Nonlinearity Error Compensation of Venturi Flow Meter using Evolutionary Optimization Algorithms

S. Murugan
Associate Professor
Department of Electrical and Electronics Engineering
Francis Xavier Engineering College affiliated to Anna University, Chennai, Tamil Nadu, India

Dr. SP. Umayal
Professor
Department of Electrical and Electronics Engineering
Muthayammal Engineering College affiliated to Anna University, Chennai, Tamil Nadu, India

Dr. K. Srinivasan
Associate Professor
Department of Electrical and Electronics Engineering
Francis Xavier Engineering College affiliated to Anna University, Chennai, Tamil Nadu, India

M. Aruna
Assistant Professor
Department of Electrical and Electronics Engineering
Francis Xavier Engineering College affiliated to Anna University, Chennai, Tamil Nadu, India

Abstract

Linearization of sensor is one of the significant issues that must always be considered to guarantee a measurement system's accuracy. Often in the progress of linearization, certain other errors also minimized. It is necessary for most of the sensor systems to have a linear performance. But since in practice there are some factors which brings non-linearity in a system. This paper focuses on the compensation of problems faced due to the non- linear response characteristics of venturi. The evolutionary algorithms used in this work are extreme learning machine (ELM), differential evolution (DE) and artificial neural network trained by genetic algorithm (GA-ANN). These algorithms when connected in series with the sensor offers extended linearity characteristics. The overall system provides accurate measurement for the whole range. A computer simulation is carried out using the experimental dataset of venturi sensor. It is observed that ELM method yields the lowest training time of zero seconds to obtain best linearity in the overall response when compared to others. At the same time DE algorithm and GA-ANN produces the lowest MSE value and better linearity. The proposed algorithm offers a less complexity structure and simple in testing and validation procedure. This hybrid technique is used to make a sensor output as more linear as possible. Further this adaptive algorithm is preferable for real time implementation also.
Keywords: Venturi, Nonlinearity, Extreme Learning Machine (ELM), Differential Evolution (DE) algorithm, ANN trained by Genetic Algorithm (GA)

I. Introduction

Accurate measurement of liquid flow using venturi in industry is essential to control many parameters. According to Bernoulli laws, the flow rate is determined inferentially by measuring the liquid's velocity or the change in kinetic energy. Velocity depends on the pressure differential that is forcing the liquid through a pipe. Because the pipe's cross-sectional area is known, the average velocity is an indication of the flow rate. Because of its high sensitivity and ruggednessventuri finds a very wide application. However the problem of offset, high non-linear response characteristics, dependenceof output on the ratio between venturi and pipe diameter, liquid density and temperature have limited its use andfurther imposing difficulties. To overcome the difficultiesfaced due to the nonlinear response characteristics of theventuri, several techniques have been suggested which aretedious and time consuming. Further, the process of calibration needs to be repeated every time the diameter ratio or liquid is changed. The problem of nonlinear response characteristics of a venturi further aggravates the situation when there is change in liquid temperature. Since the output of venturi is dependent on flow rate as well as temperature of the liquid. To overcome the above difficulties, nonlinearity compensation of flow transducer is proposed in this paper using evolutionary optimization techniques. This network is to train the systemto extend linearity range and makes the output independent of ratio of diameter between venturi and pipe, liquid density and temperature.

In [1], calibration of orifice is discussed. In [2],measurement of flow for different area of venturi nozzle isdiscussed. In [3] Calibration of flow meter is done with thehelp of microcontrollers. In [4] & [11], a simulation modelof venturi flow meter for measurement of flow rate isdiscussed. In [5], [7], [8], [10] & [12], linearization of venturi is discussed using neural network algorithms. In [6],different flow measurements are discussed. In [9],linearization of venturi flow meter is discussed usingmathematical computations. In [13], linearization ofcapacitive level sensor and making the output independentof liquid using neural network algorithm is discussed.Linearization of capacitive pressure sensor and making theoutput independent of physical parameters of diaphragmusing neural network algorithm is discussed in [14].

This paper is organized as follows: after introduction in Section 1, a brief description on flow measuring transducer, venturi is given in Section 2. Experimental observation of venturi is 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 venturi 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.

