

# An Evolutionary Optimized Algorithm Approach to Compensate the Non-linearity in Linear Variable Displacement Transducer Characteristics

S. Murugan<sup>†</sup> and SP. Umayal\*

**Abstract** – Linearization of transducer characteristic plays a vital role in electronic instrumentation because all transducers have outputs nonlinearly related to the physical variables they sense. If the transducer output is nonlinear, it will produce a whole assortment of problems. Transducers rarely possess a perfectly linear transfer characteristic, but always have some degree of non-linearity over their range of operation. Attempts have been made by many researchers to increase the range of linearity of transducers. This paper presents a method to compensate nonlinearity of Linear Variable Displacement Transducer (LVDT) based on Extreme Learning Machine (ELM) method, Differential Evolution (DE) algorithm and Artificial Neural Network (ANN) trained by Genetic Algorithm (GA). Because of the mechanism structure, LVDT often exhibit inherent nonlinear input-output characteristics. The best approximation capability of optimized ANN technique is beneficial to this. The use of this proposed method is demonstrated through computer simulation with the experimental data of two different LVDTs. The results reveal that the proposed method compensated the presence of nonlinearity in the displacement transducer with very low training time, lowest Mean Square Error (MSE) value and better linearity. This research work involves less computational complexity and it behaves a good performance for nonlinearity compensation for LVDT and has good application prospect.

**Keywords:** Linear Variable Displacement Transducer (LVDT), Nonlinearity, Extreme Learning Machine (ELM), Differential Evolution (DE) algorithm, ANN trained by Genetic Algorithm (GA)

## 1. Introduction

Linear Variable Displacement Transducer (LVDT), a patent by G.B.Hoadley in 1940. It is arranged with two sets of coil, one as the primary and the other secondary having two coils connected differentially for providing the output. The coupling between the primary and the secondary coils varies with the core plunger moving linearly and the output differential voltage varies linearly. The displacement produced by the plunger will be gained by calculating the differential voltage. So LVDT is widely used in the measurement and control system which is associated with displacement. Because of the mechanism structure and others, LVDT often exhibit inherent nonlinear input-output characteristics. Complicated and accurate winding machines are used to solve this. It is difficult to have all LVDT to be equally linear. Nonlinearity also arises in due to change in environment conditions such as temperature and humidity. Due to such nonlinearities direct digital readout is not possible. Their usable range gets restricted due to the

presence of nonlinearity. If a transducer is used for full range of its nonlinear characteristics, accuracy and sensitivity of measurement is severely affected. The nonlinearity present is usually time-varying and unpredictable as it depends on many uncertain factors.

Literature survey suggests that in [1], the authors proposed a Functional Link Artificial Neural Network (FLANN) with the practical setup for the development of a linear LVDT. In the conventional design, sophisticated and precise winding machines are used to achieve the nonlinearity compensation [2-4]. Some digital signal processing techniques have been suggested to achieve better sensitivity and to implement the signal conditioning circuits [5, 6, 13]. It is reported in [7-9] that the artificial neural network (ANN)-based inverse model can effectively compensate for the nonlinearity effect of the sensors. LVDTs show a nonlinearity behavior when the core is moved toward any one of the secondary coils. In the primary coil region (middle) of the characteristics, core movement is almost linear. Because of that, the range of operation is limited within the primary coil. The nonlinearity estimation and its compensation in the case of a capacitive pressure sensor and an LVDT using different ANNs are proposed in [7-9]. In [14], compensation of Capacitive Pressure Sensor (CPS) nonlinearities is done using neuro-fuzzy algorithms. In [15], Calibration of CPS is discussed

<sup>†</sup> Corresponding Author: Dept. of Electrical and Electronics Engineering, Einstein College of Engineering, Tirunelveli, Tamil Nadu, India. (srivimurugan@gmail.com)

\* Dept. of Electrical and Electronics Engineering, ULTRA College of Engineering and Technology for Women, Madurai, Tamil Nadu, India (umayalbabu@gmail.com)

Received: June 16, 2014; Accepted: August 13, 2014

using circuits. In [16], calibration of CPS is done using least square support vector regression, and for temperature compensation one more CPS is used. In [17], extension of linearity is achieved using Hermite neural network algorithm. In [18], Chebyshev neural network algorithm is used for extension of linearity. In [19], non linearity of CPS is compensated by using Hybrid Genetic Algorithm-Radial Basis Function neural network (HGA-RBF). In [20], calibration of CPS is done using DSP algorithms. In [21], Functional Link ANN (FLANN) algorithm is used for calibrations of CPS. In [22], Laguerre neural network is used for calibration of CPS. In [23], Calibration of CPS is achieved using ANN. Adaptation to physical properties of diaphragm, and temperature is also discussed. In [24], relation between diaphragm properties and CPS output is discussed. In [25], effect of dielectric properties on CPS output is discussed. In [26], effect of temperature on CPS output is discussed. An intelligent pressure measurement technique is proposed as an improvement to the earlier reported works [23]. The technique is designed to obtain full scale linearity of input range and makes the output adaptive to variations in physical properties of diaphragm, dielectric constant, and temperature, all using the optimized ANN model.

This paper is organized as follows: after introduction in Section 1, a brief description on LVDT is given in Section 2. Specifications and experimental observations of two different LVDTs are also discussed in this section. Section 3 deals with the mathematical analysis of ELM, DE and GA. The computer simulation study of the proposed models by using the experimental data of two different LVDTs are carried out in this Section. Results and discussion with output performance curves before and after compensation of nonlinearity using the specified algorithms are mentioned in Section 4. Finally conclusion and future scope are discussed in Section 5.

## 2. Linear Variable Displacement Transducer (LVDT)

The LVDT consists of a primary coil and two secondary coils. The two secondary coils are connected differentially for providing the output. The secondary coils are located on the two sides of the primary coil on the bobbin or sleeve, and these two output windings (secondary coils) are connected in opposition to produce zero output at the middle position of the armature.

The lengths of primary and two identical halves of the secondary coils are  $b$  and  $m$ , respectively. The coils have an inside radius  $r_i$  and an outside radius of  $r_o$ . The spacing between the coils is  $d$ . Inside the coils, a ferromagnetic armature of length  $L_a$  and radius  $r_i$  (neglecting the bobbin thickness) moves in an axial direction. The number of turns in the primary coil is  $n_p$ , and  $n_s$  is the number of turns in each secondary coils. The cross-sectional view of LVDT is

