A Cognitive Neural Linearization Model Design for Temperature Measurement System based on Optimization Algorithm

Dr. Hayder Abd Dahad

Mechanical Engineering Department, University of Technology, Baghdad

e-mail: hayder_abed2002@yahoo.com

Received: 29/12/2014

Accepted: 19/5/2015

Abstract – The main core of this paper is to design an experimental method for estimating of the nonlinearity, calibrating and testing of the different types of thermocouples temperature sensors (J, K, T, S and R) using multi-layer perceptron (MLP) neural network based on slice genetic (SG) optimization learning algorithm. Temperature sensor has a nonlinearity behavior nature in its output response but it requires a linear behavior output with accepts approximation in accuracy level, noise and measurement errors. Therefore, neural network topology is proposed with five main steps algorithm to reduce the effected noise and minimize the measured errors. Matlab simulation results and laboratory work (LabVIEW) validate the preciously of the proposed cognitive neural linearization algorithm in terms of calculating the temperature from the different types of thermocouples temperature sensors and minimizing the error between the actual temperature output and neural linearization temperature output as well as overcoming the problem of the over learning in the linearization model with the minimum number of fitness evaluation for the learning algorithm..

.

Keywords: – Thermocouple Temperature Sensors, Neural Network Topology, Slice Genetic Algorithm, Matlab, LabVIEW.

1. Introduction

Temperature sensors are essential elements in many of the industrial applications in order to monitor and control the temperature [1]. In general, sensors take a certain form of input (temperature, pressure, altitude, etc.) and convert it, through read-out circuitry, into readings that can be interpreted. However, many types of sensors are nonlinear in nature from which a linear output is desired. There are many different sensors for temperature measurement such as thermocouple types (J, K, T, S and R) which are the most commonly used [2].

They are preferred in industrial applications due to their low cost, wide operation range and fast response time. Thermocouples also have nonlinear outputs related to temperature. Therefore, the proposed a cognitive neural network linearization technique is necessary in order to solve the problem of linearizing a sensor. On the other hand, the analog circuits are frequently used for improving the linearity of the sensor characteristics, which implies additional analog hardware and typical problems particular to analog circuits such as temperature drift, gain and offset error [3].

In recent years, application of artificial neural networks (ANNs) has emerged as a promising area of research in the field of instrumentation and measurement such as explained in [4], [5] and [6] because whenever the system has a relation-ship between the input and output, and these relation-ships have a capability for learning, re-planning, re-assembly, re-origination based neural networks this system can be called a cognitive neural network [7].

The contribution of this paper is a precision linearization temperature system based on the cognitive neural network topology with slice genetic learning algorithm used as fast and stable optimization technique.

The following section contains the description of the temperature measurement system based National Instrument Data Acquisition. Section three, derives the proposed cognitive neural network linearization topology with slice genetic algorithm. Section four, shows the simulation results and laboratory work of the proposed cognitive neural linearization algorithm and the last section explains the conclusions of the research.

2. Temperature Measurement System

Thermocouples are the most popular temperature sensors used in applications because there are many advantages such as cheap, standard connectors, measure a wide range of temperatures, self-powered, requiring no external power supply, extremely rugged, not fragile, and can withstand harsh environments over other types of temperature sensors such as RTDs and thermistors [8].

A thermocouple generates a voltage proportional to the measurement junction temperature at mV levels while the cold junction temperature must be known and constant in order to make an accurate measurement [3]. The block diagram of the temperature measurement system is shown in Fig. 1 based on data acquisition system from National Instrument (NI) with LabVIEW programming [9] and [10].

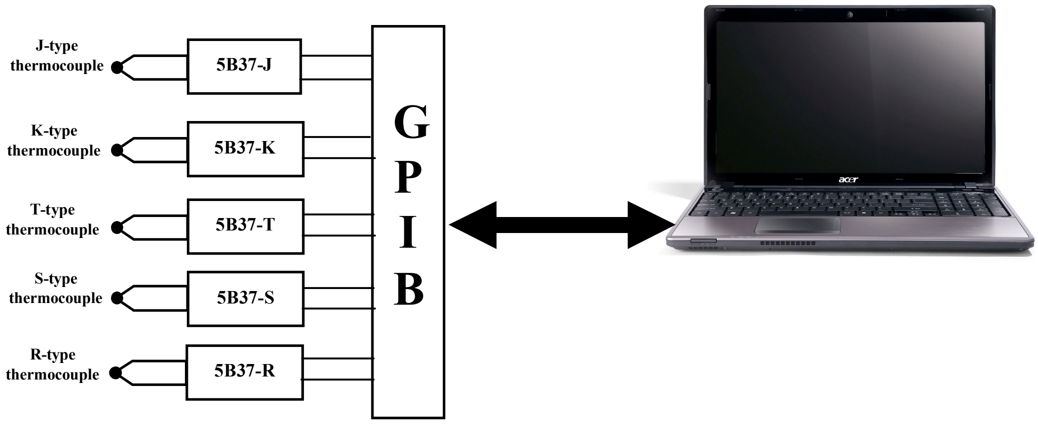

Figure 1. Temperature measurement system

It consists of the following:

- Five different thermocouple types (J, K, T, S and R).
- Thermocouple signal conditioner 5B37 module, as shown in Fig. 2.
- General purpose interface bus GPIB.