A. Concept of Linearity

There are many definitions of linearity that exist. However, linearity defined in terms of independent linearity is the most preferred, in many cases. The computation of independent linearity is done with reference to a straight line representing the ideal 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 point from this straight line. The independent non-linearity or linearity error for any particular value of the input, may be defined as

$$\text{Nonlinearity}=\frac{\text{Deviation of the output from the idealized straight line}}{\text{Actual output (or)Full scale output}}\times 100 \quad (1)$$

Equation (1) expresses the non-linearity in terms of the percentage of the instrument reading and therefore recognizes the desirability of a constant percentage nonlinearity. The least squares best fit straight line method is preferred by most sensor manufacturers because it provides the closest possible best fit to all data points on the curve, and can be most readily adapted to the computerized calibration systems in common use and the characteristics of the sensors are correctly optimized at the design and development stage and are represented by a continuous smooth curve, the assessment is meaningful and accurate.

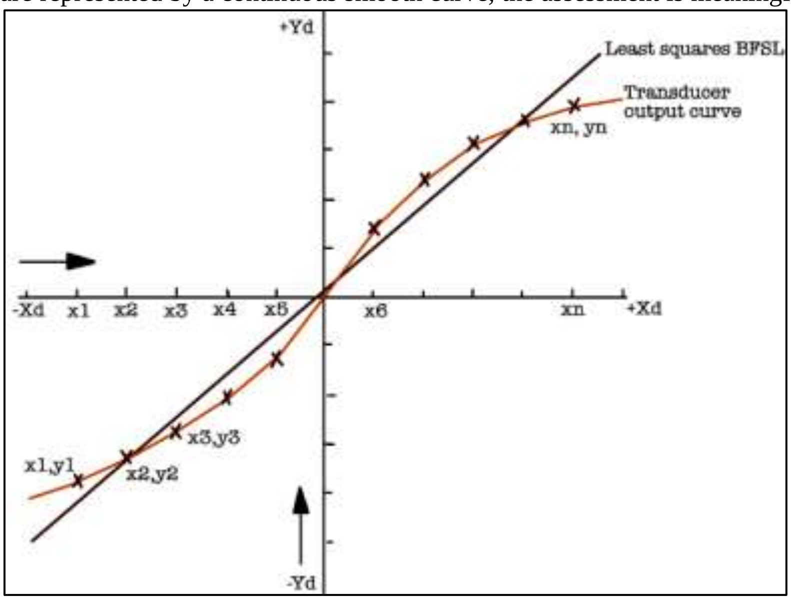

Fig. 1: Least square best fit straight line method

Equation (2) expresses the nonlinearity as a percentage of full scale reading.

$$\% \text{ of nonlinearity}=\left(\frac{Yd_{max}-X_d}{X_{fr}}\right)100 \qquad (2)$$

Where Xf_r = full scale reading
X_d= Known input data points
Y_d= actual transducer output at each X_d data point

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. Because of the mechanism structure and others, sensor often exhibit inherent nonlinear input-output characteristics. Complicated and accurate winding machines are used to solve this. It is difficult to have all sensor 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 sensor 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.

B. *Venturi Flow Meter*

A venturi flow meteris a device used for measuring the volumetric flow rate, follows the Bernoulli's principle which gives a relationship between the pressure and the velocity of the fluid. When the velocity increases, the pressure decreases and vice versa. A venturi nozzle is a tapered structure. It is usually placed in a pipe in which fluid flows. When the fluid reaches the venturi nozzle, the fluid is forced to converge to go through the small hole. The point of maximum convergence actually occurs shortly downstream of the venturi nozzle, which is called vena-contracta point. As it flows so, the velocity and the pressurechanges. Beyond the vena-contracta, the fluid expands and the velocity and pressure changes once again. Thevolumetric and mass flow rates can be obtained fromBernoulli's equation by measuring the difference in fluidpressure between the normal pipe section and at the vena-contracta[9], [15] and [16].

$$Q=C_d A_b \sqrt{\frac{1}{\left(1-\beta^4\right)}}\sqrt{\frac{2\Delta P}{\rho}} \qquad (3)$$

Where
C_d-Discharge coefficient
A_b-Area of the flow meter cross section
β-Ratio of D_b to D_a
P_a-Pressure at steady flow
P_b-Pressure at vena-contracta
ρ-Density of liquid