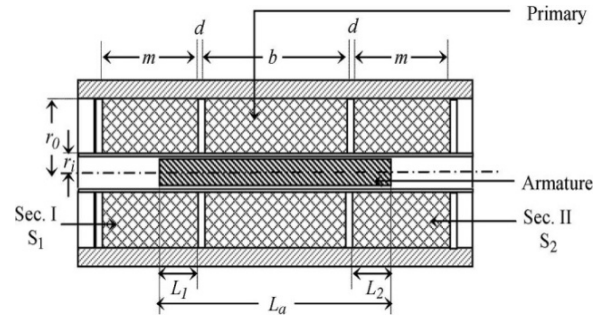


Fig. 1. Cross-sectional view of LVDT

shown in Fig. 1. With a primary sinusoidal excitation voltage  $V_p$  and a current  $I_p$  (RMS) of frequency  $f$ , the RMS voltage  $v_1$  induced in the secondary coil  $S_1$  is

$$v_1 = \frac{4\pi^3}{10^7} \cdot \frac{f I_p n_p n_s}{\ln\left(\frac{r_o}{r_i}\right)} \cdot \frac{2L_2 + b}{mL_a} x_1^2 \quad (1)$$

and that in coil  $S_2$  is

$$v_2 = \frac{4\pi^3}{10^7} \cdot \frac{f I_p n_p n_s}{\ln\left(\frac{r_o}{r_i}\right)} \cdot \frac{2L_1 + b}{mL_a} x_2^2 \quad (2)$$

Where

- $x_1$  – distance penetrated by the armature toward the secondary coil  $S_1$  ;
- $x_2$  – distance penetrated by the armature toward the secondary coil  $S_2$

The differential voltage  $v = v_1 - v_2$  is thus given by

$$v = k_1 x (1 - k_2 x^2) \quad (3)$$

Where  $x = \left(\frac{1}{2}\right)(x_1 - x_2)$  is the armature displacement and

$$k_1 = \frac{16\pi^3 f I_p n_p n_s (b + 2d + x_0) x_0}{10^7 \ln\left(\frac{r_o}{r_i}\right) mL_a} (Vm^{-1}) \quad (4)$$

$$\text{with } x_0 = \left(\frac{1}{2}\right)(x_1 + x_2) \quad (5)$$

$$\text{and } k_2 = \frac{1}{(b + 2d + x_0) x_0} \quad (6)$$

$k_2$  is a nonlinearity factor in (3), with the nonlinearity term  $\epsilon$  being

$$\epsilon = k_2 x^2 \quad (7)$$

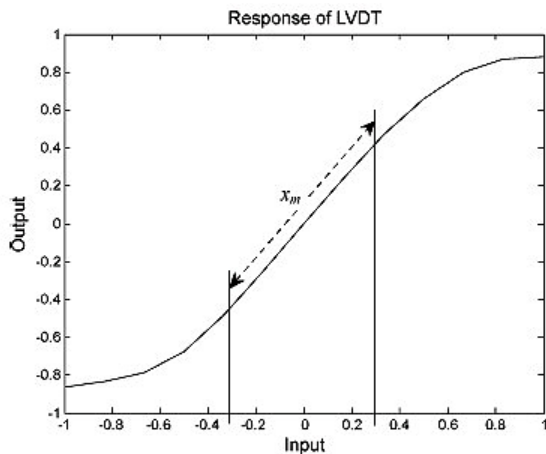


Fig. 2. Range of linear region of LVDT

The nonlinearity factor and nonlinearity term of (6) and (7) are calculated from the core movement of the LVDT. These two terms depends on the geometric parameters taken from the corresponding LVDT. For a given accuracy and maximum displacement, the overall length of the transducer is minimum for  $x_1=b$ , assuming that at maximum displacement, the armature does not emerge from the secondary coils. Taking the length of armature  $L_a = 3b + 2d$ , neglecting  $2d$  compared with  $b$ , and using (4), (3) can be simplified as

$$v = \frac{16\pi^3 f I_p n_p n_s}{10^7 \ln\left(\frac{r_o}{r_i}\right)} \cdot \frac{2b}{3m} \left(1 - \frac{x^2}{2b^2}\right) \quad (8)$$

For a given primary sinusoidal excitation, the secondary output voltage  $v$  is nonlinear with respect to displacement  $x$ . This is shown in Fig. 2 in which the linear region of the plot is indicated as  $x_m$ .

This limitation is inherent in all differential systems, and methods of nonlinearity compensation are proposed mainly by appropriate design and arrangement of the coils. Some of these are given as follows.

- 1) Balanced linear tapered secondary coils: improvement in linearity range is not significant
- 2) Overwound linear tapered secondary coils: linearity is improved to a certain range
- 3) Balanced overwound linear tapered secondary coils: the range specification is similar to 2)
- 4) Balanced profile secondary coils: helps in extending linearity range by proper profiling of the secondary coils
- 5) Complementary tapered windings method: extends the linearity range as well, but the winding is quite complicated as sectionalized winding is done [1]

## 2.1 Linearity

One of the best characteristics of a transducer is

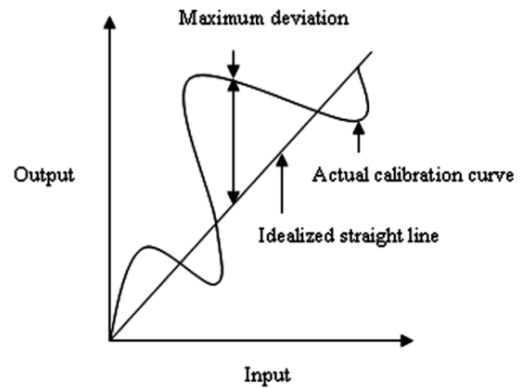


Fig.3. Actual calibration curve

considered to be linearity, that is, the output is linearly proportional to the input. The computation of linearity is done with reference to a straight line showing the relationship between output and input. This straight line is drawn by using the method of least squares from the given calibration data. This straight line is sometimes called an idealized straight line expressing the input-output relationship. The linearity is simply a measure of maximum deviation of any of the calibration points from this straight line.

Fig. 3 shows the actual calibration curve i.e., a relationship between input and output and a straight line drawn from the origin using the method of least squares.

$$\%NL = \frac{\text{Max deviation from the idealized straight line}}{\text{Full scale deviation}} \times 100 \quad (9)$$

Eq. (9) expresses the nonlinearity as a percentage of full scale reading. It is desirable to keep the nonlinearity as small as possible as it would in that case result in small errors.

## 2.2 Geometric parameters and experimental observations of LVDT

The performance of the LVDT is highly influenced by transducer geometry, arrangement of primary and secondary windings, quality of core material, variations in excitation current and frequency, and changes in ambient and winding temperatures. The geometric parameters and specifications of a conventional LVDT is listed in the below Table 1.

In this research work, we have taken the performance of two different LVDTs. The experimental data are collected from two different LVDTs having the specifications listed in Table 1. The variable parameters of the conventional LVDT are chosen as lowest range for LVDT-1 and highest range for LVDT-2. The data obtained by conducting experiments on the two LVDTs are given in Table 2 and Table 3. The output response curves of two LVDTs are shown in Figs. 4 and Fig. 5. It is clear that the output response of the two LVDTs shows the presence of nonlinearity.