Figure 2 Thermocouple signal conditioner 5B37 module [11].

To determine the output voltage from a 5B37 module, three parameters must be known: the thermocouple input voltage at measurement temperature, the thermocouple input signal at the minimum point of the 5B37 module temperature range and the 5B37 (G). So it can be used the following equation (1) [11].

$$V_{out} = (V_{TC} - V_{Zero}) \times G \tag{1}$$

where:

V_{out} : is the 5B37 type (J, K, T, S and R) module output from (0 to 5Volt).

V_{TC} : is the thermocouple output voltage in (mV) at the temperature being measured.

Vzero : is the thermocouple output voltage in (mV) at the minimum temperature span specified for the (5B37-J, 5B37-K, 5B37-T, 5B37-S and 5B37-R) are equal to (-4.632, -3.553, -3.378, 0 and 0) mVolt, respectively.

Gain (G) : is the throughput gain in (V/mV) of the (5B37-J, 5B37-K, 5B37-T, 5B37-S and 5B37-R) modules are equal to (0.105, 0.087, 0.206, 0.271 and 0.239) respectively.

The I/O interface card uses GPIB as shown in Fig. 3 and connects with Laptop computer.

To measure the thermocouple output voltage in (mV), it is used equation (2) in order to find the learning and testing data sets for the linearized neural model based on the polynomial form.

$$V_{TC} = \frac{V_{out}}{G} + V_{Zero} \tag{2}$$

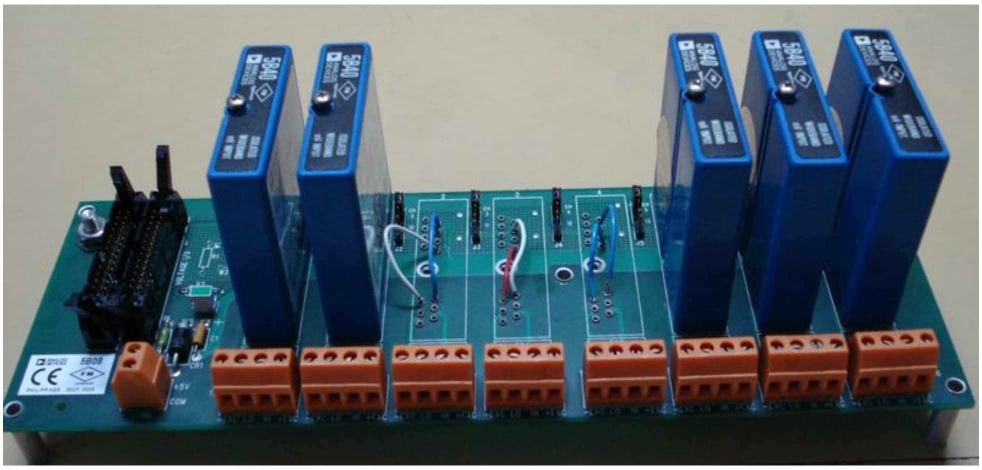

Figure 3 The GPIB card with thermocouple modules.

The polynomial is in the following form [6]:

$$T = a_0 + a_1 v + a_2 v^2 + \ldots + a_n v^n \qquad (3)$$

where

v : is the thermocouple voltage in volts.

T : is the temperature in degrees Celsius,

a_n : are coefficients for each thermocouple type.

The National Institute of Standards and Technology (NIST) polynomial coefficients for several popular thermocouple types over a selected range of temperature with errors are listed in Table 1 [6].

Table 1. NIST Polynomial Coefficients for Voltage-to-Temperature Conversion ($T = a_0 + a_1 v + a_2 v^2 + \ldots + a_n v^n$)

	Thermocouple Type					
	E	J	K	R	S	T
Range	0° to 1,000 °C	0° to 760 °C	0° to 500 °C	-50° to 250 °C	-50° to 250 °C	0° to 400 °C
a_0	0.0	0.0	0.0	0.0	0.0	0.0
a_1	1.7057035E -2	1.978425E -2	2.508355E -2	1.3891280 E-1	1.84949460E -1	2.592800E -2
a_2	-2.3301759E -7	-2.001204E -7	7.860106E -8	-9.3835290E -5	-8.00504062E -5	-7.602961E -7
a_3	6.5435585E -12	1.036969E -11	-2.503131E -10	1.3068619E -7	1.02237430E -7	4.637791E -11
a_4	7.3562749E 17	2.549687E 16	8.315270E 14	2.2703580E 10	1.52248592E 10	2.165394E 15
a_5	-1.7896001E -21	3.585153E -21	- 228034E -17	3.5145659E -13	1.88821343E -13	6.048144E -20
a_6	8.4036165E -26	-5.344285E -26	9.804036E -22	-5.8953900E -16	-1.39085941E -16	-7.293422E -25
a_7	1.3735879E 30	5.099890E 31	4.413030E 26	2.8230471E 19	8.23027880E 20	
a_8	1.0629823E -35		1.057734E -30	-1.2607281E -22	-2.34181944E -23	
a_9	-3.2447087E -41		-1.052755E -35	3.1353611E -26	2.79786260E -27	
a_{10}				-3.3187769E -30		
Error	±0.02° C	±0.05° C	±0.05° C	±0.02° C	±0.02° C	±0.05° C

3. Cognitive Neural Network Linearization Topology

The cognitive neural linearization algorithm for the temperature measurement system can be described as the proposed five steps algorithm based on multi-layered neural networks, as shown in Fig. 4.