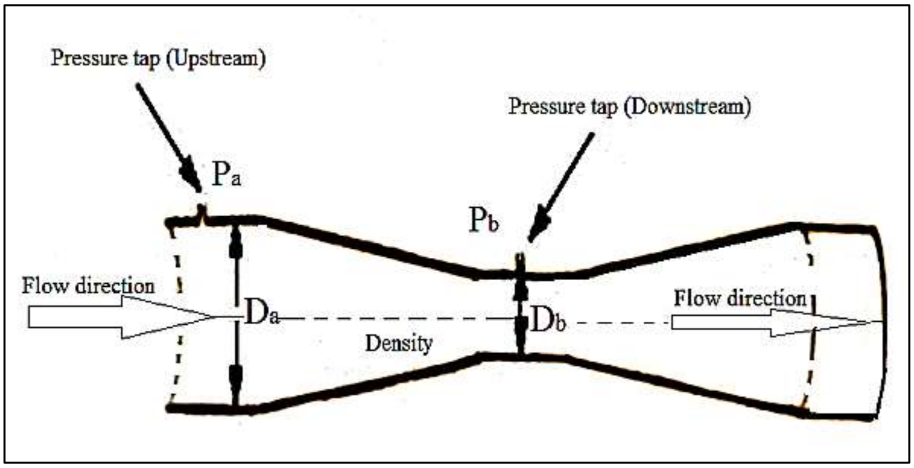

Fig. 2: Venturi nozzle

The block diagram representation of the proposed instrument is shown in Fig 2.

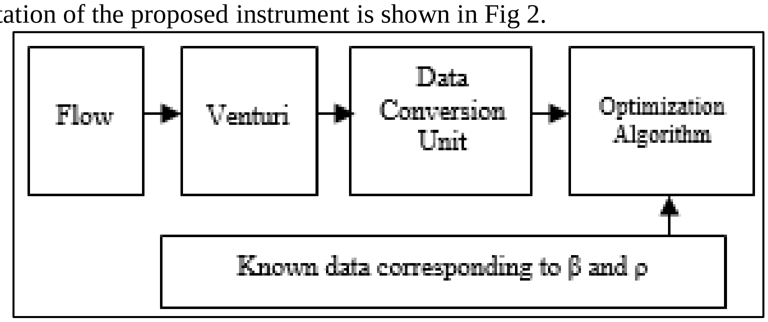

Fig. 3: Proposed block diagram

In this research work, we have taken the performance of venturi flow transducer. The experimental data is collected from the transducer and it is given in Table 1. The output response curve of transducer is shown in Fig.3. It is clear that the output response of the flow transducer shows the presence of nonlinearity.

Table - 1
Experimental observation of flow transducer

Actual flow rate (in cubic centimeter per second)	*Voltage output from data conversion unit (in volts)*
0.1	*0.180*
0.2	*0.410*
0.3	*0.685*
0.4	*0.912*
0.5	*1.165*

0.6	*1.448*
0.7	*1.582*
0.8	*1.862*
0.9	*1.912*
1.0	*2.000*
1.1	*2.342*
1.2	*2.658*
1.3	*2.985*
1.4	*3.258*
1.5	*3.541*
1.6	*3.758*
1.7	*3.895*
1.8	*4.232*
1.9	*4.652*
2.0	*4.880*

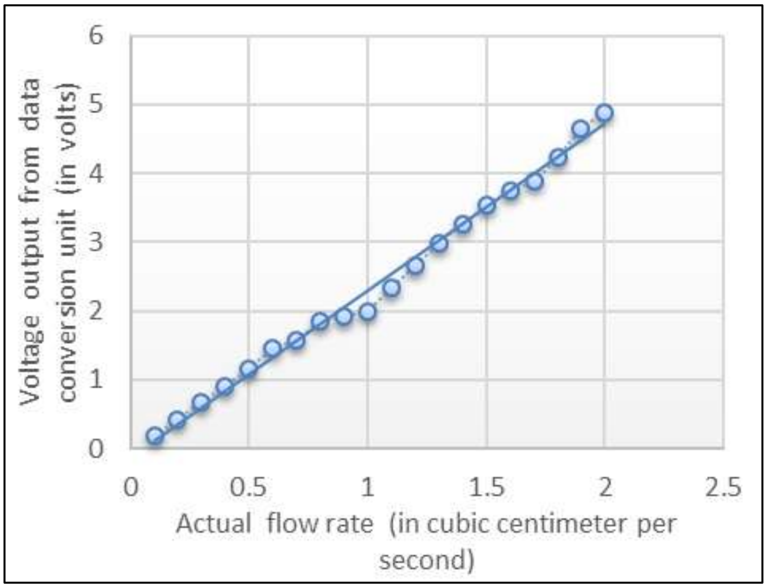

Fig. 3: Input-Output characteristics of flow transducer

It has been observed from the above graph (Fig.3), that the relation between input flow rate and voltage output of venturi flow transducer are nonlinear. The following algorithms are used to compensate the nonlinearity of transducer 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)