**Table 1.** Geometric parameters and specifications of LVDT

Parameter	Range (mm)	Variable/Fixed	LVDT-1	LVDT-2
Coil diameter	5 – 20	Variable	5 mm	20 mm
Core length	62 – 216	Variable	60 mm	140 mm
Core diameter	4.1 – 10	Variable	4.4 mm	10 mm
Number of turns P	200-2000	Variable	1500	200
Number of turns S <sub>1</sub>	100-4000	Variable	3300	100
Number of turns S <sub>2</sub>	100-4000	Variable	3300	100
Excitation voltage	3V<V <sub>e</sub> <15V <sub>pp</sub>	Variable	10 V <sub>pp</sub>	5 V <sub>pp</sub>
Excitation Frequency	5-10kHz	Variable	5 kHz	10kHz

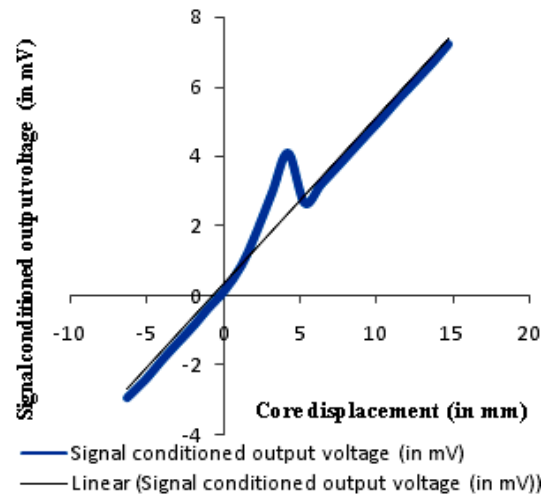
**Table 2.** Experimental observations of LVDT-1

Core Displacement (in mm)	Signal conditioned output voltage (in mV)
-6.30	-2.91
-5.08	-2.38
-4.10	-1.87
-3.02	-1.35
-1.94	-0.84
-0.99	-0.34
0.05	0.15
1.20	0.90
2.21	1.90
3.20	3.00
4.22	4.10
5.29	2.67
6.37	3.20
7.46	3.72
8.49	4.21
9.51	4.70
10.57	5.21
11.53	5.69
12.68	6.22
13.73	6.72
14.70	7.22

**Table 3.** Experimental observations of LVDT-2

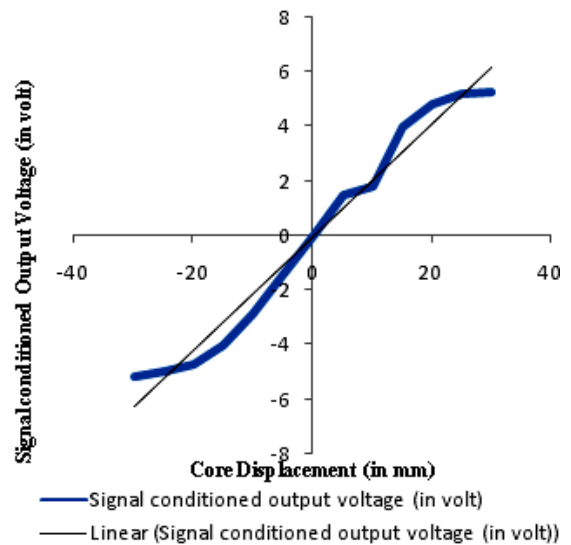
Core displacement (in mm)	Signal conditioned output voltage (in V)
-30	-5.185
-25	-5.017
-20	-4.717
-15	-4.039
-10	-2.896
-5	-1.494
0	0.001
5	1.462
10	1.810
15	3.962
20	4.799
25	5.225
30	5.276

The percentage of nonlinearity is calculated using the Eq. (9). The lowest range chosen LVDT (LVDT-1) having highest percentage of nonlinearity when compared with



**Fig. 4.** Input-Output Response of LVDT-1

$$\% \text{ of Nonlinearity for LVDT1} = \frac{2}{7.22} \times 100 = 27.70\%$$



**Fig. 5.** Input-Output Response of LVDT-2

$$\text{of Nonlinearity for LVDT2} = \frac{1.143}{5.276} \times 100 = 21.66\%$$

the highest range LVDT (LVDT-2). So it is necessary to compensate the percentage of nonlinearity present in both LVDTs.

It has been observed from the above graphs (Fig. 4 and Fig. 5), that the relation between input displacement and voltage output of LVDT are nonlinear. The following algorithms are used to compensate the nonlinearity of two different LVDTs in this work.

- AL-1: Extreme Learning Machine Method (ELM)
- AL-2: ANN trained by Differential Evolution algorithm (ANN-DE)
- AL-3: ANN trained by Genetic Algorithm (GA-ANN)

### 3. Nonlinearity Compensation using Soft Computing Techniques

#### 3.1 Extreme learning machine based nonlinearity compensation

Extreme Learning Machine (ELM) is a simple tuning-free three-step algorithm. The learning speed of is extremely fast. The hidden node parameters are not only independent of the training data but also of each other. Unlike conventional learning methods which must see the training data before generating the hidden node parameters, ELM could generate the hidden node parameters before seeing the training data. Unlike traditional gradient-based learning algorithms which only work for differentiable activation functions, ELM works for all bounded non-constant piecewise continuous activation functions. Unlike traditional gradient-based learning algorithms facing several issues like local minima, improper learning rate and over fitting, etc, ELM tends to reach the solutions straight-forward without such trivial issues. The ELM learning algorithm looks much simpler than many learning algorithms: neural networks and support vector machines. It is efficient for batch mode learning, sequential learning and incremental learning. It provides a unified learning model for regression, binary / multi-class classification. It also works with different hidden nodes including random hidden nodes (random features) and kernels.

#### 3.2 Single hidden layer feed-forward neural network

Recently, Huang et al [31, 32] proposed a new learning algorithm for Single Layer Feed forward Neural Network architecture called Extreme Learning Machine (ELM) which overcomes the problems caused by gradient descent based algorithms such as Back propagation applied in ANNs and significantly reduces the amount of time needed to train a Neural Network. It randomly chooses the input weights and analytically determines the output weights of SLFN. It has much better generalization performance with much faster learning speed. It requires less human interventions and can run thousands times faster than those conventional methods. It automatically determines all the network parameters analytically, which avoids trivial human intervention and makes it efficient in online and real time applications. Extreme Learning Machine has several advantages, Ease of use, Faster Learning Speed, Higher generalization performance, suitable for many non-linear activation function and kernel functions.

Single Hidden Layer Feed-forward Neural Network (SLFN) function with L hidden nodes [34, 35] can be represented as mathematical description of SLFN incorporating both additive and RBF hidden nodes in a unified way is given as follows.

$$f_L(x) = \sum_{i=1}^L \beta_i G(a_i, b_i, x), x \in R^n, a_i \in R^n \quad (10)$$

Where  $a_i$  and  $b_i$  are the learning parameters of hidden nodes and  $\beta_i$  the weight connecting  $i$ th hidden node to the output node.  $G(a_i, b_i, x)$  is the output of  $i$ th hidden node with respect to the input  $x$ . For additive hidden node with the activation function  $g(x): R \rightarrow R$  (e.g. sigmoid and threshold),  $G(a_i, b_i, x)$  is given by

$$G(a_i, b_i, x) = g(a_i x + b_i), b_i \in R \quad (11)$$

Where  $a_i$  is the weight vector connecting the input layer to the  $i$ th hidden node and  $b_i$  is the bias of the  $i$ th hidden node.  $a_i \cdot x$  denotes the inner product of vector  $a_i$  and  $x$  in  $R^n$ .

