USE OF SOFT COMPUTING TECHNIQUES FOR TRANSDUCER PROBLEMS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

in

Telematics & Signal Processing

By

MOHAMMAD NASEEM

Roll No. - 20607029

Department of Electronics and Communication Engineering

National Institute Of Technology

Rourkela

2006 - 2008

USE OF SOFT COMPUTING TECHNIQUES FOR TRANSDUCER PROBLEMS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

in

Telematics & Signal Processing

By

MOHAMMAD NASEEM

Roll No. - 20607029

Under the Guidance of

Prof. S. K. PATRA

Department of Electronics and Communication Engineering

National Institute Of Technology

Rourkela

2006 - 2008

Department of Electronics and Communication
NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA
ORISSA, INDIA - 769 008

CERTIFICATE

This is to certify that the thesis titled "**USE OF SOFT COMPUTING TECHNIQUES FOR TRANSDUCER PROBLEMS** ", submitted to the National Institute of Technology, Rourkela by **Mr. Mohammad Naseem**, Roll No. **20607029** for the award of the degree of Master of Technology in **Electronics & Communication Engineering** ***(Telematics & Signal Processing)***, is a bona fide record of research work carried out by him under my supervision and guidance.

The candidate has fulfilled all the prescribed requirements.

The thesis, which is based on candidate's own work, has not been submitted elsewhere for a degree/diploma.

In my opinion, the thesis is of standard required for the award of a Master of Technology degree in Electronics & Communication Engineering.

To the best of my knowledge, he bears a good moral character and decent behavior.

Date:

Prof. Sarat Kumar Patra
Department of Electronics & Communication Engineering
National Institute of Technology
Rourkela-769 008 (INDIA)
Email: skpatra@nitrkl.ac.in

ACKNOWLEDGEMENTS

This project is by far the most significant accomplishment in my life and it would be impossible without people who supported me and believed in me.

I would like to extend my gratitude and my sincere thanks to my honorable, esteemed supervisor **Prof. S. K. Patra,** Department of Electronics and Communication Engineering. He is not only a great lecturer with deep vision but also and most importantly a kind person. I sincerely thank for his exemplary guidance and encouragement. His trust and support inspired me in the most important moments of making right decisions and I am glad to work with him.

I want to thank all my teachers **Prof. G.S. Rath**, **Prof. G. Panda, Prof. K. K. Mahapatra, Prof. T.K. Dan** and **Prof. S. Meher** for providing a solid background for my studies and research thereafter. They have been great sources of inspiration to me and I thank them from the bottom of my heart.

I would like to thank all my friends and especially my classmates for all the thoughtful and mind stimulating discussions we had, which prompted us to think beyond the obvious. I've enjoyed their companionship so much during my stay at NIT, Rourkela.

I would like to thank all those who made my stay in Rourkela an unforgettable and rewarding experience.

Last but not least I would like to thank my parents, who taught me the value of hard work by their own example. They rendered me enormous support during the whole tenure of my stay in NIT Rourkela.

MOHAMMAD NASEEM

CONTENTS

ABSTRACT

In many control system applications Linear Variable Differential Transformer (LVDT) plays an important role to measure the displacement. The performance of the control system depends on the performance of the sensing element. It is observed that the LVDT exhibits the same nonlinear input-output characteristics. Due to such nonlinearities direct digital readout is not possible. As a result we employ the LVDTs only in the linear region of their characteristics. In other words their usable range gets restricted due to the presence of nonlinearity. If the LVDT is used for full range of its nonlinear characteristics, accuracy of measurement is severely affected. So, to reduce this nonlinearities different ANN techniques is being used such as single neuron structure, MLP structure, RBFNN and ANFIS structure.

Another problem considered here is with flow measurement. Generally flow measurements uses conventional flow meters for feedback on the flow-control loop cause pressure drop in the flow and in turn lead to the usage of more energy for pumping the fluid. An alternative approach for determining the flow rate without flow meters is thought. The restriction characteristics of the flow-control valve are captured by a neural network (NN) model. The relationship between the flow rate and the physical properties of the flow as well as flow-control valve, that is, pressure drop, pressure, temperature, and flow-control valve coefficient (valve position) is found. With these accessible properties, the NN model yields the flow rate of fluid across the flow-control valve, which acts as a flow meter. The viability of the methodology proposed is illustrated by real flow measurements of water flow which is widely used in hydraulic systems.

Control of fluid flow is essential in process-control plants. The signal of flow measured using the flow meter is compared with the signal of the desired flow by the controller. The controller output accordingly adjusts the opening/closing actuator of the flow-control valve in order to maintain the actual flow close to the desired flow. Typically, flow meters of comparatively low cost such as turbine-type flow meters and venturi-type meters are used to measure the volumetric quantity of fluid flow in unit time in a flow process. However, the flow

meter inevitably induces a pressure drop in the flow. In turn, this results in the use of more energy for pumping the fluid. To avoid this problem, non-contact flow meters, i.e. electromagnetic-type flow meters, have been developed and are widely used in process plants not only because there is no requirement for installation in the pipeline but also because introduction to the differential pressure across pipelines is not necessitated. Unfortunately, the cost of such non-contact measurement is comparatively much higher than that of its conventional counterparts.

Here, an alternative approach is proposed to obtain the fluid flow measurement for flow-control valves without the pressure drop and the consequent power loss that appear in conventional flow meters. Without the flow meter, it is a fact that the flow rate can be determined from the characteristics of the control valve for flow measurements. In this method, the restriction characteristics of the control valve embedded in a neural network (NN) model are used in determining the flow rate instead of actual measurement using a conventional flow meter.

LIST OF ABBREVIATIONS

ADC	Analog-to-Digital Converter
ANFIS	Adaptive Neuro Fuzzy Inference System
ANN	Artificial Neural Network
ASP	Adaptive Signal Processing
BP	Back-Propagation algorithm
DAC	Digital-to-Analog Converter
FL	Fuzzy Logic
FPGA	Field Programmable Gate Array
FSO	Full-Scale Output
LMS	Least Mean Square
LVDT	Linear Variable Differential Transformer
MDS	Minimum Detectable Signal
MLP	Multi-Layer Perceptron
MR	Measured Range
MSE	Mean Square Error
LPH	Liter per Hour
PIM	Plug-In-Module
RBFNN	Radial Basis Function based Neural Network
VLSI	Very Large Scale Integration

LIST OF FIGURES

LIST OF TABLE

Chapter 1

INTRODUCTION

The sensors are devices which, for the purpose of measurement, turn physical input quantities into electrical output signals, their output-input and output-time relationship being predictable to a known degree of accuracy at specified environmental conditions. The definition of sensors or transducers according to The Instrument Society of America is *"a device which provides a usable output in response to a specified measurand"*. Here the output is an 'electrical quantity' and measurand is a 'physical quantity'. It can also be defined as an element that senses a variation in input energy to produce a variation in another or same form of energy is called a sensor, whereas, transducer involves a transduction principle which converts a specified measurand into an usable output.

This Chapter deals with the fundamental of the sensors and their characteristics. Section 1.1 deals with the sensor fundamental along with its characteristics. The literature survey is discussed in Section 1.2. Problem formulation of the thesis is depicted in Section 1.3. Finally the Chapter-wise organization of the thesis is presented in Section 1.4.

1.1 Sensors Fundamentals

The sensor consists of several elements or blocks such as sensing element, signal conditioning element, signal processing element and data presentation element.

Sensing element: This is in contact with the process and gives an output which depends in some way on the variable to be measured. Examples are: thermocouple where millivolt emf depends on temperature, capacitive pressure sensor where capacitance of a chamber depends on pressure, linear variable differential transformer where emf at the secondary coil depends on displacement, etc.

If there is more than one sensing element in a system, the element in contact with the process is termed the primary sensing element, the others secondary sensing elements.

Signal conditioning element: This takes the output of the sensing element and converts it into a form more suitable for further processing, usually a dc voltage, dc current or frequency signal. Examples are: deflection bridge which converts an impedance change into a voltage change, amplifier which amplifies millivolts to volts, etc.

Signal processing element: This takes the output of the conditioning element and converts it into a form more suitable for presentation. Examples are: analog-to-digital converter (ADC) which converts a voltage into a digital form for input to a computer; a microcontroller which calculates the measured value of the variable from the incoming digital data.

Data presentation element: This presents the measured value in a form which can be easily recognized by the observer. Examples are: a simple pointer-scale indicator, chart recorder, and alphanumeric display etc.

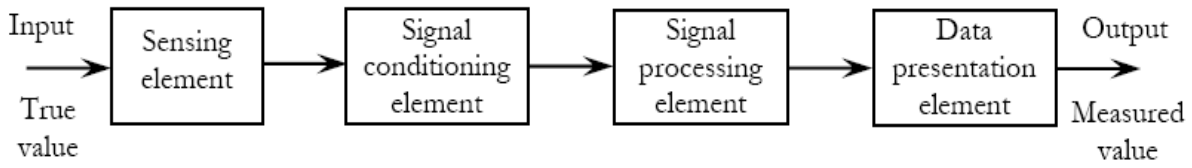

Fig 1.1 General structure of sensor

1.1.1 Characteristics of sensor

Sensors or all measurement systems, have two general characteristics, i.e. (i) *static characteristics*, and (ii) *dynamic characteristics*.

The static characteristics comprise of:

(a) Accuracy: These characteristics specified by error. And it is given by,

$$\varepsilon_a\% = (x_m - x_t)\frac{100}{x_t}$$

Where *t* is for true value, *m* for measured value and variable *x* stands for measurand. This is often expressed for the full scale output and is given by,

$$\varepsilon_{fso}\% = (x_m - x_t)\cdot 100 \Big/ x_{fso}$$

(b) Precision: It describes how far a measured quantity is reproducible as also how close it is to the true value.

Term *repeatability* is similar to precision which is the difference in output *y* at a given value of the input *x* when this is obtained in two consecutive measurements. It may be expressed as *% Full-scale Output (FSO)*. Fig 1.2 shows the repeatability.

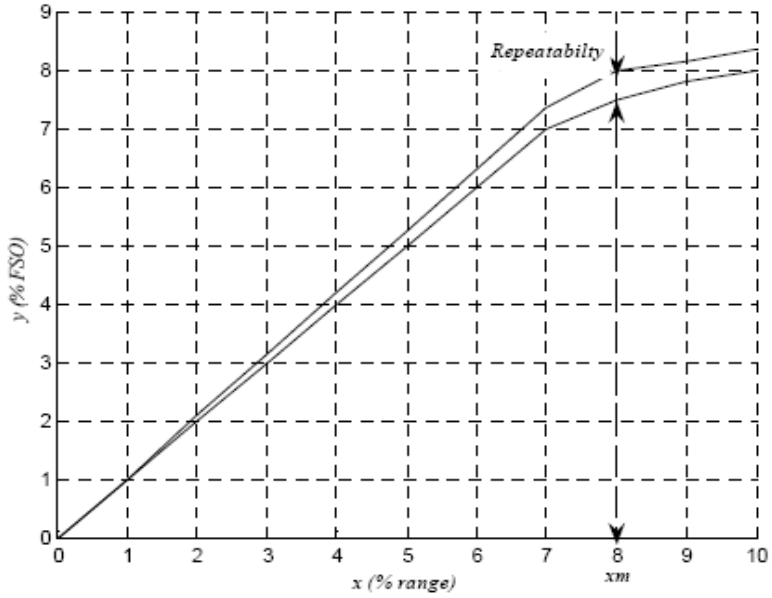

Fig 1.2 Repeatability

(c) Resolution: It is defined as the smallest incremental change in the input that would produce a detectable change in the output. This is often expressed as percentage of the measured range, MR. The measured range is defined as the difference of the maximum input and the minimum input, x_{max}-x_{min} = MR. For a detectable output Δy, if the minimum change in x is Δxmin, then the maximum resolution is

$$R_{\max}(\%) = 100(\Delta x)_{\min} / MR$$

(d)Threshold: This is the smallest input change that produces a detectable output at zero value condition of the measured.

(e)Sensitivity: This is defined as the ratio of the incremental output (Δy) to incremental input (Δx), i.e

$$S = \Delta y / \Delta x$$

(f)Nonlinearity: The deviation from linearity, which itself is defined in terms of superposition principles, is expressed as a percentage of the full scale output at a given value of the input. Nonlinearity can, however, be specified here.

• deviation from a straight line joining the end points of the scale.

These are shown in Figs. below

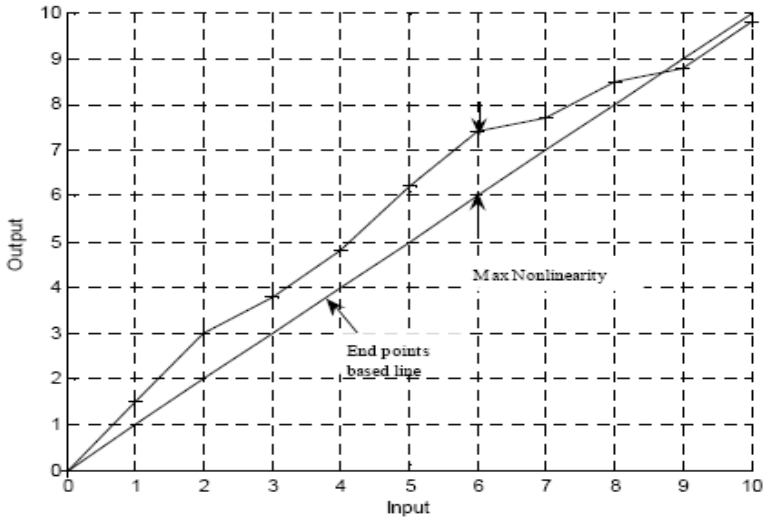

Fig 1.3 Nonlinearity curve

(g) Hysteresis: It is the difference in the output of the sensor *y* for a given input *x* when *x* reaches the value in upscale and downscale directions as shown in Fig. below. The causes are different for different types of sensors. In magnetic types, it is lag in alignment of the dipoles, while in semiconductor types it is the injection type slow traps producing the effect, and so on.

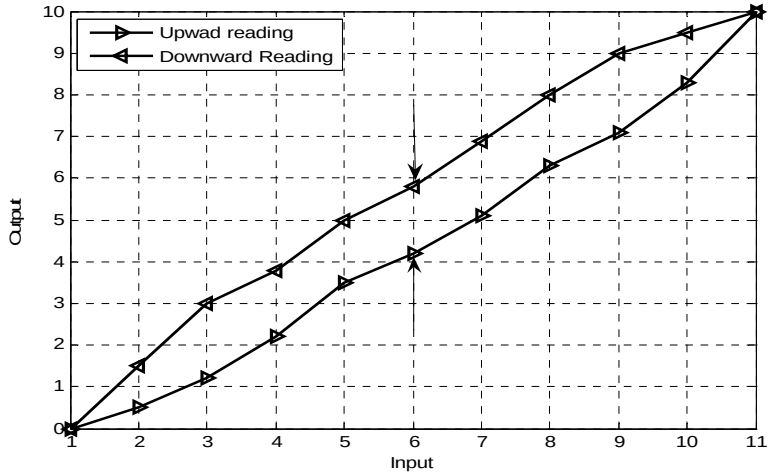

Fig 1.4 Hysteresis curve

The **dynamic characteristics** involve determination of transfer function, frequency response, impulse response and step response and then evaluation of the time-dependent outputs. The two important parameters in these connections are:

• Fidelity determined by dynamic error, and

• Speed of response determined by lag.

For determining the *dynamic characteristics* different inputs are given to the sensor and the response characteristics are to be studied. With step input, the specifications in terms of the time constant of the sensor are to be made. Impulse response as well as its Fourier transform is also to be considered for time domain as well as frequency domain studies.

Environmental Parameters: The external variables such as temperatures, pressure, humidity, vibration, etc. which affects the performance of the sensor. Aging is also an important parameter of the sensor. These parameters are not the ones that are to be sensed. For any environmental parameters, the performance of the sensor can be studied in terms of its effect on the *static* and *dynamic characteristics*. For this study, one environmental parameter at a time is considered variable while others are fixed.

1.2 Literature Survey

In many control system applications Linear Variable Differential Transformer (LVDT) plays an important role to measure the displacement. The performance of the control system depends on the performance of the sensing element. It is observed that the LVDT exhibits the same nonlinear input-output characteristics. Due to such nonlinearities direct digital readout is not possible. As a result we employ the LVDTs only in the linear region of their characteristics. In other words their usable range gets restricted due to the presence of nonlinearity. If the LVDT is used for full range of its nonlinear characteristics, accuracy of measurement is severely affected.

The nonlinearity present is usually time-varying and unpredictable as it depends on many uncertain factors. Attempts have been made by many researchers to increase the range of linearity of LVDT. In the conventional design of LVDT sophisticated and precise winding machines are used to compensate the nonlinearity effects. The nonlinearity compensation can also be achieved by square coil method in which the core moves perpendicular to the axis instead of along the axis. A self-compensated LVDT has been modeled using a dual secondary coil which is insensitive to the variation in excitation current and frequency. Some soft computing

techniques have been employed on LVDT to achieve better sensitivity and to implement the signal conditioning circuits.

Flow measurements using conventional flow meters for feedback on the flow-control loop cause pressure drop in the flow and in turn lead to the usage of more energy for pumping the fluid. An alternative approach for determining the flow rate without flow meters is thought. The restriction characteristics of the flow-control valve are captured by a neural network (NN) model. The relationship between the flow rate and the physical properties of the flow as well as flow-control valve, that is, pressure drop, pressure, temperature, and flow-control valve coefficient (valve position) is found. With these accessible properties, the NN model yields the flow rate of fluid across the flow-control valve, which acts as a flow meter. The viability of the methodology proposed is illustrated by real flow measurements of water flow which is widely used in hydraulic systems.

Control of fluid flow is essential in process-control plants. The signal of flow measured using the flow meter is compared with the signal of the desired flow by the controller. The controller output accordingly adjusts the opening/closing actuator of the flow-control valve in order to maintain the actual flow close to the desired flow. Typically, flow meters of comparatively low cost such as turbine-type flow meters and venturi-type meters are used to measure the volumetric quantity of fluid flow in unit time in a flow process. However, the flow meter inevitably induces a pressure drop in the flow. In turn, this results in the use of more energy for pumping the fluid. To avoid this problem, non-contact flow meters, i.e. electromagnetic-type flow meters, have been developed and are widely used in process plants not only because there is no requirement for installation in the pipeline but also because introduction to the differential pressure across pipelines is not necessitated. Unfortunately, the cost of such non-contact measurement is comparatively much higher than that of its conventional counterparts.

Here, an alternative approach is proposed to obtain the fluid flow measurement for flow-control valves without the pressure drop and the consequent power loss that appear in conventional flow meters. Without the flow meter, it is a fact that the flow rate can be determined from the characteristics of the control valve for flow measurements. In this method,

the restriction characteristics of the control valve embedded in a neural network (NN) model are used in determining the flow rate instead of actual measurement using a conventional flow meter.

1.3 Problem Formulation

The sensors exhibit nonlinear characteristics which limit the dynamic range of these devices. As a result the direct digital readout of the output is not possible for the whole input range of the sensors. In addition the full potential of the sensors cannot be utilized. Thus there is a real challenge for designing and implementing novel sensors which circumvent the nonlinearity problems associated with them. In recent past few attempts have been made but not much has been achieved in this direction. The second problem is the accuracy of measurements in these sensors which is greatly affected by aging of the sensor, temperature and humidity variations.