II. NONLINEARITY COMPENSATION USING OPTIMIZATION TECHNIQUES

A. Extreme Learning Machine based Nonlinearity Compensation

A new learning algorithm for Single Layer Feed forward Neural Network (SLFN) architecture called Extreme Learning Machine (ELM) which overcomes the problems caused by gradient descent based algorithms. 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 conventional methods. ELM automatically determines all the network parameters analytically, which avoids trivial human intervention and makes it efficient in online and real time applications.

1) Mathematical Model:

$$\sum_{i=1}^{L} \beta_i G(a_i,b_i,X_j) = t_j, j=1,\ldots,N, \text{ is equivalent to } H\beta=T,$$

Where

$$H=\begin{bmatrix} h(X_1) \\ . \\ . \\ . \\ h(X_N) \end{bmatrix}=\begin{bmatrix} G(a_1,b_1,X_1) & . & . & G(a_L,b_L,X_1) \\ & & . & \\ & & . & \\ & & . & \\ G(a_1,b_1,X_N). & & . & G(a_L,b_L,X_N) \end{bmatrix} \qquad (2)$$

$$\beta=\begin{bmatrix} \beta_1^T \\ . \\ . \\ . \\ \beta_L^T \end{bmatrix} \text{and} T=\begin{bmatrix} t_1^T \\ . \\ . \\ . \\ t_L^T \end{bmatrix} \qquad (4)$$

H is called the hidden layer output matrix of SLFN, the i^{th} column of H is the output of the i^{th} hidden node with respect to inputs $x_1,x_2,\ldots,x_N$

2) *Three Step Learning Model:*

Given a training set $\aleph=\{(X_i,t_i)|X_i\epsilon R^n,t_i\epsilon R^m,i=1,\ldots,N\}$, hidden node output function G(a,b,X), and the number of hidden nodes L,
Generate random hidden nodes $(a_i,b_i),i=1,\ldots.,L$
Calculate the hidden layer output matrix H
Calculate the output weight vector β:$\beta=H\dagger T$
Where H† is the Moore-Penrose generalized inverse of hidden layer output matrix H.
From the observed readings of flow transducer shown in Table.1, the simulation study has been carried out and the following results have been obtained.

Table - 2
Simulation results of ELM based nonlinearity compensation

Activation function	*Training Time (sec)*	*Testing time (sec)*	*Testing accuracy*	*RMSE*
sine	*0.2652*	*0*	*0.0066*	*0.0066*
sigmoid	*0*	*0*	*0.3436*	*0.3436*

The results obtained by ELM based nonlinearity compensation of flow transduceris listed in Table 2. Two different activations functions namely sine and sigmoid are used here. The training time, testing time and Root Mean Squared Error (RMSE) values are tabulated. The testing and training time are zero by using sine function and sigmoid function respectively. There are 20 hidden nodes assigned for ELM algorithm. 50 trials have been conducted for the algorithm and the average results are shown in Table 2. It can be seen from Table 2 that ELM learning algorithm spent zero seconds CPU time obtaining the testing root mean square error (RMSE) 0.0066 with sine activation function, and zero seconds CPU time obtaining the RMSE value of 0.3436 with sigmoid activation function. The new ELM runs 170 times faster than the conventional BP algorithms.

B. Differential Evolution Algorithm Based Nonlinearity Compensation

The DE algorithm has a few control parameters: number of population NP, scaling factor $F$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. 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 effectively improved by applying mutation, crossover and selection operators.
The common problem formulation is:

For an objective function $f\colon X\subseteq \mathbb{R}^D \longrightarrow \mathbb{R}$ where the achievable 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 $f(x^*)\neq -\infty$