For N arbitrary distinct samples  $(x_i, t_i) \in R^n \times R^m$  Here,  $x_i$  is a  $n \times 1$  input vector and  $t_i$  is a  $m \times 1$  target vector. If an SLFN with L hidden nodes can approximate these N samples with zero error. It then implies that there exist  $\beta_i, a_i$  and  $b_i$  such that

$$f_L(x_j) = \sum_{i=1}^L \beta_i G(a_i, b_i, x_j), j = 1, 2, \dots, N \quad (12)$$

The above equation can be written as

$$H\beta = T$$

Where

$$H \begin{pmatrix} \bar{a} & \bar{b} & \bar{x} \end{pmatrix} = \begin{pmatrix} G(a_1, b_1, x_1) \dots G(a_L, b_L, x_1) \\ G(a_1, b_1, x_N) \dots G(a_L, b_L, x_N) \end{pmatrix} \\ N \times L$$

With  $\bar{a} = a_1, \dots, a_L; \bar{b} = b_1, \dots, b_L; \bar{x} = x_1, \dots, x_N$ .

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix} L \times m \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix} N \times m \quad (13)$$

H is the hidden layer output matrix of SLFN with  $i$ th column of H being the  $i$ th hidden node's output with respect to inputs  $x_1, x_1, \dots, x_N$

From the observed readings of LVDT-1 and LVDT-2 shown in Tables 2 and 3, the simulation study has been carried out and the following results have been obtained.

The results obtained by ELM based nonlinearity compensation of two different LVDTs are listed in Table 4 and Table 5. Two different activations functions namely sine and sigmoid are used here. The training time, testing time and Root Mean Square Error (RMSE) values are tabulated. The testing and training time are zero by using sine function for both LVDTs. There are 20 hidden nodes

**Table 4.** ELM based nonlinearity compensation of LVDT-1

Activation function	Training time (sec)	Testing time (sec)	Testing accuracy	RMSE
Sine	0	0	0.0087	0.0087
sigmoid	0.0156	0.0156	0.0088	0.0088

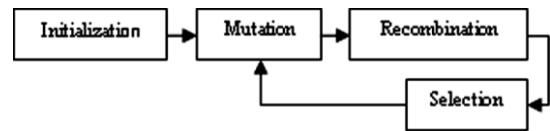
**Table 5.** ELM based nonlinearity compensation of LVDT-2

Activation function	Training Time (sec)	Testing time (sec)	Testing accuracy	RMSE
Sine	0	0	0.0265	0.0265
sigmoid	0.0156	0.0156	0.5513	0.5513

assigned for ELM algorithm. 50 trials have been conducted for the algorithm and the average results are shown in Tables 4 and Table 5. It can be seen from Table 4 that ELM learning algorithm spent 0 seconds CPU time obtaining the testing root mean square error (RMSE) 0:0087 with sine activation function, and 0.0156 seconds CPU time obtaining the RMSE value of 0.0088 with sigmoid activation function. Similarly from Table 5, the ELM algorithm spent 0 seconds CPU time obtaining RMSE value of 0.0265 with sine activation function and 0.0156 seconds CPU time obtaining RMSE value of 0.5513 with sigmoid function. The new ELM runs 170 times faster than the conventional BP algorithms.

**3.3 Differential evolution algorithm based nonlinearity compensation**

The Differential Evolution (DE) algorithm is a stochastic, population-based optimization algorithm introduced by Storn and Price in 1996. It is developed to optimize real parameter and real valued functions. It is a population based algorithm like genetic algorithms using the similar operators; crossover, mutation and selection. The main difference in constructing better solutions is that genetic algorithms rely on crossover while DE relies on mutation operation. This main operation is based on the differences of randomly sampled pairs of solutions in the population. The algorithm uses mutation operation as a search mechanism and selection operation to direct the search toward the prospective regions in the search space. The DE algorithm also uses a non-uniform crossover that can take child vector parameters from one parent more often than it does from others. By using the components of the existing population members to construct trial vectors, the recombination (crossover) operator efficiently shuffles information about successful combinations, enabling the search for a better solution space. An optimization task consisting of D parameters can be represented by a D-dimensional vector. In DE, a population of NP solution vectors is randomly created at the start. This population is successfully improved by applying mutation, crossover and selection operators. The main steps of the DE algorithm are given as follows:



**Fig. 6.** General evolutionary algorithm procedure

**The general problem formulation is:**

For an objective function  $f: X \subseteq \mathbb{R}^D \rightarrow \mathbb{R}$  where the feasible region  $X \neq \emptyset$ , the minimization problem is to find  $x^* \in X$  Such that  $f(x^*) \leq f(x) \forall x \in X$  where  $f(x^*) \neq -\infty$

Suppose we want to optimize a function with  $D$  real parameters, we must select the size of the population  $N$  (it must be at least 4). The parameter vectors have the form:

$$x_{i,G} = [x_{1,i,G}, x_{2,i,G}, \dots \dots x_{D,i,G}] \quad i = 1, 2, \dots \dots N \quad (14)$$

Where,  $G$  is the generation number.

**Initialization:**

Define upper and lower bounds for each parameter:

$$x_j^L \leq x_{j,i,1} \leq x_j^U \quad (15)$$

Randomly select the initial parameter values uniformly on the intervals:

$$[x_j^L, x_j^U] \quad (16)$$

**Mutation:**

Each of the  $N$  parameter vectors undergoes mutation, recombination and selection. Mutation expands the search space.

For a given parameter vector  $x_{i,G}$  randomly select three vectors  $x_{r1,G}, x_{r2,G}$  an  $x_{r3,G}$  such that the indices  $i, r1, r2$  and  $r3$  are distinct.

Add the weighted difference of two of the vectors to the third

$$v_{i,G+1} = x_{r1,G} + F(x_{r2,G} - x_{r3,G}) \quad (17)$$

The mutation factor  $F$  is a constant from  $[0,2]$   $v_{i,G+1}$  is called the donor vector

**Recombination:**

Recombination incorporates successful solutions from the previous generation. The trial vector  $u_{i,G+1}$  is developed from the elements of the target vector,  $x_{i,G}$  and the elements of the donor vector,  $v_{i,G+1}$

Elements of the donor vector enter the trial vector with probability  $CR$

$$u_{j,i,G+1} = \begin{cases} v_{j,i,G+1}, & \text{if } rand_{j,i} \leq CR \text{ or } j = I_{rand} \\ x_{j,i,G}, & \text{if } rand_{j,i} > CR \text{ and } j \neq I_{rand} \end{cases} \quad i = 1, 2, \dots \dots N; j = 1, 2, \dots \dots D \quad (18)$$

$rand_{j,i} \sim U[0,1]$ ,  $I_{rand}$  is a random integer from  $[1, 2, \dots, D]$   
 $I_{rand}$  ensures that  $v_{i,G+1} \neq x_{i,G}$

**Selection:**

The target vector  $x_{i,G}$  is compared with the trial vector  $v_{i,G+1}$  and the one with the lowest function value is admitted to the next generation

$$x_{i,G+1} = \begin{cases} u_{i,G+1}, & \text{if } f(u_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G}, & \text{otherwise} \end{cases} \quad (19)$$

$i = 1, 2, \dots, N$

Mutation, recombination and selection continue until some stopping criterion is reached.