Both these issues detoriate the accuracy of measurement. Therefore nonlinearity compensation is required to achieve accurate measurement under adverse conditions. Most of the existing nonlinearity compensation techniques work well for fixed and known type of nonlinearities. But in actual practice the nonlinearity behavior of the sensors changes with time and is usually unknown. The adaptive inverse model of the LVDTs can be designed and connected in series with these devices, so that the associated nonlinearity can be compensated and direct digital readout is possible for the whole input range. A scheme of direct modeling and inverse modeling of the LVDTs using different ANN structure has been developed. The direct modeling is proposed for calibration of inputs and estimation of internal parameters of the LVDTs. The purpose of the direct model is to obtain an ANN model of the LVDTs in such a way that the outputs of the LVDTs and the ANN match closely. Once a model of the LVDTs is available, it may be used for fault detection of the sensor. The LVDTs output provides a voltage signal proportional to the displacement change.. And the inverse modeling is proposed for estimation of applied input displacement. The adaptive inverse model can be developed using soft-computing techniques such as Artificial Neural Networks (ANN) and Neuro-Fuzzy algorithms are potential candidates for developing adaptive inverse model of these devices.

Another problem for flow measurements using conventional flow meters for feedback on the flow-control loop cause pressure drop in the flow and in turn lead to the usage of more

energy for pumping the fluid. An alternative approach for determining the flow rate without flow meters is thought. The restriction characteristics of the flow-control valve are captured by a

neural network (NN) model. The relationship between the flow rate and the physical properties of the flow as well as flow-control valve, that is, pressure drop, pressure, temperature, and flow-control valve coefficient (valve position) is found. With these accessible properties, the NN model yields the flow rate of fluid across the flow-control valve, which acts as a flow meter. The viability of the methodology proposed is illustrated by real flow measurements of water flow which is widely used in hydraulic systems.

Control of fluid flow is essential in process-control plants. The signal of flow measured using the flow meter is compared with the signal of the desired flow by the controller. The controller output accordingly adjusts the opening/closing actuator of the flow-control valve in order to maintain the actual flow close to the desired flow. Typically, flow meters of comparatively low cost such as turbine-type flow meters and venturi-type meters are used to measure the volumetric quantity of fluid flow in unit time in a flow process. However, the flow meter inevitably induces a pressure drop in the flow. In turn, this results in the use of more energy for pumping the fluid. To avoid this problem, non-contact flow meters, i.e. electromagnetic-type flow meters, have been developed and are widely used in process plants not only because there is no requirement for installation in the pipeline but also because introduction to the differential pressure across pipelines is not necessitated. Unfortunately, the cost of such non-contact measurement is comparatively much higher than that of its conventional counterparts. Here, an alternative approach is proposed to obtain the fluid flow measurement for flow-control valves without the pressure drop and the consequent power loss that appear in conventional flow meters. Without the flow meter, it is a fact that the flow rate can be determined from the characteristics of the control valve for flow measurements. In this method, the restriction characteristics of the control valve embedded in a neural network (NN) model are used in determining the flow rate instead of actual measurement using a conventional flow meter.

1.4 Chapter-wise Organization

The chapter-wise organization of the thesis is outlined below.

CHAPTER 1 Introduction

Brief Introduction

1.1 Sensors Fundamentals

1.1.1Characteristics of sensor

1.2 Literature Survey

1.3 Problem Formulation

1.4 Chapter-wise Organization

CHAPTER 2 Soft Computing Techniques and Algorithm

2.1 Introduction

2.2 Artificial Neural Network

2.2.1 Single Neuron Structure

2.2.2 Multi-Layer Perceptron (MLP)

2.2.3 Radial Basis Function Neural Network (RBF-NN)

2.2.4 Adaptive Neuro Fuzzy Inference System (ANFIS)

2.3 Summary and Discussion

CHAPTER 3 Nonlinear Compensation of Linear Variable Differential Transformer (LVDT) Using Soft Computing Techniques

Brief Introduction

3.1 Linear Variable Differential Transformer (LVDT)

3.1.1 Signal Conditioning for LVDTs

3.1.2 Common Specifications

3.1.3 Pros and Cons

3.2 Nonlinearity Compensation of LVDTs

3.3 Simulation Studies

3.3.1 ADALIN Based Non-linearity Compensation

3.3.2 MLP Based Non-linearity Compensation

3.3.3 RBFNN Based Non-linearity Compensation

3.3.4 ANFIS Based Non-linearity Compensation

3.4 Comparison and Discussion

Chapter 2

SOFT COMPUTING TECHNIQUES AND ALGORITHMS

2.1 Introduction

In recent years, a growing field of research in "Adaptive Systems" has resulted in a variety of adaptive automations whose characteristics in limited ways resemble certain characteristics of living systems and biological adaptive processes. An adaptive automation is a system whose structure is alterable or adjustable in such a way that its behavior and performance improves by its environment. A simple example of an adaptive system is the automatic gain control used in radio and television receiver. The most important factor in adaptive system is its time-varying and self-adjusting performance. Their characteristic depends upon the input signal. If a signal is applied to the input of adaptive system to test its response characteristic, the system adapts to this specific input and thereby changes its parameters. Based on the different neural architecture of human brain, different Artificial Neural Algorithms are developed such as Artificial Neural Network (ANN), Multi-Layer Perceptron (MLP), Radial Basis Function (RBF), Adaptive Neuro Fuzzy Inference System (ANFIS) etc. These are capable of mapping the input and output nonlinearly.

This Chapter deals with different types of adaptive algorithms, which are used as a tool to compensate nonlinearity problem of different sensors. The present Chapter is organized as follows: Section 2.2 deals with different types of Artificial Neural Network (ANN) such as Single Neuron Network (ADALIN), Multi-Layer Perceptron (MLP), Radial Basis Function based Neural Network (RBFNN) and Adaptive Neuro Fuzzy Inference System (ANFIS). Finally the summary and discussion is presented in Section 2.3.

2.2 Artificial Neural Network

Artificial neural network (ANN) takes their name from the network of nerve cells in the brain. Recently, ANN has been found to be an important technique for classification and optimization problem. McCulloch and Pitts have developed the neural networks for different computing machines. There are extensive applications of various types of ANN in the field of communication, control and instrumentation. The ANN is capable of performing nonlinear mapping between the input and output space due to its

large parallel interconnection between different layers and the nonlinear processing characteristics. An artificial neuron basically consists of a computing element that performs the weighted sum of the input signal and the connecting weight. The sum is added with the bias or threshold and the resultant signal is then passed through a nonlinear function of sigmoid or hyperbolic tangent type. Each neuron is associated with three parameters whose learning can be adjusted; these are the connecting weights, the bias and the slope of the nonlinear function. For the structural point of view a NN may be single layer or it may be multilayer.

In multilayer structure, there is one or many artificial neurons in each layer and for a practical case there may be a number of layers. Each neuron of the one layer is connected to each and every neuron of the next layer. The functional-link ANN is another type of single layer NN. In this type of network the input data is allowed to pass through a functional expansion block where the input data are nonlinearly mapped to more number of points. This is achieved by using trigonometric functions, tensor products or power terms of the input. The output of the functional expansion is then passed through a single neuron.

The learning of the NN may be supervised in the presence of the desired signal or it may be unsupervised when the desired signal is not accessible. Rumelhart developed the Back-propagation (BP) algorithm, which is central to much work on supervised learning in MLP. A feed-forward structure with input, output, hidden layers and nonlinear sigmoid functions are used in this type of network. In recent years many different types of learning algorithm using the incremental back-propagation algorithm, evolutionary learning using the nearest neighbor MLP and a fast learning algorithm based on the layer-by-layer optimization procedure are suggested in literature. In case of unsupervised learning the input vectors are classified into different clusters such that elements of a cluster are similar to each other in some sense. The method is called competitive learning, because during learning sets of hidden units compete with each other to become active and perform the weight change. The winning unit increases its weights on those links with high input values and decreases them on those with low input

values. This process allows the winning unit to be selective to some input values. Different types of NNs and their learning algorithms are discussed in sequel.

2.2.1 Single Neuron Structure

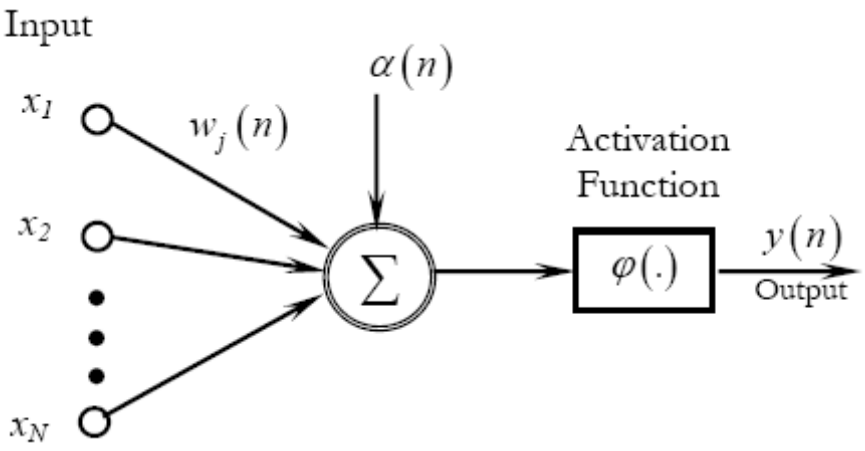

Fig 2.1 Structure of Single Neuron

The basic structure of an artificial neuron is presented in Fig.. The operation in a neuron involves the computation of the weighted sum of inputs and threshold. The resultant signal is then passed through a nonlinear activation function. This is also called as a perceptron, which is built around a nonlinear neuron; whereas the LMS algorithm described in the preceding sections is built around a linear neuron. The output of the neuron may be represented as,

$$y(n) = \varphi\left[\sum_{j=1}^{N} w_j(n)\, x_j(n) + \alpha(n)\right] \tag{2.1}$$

Where is the threshold to the neurons at the first layer, $w_j(n)$ is the weight associated with j^{th} the input, *N* is the no. of inputs to the neuron and φ(.) is the nonlinear activation function. Different types of nonlinear function are shown in Fig.

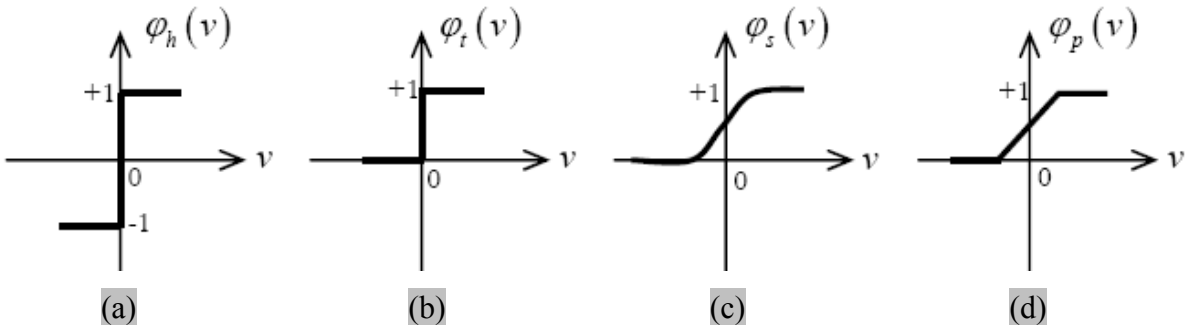

Fig 2.2 Different types of nonlinear activation function,

(a) Signum function or hard limiter,

(b) Threshold function,

(c) Sigmoid function,

(d) Piecewise Linear

Signum Function: For this type of activation function, we have

$$\varphi(v) = \begin{cases} 1 & if\ v > 0 \\ 0 & if\ v = 0 \\ -1 & if\ v < 0 \end{cases} \tag{2.2}$$

Threshold Function: This function is represented as,

$$\varphi(v) = \begin{cases} 1 & if\ v \geq 0 \\ 0 & if\ v < 0 \end{cases} \tag{2.3}$$

Sigmoid Function: This function is S-shaped, is the most common form of the activation function used in artificial neural network. It is a function that exhibits a graceful balance between linear and nonlinear behavior.

$$\varphi(v) = \frac{1}{1+e^{-av}} \tag{2.4}$$

Where v is the input to the sigmoid function and a is the slope of the sigmoid function. For the steady convergence a proper choice of a is required.

Piecewise-Linear Function: This function is

$$\varphi(v)=\begin{cases}1, & v\geq+\frac{1}{2}\\ v, & +\frac{1}{2}>v>-\frac{1}{2}\\ 0, & v\leq+\frac{1}{2}\end{cases} \qquad (2.5)$$

Where the amplification factor inside the linear region of operation is assumed to be unity. This can be viewed as an approximation to a nonlinear amplifier.

2.2.2 Multi-Layer Perceptron (MLP)

In the multilayer neural network or multilayer perceptron (MLP), the input signal propagates through the network in a forward direction, on a layer-by-layer basis. This network has been applied successfully to solve some difficult and diverse problems by training in a supervised manner with a highly popular algorithm known as the error back-propagation algorithm. The scheme of MLP using four layers is shown in Fig. below. $x_i(n)$ represent the input to the network, f_j and f_k represent the output of the two hidden layers and $y_l(n)$ represents the output of the final layer of the neural network. The connecting weights between the input to the first hidden layer, first to second hidden layer and the second hidden layer to the output layers are represented by w_{ij}, w_{jk} and w_{kl} respectively.

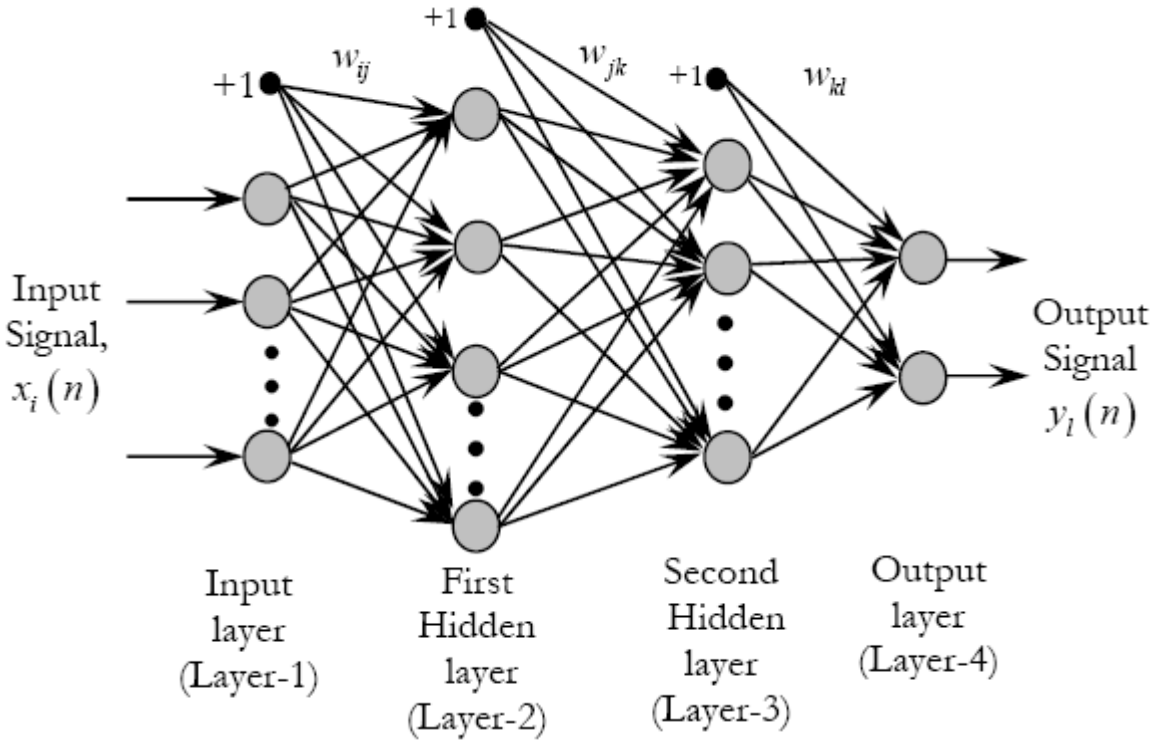

Fig 2.3 Structure of multilayer perceptron

If P_1 is the number of neurons in the first hidden layer, each element of the output vector of first hidden layer may be calculated as,

$$f_j = \varphi_j\left[\sum_{i=1}^{N} w_{ij}x_i(n) + \alpha_j\right], \quad i = 1,2,3,\ldots N,\ j = 1,2,3,\ldots P_1 \tag{2.6}$$

where α_j is the threshold to the neurons of the first hidden layer, N is the no. of inputs and $\varphi(.)$ is the nonlinear activation function in the first hidden layer of these type. The time index n has been dropped to make the equations simpler. Let P_2 be the number of neurons in the second hidden layer. The output of this layer is represented as, f_k and may be written as

$$f_k = \varphi_k\left[\sum_{j=1}^{P_1} w_{jk}f_j + \alpha_k\right], \quad k=1,\ 2,\ 3,\ \ldots,\ P_2 \tag{2.7}$$

where, α_k is the threshold to the neurons of the second hidden layer. The output of the final output layer can be calculated as

$$y_l(n) = \varphi_l\left[\sum_{k=1}^{P_2} w_{kl} f_k + \alpha_l\right], \quad l=1, 2, 3, \ldots, P_3 \tag{2.8}$$

Where, α_l is the threshold to the neuron of the final layer and P_3 is the no. of neurons in the output layer. The output of the MLP may be expressed as

$$y_l(n) = \varphi_n\left[\sum_{k=1}^{P_2} w_{kl}\varphi_k\left(\sum_{j=1}^{P_1} w_{jk}\varphi_j\left\{\sum_{i=1}^{N} w_{ij}x_i(n) + \alpha_j\right\} + \alpha_k\right) + \alpha_l\right] \tag{2.9}$$

2.2.3 Radial Basis Function Neural Network (RBF-NN)

The Radial Basis Function based neural network (RBFNN) consists of an input layer made up of source nodes and a hidden layer of large dimension. The number of input and output nodes is maintained same and while training the same pattern is simultaneously applied at the input and the output. The nodes within each layer are fully connected to the previous layer as shown in the Fig. 2.4. The input variables are each assigned to a node in the input layer and pass directly to the hidden layer without weights. The hidden nodes contain the radial basis functions (RBFs) which are Gaussian in characteristics. Each hidden unit in the network has two parameters called a center (μ), and a width (σ) associated with it. The Gaussian function of the hidden units is radially symmetric in the input space and the output of each hidden unit depends only on the radial distance between the input vector *x* and the center parameter μ for the hidden unit.