- Step1: Population - To optimize a function with D real parameters, select the size of the population N (it must be at least 4). The parameter vectors have the form: $x_{i,G}=[x_{1,i,G},x_{2,i,G},\ldots\ldots.x_{D,i,G}]$

$$i=1,2,\ldots\ldots N \qquad (5)$$

Where, G is the generation number

- Step2: Initialization - Express upper and lower limits for each parameter

$$x_j^L\leq x_{j,i,1}\leq x_j^U \qquad (6)$$

Arbitrarily select the initial parameter values frequently on the intervals

$$[x_j^L,x_j^U] \qquad (7)$$

- Step3: Mutation - Mutation enlarges the search planetary. For a known parameter vector $x_{i,G}$ arbitrarily select three vectors $x_{r1,G},x_{r2,G}$ and $x_{r3,G}$ such that the indices $i,r1,r2$ and $r3$ are different. Add the weighted difference of two of the vectors to the third.

$$v_{i,G+1}=x_{r1,G}+F(x_{r2,G}-x_{r3,G}) \quad (8)$$

- Step4: Recombination - Recombination combines effective 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

$$u_{j,i,G+1} = \begin{cases} v_{j,i,G+1}, & if\ rand_{j,i} \le CR\ or\ j = I_{rand} \\ x_{j,i,G}, & if\ rand_{j,i} > CR\ and\ j \ne I_{rand} \end{cases} \quad (9)$$

$i = 1,2, \ldots \ldots N; j = 1,2, \ldots \ldots D$

$rand_{j,i}$~U[0,1], I_{rand} is a random integer from [1,2,……D]

I_{rand} ensures that $v_{i,G+1} \ne x_{i,G}$

- Step5: 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.

$$x_{i,G+1} = \begin{cases} u_{i,G+1}, & if\ f(u_{i,G+1}) \le f(x_{i,G}) \\ x_{i,G}, & otherwise \end{cases} \quad (10)$$

It has been observed from the above graphs (Fig.2) that, the relation between input flow rate and voltage output of flow transducer 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 flow transducer shown in Table 2, the simulation study has been carried out and the following results have been obtained.

Table – 3
DE based nonlinearity compensation for venturi flow meter

Iterations	*NP*	*F*	*CR*	*MSE*	*Average Training Time (seconds)*
100	*100*	*0.9*	*0.2*	*0.0019*	*1.495151e+02*
		0.8	*0.4*	*0.0035*	*1.502282e+02*

C. *ANN Trained by Genetic Algorithm Based Nonlinearity 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. The flowchart of GANN for weight optimization is shown in Fig.5. 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.

$$Fitness = \frac{1}{MSE} \text{or} Fitness = \frac{1}{1+MSE} \quad (11)$$

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 (η) and the momentum (α). 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:

$$Fitness = \frac{1}{e_p} or \frac{1}{1+e_p} \quad (12)$$

The parameters of GANN training algorithm are listed in Table 4. 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.