It has been observed from the above graphs (Figs. 4 and Fig. 5) that, the relation between input displacement and voltage output of LVDT are nonlinear before compensation. After compensation by DE algorithm, the nonlinearity is successfully compensated. The DE algorithm has a few control parameters: number of population  $NP$ , scaling factor  $F$ , combination coefficient  $K$ , and crossover rate  $CR$ . The problem specific parameters of the DE algorithm are the maximum generation numbers  $G_{max}$  and the number of parameters designing the problem dimension  $D$ . The values of these two parameters depend on the problem to be optimized. The following results were obtained by using DE algorithm in this research work. From the observed readings of LVDT-1 and LVDT-2 shown in Table 2 and Table 3, the simulation study has been carried out and the following results have been obtained.

The DE algorithm has a few control parameters: number of population  $NP$ , scaling factor  $F$  and crossover rate  $CR$ . In the simulations, it was observed that the value of scaling factor significantly affected the performance of DE. This can be seen in Table 6 and Table 7. In order to get the best performance from the DE, the scaling factor value  $F$  and cross over value  $CR$  must be optimally tuned for each function. Of course, this is a time-consuming task. For the simplicity and flexibility, the value of  $F$  was randomly chosen between  $[0 \ 2]$  and the value of  $CR$  was chosen between  $[0 \ 1]$  for each generation instead of using a constant value. DE algorithm was run 1000 times for each function to achieve average results. For each run, the initial population was randomly created by means of using different seed numbers. The corresponding MSE values and average training time are calculated and listed in Table 6 and Table 7.

**3.4 ANN trained by Genetic algorithm based non-linearity compensation**

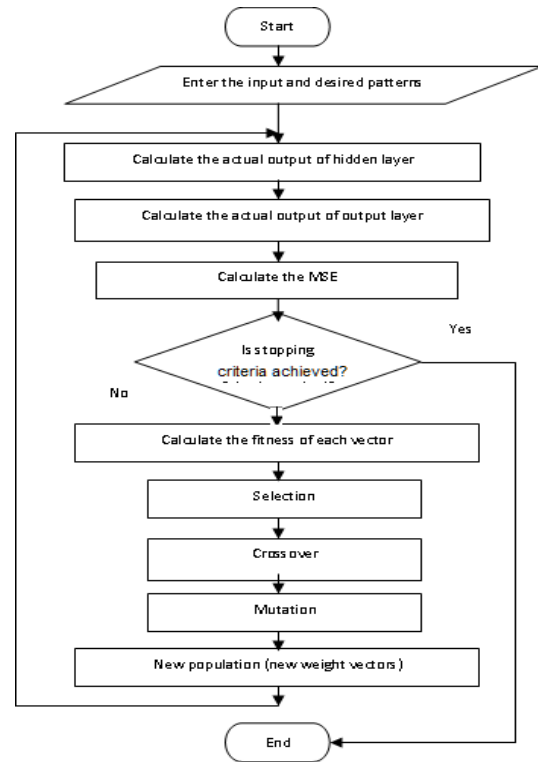
To guide ANN learning, GA is employed to determine the best number of hidden layers and nodes, learning rate, momentum rate and weight optimization. With GA, it is proven that the learning becomes faster and effective.

**Table 6.** DE based nonlinearity compensation for LVDT-1

Iterations	NP	F	CR	MSE	Average training time
1000	100	0.8	0.5	0.0097	7.141830e+02
		0.7	0.4	0.0108	7.282330e+02
		0.6	0.3	0.0084	7.358051e+02
		0.5	0.2	0.0102	7.280086e+02
		1.1	0.8	0.0301	7.326111e+02
		1.2	0.9	0.0094	7.285723e+02
		0.4	0.9	0.0118	7.306492e+02

**Table 7.** DE based nonlinearity compensation for LVDT-2

Iterations	NP	F	CR	MSE	Average training time
1000	100	0.8	0.5	0.0035	7.37e+01
		0.7	0.4	0.0044	7.51e+01
		0.6	0.3	0.0043	7.44e+01
		0.5	0.2	0.0032	7.56e+01
		0.4	0.1	0.0054	7.58e+01
		0.9	0.6	0.0062	7.58e+01
		1.0	0.7	0.0133	7.48e+01
		1.1	0.8	0.0083	7.71e+01
		1.2	0.9	0.0209	7.64e+01
				0.4	0.9



**Fig.7.** Flow chart of GANN weight optimization

The flowchart of GANN for weight optimization is shown in Fig. 7. In the first step, weights are encoded into chromosome format and the second step is to define a fitness function for evaluating the chromosome's performance. This function must estimate the performance of a given neural network. The function usually used is the Mean Squared Errors (MSE). The error can be transformed by using one of the two equations below as fitness value.

$$Fitness = \frac{1}{MSE} \text{ or } Fitness = \frac{1}{1+MSE} \quad (20)$$

In GANN for optimum topology, the neural network is defined by a “genetic encoding” in which the genotype is the encoding of the different characteristics of the MLP and the phenotype is the MLP itself. Therefore, the genotype contains the parameters related to the network architecture, i.e. number of hidden layers (H), number of neurons in each hidden layer (NH), and other genes representing the Bp parameters. The most common parameters to be optimized are the learning rate ( $\eta$ ) and the momentum ( $\alpha$ ). They are encoded as binary numbers. The parameter, which seems to best describe the goodness of a network configuration, is the number of epochs (ep) needed for the learning. The goal is to minimize the ep. The fitness function is:

$$Fitness = \frac{1}{e_p} \text{ or } \frac{1}{1+e_p} \quad (21)$$

The parameters of GANN training algorithm are listed in Table 8 and Table 9. After several runs the genetic search returns approximately the same result each time as the best solution despite the use of different random generated populations and a different population size reaching the lowest value of MSE with a very few number of generations are carried out. The maximum number of training cycles may be set relative to the size of the network.