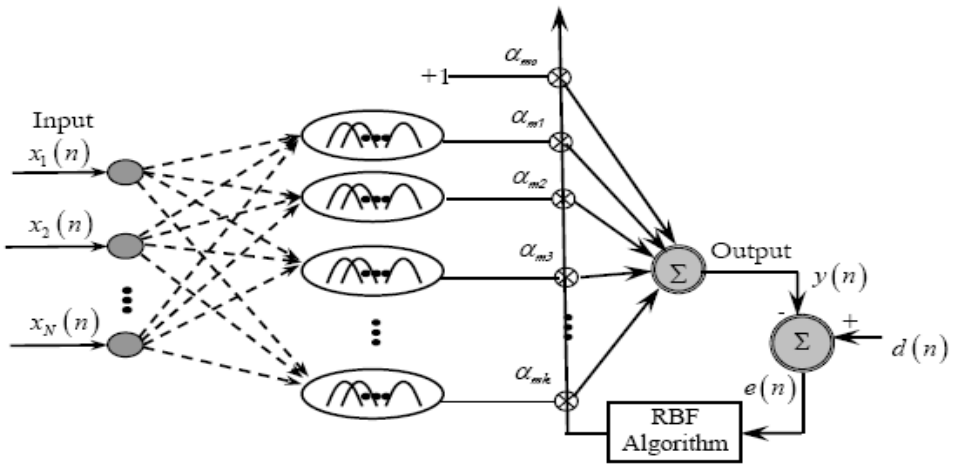

Fig 2.4 Radial Basis Function neural network structure

The Gaussian function gives the highest output when the incoming variables are closest to the center position and decreases monotonically as the distance from the center decreases. The response of each hidden unit is scaled by its connecting weights (α_{mi}'s) to the output units and then summed to produce the final network output and. The overall network output at the time index *n* is therefore

$$y(n) = \alpha_{m0} + \sum_{k=1}^{K} \alpha_{mk} \phi_k (x_j); \qquad j = 1, 2, ..., N \tag{2.10}$$

For each input x_j, *N* represents the no. of inputs, *K* = number of hidden units, α_{mk} = connecting weight of the k^{th} hidden unit to output layer, α_{mo} = bias term, *m* is the number of output.

The value of $\varphi_k (x_j)$ is given by

$$\phi_k(x_j) = \exp(-\frac{1}{\sigma_k^2} \left\| x_j - \mu_k \right\|^2) \tag{2.11}$$

Where μ_k is the center vector for the k^{th} hidden unit and σ_k is the width of the Gaussian function and denotes the Euclidean norm.

The parameters of the RBFNN are updated using the RBF algorithm. The RBF algorithm is an exact analytical procedure for evaluation of the first derivative of the output error with respect to network parameters. In the present paper we apply a three layer RBF network with nonlinear output units.

Let the error vector at the n^{th} instant is e(n) = d(n)-y(n) where d(n) = desired output vector and y(n) = estimated output vector. Let $\xi(n) = \frac{1}{2} \Sigma e^2(n)$. The update equations for the center and width of the Gaussian function as well as the connecting and bias weights are derived as

$$\mu_k(n+1) = \mu_k(n) + \Delta\mu_k(n) \tag{2.12}$$

$$\sigma_k(n+1) = \sigma_k(n) + \Delta\sigma_k(n) \tag{2.13}$$

$$\alpha_{mk}(n+1) = \alpha_{mk}(n) + \Delta\alpha_{mk}(n) \tag{2.14}$$

$$\alpha_{m0}(n+1) = \alpha_{m0}(n) + \Delta\alpha_{m0}(n) \tag{2.15}$$

where $\Delta\mu_k(n)$, are the change of the centers and spread of the Gaussian functions; the $\Delta\alpha_{mk}(n)$ and $\Delta\alpha_{mk0}(n)$ are the change in weights and the threshold of the RBFNN. These are computed by taking the partial derivative of $\xi(n)$ with respect to different network parameters. The key equations obtained are stated in (2.16) to (2.19).

$$\Delta\mu_k(n) = -\eta\frac{\partial\xi}{\partial\mu_k} = 2\eta e(n)\phi_k(x_j)\alpha_{mk}\frac{(x_j-\mu_k)}{\sigma_k^2} \tag{2.16}$$

$$\Delta\sigma_k(n) = -\eta\frac{\partial\xi}{\partial\sigma_k} = 2\eta e(n)\phi_k(x_j)\alpha_{mk}\frac{(x_j-\mu_k)^2}{\sigma_k^3} \tag{2.17}$$

$$\Delta\alpha_{mk}(n) = -\eta\frac{\partial\xi}{\partial\alpha_{mk}} = \eta e(n)\phi_k(x_j) \tag{2.18}$$

$$\Delta\alpha_{m0}(n) = -\eta\frac{\partial\xi}{\partial\alpha_{m0}} = \eta e(n) \tag{2.19}$$

where η is the learning rate parameter ($0 \leq \eta \leq 1$). By applying each input patterns, the change in center location, width of the Gaussian function as well as the connecting weights and bias weights are calculated.

2.2.4 Adaptive Neuro Fuzzy Inference System (ANFIS)

ANFIS is an adaptive network based on typical fuzzy inference systems, in which outputs have been obtained using fuzzy rules on inputs. The scheme of a 2 inputs 1 output system is depicted in Fig. 2.5; it is referred to type-3 reasoning, but can be extended, by means of simple adjustments, to type-1 and type-2 reasoning too. A node represented by a square has parameters and it is called adaptive node, while the circle shaped one has none and it is called fixed node; it is clear that the network in Fig. 2.5 has 5 layers, but only nodes of layer 1 and 4 are adaptive ones.

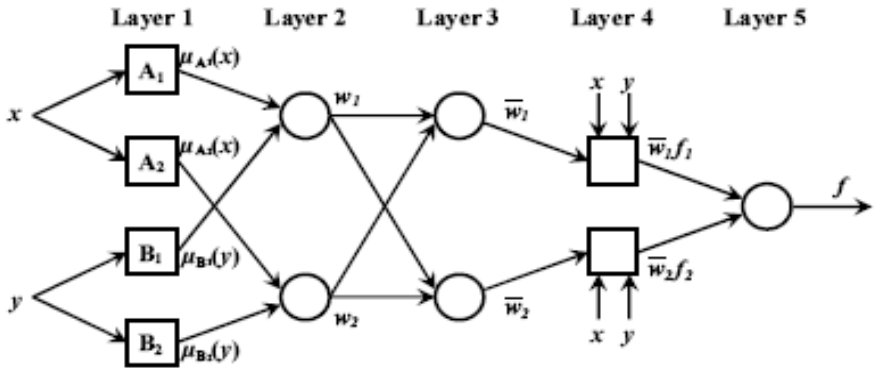

Fig 2.5 ANFIS Structure

Parameters of the first layer are called premise parameters and define the membership function of the node; normally this function is bell-shaped with maximum equal to 1 and minimum equal to 0. The bell-shaped function is given as:

$$\mu(x) = \frac{1}{1 + e^{-\frac{(x-c)^{2b}}{a^{2b}}}} \qquad (2.20)$$

Where x is the node input and a, b and c are the premise parameters

(A_i or $B_i = \{a_i, b_i, c_i\}$).

Layers 2 and 3 have no parameters; nodes on layer 2 apply a scaling factor to their inputs, while nodes on layer 3 generate the weights according to:

$$\overline{w}_i = \frac{w_i}{w_1 + w_2}, i = 1, 2 \qquad (2.21)$$

Where w_1 and w_2 are the outputs of the node 1 and 2 of second layer. Nodes of layer 4 perform a simple linear combination of system inputs x and y by means of parameters p, q, r, called consequent parameters, as reported in the following equation:

$$\overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i), i = 1, 2 \qquad (2.22)$$

These four layers represent the equivalent of the Takagi and Sugeno fuzzy "if-then" rule. The premise part is a linguistic label characterized by an appropriate membership function, thus allowing to capture the imprecise modes of reasoning typical of the human ability to make decisions in an environment of uncertainty and imprecision. On the contrary, the consequent part is described by a non-fuzzy equation of the input, as shown in the next equation:

If x is A and y is B then f = p x + q y + r (2.23)

Layer 5 simply computes the sum of the layer 4 outputs in order to obtain the system output; in a fuzzy system, generally, the output is the weighted sum of the rules results, but in this case the weights have already been considered by previous layers, so a simple sum is enough. The number of nodes in layer 1 represents the number of fuzzy

sets, while the dimension of layer 4 determines the number of fuzzy rules used in the system. Compared to neural networks, fuzzy rules can be considered as the equivalent of neurons, therefore, the more the fuzzy rules are employed, the more the elaboration performed by the system could result complex and flexible. Obviously, also the training

process results more complicated and more calibration points are needed to accomplish it. Since the ANFIS is equivalent to an adaptive network, typical training procedure of neural network, as the gradient descent method, can be applied. This technique uses the gradient method to identify parameters. The output **f** can be expressed as:

$$\begin{aligned} f &= \overline{w}_1 f_1 + \overline{w}_2 f_2 = \\ &(\overline{w}_1 x)p_1 + (\overline{w}_1 y)q_1 + \overline{w}_1 r_1 + \\ &+ (\overline{w}_2 x)p_2 + (\overline{w}_2 y)q_2 + \overline{w}_2 r_2 \end{aligned} \qquad (2.24)$$

2.3 Summary and Discussion

The various adaptive techniques used in the thesis are presented in this Chapter. The single layer network or perceptron and an adaptive filter using the LMS algorithm are naturally related, as evidenced by their weights updates. However, the perceptron and LMS algorithm differ from each other in some fundamental respects:

- The LMS algorithm uses a linear neuron, whereas the perceptron uses the nonlinear neuron.

- The learning process in the perceptron is performed for a finite number of iterations and then stops. In contrast, continuous learning takes place in the LMS algorithm.

For MLP, the back propagation learning is the standard algorithm. The back-propagation algorithm derives its name from the fact that the partial derivatives of the cost function (performance measure) with respect to the free parameters (synaptic weights and biases) of the network are determined by back-propagating the error signals (computed by output neurons) through the network, layer by layer.

The structure of an RBFNN is unusual in that the constitution of its hidden units is entirely different from that of its output units. Each hidden node consists of a Gaussian function. The learning algorithm of RBFNN is more or less equivalent to LMS algorithm.

Chapter 3

NONLINEAR COMPENSATION OF LINEAR VARIABLE DIFFERENTIAL TRANSFORMER (LVDT) USING SOFT COMPUTING TECHNIQUES

This chapter deals with the nonlinearity compensation of Linear Variable Differential Transformer (LVDT). In many practical control systems LVDT is used as the sensing element for displacement. The performance of the control system depends upon the performance of the sensing elements. Many researchers have worked to design LVDT with high linearity. In its conventional design methodology achieving high linearity involves complex design task. Sophisticated and precise winding machines are used to achieve that. It is difficult to have all LVDT fabricated in a factory at a time to be equally linear. LVDT having different nonlinearity present in a control system malfunctions at times because of the difference in sensor characteristics.

In Section 3.1, the electrical characteristics of LVDT are presented. Section 3.2 deals with the nonlinearity compensation of LVDT and its experimental set-up. The simulation studies are carried out using different ANN models (ADALIN, MLP, RBFNN and ANFIS) and the experimental dataset of a standard LVDT is dealt in Section 3.3. The result obtained from different models is compared in Section 3.4.

3.1 Linear Variable Differential Transformer (LVDT)

Linear displacement is movement in one direction along a single axis. A position or linear displacement sensor is a device whose output signal represents the distance an object has traveled from a reference point. A displacement measurement also indicates the direction of motion (See Figure 1).

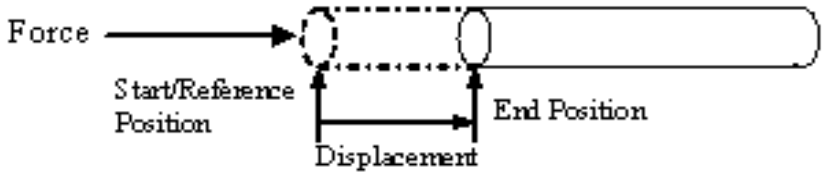

Fig 3.1 Linear Displacement measurement

A linear displacement typically has units of millimeters (mm) or inches (in.) and a negative or positive direction associated with it.

Linear variable differential transformers (LVDT) are used to measure displacement. LVDTs operate on the principle of a transformer. As shown in Figure 2, an LVDT consists of a

coil assembly and a core. The coil assembly is typically mounted to a stationary form, while the core is secured to the object whose position is being measured. The coil assembly consists of three coils of wire wound on the hollow form. A core of permeable material can slide freely through the center of the form. The inner coil is the primary, which is excited by an AC source as shown. Magnetic flux produced by the primary is coupled to the two secondary coils, inducing an AC voltage in each coil.

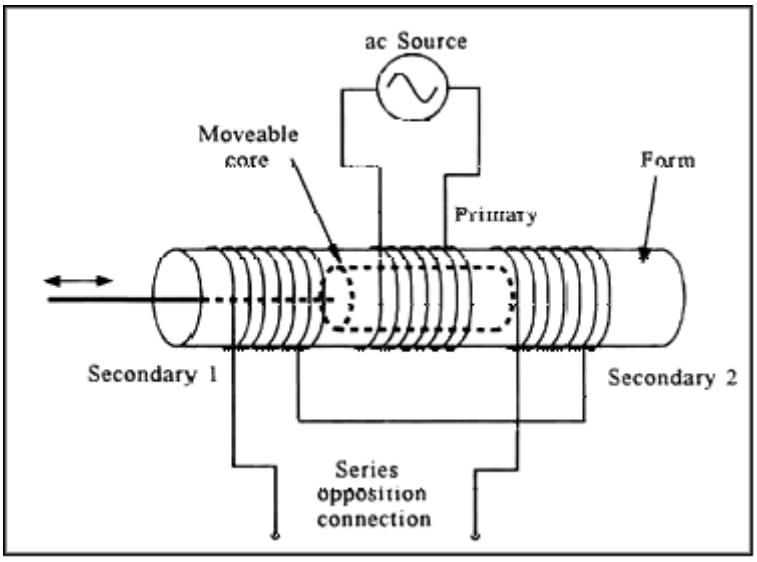

Figure 3.2 General LVDT Assembly

The main advantage of the LVDT transducer over other types of displacement transducer is the high degree of robustness. Because there is no physical contact across the sensing element, there is no wear in the sensing element.

Because the device relies on the coupling of magnetic flux, an LVDT can have infinite resolution. Therefore the smallest fraction of movement can be detected by suitable signal conditioning hardware, and the resolution of the transducer is solely determined by the resolution of the data acquisition system.

An LVDT measures displacement by associating a specific signal value for any given position of the core. This association of a signal value to a position occurs through electromagnetic coupling of an AC excitation signal on the primary winding to the core and back to the secondary windings. The position of the core determines how tightly the signal of the primary coil is coupled to each of the secondary coils. The two secondary coils are series-opposed, which means wound in series but in opposite directions. This results in the two signals

on each secondary being 180 deg out of phase. Therefore phase of the output signal determines direction and its amplitude, distance.

Figure 3.3 depicts a cross-sectional view of an LVDT. The core causes the magnetic field generated by the primary winding to be coupled to the secondaries. When the core is centered perfectly between both secondaries and the primary, as shown, the voltage induced in each secondary is equal in amplitude and 180 deg out of phase. Thus the LVDT output (for the series-opposed connection shown in this case) is zero because the voltages cancel each other.

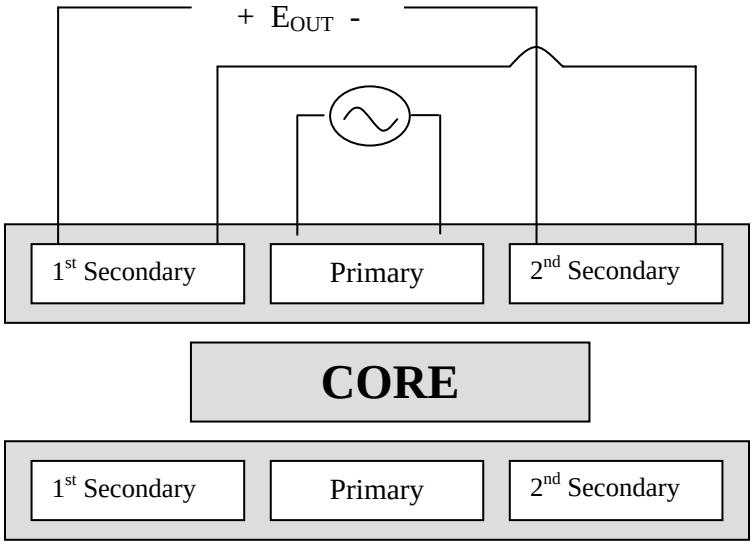

Figure 3.3 Cross-Sectional Views of LVDT Core and Windings

Displacing the core to the left (Figure 3.4) causes the first secondary to be more strongly coupled to the primary than the second secondary. The resulting higher voltage of the first secondary in relation to the second secondary causes an output voltage that is in phase with the primary voltage.

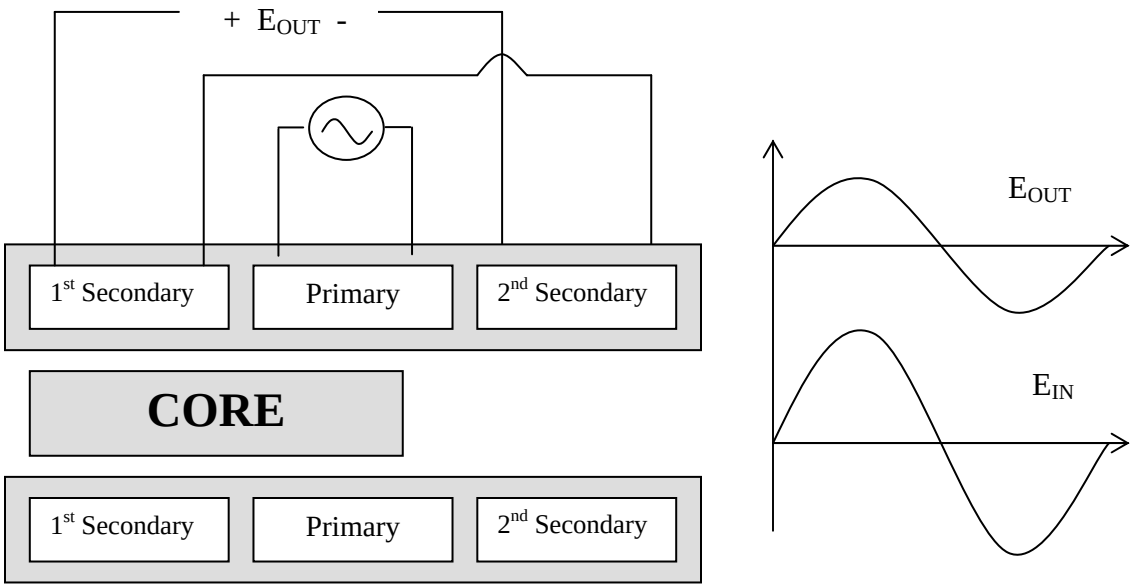

Figure 3.4 Coupling to First Secondary Caused by Associated Core Displacement

Likewise, displacing the core to the right causes the second secondary to be more strongly coupled to the primary than the first secondary. The greater voltage of the second secondary causes an output voltage to be out of phase with the primary voltage.