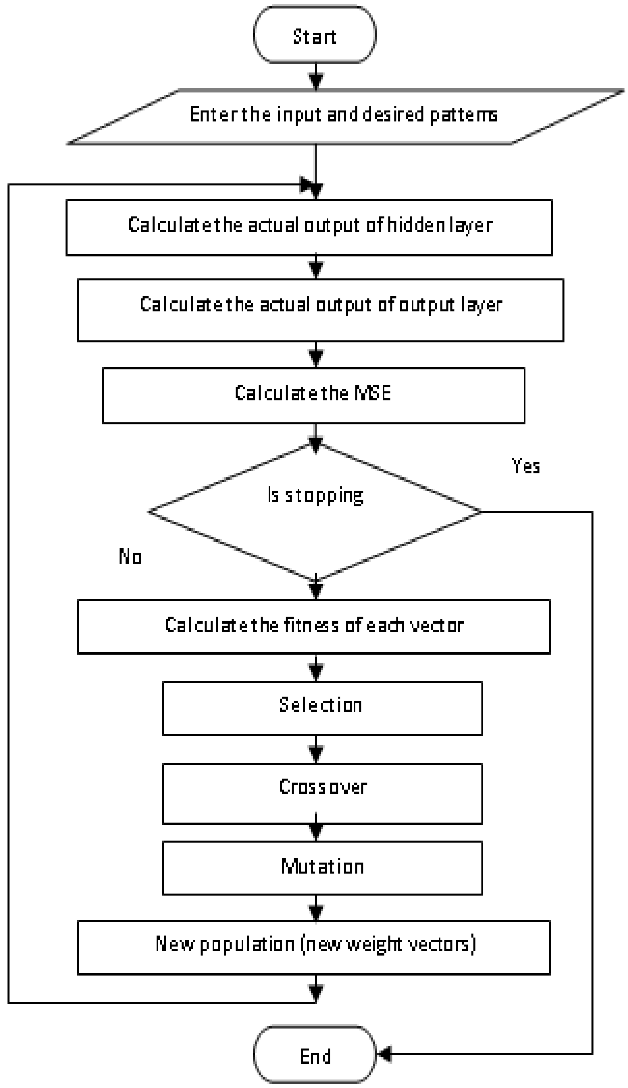

Fig. 4: Flow chart of GANN weight optimization

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

Table – 4
ANN Trained by GA based nonlinearity compensation of flow transducer

Number of Population	*Average training time (seconds)*	*MSE value*
100	*3.196334e+01*	*0.0036*
35	*1.147918e+01*	*0.0025*
20	*6.911981e+00*	*0.0029*

III. Results

A computer simulation is carried out in the MATLAB.12 environment using an experimental dataset. The experimental data are collected from flow transducer is shown in Table 1. 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 and GA-ANN algorithm produces the lowest MSE value.

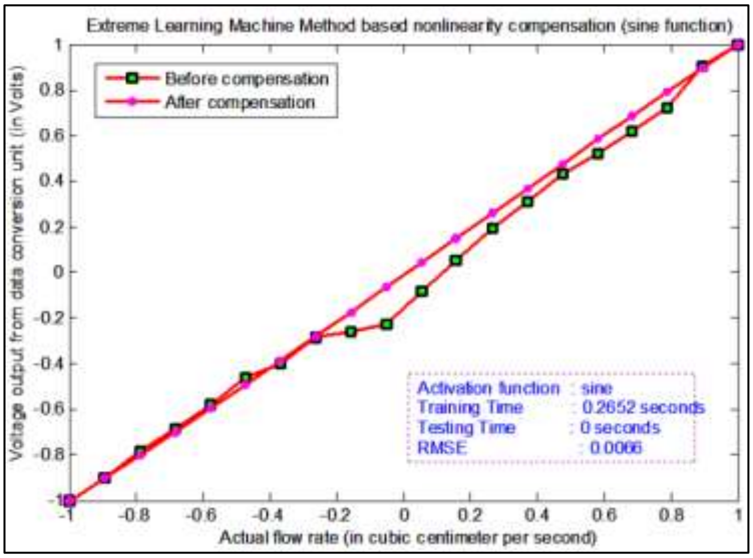

Fig. 5: ELM based nonlinearity compensation (sine)

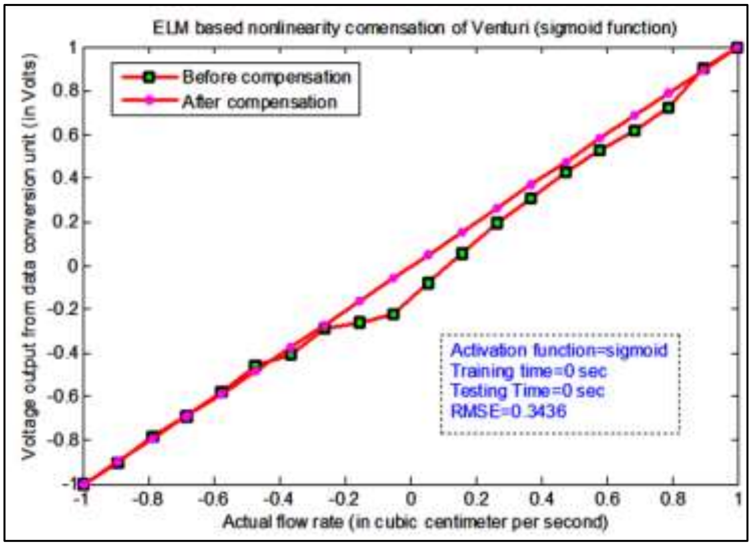

Fig. 6: ELM based nonlinearity compensation (sigmoid)