The first step in developing a neural network is to create a database for its training, testing and validation. The

**Table 8.** ANN Trained by GA based nonlinearity compensation for LVDT-1

Number of population	Average training time	MSE value
10	3.738543e+00	0.0097
20	6.660850e+00	0.0190
30	9.670067e+00	0.0096
50	1.546629e+01	0.0099
60	1.866235e+01	0.0085
70	2.135233e+01	0.0096
80	2.497642e+01	0.0066
100	3.034764e+01	0.0079
500	2.225206e+01	0.0160

**Table 9.** ANN Trained by GA based nonlinearity compensation for LVDT-2

Number of population	Average training time	MSE value
10	3.692170e+00	0.0305
20	6.715500e+00	0.0144
30	9.644655e+00	0.0038
50	1.661117e+01	0.0033
60	1.844437e+01	0.0057
70	2.150443e+01	0.0038
80	2.433312e+01	0.0131
100	3.000041e+01	0.0138
500	3.284778e+01	0.0093

output voltage of LVDT is used to form the other rows of input data matrix. The output matrix is the target matrix consisting of data having a linear relation with the displacement. The process of finding the weights to achieve the desired output is called training. The optimized ANN is found by considering different algorithms with varying number of hidden layers, iterations and epochs. Mean Square Error (MSE) is the average squared difference between outputs and targets. Lower values of MSE are better. Zero means no error. For ANN trained by GA, the number of iterations is assumed initially as 10 and corresponding MSE and training time are noted. Then the iterations are increased to 20 and training is repeated. The process is repeated up to 100 iterations and MSE and training time is noted.

From the observed readings of LVDT-1 and LVDT-2, the simulation study has been carried out and the following results have been obtained.

#### 4. Results

A computer simulation is carried out in the MATLAB. 12 environment using an experimental dataset. The experimental data are collected from two different LVDTs having different specifications shown in Table 1. The data obtained by conducting experiments on the two LVDTs are given in Tables 2 and Table 3. The observed simulation results are shown in various figures listed below. It is observed that ELM model yields the lowest training time of zero seconds to obtain better linearity in the overall response when compared to others. At the same time DE algorithm produces the lowest MSE value of 0.000311 for F=0.4, CR=0.9 and NP=100. The average values of training time and MSE values are compared and listed in Table 10.

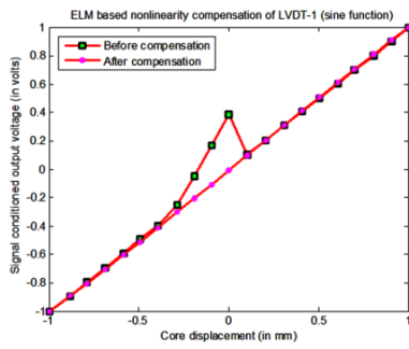
**Table 10.** Comparison of different methodologies for nonlinearity compensation of two different LVDTs (average best values)

Methodology		Training Time (seconds)	MSE/RMSE
ELM Method	Sine	0	0.0087
	Sigmoid	0.156	0.0088
DE algorithm	F = 0.4	1.49e+03	0.000311
	CR = 0.9		
	NP = 100		
GA-ANN	NP=500	3.284778e+01	0.0093

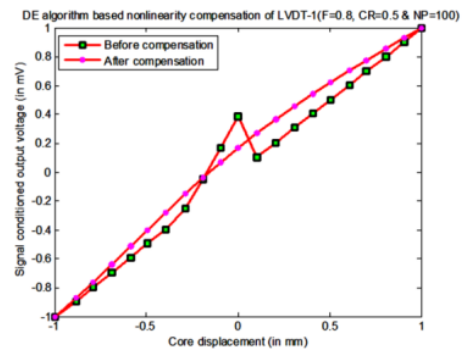
#### 5. Conclusion and Future Scope

This paper has proposed Extreme Learning Machine (ELM) method and two optimized ANN models to adaptively compensate for the nonlinearity offered by two different LVDTs. On comparison, ELM method based nonlinearity compensation produces a less training time and Differential

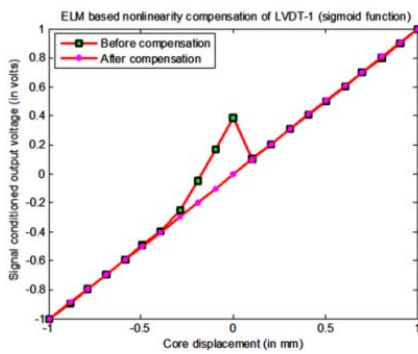




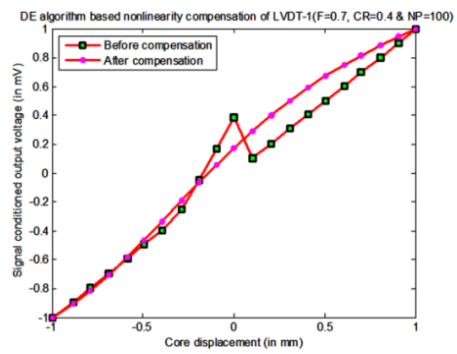
**Fig. 8.** ELM based nonlinearity compensation of LVDT-1 (sine function)



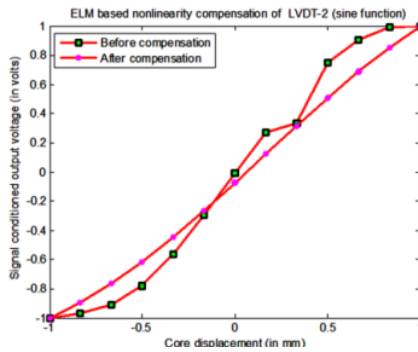
**Fig. 12.** DE algorithm based nonlinearity compensation of LVDT-1(F=0.8, CR=0.5 & NP=100)



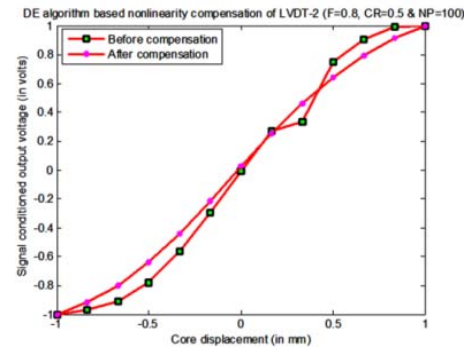
**Fig. 9.** ELM based nonlinearity compensation of LVDT-1(sigmoid function)



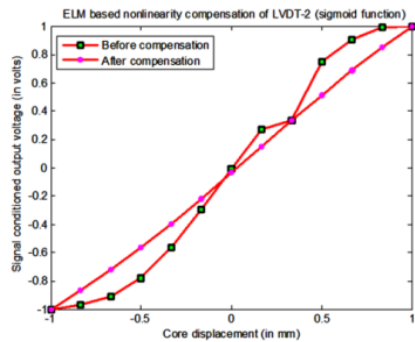
**Fig. 13.** DE algorithm based nonlinearity compensation of LVDT-1(F=0.7, CR=0.4 & NP=100)



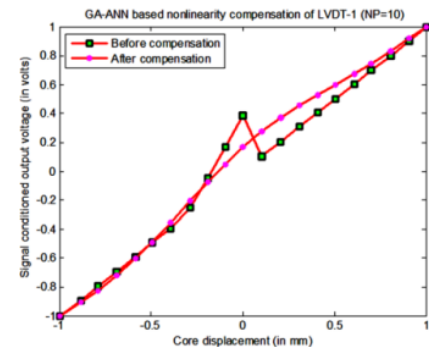
**Fig. 10.** ELM based nonlinearity compensation of LVDT-2(sine function)