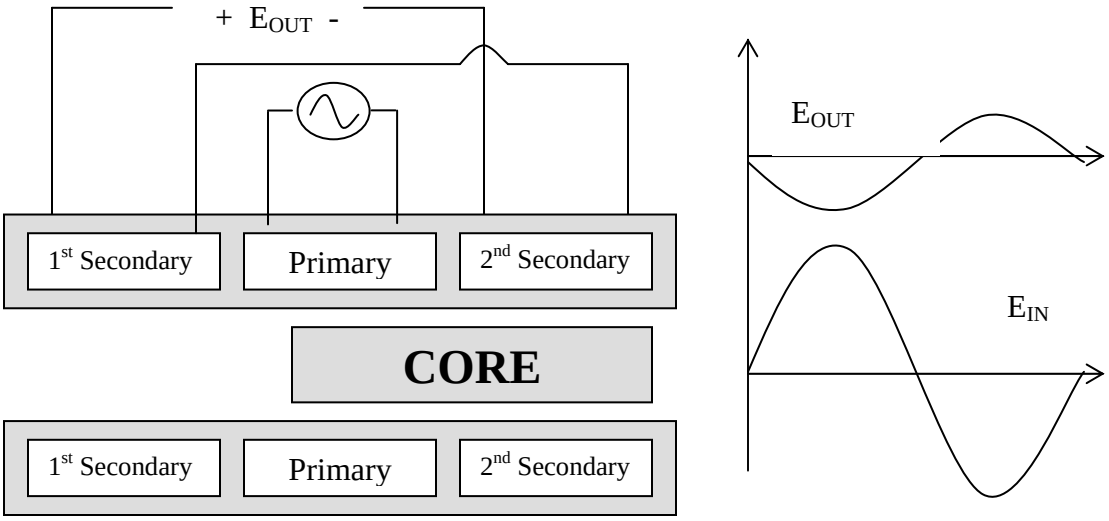

Figure 3.5 Coupling to Second Secondary Caused by Associated Core Displacement

To summarize, "The LVDT closely models an ideal zeroth-order displacement sensor structure at low frequency, where the output is a direct and linear function of the input. It is a variable-reluctance device, where a primary center coil establishes a magnetic flux that is coupled through a center core (mobile armature) to a symmetrically wound secondary coil on either side of the primary. Thus, by measurement of the voltage amplitude and phase, one can determine the extent of the core motion and the direction, that is, the displacement." Figure 3.6 shows the linearity of the device within a range of core displacement. Note that the output is not linear as the core travels near the boundaries of its range. This is because less magnetic flux is coupled to the core from the primary. However, because LVDTs have excellent repeatability, nonlinearity near the boundaries of the range of the device can be predicted by a table or polynomial curve-fitting function, thus extending the range of the device.

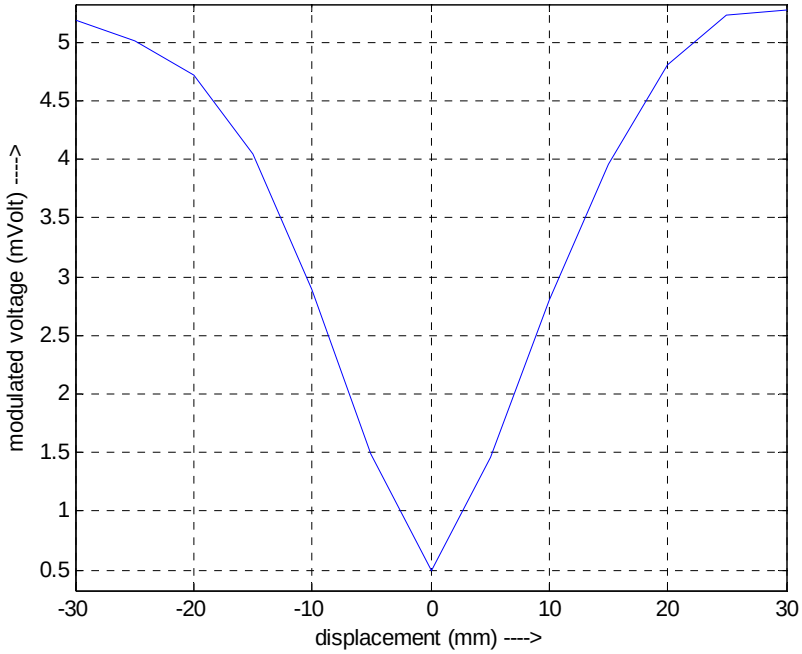

Figure 3. 6 Proportionally Linear LVDT Responses to Core Displacement

3.1.1 Signal Conditioning for LVDTs

Because the output of an LVDT is an AC waveform, it has no polarity. The magnitude of the output of an LVDT increases regardless of the direction of movement from the electrical zero position.

In order to know in which half of the device the center of the core is located, one must consider the phase of the output as well as the magnitude as compared to the AC excitation source on the primary winding. The output phase is compared with the excitation phase and it can be either in or out of phase with the excitation source, depending upon which half of the coil the center of the core is in.

The signal conditioning electronics must combine information on the phase of the output with information on the magnitude of the output, so the user can know the direction the core has moved as well as how far from the electrical zero position it has moved.

LVDT signal conditioners generate a sinusoidal signal as an excitation source for the primary coil. “This signal is typically between 50 Hz and 25 kHz. The carrier frequency is generally selected to be at least 10 times greater than the highest expected frequency of the core motion.” The signal conditioning circuitry synchronously demodulates the secondary output signal with the same primary excitation source. The resulting DC voltage is proportional to core displacement. The polarity of the DC voltage indicates whether the displacement is toward or away from the first secondary (displacement left or right).

Figure 3.7 shows a practical detection scheme, typically provided as a single integrated circuit (IC) manufactured specifically for LVDTs. The system contains a signal generator for the primary, a phase-sensitive detector (PSD) and amplifier/filter circuitry.

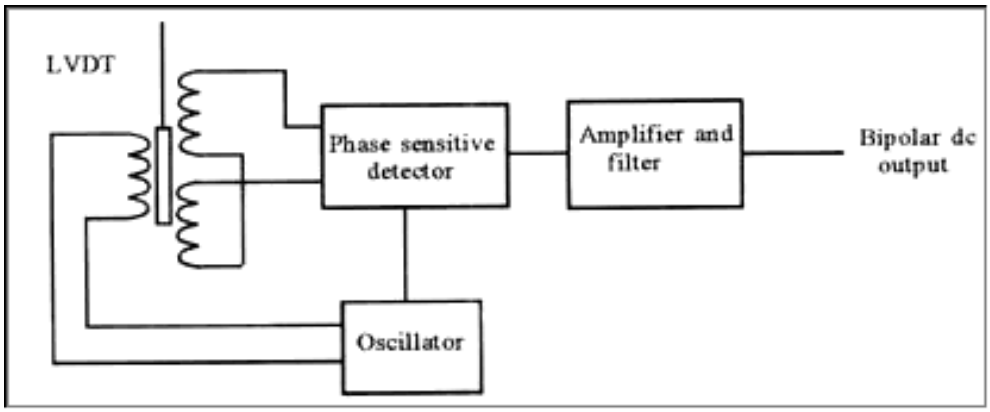

Figure 3.7 Sophisticated Phase-Sensitive LVDT Signal Conditioning Circuit

Broad ranges of LVDTs are available with linear ranges from at least ±50 cm down to ±1 mm. The time response is dependent on the equipment to which the core is connected. The units of an LVDT measurement are typically in mV/V/mm or mV/V/in. This indicates that for every volt of stimulation applied to the LVDT there is a definite feedback in mV per unit distance. A carefully manufactured LVDT can provide an output linear within ±0.25% over a range of core motion, with very fine resolution. The resolution is limited primarily by the ability of signal conditioning hardware to measure voltage changes.

3.1.2 Common Specifications

Common specifications for commercially available translational LVDT's are listed below:

Input: Power input is a 3 to 15 V (rms) sine wave with a frequency between 60 to 20,000 Hz (the two most common signals are 3 V, 2.5 kHz and 6.3 V, 60 Hz).

Stroke: Full-range stroke ranges from ±125 μm to ±75 mm (±0.005 to ±3 in).

Sensitivity: Sensitivity usually ranges from 0.6 to 30 mV per 25 μm (0.001 in) under normal excitation of 3 to 6 V. Generally, the higher the frequency the higher the sensitivity.

Nonlinearity: Inherent nonlinearity of standard units is on the order of 0.5% of full scale.

3.1.3 Pros and Cons

Pros

- Relative low cost due to its popularity.
- Solid and robust, capable of working in a wide variety of environments.
- No friction resistance, since the iron core does not contact the transformer coils, resulting in an infinite (very long) service life.
- High signal to noise ratio and low output impedance.
- Negligible hysteresis.
- Infinitesimal resolution (theoretically). In reality, displacement resolution is limited by the resolution of the amplifiers and voltage meters used to process the output signal.

- Short response time, only limited by the inertia of the iron core and the rise time of the amplifiers.

Cons

- The core must contact directly or indirectly with the measured surface which is not always possible or desirable. However, a non-contact thickness gage can be achieved by including a pneumatic servo to maintain the air gap between the nozzle and the work piece.
- Dynamic measurements are limited to no more than 1/10 of the LVDT resonant frequency. In most cases, this results in a 2 kHz frequency cap.

3.2 Nonlinearity Compensation of LVDTs

The proposed nonlinearity compensation scheme is shown in Fig. . In this scheme the LVDT can be controlled by a displacement actuator. The main controller gives an actuating signal to the displacement actuator, which displaces the core of the LVDT. The differential voltage of the LVDT after being demodulated does not keep linear relationship with the displacement. The nonlinearity compensator can be developed by using different ANN techniques like ADALIN, MLP, RBF-NN and ANFIS. The output of the ANN based nonlinearity compensator is compared with the desired signal (actuating signal of the displacement actuator) to produce an error signal. With this error signal, the weight vectors of the ANN model are updated. This process is repeated till the mean square error (MSE) is minimized. Once the training is complete, the LVDT together with the ANN model acts like a linear LVDT with enhanced dynamic range.

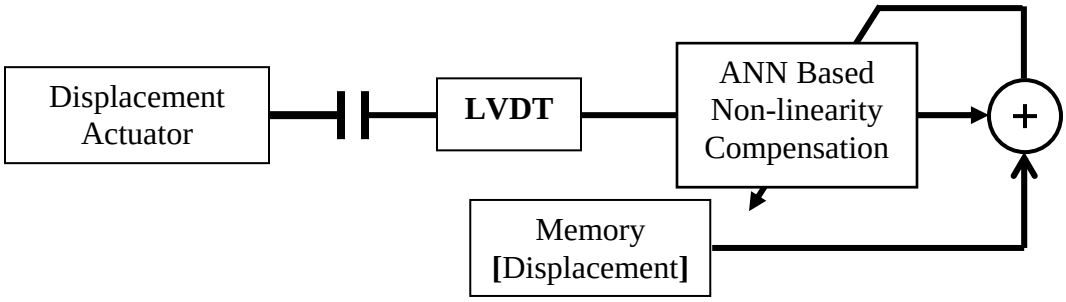

Fig 3.8 Scheme of nonlinearity compensation of LVDT

The practical set-up of the LVDT along with the ANN based nonlinearity compensator after the training is shown in Fig. 3.9.

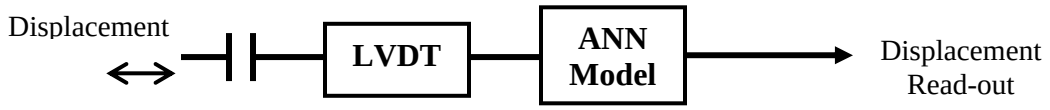

Fig. 3.9 Practical set-up of LVDT after training

3.3 Simulation Studies

To demonstrate the effectiveness these ANN based nonlinear compensators, computer simulation studies are carried out using experimental data obtained from a typical LVDT. The observation readings are:

Table 3.1 Experimental Measured Data

Displacement (in mm)	**Differential Output Voltage (e_{rms} in mVolt)**	**Demodulated Voltage Output (e in mVolt)**
-30	4.085	5.185
-25	3.956	5.017
-20	3.731	4.717
-15	3.221	4.039
-10	2.359	2.896
-5	1.273	1.494
Null position (0)	0.204	0.001
5	1.153	1.462
10	2.226	1.810
15	3.118	3.962
20	3.748	4.799
25	4.050	5.225
30	4.085	5.276

3.3.1 ADALIN Based Non-linearity Compensation

The differential or demodulated voltage *e* at the output of LVDT is normalized by dividing each value with the maximum value. The normalized voltage output *e* is subjected to functional expansion and than input to the single neuron perceptron based nonlinearity compensator. The output of neuron contains tanh(.) type activation function. The output of the ADALIN based nonlinearity compensator is compared with the normalized input displacement of the LVDT. The widrow-hoff algorithm, in which both the learning rates are chosen as 0.07, is used to adapt the weights of the Neuron. Applying various input patterns, the ANN weights are updated using the widrow-hoff algorithm. To enable complete learning, 1000 iterations are made. Then, the weights of neurons are frozen and stored in the memory. During the testing phase, the frozen weights are used in the ADALIN model. Fig. shows the nonlinearity compensation of LVDT by ADALIN. In this model the *MSE* is obtained to be 0.34 %

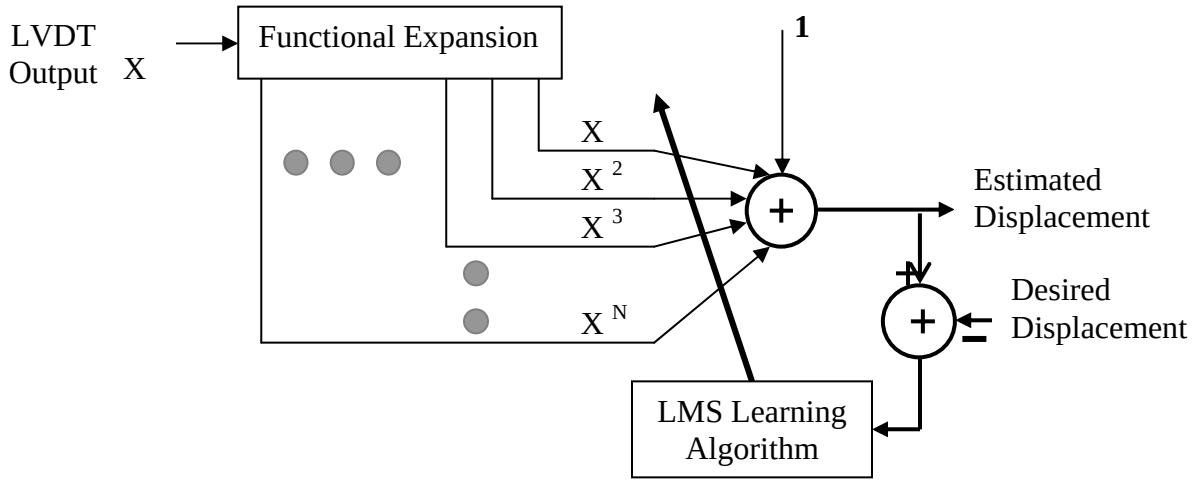

Fig 3.10 Architecture of ADALIN Network

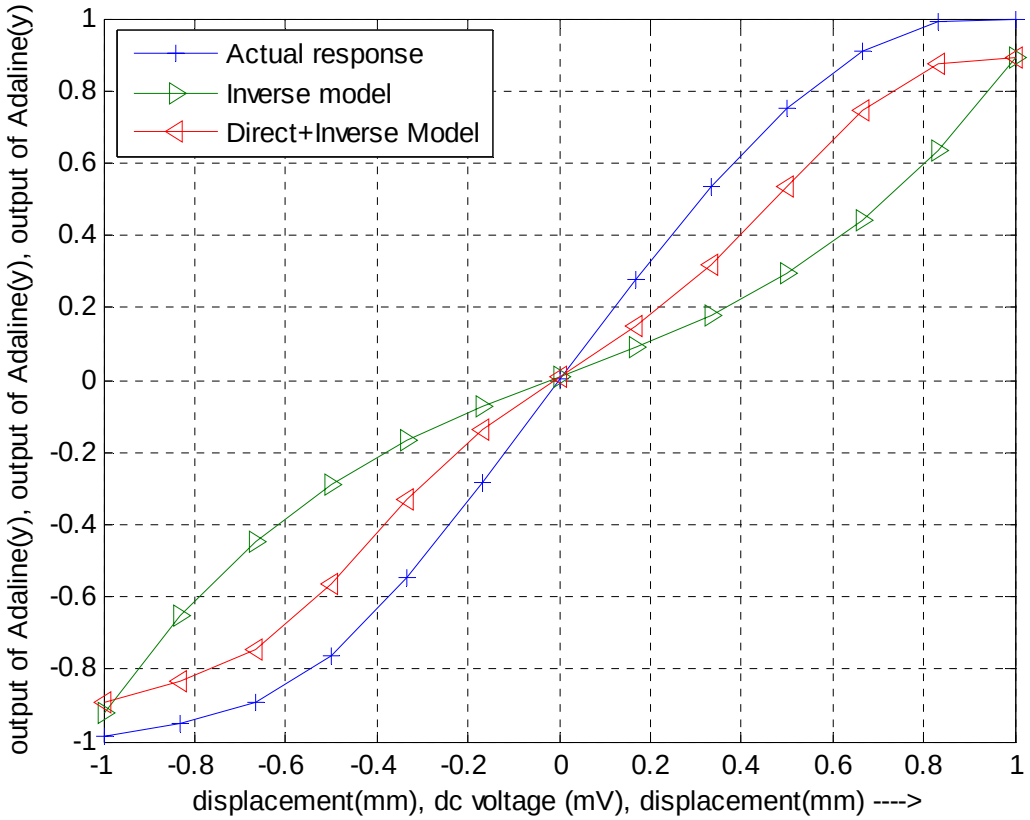

Fig 3.11 Response of ADALIN Network for LVDT nonlinearity compensation

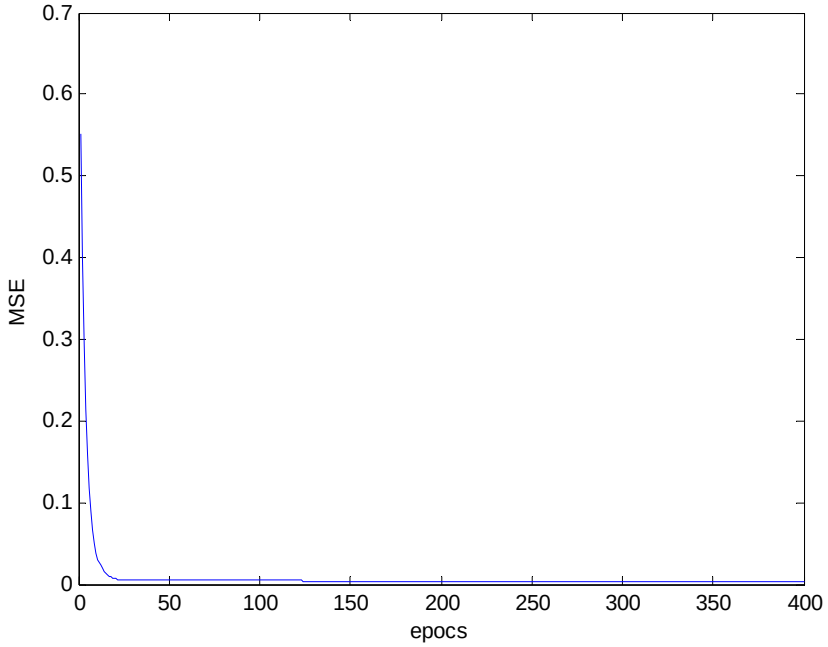

Fig 3.12 Mean Square Error (MSE) Plot of ADALIN for LVDT nonlinearity compensation

Table 3.2 ADALIN Simulation Validation

Input to ADALIN Model (mVolt)	Actual Displacement in mm d(n)	Output of ADALIN model in mm y(n)	Error	Mean Square Error
-5.0170	-25.000	-25.3630	0.3630	
-2.8960	-10.000	-9.34840	-0.6516	
0.0010	0	0.2590	-0.2590	0.0034
1.4620	5.0000	4.30340	0.6966	
4.7990	20.000	21.8853	-1.8853	

3.3.2 MLP Based Non-linearity Compensation

The differential or demodulated voltage *e* at the output of LVDT is normalized by dividing each value with the maximum value. The normalized voltage output *e* is subjected to input to the MLP based nonlinearity compensator. In case of the MLP, we used different neurons with different layers. However, the 1-30-50-1 network is observed to perform better hence it is chosen for simulation. Each hidden layer and the output layer contains tanh(.) type activation function. The output of the MLP based nonlinearity compensator is compared with the normalized input displacement of the LVDT. The BP algorithm, in which both the learning rate and the momentum rate are chosen as 0.01 and 1 respectively, is used to adapt the weights of the MLP. Applying various input patterns, the ANN weights are updated using the BP algorithm. To enable complete learning, 400 iterations are made. Then, the weights of the various layers of the MLP are frozen and stored in the memory. During the testing phase, the frozen weights are used in the MLP model. Fig 3.13 shows the nonlinearity compensation of LVDT by MLP. In this model the *MSE* is obtained to be 0.0022.

