanced Computing System: Technology and Applications 8-10 September ZW3, Lviv, U b w0-7803-8138-6/03 02003 IEEE *272-76*

132. R. van der Merwe, J. F. G. de Freitas, A. Doucet, and E. A. Wan, "*The Unscented Particle Filter*", Technical report, Dept. of Engineering, University of Cambridge, 2000. In preparation

133. Doucet, A., N. Gordon and V. Krishnamurthy, "*Particle Filters for State Estimation of Jump Markov Linear Systems*". IEEE Transactions on Signal Processing 49(3), 613–624, 2001

134. J. Jansson, R. Karlsson and Nordlund P-J , " *Particle filters for positioning, navigation and tracking*", IEEE Transactions on Signal Processing, 2002

135. M.S Arulampalm, B.Ristic,N.Gordon and T.Mansell, "*Bearing Only Tracking of Manoeuvring Targets Using Particle Filters*," Eurasip Journal of Applied Signal Processing, Vol. 15, 2004,pp. 2351-2365

136. V. Kadirkamanathant, M. H. Jawaxdt, S. G. Fabri and M. Kadirkamanathan "*Particle Filters for Recursive Model Selection in Linear and Nonlinear System Identification*", Proceedings of the 39th IEEE Conference on Decision and Control Sydney, Australia December, 2000-7803-6638-7/2000 IEEE 2391-96

137. Marcos del Toro Peral, Fernando Gomez Bravo Alberto Martinho Vale, "*State Variables Estimation Using Particle Filter: Experimental Comparison with Kalman Filter*"-4244-0830-X/07 ©2007 IEEE

138. TIAN-Zengshan, LUO Lei "*Particle Filter positioning and tracking Based on dynamic model*", 4th IEEE Conference on Automation Science and Engineering Key Bridge Marriott, Washington DC, USA August 23-26, 2008978-1-4244-2023-0/08/©2008 IEEE. 756-759

139. *Jayesh H. Kotecha and Petar M. Djuri C*, "Gaussian sum particle filtering for dynamic state space models",0-7803-7041-4/0102001 IEEE 3465-68

140. Z. Zhu and H. Leung, "*Adaptive identification of nonlinear systems with application to chaotic communications*," *IEEE Trans. Circuits Sys t. I.*, Vol. 47, No. 7, pp. 1072–1080, Jul. 2000

141. J. C. Patra and R. N. Pal, "*A functional link artificial neural network for adaptive channel equalization*," Signal Process., vol. 43, pp. 181–195, May 1995

142. Bergman, N. (1999), "*Recursive Bayesian Estimation: Navigation and Tracking Applications*", Dissertation nr. 579. Link¨oping University, Sweden

143. Doucet, A., de Freitas, N. and Gordon, N., Eds (2001a)., "*Sequential Monte Carlo Methods in Practice*", Springer Verlag.

144. Nordlund, P.J. (2002), "*Sequential Monte Carlo Filters and Integrated navigation*", Licenciate thesis, Thesis No. 945

145.Yaakov Bar-Shalom and Xiao_Rong Li, "*Estimation and Tracking: Principles, Techniques and Software*", Artech House, London

146.Chuan Li, Yun Bai, Xianming Zhang, Hongjun Xia and Jing Chen, "*Inverse Dynamic System Identification Using Multiple Support Vector Machines*", 2009 International Association of Computer Science and Information Technology - Spring Conference (35-39)

147.Katsumi Konishi, Hiroaki Kato, "*A System Identification Method with Roughly Quantized Data using Semi definite Programming*"'978-1-4244-4649-0/09/2009 IEEE 1426-1431

148.Cao Wen-Mmgl Lu Feil Faig Hao' Wang sliiiojiie, "*Nonlinear Systems Using Two-Layer DBF Neural Networks and Application for Model identification Structure*", ICSP'O4 Proceedings 0-7X03-X406-7104/*GI* 2004 IEEE

149.Issam AMRI 1, Dhaou SOUDANI 1 and Mohamed Benrejeb, "*Exponential stability and stabilization of linear systems with time varying delays*", 2009 6th International Multi-Conference on Systems, Signals and Devices978-1-4244-4346-8/09©2009 IEEE

150.Gustavo Camps-Valls*, Member, IEEE*, Manel Martínez-Ramón*, Senior Member, IEEE*, José Luis Rojo-Álvarez*, Member, IEEE*, and Jordi Muñoz-Marí, "*Nonlinear System Identification With Composite Relevance Vector Machines*", IEEE Signal Processing Letters, vol. 14, No. 4, April 2007(279-282)

151. Wlodzimierz Greblicki, "*Nonparametric Identification of Wiener Systems*", IEEE Transaction on Information theory, vol. 38, No. 5, September1992 1487-1493

152. Byung-hwa Lee, Sang-un Kim, Jin-wook Seok and Sangchul Won, " *Nonlinear System Identification based on Support Vector Machine using Particle Swarm Optimization*", SICE-ICASE International Joint Conference 2006 Oct. 18-21, 2006 in Bexco, Busan, Korea89-950038-5-5 98560/06/ ICASE 5614-18

153. Ali, M. Ashfaq and Chr. Schmid, "*Model reference adaptive control of nonlinear systems using RLLM networks*", Proceedings of the American Control Conference Chicago, Illinois June 20000-7803-551 9-9/*0 2000* AACC 1659-63

154. Xiaodong Wang, Weifeng Liang, Xiushan Cai, Ganyun Lv, Changjiang Zhang and Haoran Zhang, "*Application of Adaptive Least Square Support Vector Machines in Nonlinear System Identification*", Proceedings of the 6th World Congress on Intelligent Controland Automation, June 21 - 23, 2006, Dalian, China1-4244-0332-4/062006 IEEE 1897-900