**Fig. 14.** DE algorithm based nonlinearity compensation of LVDT-2 (F=0.8, CR=0.5 & NP=100)



**Fig. 11.** ELM based nonlinearity compensation of LVDT-2(sigmoid function)



**Fig. 15.** GA-ANN based nonlinearity compensation of LVDT-1 (NP=10)

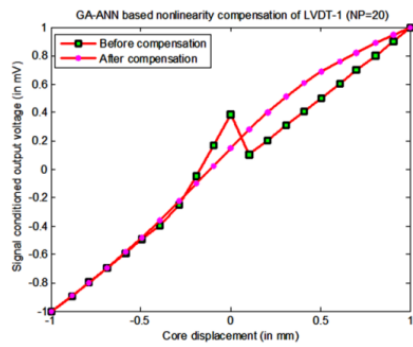


Fig. 16. GA-ANN based nonlinearity compensation of LVDT-1 (NP=20)

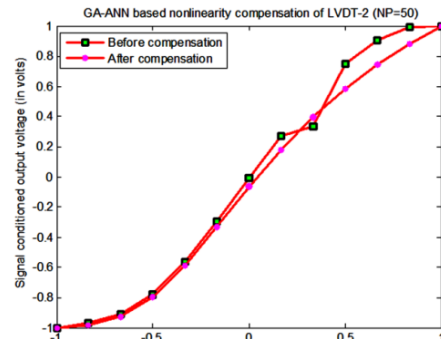


Fig. 20. GA-ANN based nonlinearity compensation of LVDT-2 (NP=50)

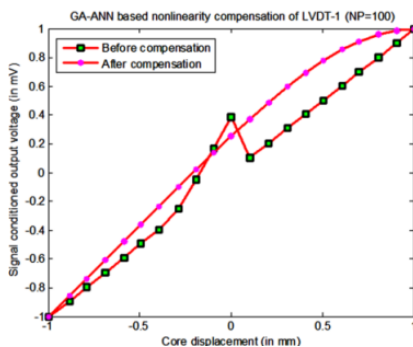


Fig. 17. GA-ANN based nonlinearity compensation of LVDT-1 (NP=100)

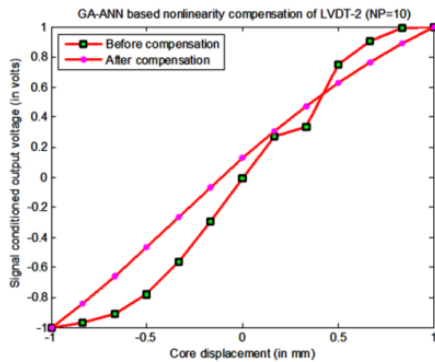


Fig. 18. GA-ANN based nonlinearity compensation of LVDT-2 (NP=10)

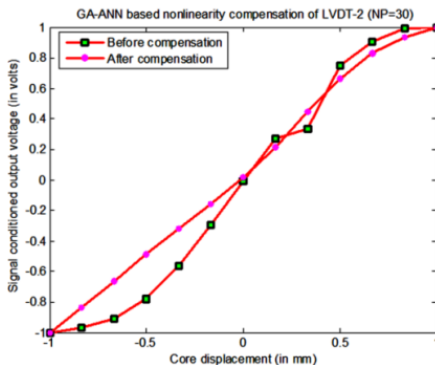


Fig. 19. GA-ANN based nonlinearity compensation of LVDT-2 (NP=30)

Evolution (DE) algorithm based nonlinearity compensation yields better mean square error value when compared to others. Results reveal that ELM method has given best linearization approximation and compensated the nonlinearity with very less training time and lowest MSE among the proposed tools. The proposed algorithm offers a less complexity structure and simple in testing and validation procedure. This adaptive algorithm can also be applied to any transducer having a nonlinear characteristic. This hybrid technique is used to make a transducer output as more linear as possible. Further this adaptive algorithm is preferable for real time implementation also.

### Acknowledgement

The authors would like thank the management of Einstein College of Engineering, Tirunelveli and Ultra College of Engineering and Technology for women, Madurai for providing us the support to publish this completion of research work.

### References

- [1] Saroj Kumar Mishra and Ganapati Panda, IEEE, and Debi Prasad Das, 'A Novel Method of Extending the Linearity Range of Linear Variable Differential Transformer Using Artificial Neural Network', IEEE Transactions on Instrumentation and Measurement, vol. 59, no. 4, pp. 947-953 April 2010.
- [2] Y. Kano, S. Hasebe, and H. Miyaji, "New linear variable differential transformer with square coils," IEEE Trans. Magn., vol. 26, no. 5, pp. 2020-2022, Sep. 1990.
- [3] S. C. Saxena and S. B. L. Seksena, "A self-compensated smart LVDT transducer," IEEE Trans. Instrum. Meas., vol. 38, no. 3, pp. 748-753, Jun. 1989.
- [4] G. Y. Tian, Z. X. Zhao, R. W. Baines, and N. Zhang, "Computational algorithms foe linear variable dif-