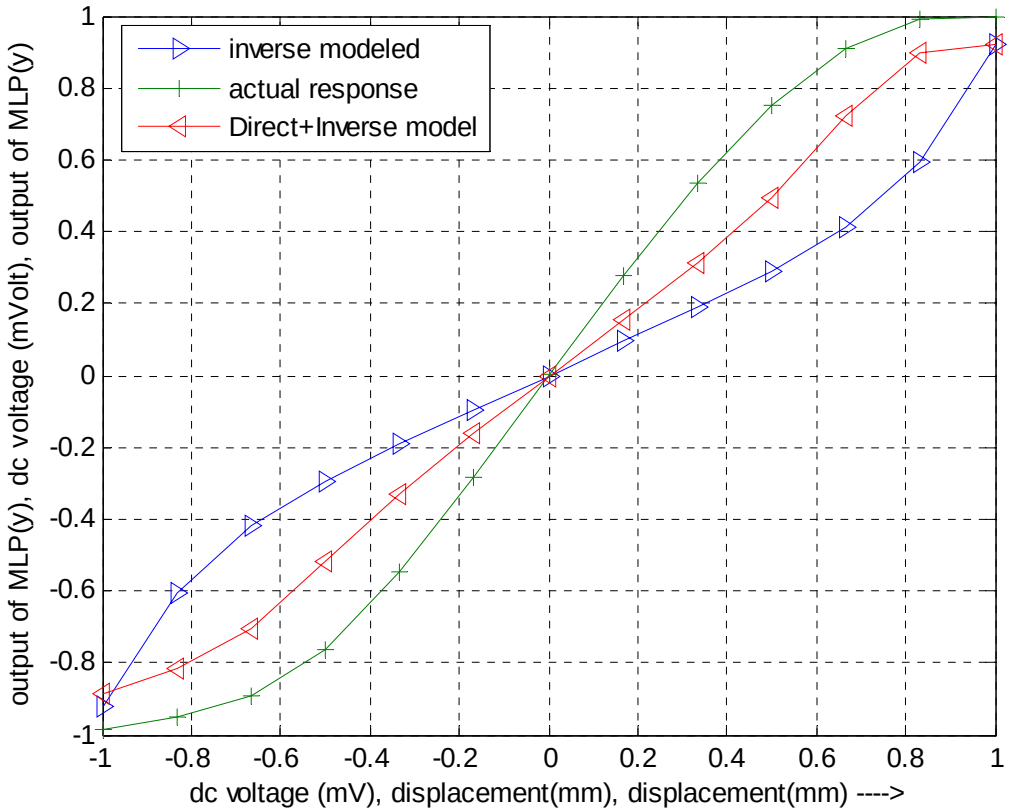

Fig 3.13 Response of MLP Network for LVDT nonlinearity compensation

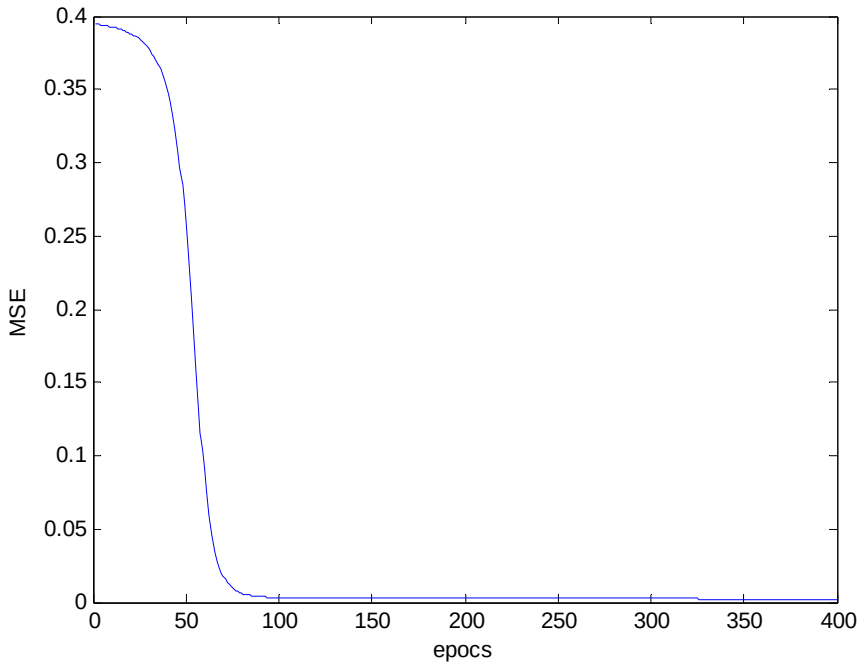

Fig 3.14 Mean Square Error (MSE) Plot of MLP for LVDT nonlinearity compensation

Table 3.3 MLP Simulation Validation

Input to MLP Model (mVolt)	Actual Displacement in mm d(n)	Output of MLP model in mm y(n)	Error	Mean Square Error
-5.0170	-25.0000	-24.5550	-0.4450	
-4.0390	-15.0000	-15.4966	0.4966	
0.0010	0	-0.0108	0.0108	0.0022
1.4620	5.0000	4.7072	0.2928	
4.7990	20.0000	21.6670	-1.6670	

3.3.3 RBFNN Based Non-linearity Compensation

Unlike the MLP, the RBFNN is a single layered network. The detailed theory of RBFNN is depicted in Chapter 2. For the simulation, an RBFNN with a 1-5-1 structure is chosen for inverse modeling of a LVDT. (1, 5, and 1 denote the number of nodes in the input layer, the first layer including the bias units, i.e. the hidden layer and the output layer of the ANN, respectively). The update algorithm, in which both the learning rate is chosen as 0.07, is used to adapt the weights of the RBFNN. The normalized LVDT voltage ()is used as input pattern, and the LVDT input displacement () is used as the desired pattern to the RBFNN. After application of all patterns, the ANN weights are updated using the update algorithm. This process is repeated till the mean square error (MSE) is minimized. Once the training is complete, the RBFNN model will work as inverse model of LVDTs. To make the training successful, 2,000 iterations are needed. Once the training is complete the weights of the RBFNN can be frozen. During the testing phase, the frozen weights of RBFNN model are used for testing. Then the inputs, from the test set are fed to this model. The model output is computed and compared with the actual displacement to verify the effectiveness of the model. Here the centers of RBFNN were selected by hit and trial in order to get lowest mean square error.

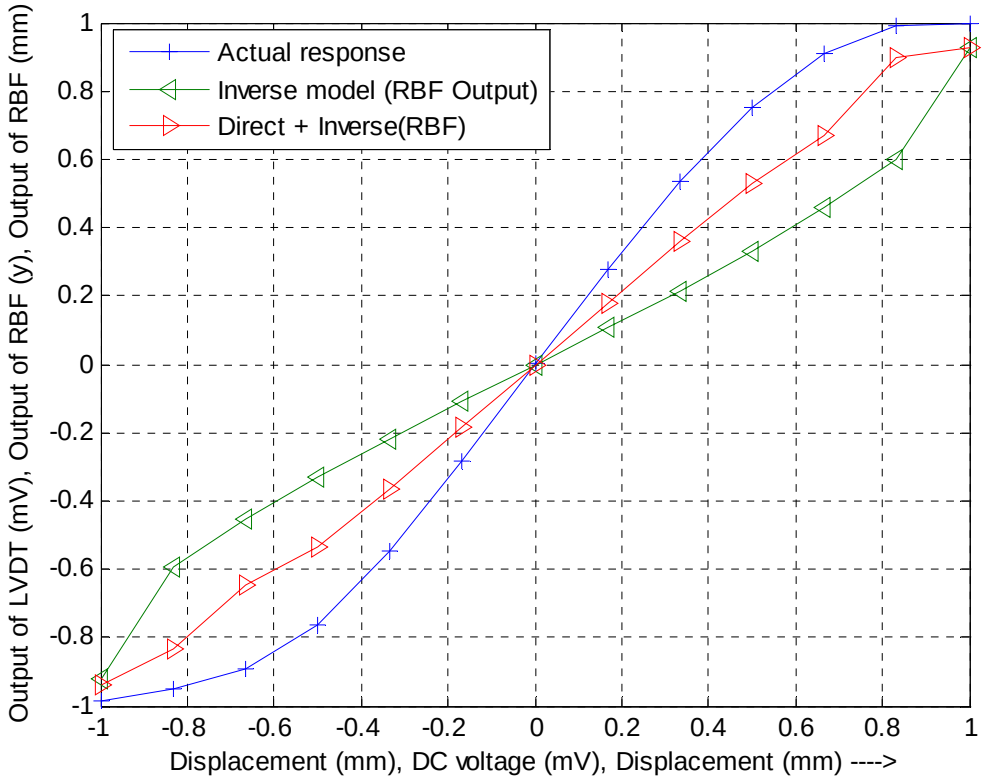

Fig 3.15 Response of RBFNN Network for LVDT nonlinearity compensation

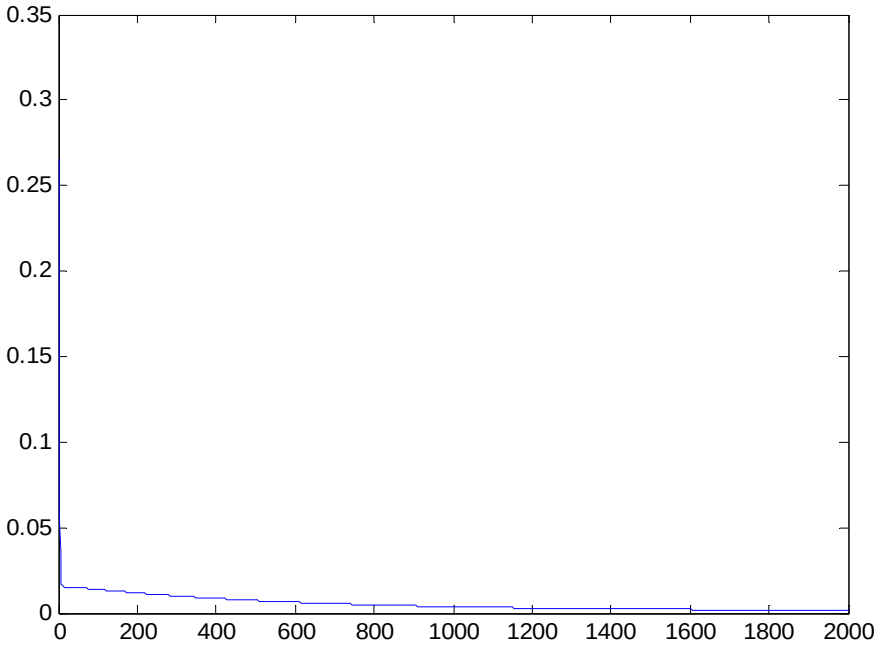

Fig 3.16 Mean Square Error (MSE) Plot of RBFNN for LVDT nonlinearity compensation

Table 3.4 RBFNN Simulation Validation

Input to RBFNN Model (mVolt)	Actual Displacement in mm d(n)	Output of RBFNN model in mm y(n)	Error	Mean Square Error
-5.0170	-25.0000	-24.9626	-0.0374	
-2.8960	-10.0000	-10.9876	0.9876	
0.0010	0	-0.0643	0.0643	0.0014
1.4620	5.0000	5.3526	-0.3526	
3.9620	15.0000	15.8476	-0.8476	

3.3.4 ANFIS Based Non-linearity Compensation

Like the MLP, the ANFIS is a two layered network. The detailed theory of RBFNN is depicted in Chapter 2. For the simulation, an ANFIS with a 1 × 8 Fuzzy input triangular member (Rules) as shown in fig 3.19 is chosen for inverse modeling of a LVDT. The ANFIS structure chosen for nonlinearity compensation is shown in fig 3.16. The LMS update algorithm is used for update of adaptive weights as it is of sugeno model. The normalized LVDT voltage ()is used as input pattern, and the LVDT input displacement () is used as the desired pattern to the ANFIS. After application of all patterns, the weights (W_i) are updated using the update algorithm. This process is repeated till the mean square error (MSE) is minimized. Once the training is complete, the RBFNN model will work as inverse model of LVDTs. To make the training successful, 200 iterations are needed. Once the training is complete the weights of the ANFIS can be frozen. During the testing phase, the frozen weights of ANFIS model are used for testing. Then the inputs, from the test set are fed to this model. The model output is computed and compared with the actual displacement to verify the effectiveness of the model. In this model the *MSE* is obtained to be 0.00077.

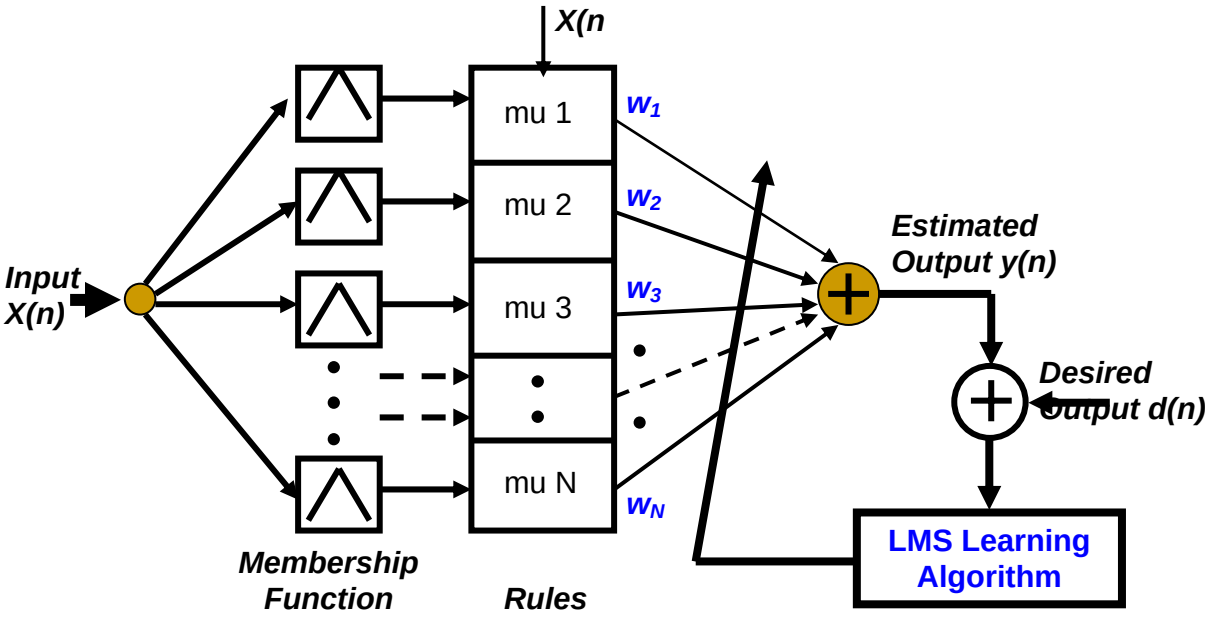

Fig 3.17 ANFIS model for LVDTs nonlinearity compensation

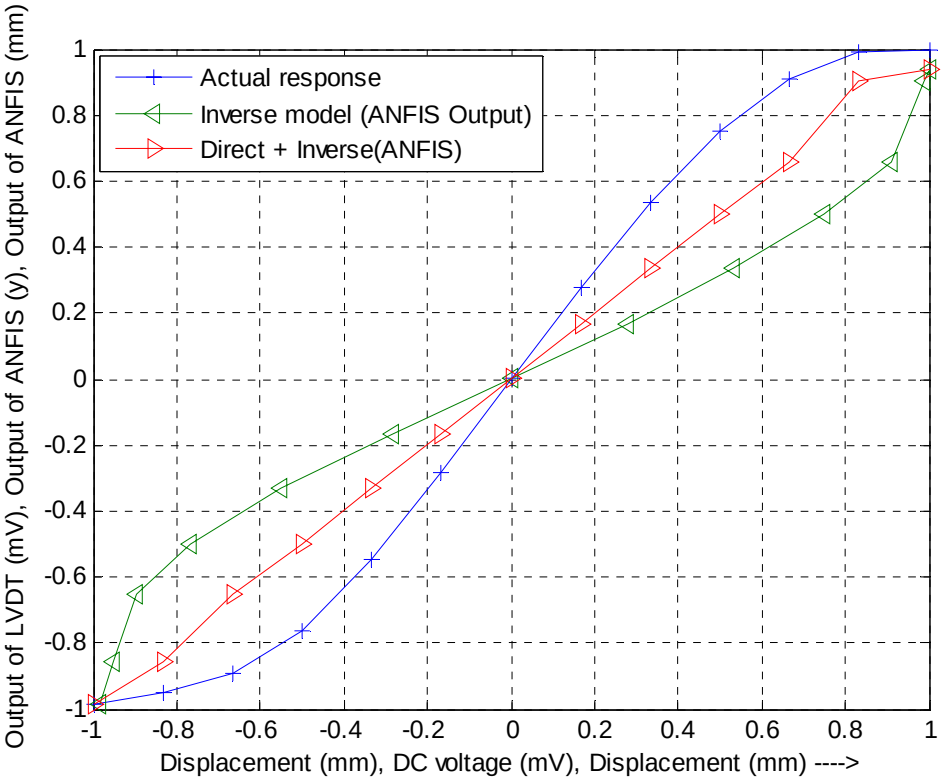

Fig 3.18 Response of ANFIS Network for LVDT nonlinearity compensation

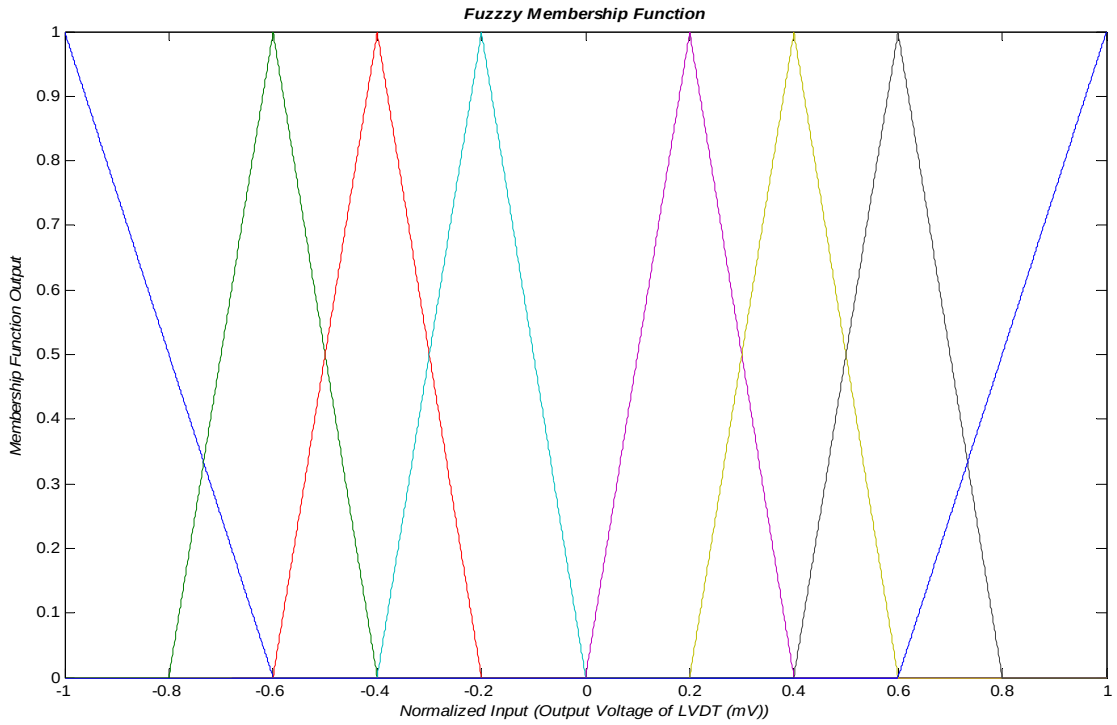

Fig 3.19 Input membership function for ANFIS structure used for nonlinearity compensation of LVDT