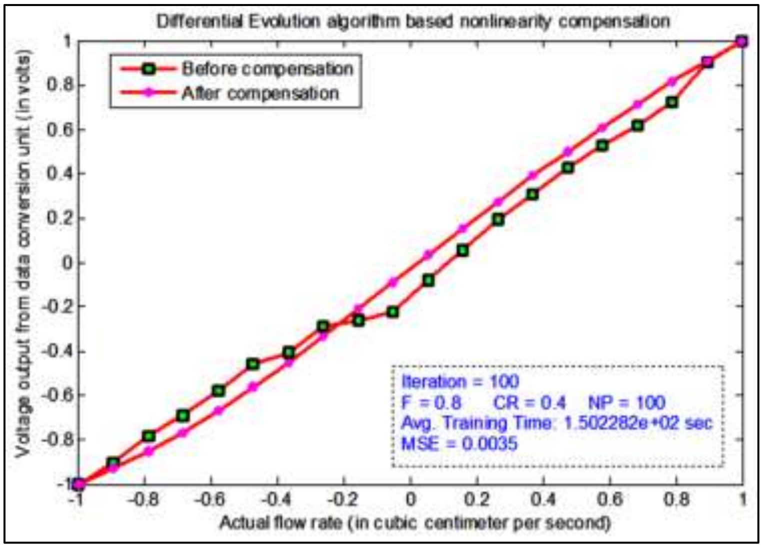

Fig. 8: DE based nonlinearity compensation
(MSE = 0.0035)

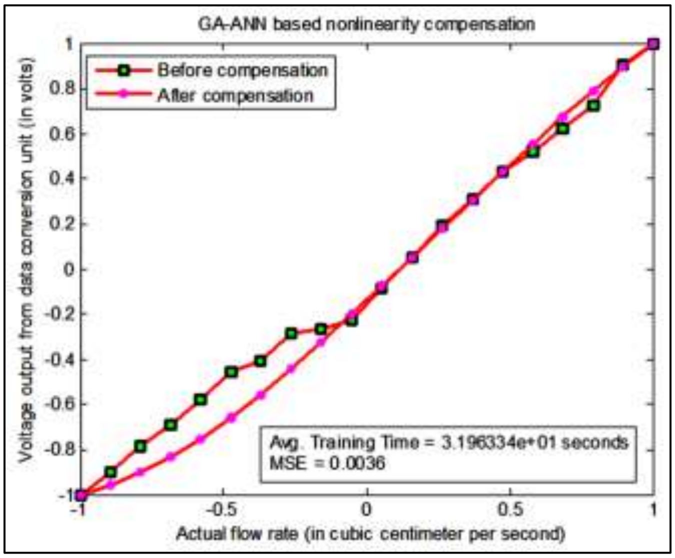

Fig. 9: GA-ANN based nonlinearity compensation
(NP = 100)

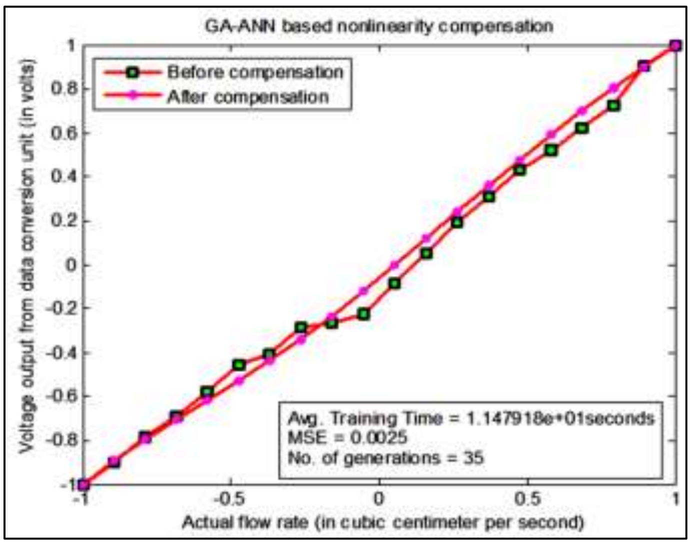

Fig. 10: GA-ANN based nonlinearity compensation
(NP = 35)

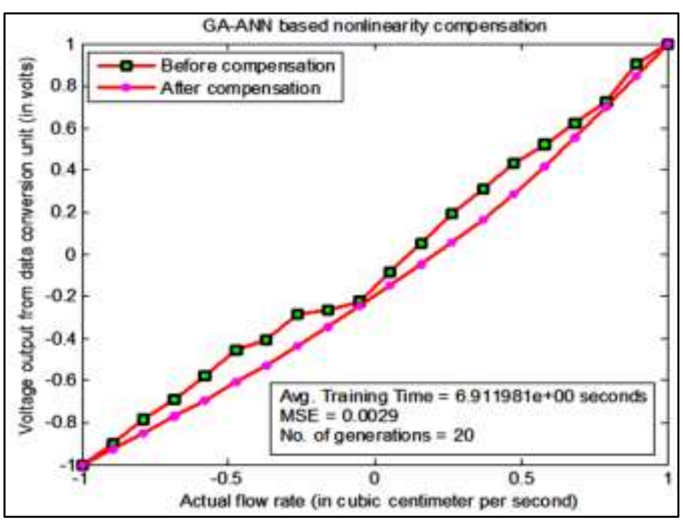

Fig. 11: GA-ANN based nonlinearity compensation
(NP = 20)