- ferential transformers (LVDTs),” *Proc. Inst. Elect. Eng. -Sci. Meas. Technol.*, vol. 144, no. 4, pp. 189-192, Jul. 1997.
- [5] D. Crescini, A. Flammini, D. Marioli, and A. Taroni, “Application of an FFT-based algorithm to signal processing of LVDT position sensors,” *IEEE Trans. Instrum. Meas.*, vol. 47, no. 5, pp. 1119-1123, Oct. 1998.
- [6] R.M. Ford, R. S.Weissbach, and D. R. Loker, “A novel DSP-based LVDT signal conditioner,” *IEEE Trans. Instrum. Meas.*, vol. 50, no. 3, pp. 768-774, Jun. 2001.
- [7] J. C. Patra, A. C. Kot, and G. Panda, “An intelligent pressure sensor using neural networks,” *IEEE Trans. Instrum. Meas.*, vol. 49, no. 4, pp. 829-834, Aug. 2000.
- [8] J. C. Patra, A. van den Bos, and A. C. Kot, “An ANN-based smart capacitive pressure sensor in dynamic environment,” *Sens. Actuators A, Phys.*, vol. 86, no. 1/2, pp. 26-38, Oct. 30, 2000.
- [9] 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,” in *Proc. IEEE Conf. ICISIP*, Jan. 2005, pp. 223-227.
- [10] J. C. Patra and R. N. Pal, “A functional link artificial neural network for adaptive channel equalization,” *Signal Process.*, vol. 43, no. 2, pp. 181-195, May 1995.
- [11] J. C. Patra, R. N. Pal, R. Baliarsingh, and G. Panda, “Nonlinear channel equalization for QAM signal constellation using artificial neural network,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 2, pp. 262-271, Apr. 1999.
- [12] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
- [13] A. Flammini, D. Marioli, E. Sisinni, and A. Taroni, “Least mean square method for LVDT signal processing,” *IEEE Trans. Instrum. Meas.*, vol. 56, no. 6, pp. 2294-2300, Dec. 2007.
- [14] Jun Li, Feng Zhao, “Nonlinear Inverse Modeling of Sensor Characteristics Based on Compensatory Neuro-fuzzy Systems”, *Proc. 1st International Symposium on System and Control in Aerospace and Astronautics*, Harbin, China, January 2006.
- [15] Yi Leng, Genbao Zhao, Qingxia Li, Chunyu Sun, Sheng Li, “A High Accuracy Signal Conditioning Method and Sensor Calibration System for Wireless Sensor in Automotive Tire Pressure Monitoring System”, *Proc. International Conference on Wireless Communications, Networking and Mobile Computing*, Shanghai, China, September 2007.
- [16] Xiaoh Wang, “Non-linearity Estimation and Temperature Compensation of Capacitor Pressure Sensor Using Least Square Support Vector Regression”, *Proc. International Symposium on Knowledge Acquisition and Modeling Workshop*, Beijing, China, December 2008.
- [17] Jagdish C. Patra, Cedric Bornand, Goutam Chakraborty, “Hermite Neural Network-based Intelligent Sensors for Harsh Environments”, *Proc. International Conference on Neural Networks*, Georgia, USA, June 2009.
- [18] Jagdish C Patra, Cedric Bornand, “Development of Chebyshev Neural Network-based Smart Sensors for Noisy Harsh Environment”, *Proc. World Congress on Computational Intelligence*, Barcelona, Spain, July, 2010.
- [19] Zhiqiang Wang, Ping Chen, Mingbo Zhao, “Approaches to Non-Linearity Compensation of Pressure Transducer Based on HGA-RBFNN”, *Proc. 8th World Congress on Intelligent Control and Automation*, Jinan, China, July 2010.
- [20] Yang Chuan, Li Chen, “The Intelligent Pressure Sensor System Based on DSP”, *Proc. 3rd International Conference on Advanced Computer Theory and Engineering*, Chengdu, China, August 2010.
- [21] Jiaoying Huang, Haiwen Yuan, Yong Cui, Zhiqiang Zheng, “Nonintrusive Pressure Measurement With Capacitance Method Based on FLANN”, *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 11, pp. 2914-2920, November 2010.
- [22] Jagdish Chandra Patra, Pramod Kumar Meher, Goutam Chakraborty, “Development of Laguerre Neural-Network- Based Intelligent Sensors for Wireless Sensor Networks”, *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, March 2011.
- [23] Santhosh K V, B K Roy, “An Improved Smart Pressure Measuring Technique”, *Proc. The International Joint Colloquiums on Computer Electronics Electrical Mechanical and Civil*, Muvatupuzha, India, 20-21 September 2011.
- [24] Norhayati Soin, Burhanuddin Yeop Majids, “An Analytical Study on Diaphragm Behavior for Micro-Machined Capacitance Pressure Sensor”, *Proc. International Conference on Semiconductor Electronics*, Penang, Malaysia, December, 2002.
- [25] Zongyang Zhang, Zhimin Wan, Chaojun Liu, Gang Cao, Yun Lu, Sheng Liu, “Effects of Adhesive Material on the Output Characteristics of Pressure Sensor”, *Proc. 11th International Conference on Electronic Packaging Technology and High Density Packaging*, Shanghai, China, August, 2011.
- [26] Cui Chun sheng, Ma Tie Hua, “The research of temperature compensation Technology and High-temperature Pressure Sensor”, *Proc. International Conference on Electronic & Mechanical Engineering and Information Technology*, Harbin, China, August, 2011.
- [27] Neubert, H. K. P, *Instrument Transducers: an Introduction to their Performance and Design*, Clarendon Press, Oxford 1975.
- [28] Bela G Liptak, *Instrument Engineers’ Handbook: Process Measurement and Analysis*, 4th Edition, CRC Press, June 2003.
- [29] Jeng-Bin Li, Yun-Kung Chung, “A Novel Back propagation Neural Network Training Algorithm Designed by an Ant Colony Optimization”, *IEEE/*

PES Transmission and Distribution Conference & Exhibition: Asia and Pacific Dalian, China 2005

- [30] L. Bianchi, L.M. Gambardella, M. Dorigo, "An ant colony optimization approach to the probabilistic travelling salesman problem", Proc. of PPSN-VII, Seventh Inter17 national Conference on Parallel Problem Solving from Nature, Springer Verlag, Berlin, Germany, 2002.
- [31] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks, International Joint Conference on Neural Networks, Vol. 2, pp: 985-990, 2004.
- [32] Dianhui Wang, Guang-Bin Huang, "Protein Sequence Classification Using Extreme Learning Machine, Proceedings of International Joint Conference on Neural Networks, Vol. 3, pp: 1406- 1411, 2005.
- [33] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, Extreme Learning Machine: Theory and Applications, Neurocomputing, Vol. 70, pp: 489-501, 2006.
- [34] Guang-Bin Huang, Lei Chen, Chee-Kheong Siew, Universal Approximation Using Incremental Constructive Feed forward Networks with Random Hidden Nodes, IEEE Transactions on Neural Networks, Vol. 17, No. 4, pp: 879-892, 2006.
- [35] Nan-Ying Liang, Guang-Bin Huang, Hai-Jun Rong, P. Saratchandran, N. Sundararajan, A Fast and Accurate On-line Sequential Learning Algorithm for Feed forward Networks, IEEE Transactions on Neural Networks, Vol. 17, No. 6, pp: 1411-1423, 2006.



**SP. Umayal** is a professor and Head of Electrical and Electronics Engineering of ULTRA College of Engineering and Technology for Women, Madurai. She completed her Bachelor of Engineering in Electrical and Electronics Engineering at Thiyagarajar college of Engineering in the year of 1990 and Master of Engineering in Power System in the year 1999 in the same institution. She received her Ph.D in the year 2008 in the field of Power System Optimization. Her areas of research include Power system optimization, FACTS and power quality. She has 35 publications in her research area. Presently she is guiding 10 P.G and 5 Ph.D. research scholars. She is the life member of IE (I) and ISTE.



**S. Murugan** received the B.E degree in Electronics and Instrumentation Engineering from Manonmaniam Sundaranar University, Tamil Nadu, India, in 2002 and the M.E. degree in Control and Instrumentation engineering from the Anna University, Tamil Nadu, India, in 2005. From 2005 to

2006, he was a Lecturer with the Kamaraj College of Engineering and Technology, Virudhunagar. From 2007 to 2010, he was an Assistant Professor with the Department of Electrical and Electronics Engineering, Francis Xavier Engineering College. In July 2011, he joined Ph.D as a part time research scholar at Anna University, Chennai, India. He is currently working as an Associate Professor with Department of Electrical and Electronics Engineering, Einstein College of Engineering, Tirunelveli, India. His research interests include Sensor design, Measurement system design, and Control systems. He was born in the year 1981 at Srivilliputhur, Tamil Nadu, India.