Table 3.5 ANFIS Simulation Validation

Actual Displacement in mm d(n)	Output of ANFIS model in mm y(n)	Error
-20.001	-19.6380	1.815
5.0010	4.97700	0.479
15.000	14.9640	0.240

3.4 Comparison and Discussion

Table 3.6 Comparison of Different network for nonlinearity compensation

Network	MSE
ADALIN	**0.00340**
MLP (1:30:50:1)	**0.00220**
RBFNN (1:4:1)	**0.00140**
ANFIS (sugeno)	**0.00077**

Different efficient ANN based nonlinearity compensators for LVDT are developed in this Chapter. The nonlinearity compensation capability of ADALIN and MLP is poor; the *MSE* is about 0.0034 and 0.0022. The ANFIS based nonlinearity compensator improves the linearity quite appreciably about 0.00077.

Chapter 4

FLOW- SENSORLESS CONTROL VALVE (EQUAL PERCENTAGE TYPE): USING SOFT COMPUTING TECHNIQUES

Flow measurements using conventional flow meters for feedback on the flow-control loop cause pressure drop in the flow and in turn lead to the usage of more energy for pumping the fluid. This chapter presents an alternative approach for determining the flow rate without flow meters. The restriction characteristics of the flow-control valve are captured by a neural network (NN) model. The relationship between the flow rate and the physical properties of the flow as well as flow-control valve, that is, pressure drop, pressure, temperature, and flow-control valve coefficient (valve position) is found. With these accessible properties, the NN model yields the flow rate of fluid across the flow-control valve, which acts as a flow meter. The viability of the methodology proposed is illustrated by real flow measurements of water flow which is widely used in hydraulic systems.

4.1 Introduction

Control of fluid flow is essential in process-control plants. Fig. 4.1 shows the scheme of the conventional flow-control loop. The signal of flow measured using the flow meter is compared with the signal of the desired flow by the controller. The controller output accordingly adjusts the opening/closing actuator of the flow-control valve in order to maintain the actual flow close to the desired flow. Typically, flow meters of comparatively low cost such as turbine-type flow meters and venturi-type meters are used to measure the volumetric quantity of fluid flow in unit time in a flow process. However, the flow meter inevitably induces a pressure drop in the flow. In turn, this results in the use of more energy for pumping the fluid. To avoid this problem, non-contact flow meters, i.e. electromagnetic-type flow meters, have been developed and are widely used in process plants not only because there is no requirement for installation in the pipeline but also because introduction to the differential pressure across pipelines is not necessitated. Unfortunately, the cost of such non-contact measurement is comparatively much higher than that of its conventional counterparts.

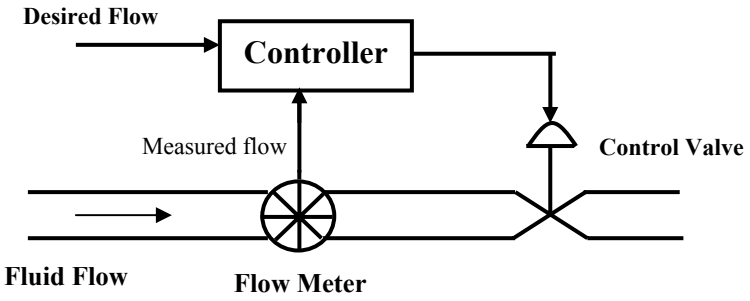

Fig 4.1 Conventional Flow Control

In this study, an alternative approach is proposed to obtain the fluid flow measurement for flow-control valves without the pressure drop and the consequent power loss that appear in conventional flow meters. Without the flow meter, it is a fact that the flow rate can be determined from the characteristics of the control valve for flow measurements. In this method, the restriction characteristics of the control valve embedded in a neural network (NN) model are used in determining the flow rate instead of actual measurement using a conventional flow meter.

4.2 Background principle

For the restriction characteristics of the valve, the flow rate of fluid passing through the flow-control valve can be determined by the physical properties of the flow and the control valve, that is,

$$Q = \eta C_v P Y \sqrt{\frac{x}{\gamma T Z}} \qquad (4.1)$$

With the ratio of pressure drop to the upstream pressure,

$$x = \frac{\Delta P}{P} \qquad (4.2)$$

Where Q is the flow rate, η is the numerical multiplier for compatible units, C_v is the valve coefficient, P is the upstream pressure, Y is the expansion factor, γ is the specific gravity, T

is the upstream temperature, Z is the compressibility factor and ΔP is the pressure drop across the flow-control valve.

The expansion factor can be calculated by:

$$Y = 1 - \frac{x}{3F_k x_T} \qquad (4.3)$$

Where F_k is the ratio of specific heat factor and x_T is the terminal pressure-drop ratio. The ratio of specific heat factor is defined as:

$$F_k = \frac{k}{1.4} \qquad (4.4)$$

With the specific heat ratio, k, expressed as:

$$k = \frac{c_p}{c_v} \qquad (4.5)$$

Where c_p is the specific heat at constant pressure and c_v is the specific heat at constant volume. For example, the value of k is equal to 1.4 for air.

The compressibility ratio Z is determined by the pressure and temperature normalized with respect to the critical pressure and temperature. The normalized pressure and temperature are called the reduced pressure P_r and the reduced temperature T_r, respectively. The value of Z can be expressed as:

$$\boldsymbol{Z=f(P_r,T_r)} \qquad (4.6)$$

with

$$P_r = \frac{P}{P_c} \qquad (4.7)$$

And

$$T_r = \frac{T}{T_c} \tag{4.8}$$

Where P_c is the critical pressure and T_c is the critical temperature. For air, the critical pressure and temperature are specified as 3.39 MPa and −147 °C, respectively.

Now, let us consider equation (4.1), (4.2), (4.3), (4.4), (4.5), (4.6), (4.7) and (4.8). It can be concluded that the flow rate of fluid flowing through the control valve is dependent upon the properties of both the fluid and the control valve. These are the pressure drop ΔP, the upstream pressure P, the upstream temperature T, and the valve coefficient C_v that is given by Eq. (4.9).

$$C_v = \frac{Q}{\eta P Y}\sqrt{\frac{\gamma T Z}{x}} \tag{4.9}$$

From Eq. (4.9), it should be noticed that the valve coefficient depends on not only the physical properties of the fluid but also the degree of variation of fluid flow in the opening of the control valve (%). Hence, the general relation of the flow rate can be written as a function in terms of the physical properties of the fluid and control valve in Eq. (4.10).

$$\boldsymbol{Q=f(\Delta P,P,T,\%)} \tag{4.10}$$

Now, if the non-linear relation in Eq. (4.10) is known, the flow rate of fluid can be obtained by simply measuring pressure, temperature and the percentage of opening in the control valve. However, the computational calculations of the flow rate from (4.1), (4.2), (4.3), (4.4), (4.5), (4.6), (4.7) and (4.8) may not be practical for the real-time control of the flow-control valve due to strong non-linearity and complexity. To overcome this problem, the NN is used to capture the relationship between the flow rate and the properties of the flow and the control valve.

4.3 Equal Percentage type Control Valve Actuators

Valve is essentially a variable orifice. Control valve is a valve with a pneumatic, hydraulic, electric (excluding solenoids) or other externally powered actuator that automatically, fully or partially opens or closes the valve to a position dictated by primarily to throttle energy in a fluid system and not for shutoff purpose.

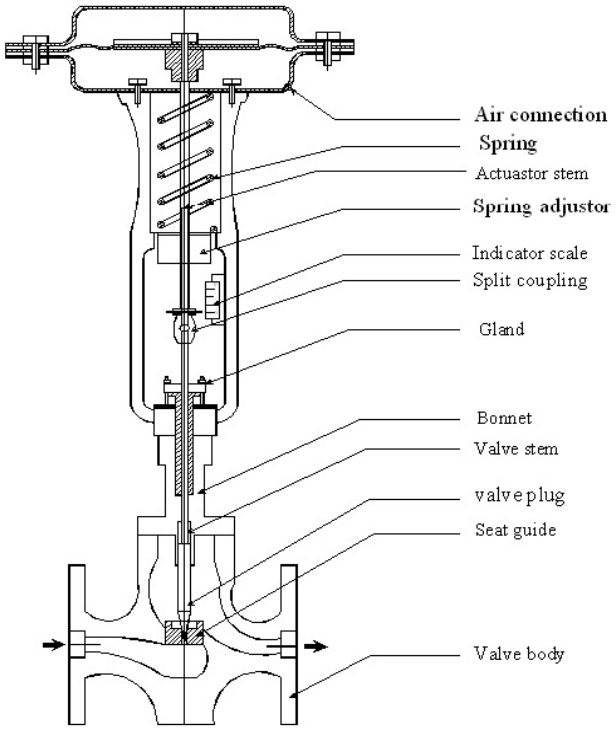

Fig 4.2 Cross sectional view of Control valve (Action Air to open)

The figure 4.2 shows basic elements and internal parts of typical pneumatic control valve. Depending upon the valve plug design the control valves can be classified as quick opening, linear and equal percent type.

The relation between the flow through the valve and the valve stem position (or lift) is called the valve characteristics. It may be differentiated as *inherent flow characteristics* and *installed flow characteristics*. The *inherent flow characteristics* refer to the characteristics observed with constant pressure drop across the valve. Where as the *installed flow characteristics* refers to the characteristic observed when the valve is in service with varying

pressure drop and other changes in the system. The relation between the flow through the valve and the valve stem position (or lift) is called the valve characteristics.

For the equal percentage valve, the sensitivity can be

$$\frac{dm}{dx} = \beta m \tag{4.11}$$

Where β is constant, m is flow and x is valve opening.

Integration of (4.11) gives

$$\int_{m_0}^{m} \frac{dm}{m} = \int_{0}^{x} \beta dx \tag{4.12}$$

Or

$$\ln \frac{m}{m_0} = \beta x \tag{4.13}$$

Where m_0 is the flow at $x = 0$.

The basis for calling the valve characteristics equal percentage can be by rearranging (4.11) in the form

$$\frac{dm}{m} = \beta dx \quad \text{and} \quad \frac{\Delta m}{m} = \beta \Delta x \tag{4.14}$$

in this form it can be seen that an equal fractional (or percentage) change in flow $\Delta m/m$ occurs for a specified increment of change in stem position Δx, regardless of where the change in stem position occurs along the change the characteristic curve.

The term β can be expressed in terms of m_0 by inserting m=1 at x=1 into (4.13). The result is

$$\beta = \ln\left(\frac{1}{m_0}\right)$$

Then (4.12) for *m* gives

$$m = m_0 e^{\beta x} \qquad \text{(Equal percentage valve)}$$

m_0 is the flow at x=0. m_0 cannot be zero since there may be some leakage when the stem is at its lowest position. For some valves, especially large ones, the valve manufacturer intentionally allows some leakage at minimum lift (x=0) to prevent binding and wearing of the plug and seat surfaces.

4.4 Experimental Set-up

Experimental data were collected from laboratory control valve trainee set-up as shown in figure below.

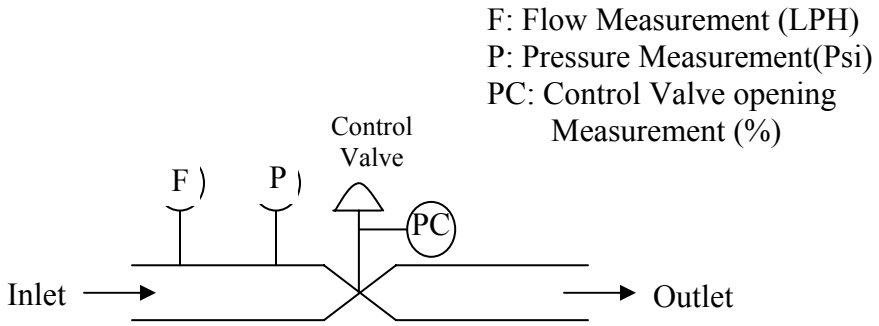

Fig 4.3 Arrangement for Control valve measurement

The flow of water inside the pipe depends upon the opening of control valve. The opening of control valve is pneumatically controlled using pneumatic pressure ranges from 3 to 15 Psi, and depending upon the opening of control valve position, flow inside the pipe changes from 100 to 1000 liter/hour (LPH). Relation between valve opening position (%) and flow (LPH) is shown below which is nonlinear.

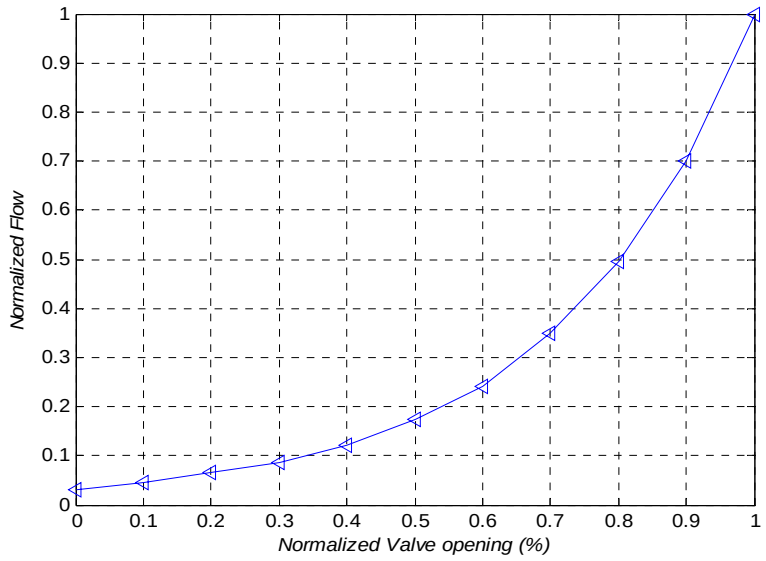

Fig 4.4 Normalized Valve position Vs Normalized flow

For each valve position its respective differential pressure (Psi) were also measured. For valve position (0 % to 100 %), differential pressure changes from 0 to 30 Psi. Relationship between valve position (%) and differential pressure (Psi) is shown in figure below. Thus using the relationship between flows, differential pressure and valve opening position the neural network (NN) can be trained to estimate the flow in LPH.

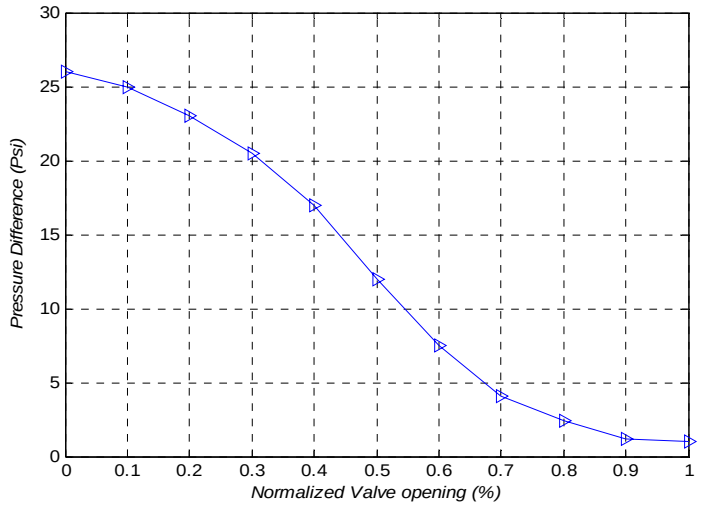

Fig 4.5 Normalized valve position Vs differential pressure (Psi)

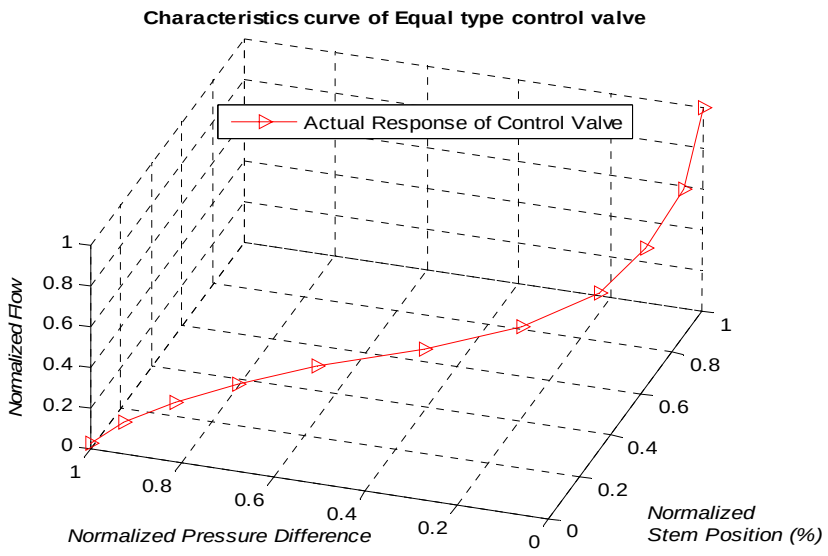

Fig 4. 6 Relation between Stem position, Differential pressure and Flow of control Valve

4.5 Direct Modeling

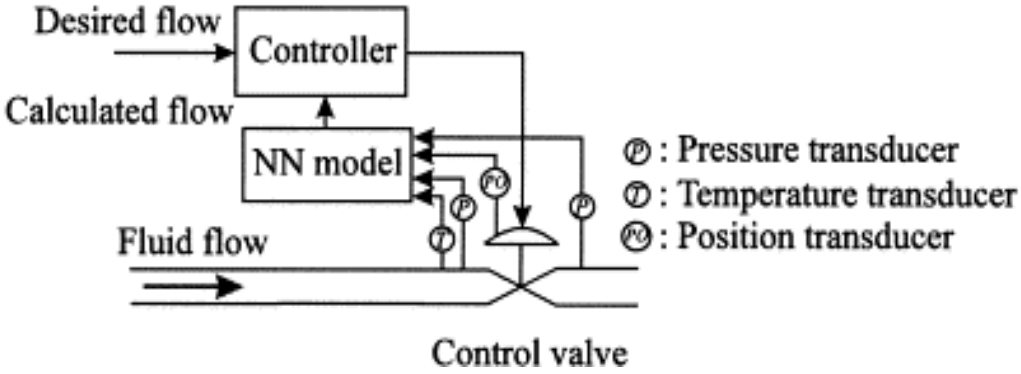

Fig 4.7 Real-time implementation of proposed flow-control valve.