IV. CONCLUSION

This paper has proposed Extreme Learning Machine (ELM) method and two optimized ANN models to adaptively compensate for the nonlinearity offered by flow transducer. On comparison, ELM method based nonlinearity compensation produces a less training time and Differential Evolution (DE) algorithm and GA-ANN based nonlinearity compensation yields better mean squared 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.

REFERENCES

[1] J. E. Lawley, “Orifice meter calibration", 57th International School of Hydrocarbon Measurement, document ID: 16233, 1982.
[2] Shigeru Yanagiharaa, Osamu Mochizukia, KyosukeSatoa, KeizoSaitob, “Variable area venturi type exhaust gas flow meter", JSAE Review, Science direct, vol 20, No 2 pp 265 - 267, April 1999.
[3] JianQiu Zhang, Yong Yan, “A self validating differential pressure flow sensor", IEEE conference Proceedings on Instrumentation and Measurement Technology, Budapest, May 2001.
[4] Man Gyun Na, Yoon Joon Lee, In Joon Hwang, “A smart software sensor for feedwater flow measurement monitoring", IEEE transactions on Nuclear Science,vol 52, No 6, pp 3026 - 3034, December 2005.
[5] LijunXu, Hui Li, Shaliang Tang, Cheng Tan, Bo Hu, “Wet Gas Metering Using a Venturi-meter and Neural Networks", IEEE conference Proceedings on Instrumentation and Measurement Technology, Canada, May 2008.
[6] Pereira M, “Flow meters: Part 1", IEEE Magazine on Instrumentation and Measurement, vol.12, No 1, pp 18 - 26, February 2009.
[7] Pereira J.M.D, Postolache O, Girao P.M.B.S, “PDF-Based Progressive Polynomial Calibration Method for Smart Sensors Linearization", IEEE transactions on Instrumentation and Measurement, vol.58, No 9, pp 3245 - 3252, September 2009
[8] Heon Young Yang, Sung Han Lee, Man Gyun Na, “Monitoring and Uncertainty Analysis of Feedwater Flow Rate Using Data-Based Modeling Methods", IEEE transactions on Nuclear Science, vol. 56, No 4, pp 2426 - 2433, August 2009.
[9] LijunXu, Shaliang Tang, “Wet gas metering using a Venturi-meter and Support Vector Machines", IEEE conference Proceedings on Instrumentation and Measurement Technology, Singapore, May 2009.
[10] Fang Lide, Cao Suosheng, Li Jinhai; Cheng Hao, “Venturi Wet Gas measurement based on homogenous and chisholm model", International Conference on Computer and Automation Engineering, Thailand, 2009.
[11] LijunXu, Wanlu Zhou, Xiaomin Li, Minghao Wang, “Wet-Gas Flow Modeling for the Straight Section of Throat-Extended Venturi Meter", IEEE transactions on Instrumentation and Measurement, vol.60, No 6, pp 2080 - 2087, 2011.
[12] LijunXu, Wanlu Zhou, Xiaomin Li, Shaliang Tang, “Wet Gas Metering Using a Revised Venturi Meter and Soft-Computing Approximation Techniques", IEEE transactions on Instrumentation and Measurement, vol.60, No 3, pp 946 - 2087, March 2011.
[13] B K Roy, Santhosh K V, “An Intelligent Instrument for Measuring Liquid Level", IEEE Conference on process automation control and computing (PACC2011), Coimbatore, India, July 2011.
[14] Santhosh K V, B K Roy, “An Intelligent Pressure Measuring Instrument", International conference on modern trends in instrumentation and control (ICIC 2011), Coimbatore, India, September 2011.
[15] Guang-Bin Huang, Qin-Yu Zhu, Chee-KheongSiew, Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks, International Joint Conference on Neural Networks, Vol. 2, pp: 985-990, 2004.
[16] 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.
[17] Guang-Bin Huang, Lei Chen, Chee-KheongSiew, 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.
[18] 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.