The direct modeling is analogous to that of the system identification problem in control system. The purpose of the direct model is to obtain an ANN model of the control valve in such a way that flow through control valve and output of ANN match closely. Once a model of the Control valve is available, it may be used for determination of flow through valve. By changing the stem position of control valve, the response is the change in flow (LPH) value with respect to the change in pressure difference across valve. Since the temperature almost remain constant (26.5^0C), both normalized pressure () and normalized stem position () are used as input to the ANN model. The output flow () through Control valve and that of the ANN model are compared to produce the error signal. This error information is used to update the ANN model. The model

of Control valve has been developed by separately applying all types of neural models such as MLP, RBFNN and ANFIS.

4.6 The MLP Based Direct Modeling

The MLP technique based direct models have been simulated extensively in MATLAB 7.0 environment. Simulation studies for MLP network are carried out to obtain a direct model of the Control valve. For the simulation study, a two-layer MLP with a 20-20-1 structure is chosen for direct modeling of a Control valve. (20, 20, and 1 denote the number of nodes including the bias units in the input layer, the first layer, i.e. the hidden layer and the output layer of the ANN, respectively). The hidden layer and the output layer contain the tanh (.) type activation function. The back-propagation (BP) algorithm, in which both the learning rate and the momentum rate are chosen as 0.08 and 1 respectively, is used to adapt the weights of the MLP. The normalized valve stem position () and the normalized differential pressure () are used as input pattern, and the respective flow through control valve () is used as the desired pattern to the MLP. After application of each pattern, the ANN weights are updated using the BP algorithm. Completion of all patterns of all the training sets constitutes one iteration of training. To make the learning complete and effective, 1000 epoch are made to train the ANN. Then, the weights of the MLP are frozen and stored in a database. During the testing phase, the frozen weights are loaded into the MLP model. Then the inputs, from the test set are fed to this model. The model output is computed and compared with the actual output to verify the effectiveness of the model which gives mean square error (MSE) of 0.0045. The Control valve response characteristic for respective valve opening is shown below.

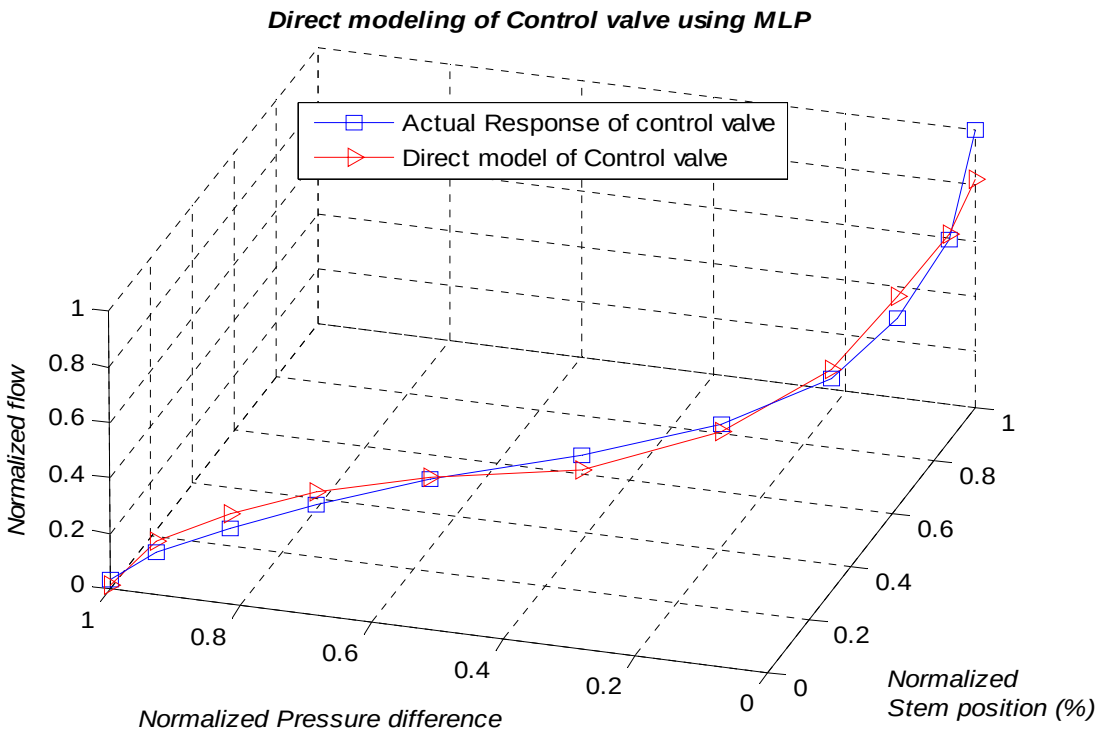

Fig 4.8 Plot of true and estimated forward characteristics of Control valve by MLP

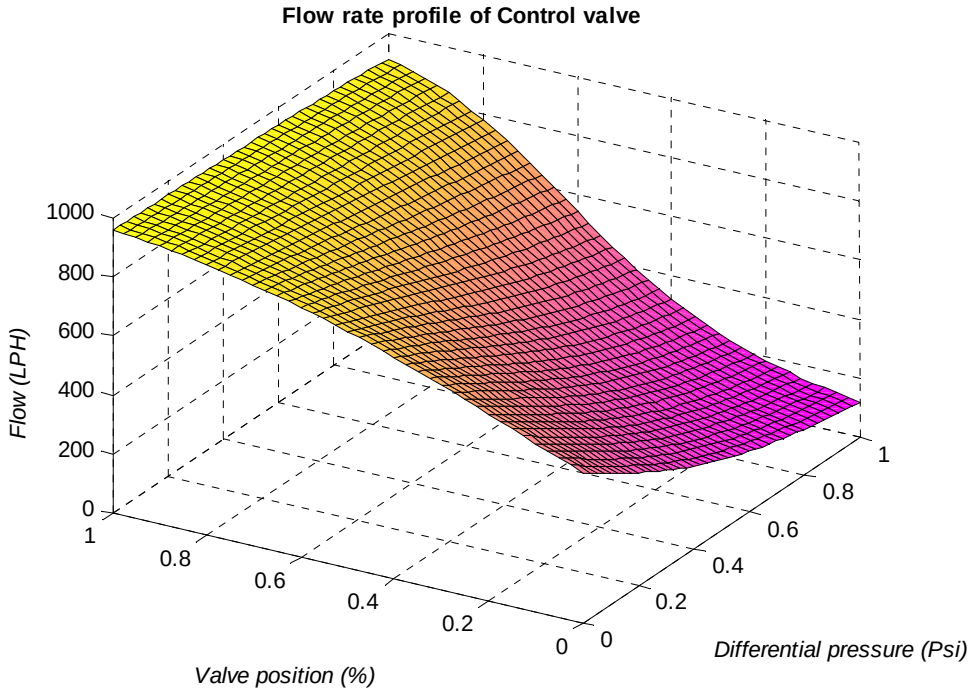

Fig 4.9 Flow-rate profile of MLP model against valve position and Pressure drop with constant upstream pressure

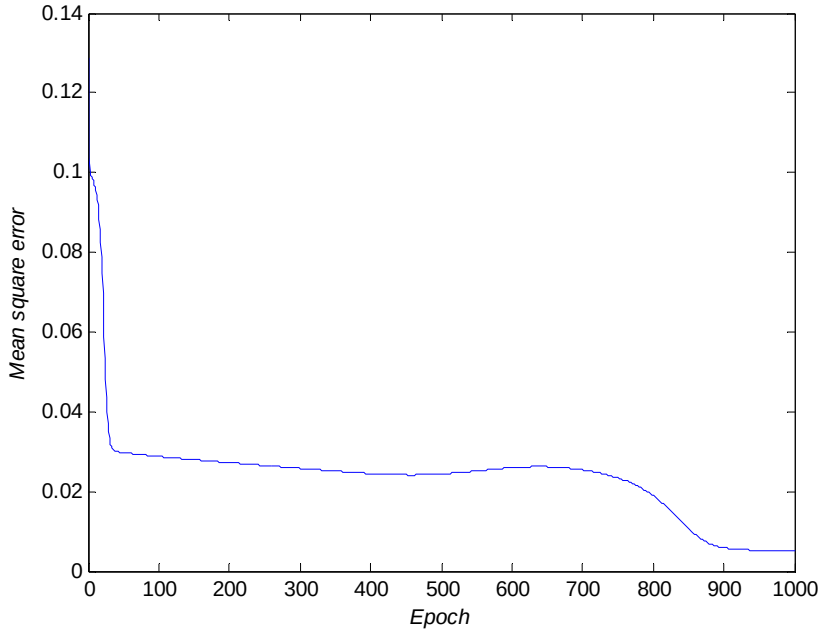

Fig 4.10 Mean Square Error plot while training MLP system for direct model of Control valve

4.7 Radial Basis Function NN based Direct modeling

The RBFNN technique based direct models have been simulated extensively in MATLAB 7.0 environment. Simulation studies for MLP network are carried out to obtain a direct model of the Control valve. For the simulation study, a RBFNN with a 1-4-1 structure is chosen for direct modeling of a Control valve. (1, 4, and 1 denote the number of nodes in the input layer, the first layer including the bias units, i.e. the hidden layer and the output layer of the ANN, respectively). The update algorithm, in which both the learning rate is chosen as 0.07, is used to adapt the weights of the RBFNN. The normalized valve stem position () and the normalized differential pressure () are used as input pattern, and the respective flow through control valve () is used as the desired pattern to the RBFNN. After application of each pattern, the ANN weights are updated using the RBF learning algorithm described in chapter 2. Completion of all patterns of all the training sets constitutes one iteration of training. Here the centers of RBFNN were selected by hit and trial in order to get lowest mean square error. To make the learning complete and effective, 1500 epoch are made to train the RBFNN. Then, the weights of the RBFNN are frozen and stored in a database. During the testing phase, the frozen

weights are loaded into the RBFNN model. Then the inputs, from the test set are fed to this model. The model output is computed and compared with the actual output to verify the effectiveness of the model which gives mean square error (MSE) of 0.00027. The Control valve response characteristic for respective valve opening using RBFNN is shown below.

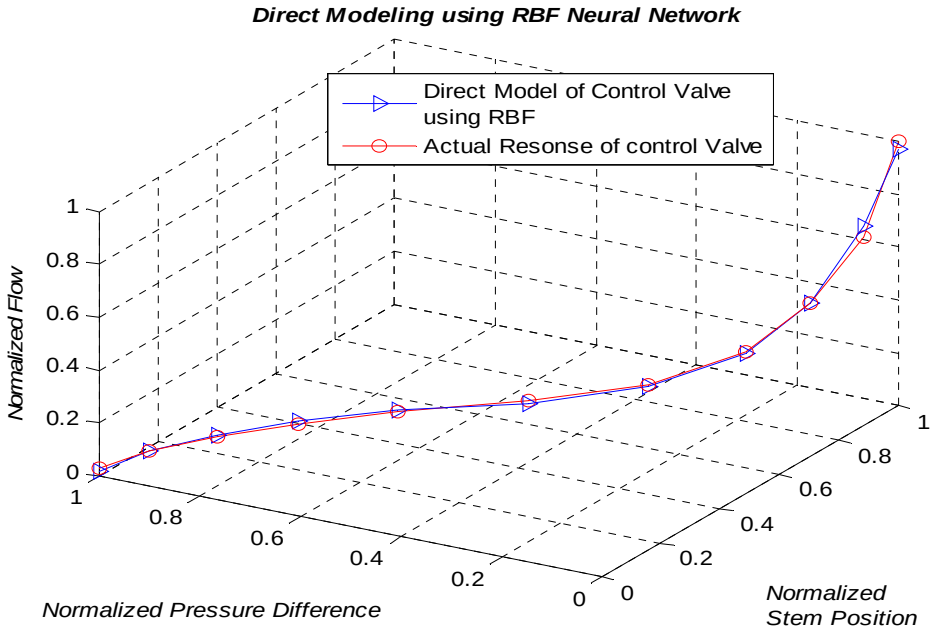

Fig 4.11 Plot of true and estimated forward characteristics of Control valve by RBF (NN)

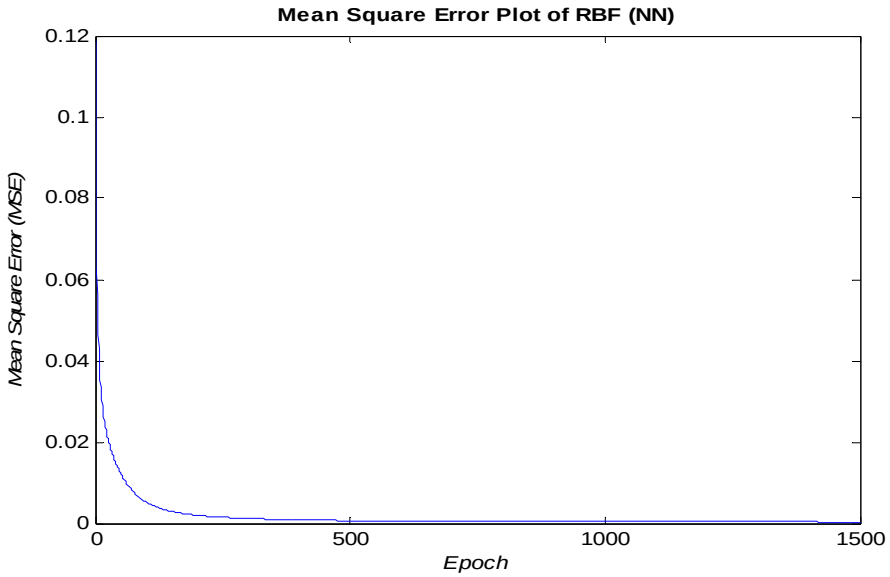

Fig 4.12 Mean Square Error plot while training RBF (NN) system for direct model of Control valve

4.8 Experimental Validation

Table 4.1 MLP Experimental results

Normalized desired Flow	Normalized MLP Output (Flow)
1.000	0.8234
0.7000	0.7207
0.4950	0.5761
0.3500	0.3882
0.2400	0.2153
0.1750	0.1407
0.1200	0.1265
0.0850	0.0911
0.0650	0.0532

The table 4.1 show the output of MLP ANN model, on the left column it represent normalized desired flow and right column shows the respective estimated flow of MLP structure. Similarly table 4.2 shows for RBFNN structure. By comparing both the table 4.1 and 4.2, it shows that RBFNN estimate more closely as compared to MLP structure.

Table 4.2 RBFNN Experimental results

Normalized desired Flow	Normalized RBFNN Output(Flow)
1.000	**0.9699**
0.7000	**0.7393**
0.4950	**0.4930**
0.3500	**0.3452**
0.2400	**0.2345**
0.1750	**0.1628**
0.1200	**0.1263**
0.0850	**0.0981**
0.0650	**0.0718**

4.9 Conclusion

Conventional flow measurement for flow-control valves can cause undesired pressured drops and increase in pumping energy. Since the flow-control valve has already been used for flow control in the pipeline and the properties of flow and the flow-control valve are accessible, neural computing is proposed to determine the flow rate of fluid instead of using a flow meter. To obtain the NN model, the input/output data of the relationship between the flow rate and the properties of fluid flow and the flow-control valve are required. With these properties available, the NN model yields the flow rate of fluid across the flow-control valve, which acts as the flow meter. The experimental results show the viability of practical implementation for flow-control processes.

Chapter 5

INVERSE MODELING OF CONTROL VALVE (EQUAL PERCENTAGE TYPE)

5.1 Introduction

Since the NN model is capable of approximating any continuous non-linear function with certain accuracy, the NN model as described in chapter 4, which has been described in Section 4.2, is used to determine the actual flow rate of the fluid across the flow-control valve after measuring the properties of the flow and flow-control valve. The flow rate obtained is compared with the desired one in order to adjust the flow-control valve such that the difference in flow rate is minimized. Fig. 4.7 shows the block schematic of the flow-control valve acting as the flow meter. Furthermore, it should be mentioned that Fig. 5.1 shows another possible scheme of fluid flow control by using the inverse of the NN model in Fig. 4.7. The inverse of the NN model provides the valve controller with the corresponding valve position (%) for the desired flow rate. The inverse of the NN model can be obtained with the same training dataset obtained in chapter 4 by switching the flow rate to be the input of the NN model and the valve position (%) to be the output.

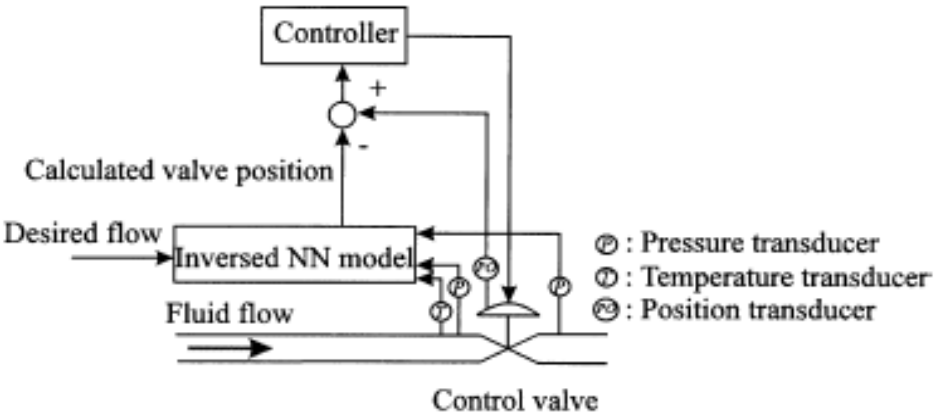

Fig 5.1 Flow-control valve with inverse of NN model.

5.2 Simulation Work

First we found out the relation between flow *V*s valve opening and flow *V*s back pressure as shown in next page. Desired flow and back pressure is given as input to Inverse NN model and position of valve is computed by NN which is further compared with actual position and then error is calculated to trained the NN model to respond actual position.

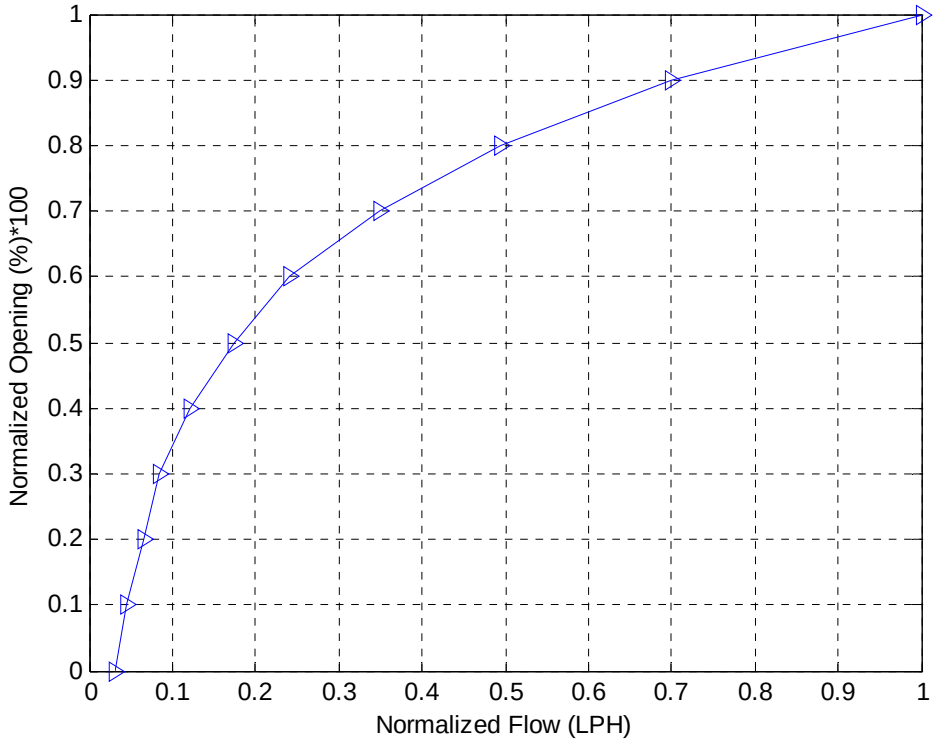

Fig 5.2 Relation between Normalized flow and Valve opening

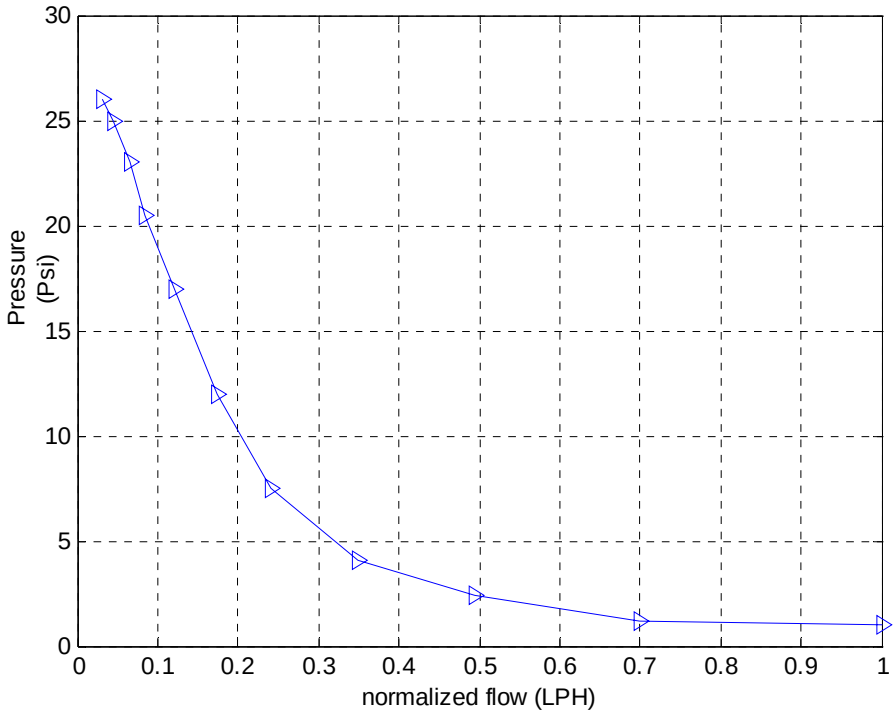

Fig 5.3 Relation between Normalized flow and Back Pressure of control valve

So, we have got the relation between flow, back pressure and valve position as,

Stem position (%) = F (flow, temperature, back pressure) (5.1)

Now, using these above relation the NN network will be trained and we have used MLP neural structure to design the inverse control valve model.

5.3 MLP based Inverse modeling of Control Valve

The MLP technique based direct models have been simulated extensively in MATLAB 7.0 environment. Simulation studies for MLP network are carried out to obtain a inverse model of the Control valve. For the simulation study, a two-layer MLP with a 50-50-1 structure is chosen for direct modeling of a Control valve. (50, 50, and 1 denote the number of nodes including the bias units in the input layer, the first layer, i.e. the hidden layer and the output layer of the ANN, respectively). The hidden layer and the output layer contain the tanh (.) type activation function. The back-propagation (BP) algorithm, in which both the learning rate and the momentum rate are chosen as 0.08 and 1 respectively, is used to adapt the weights of the MLP. The normalized desired flow () and the normalized differential pressure () are used as input pattern, and the respective normalized valve stem position () is used as the desired pattern to the MLP. After application of each pattern, the ANN weights are updated using the BP algorithm. Completion of all patterns of all the training sets constitutes one iteration of training. To make the learning complete and effective, 1000 epoch are made to train the ANN. Then, the weights of the MLP are frozen and stored in a database. During the testing phase, the frozen weights are loaded into the MLP model. Then the inputs, from the test set are fed to this model. The model output is computed and compared with the actual output to verify the effectiveness of the model which gives mean square error (MSE) of 0.0052. The inverse Control valve response characteristic for respective valve opening is shown below.

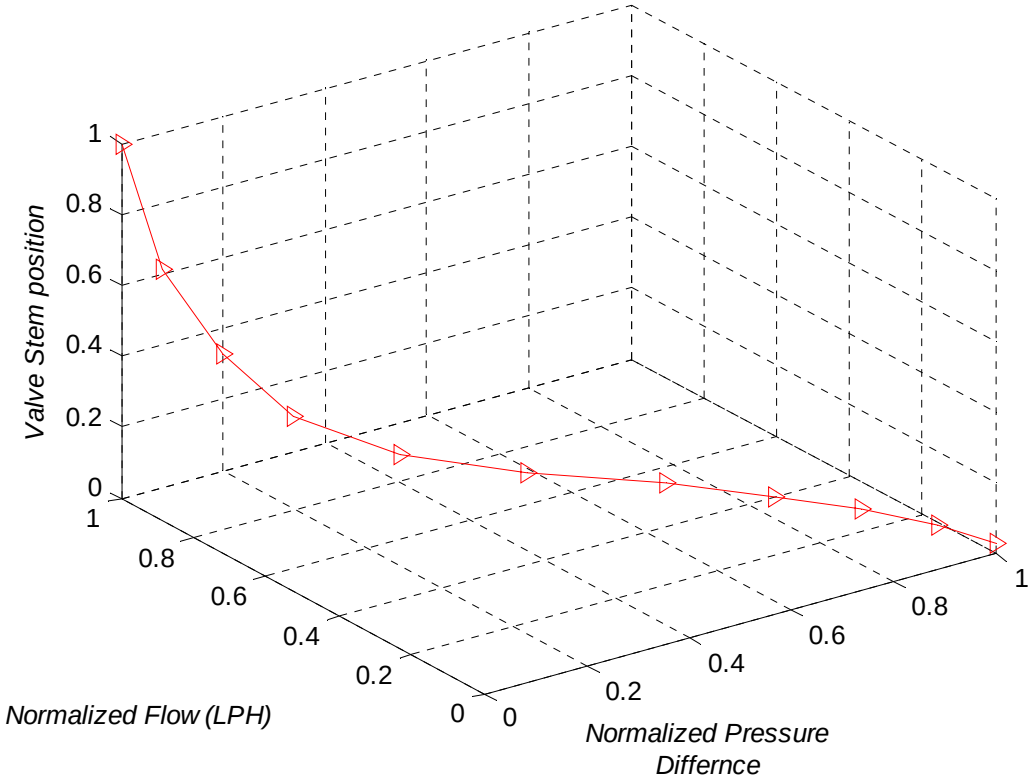

Fig 5.4 MLP based inverse control valve model

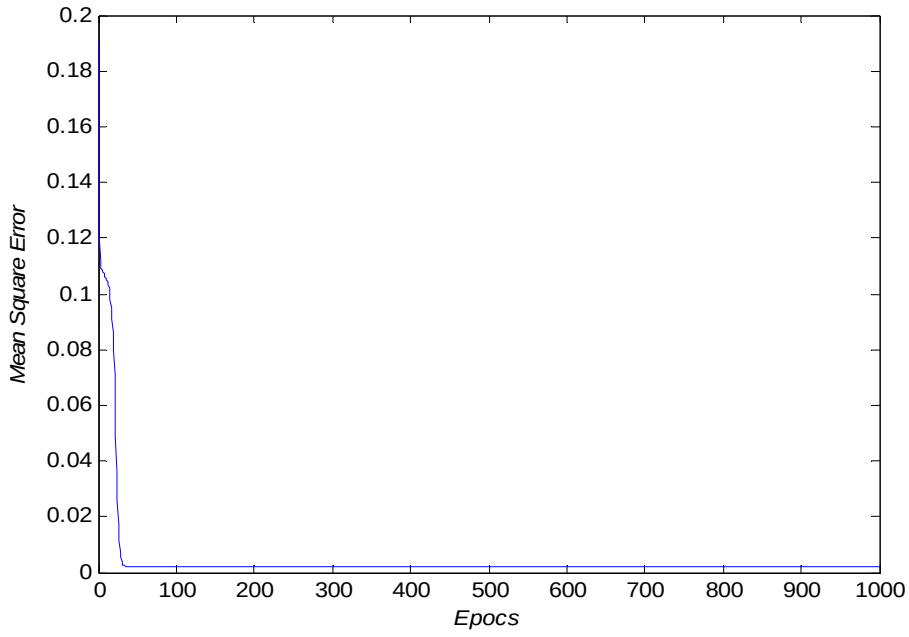

Fig 5.5 Mean Square Error plot while training MLP system for Inverse model of Control valve

5.4 Conclusion

The input to the flow controllers are generally position of the control valve. Here, the inverse model of control valve is perfectly designed using MLP network to give the position of valve, with this design we easily fulfill the input requirement of flow controller. The result has shown that the inverse ANN model of control valve totally estimate the position of control valve with the mean square error of 0.0052 which is tolerable. It will more closely estimate by designing inverse ANN model using RBF-NN or ANFIS structure.

Chapter 6

CONCLUSIONS

6.1 Conclusion

(1) It investigates on the nonlinearity issues relating to LVDT sensor.

(2) The nonlinearity problem gives rise to the following difficulties:

(i) Non accuracy in measurement

(ii) Limitation of dynamic range (linearity region)

(iii) Full potentiality of the sensor cannot be utilized.

(3) The nonlinearity problem arises due to:

(i) Environmental changes such as change in temperature, humidity and atmospheric pressure

(ii) Aging

(iii) Constructional limitations.

(4) In this thesis adaptive and intelligent methods for compensation of nonlinearities have been proposed and have been applied to three typical sensors.

(5) These methods are based on the following structures:

(i) ADALIN

(ii) MLP

(iii) RBFNN

(iv) ANFIS

The learning algorithms employed in the thesis are:

(i) LMS algorithm in ADALIN and ANFIS

(ii) BP algorithm in MLP

(iii) RBF learning algorithm

(6) Basically nonlinear compensation has been achieved through Inverse modeling of the sensors

(7) Exhaustive simulation studies of various methods show that RBFNN and ANFIS structures provide improved non-linearity compensation performance but involves more computations and tedious to implement.

(8) This thesis also include removal of conventional flow sensors which puts loss of energy by pumps in liquid flow line by implementing neural network (NN) flow sensors. The ANN includes MLP and RBFNN structure.

6.2 Scope for Future Research Work

Many further research works may be carried out on the same and related topics.

(1) For developing the inverse model, supervised learning has been used. It has employed training data. In many situations the training data is not available. In absence of such data, training can be carried out by employing blind techniques which are known as unsupervised method. Investigation is needed to develop nonlinear compensator using blind techniques such as higher order statistics (HOS).

(2) The investigation made in this thesis can also be extended to other types of sensors and instruments.

(3) Practical implementation of the inverse model as a plug-in-module, cascading it with physical sensors and using the combined ones for real-time applications is very important. Further research work can be carried out in this direction.

(4) ANFIS techniques can be used for implementing sensorless flow measurement, which can provide more estimation toward the desired flow.

(5) DSP and FPGA implementation of the proposed nonlinearity compensators and sensorless flow can be carried out as further research work.

REFERENCES

[1] G. Panda, J.C. Patra, ”ANN Based Non-Linearity Estimation of Pressure sensors,” IEEE Transactions on Instrumentation and Measurement, Vol. 43, No. 6, December 1994.

[2] J.C. Patra, A. C. Kot, and G. Panda, “An intelligent pressure sensor using neural networks,” IEEE Transactions on Instrumentation and Measurement, vol.49, No.4, pp.829-834, Aug.2000.

[3] R. M. Ford, R. S. Weissbach, and D. R. Loker, “A novel DSP-based LVDT signal conditioner” IEEE Transactions on Instrumentation and Measurement, Vol. 50, No. 3, pp. 768-774, June 2001.

[4] J.C. Patra, and R.N. Pal, “A functional link artificial neural network for adaptive channel equalization”, Signal Processing 43(1995) 181-195, vol.43, no.2, May 1995.

[5] G. Panda, S.K. Mishra, and D.P. Das, “A novel method of extending the linearity range of LVDT using artificial neural network” Sensors & Actuators: A. Physical, Elsevier, on Nov, 2005.

[6] S.C. Saxena, and S.B.L. Sexena, “A self-compensated smart LVDT transducer,” IEEE Transactions on Instrumentation and Measurement, Vol. 38, Issue: 3, pp.748 – 753, June 1989.

[7] M.A. Atmanand, M.S. Konnur, A novel method of using a control valve for measurement and control of flow, IEEE Transactions on Instrumentation and Measurement 48 (6) (1999) 1224–1226.

[8] T. Leephakpreeda, Resolving scaling problem of large-range data in neural network training with fuzzy linguistic model, Advances in Engineering Software 32 (2) (2001) 641–646.

[9] K. Hornik, M. Stinchombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (5) (1989) 359–366.

[10] S.K.Mishra, G.Panda, T.K.Dan and S.K.Pattanaik, “Artificial Neural Network based Nonlinearity Estimation of Control Valve Actuators”, Proceeding of National Conference on Soft Computing Techniques for Engineering Applications (SCT2006), pp.223-232, Mar, 2006.

[11] R. R. Selmic, V. V. Phoha and F. L. Lewis, “Intelligent control of Actuator Nonlinearities”, Proc. of the 42nd IEEE Conf. on Decis. & Contr., Maui, Hawaii USA, Dec, 2003.

[12] Thananchai Leephakpreeda," Flow-sensorless control valve: neural computing approach" Elsevier, Flow Measurement and Instrumentation 14 (2003) 261–266

[13] S.K. Mishra, G. Panda, D.P. Das, S.K. Pattanaik and M.R. Meher, "A Novel Method of Designing LVDT using Artificial Neural Network," *Proceedings of IEEE, ICISIP-2005,* pp.223-227, Jan, 2005.

[14] S.K.Mishra and G.Panda, "A New Methodology for Design of LVDT and Comparison with Conventional Design",(paper ID: 520-133) *International Conference on Signal Processing, Pattern Recognition and Applications (SPPRA 2006),* held on 15th Feb, 2006, in Innsbruck, Austria.

[15] S.K. Mishra and G. Panda, "A Novel Method for Designing LVDT and its Comparison with Conventional Design",(paper ID: 6014) *IEEE Sensors Applications Symposium (SAS-2006)* held on Houston Marriott West Loop by the Galleria, Houston, Texas, February 7-9, 2006.

[16] S.K.Mishra, G.Panda, T.K. Dan and S.K. Pattanaik, "Artificial Neural Network based Nonlinearity Estimation of Control Valve Actuators", Proceeding of National Conference on Soft Computing Techniques for Engineering Applications (SCT2006), pp.223-232, Mar, 2006.

[17] G. Panda, S.K. Mishra, and D.P. Das, "A novel method of extending the linearity range of LVDT using artificial neural network" (paper ID: SNA-D-05-00679) Sensors & Actuators: A. Physical, Elsevier, on Nov, 2005.

[18] G.Panda and S.K.Mishra, "Development of an Intelligent LVDT using CFLANN Structure", (paper ID: IM-8409) IEEE Transaction on Instrumentation & Measurement, on Jan, 2006.

[19] D. R. Coughanowr, "Process Systems Analysis and Control", 2nd edition, McGraw-Hill international editions, 1991.

[20] G. Tao and P. V. Kokotovic, "Adaptive Control of Systems with Actuator and Sensor Nonlinearities", John Wiley & Sons, New York, 1996.

[21] B. Armstrong-Htlouvry, P. Dupont, and C. Canudas de Wit, "A survey of models, analysis tools and compensation methods for the control of machines with fiction," Automatica, vol.30, no.7, pp.1083-1138, 1994.

[22] C. Canudas de Wit, P. Noel, A. Aubin, and B. Brogliato, “Adaptive friction compensation in robot manipulators: Low velocities,” Int. J. Robot. Res., vol.10, no.3, Jun. 1991.

[23] S. W. Lee and J. H. Kim, “Robust adaptive stick-slip fiction compensation,” IEEE Trans. Ind. Electron., vol. 42, no.5, pp. 474-479, Oct. 1995.

[24] Y. H. Kim and F. L. Lewis, “Reinforcement adaptive learning neural-net-based friction compensation control for high speed and precision,” IEEE Trans. Control Sysi. Technol., vol. 8, no. l, pp.118-126, Jan. 2000.

[25] C. A. Desoer and S. M. Shahruz, “Stability of dithered nonlinear systems with backlash or hysteresis,” Int. J. Contr., vol. 43, no. 4, pp. 1045-1060, 1986.

[26] R. R. Selmic, F. L. Lewis, “Deadzone Compensation in Nonlinear Systems using Neural Network”, Proc. of the 37th IEEE conference of Decision & control, Dec, 1998.

[27] Y. D. Song, T. L. Mitchell, and H. Y. Lai, “Control of a class of nonlinear uncertain systems via compensated inverse dynamics approach,” IEEE Trans. Automat. Control, vol. 39, no. 9, pp. 1866-1871, Sep. 1994.

[28] R. R. Selmic, V. V. Phoha and F. L. Lewis, “Intelligent control of Actuator Nonlinearities”, Proc. of the 42nd IEEE Conf. on Decis. & Contr., Maui, Hawaii USA, Dec, 2003.