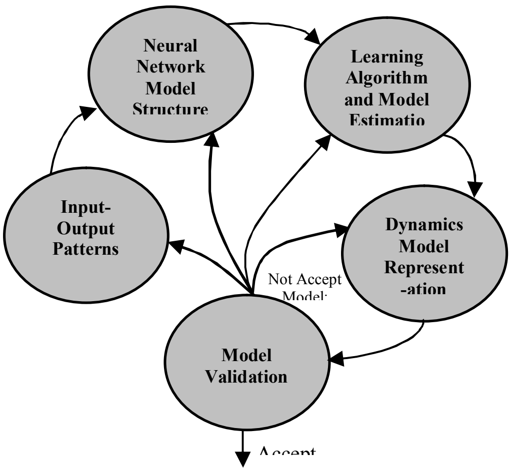

Figure 4 Five steps of cognitive neural Linearization algorithm.

3.1. The Input-Output Patterns

The training neural networks often required the existence of set of input and output patterns called the training set and this kind of learning is called supervised learning [12] and [13].

3.2. Neural Networks Model Structure

This section focuses on the neural linearization model using the multi-layer perceptron (MLP) neural network structure, as shown in Fig. 5, which consists from the nodes of input layer, hidden layer and output layer [14].

The network notations are as follows:

V_{an} : Weight matrix of the hidden layers.

$\overline{Vb}_a$: Weight vector of the hidden layers.

W_{ba} : Weight matrix of the output layer.

$\overline{Wb}_b$: Weight vector of the output layer.

To explain these calculations, consider the general a'th neuron in the hidden layer shown in Fig.5. The inputs to this neuron consist of an n– dimensional vector and (nth is the number of the input nodes). Each of these inputs has a weight V associated with it. The first calculation within the neuron consists of calculating the weighted sum net_a of the inputs as in (4) [14]:

$$net_a = \sum_{a=1}^{nh} V_{an} \times \overline{Z_n} + bias \times \overline{Vb_a} \tag{4}$$

where

nh : is number of the hidden nodes.

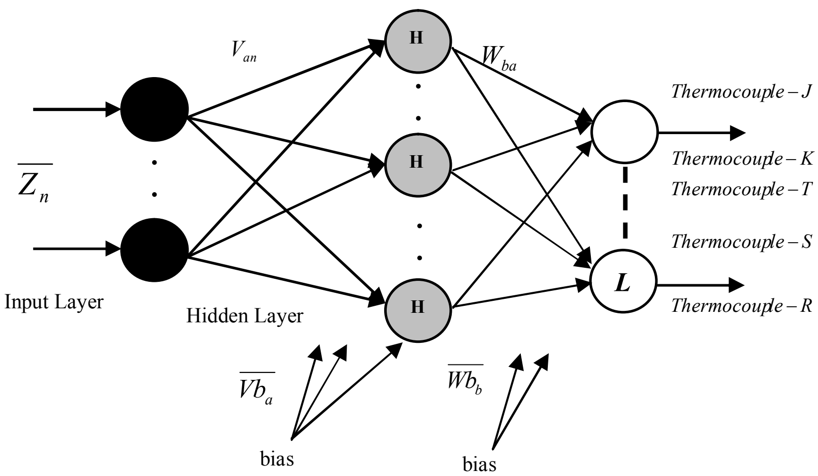

Figure 5 The multi-layer perceptron neural networks act as cognitive linearization model.

Next the output of the neuron h_a is calculated as the continuous sigmoid function of the net_a as in (5):

$$h_a = \mathrm{H}(net_a) \tag{5}$$

$$\mathrm{H}(net_a) = \frac{2}{1+e^{-net_a}} - 1 \tag{6}$$

Once the outputs of the hidden layer are calculated, they are passed to the output layer.

In the output layer, five linear neurons are used to calculate the weighted sum (net_o) of its inputs (the output of the hidden layer as indicated in (7)).

$$\mathrm{neto}_b = \sum_{a=1}^{nh} W_{ba} \times \overline{h_a} + bias \times \overline{Wb_b} \tag{7}$$

where

W_{ba} : is the weight between the hidden neuron h_a and the output neuron.

The five linear neurons, then, passes the sum (neto_b) through a linear function of slope 1 (another slope can be used to scale the output) as:

$$O_b = L(netc_b) \tag{8}$$

The outputs of the neural network linearization model represent the temperature of thermocouple types (J, K, T, S and R).

3.3. Learning Algorithm and Model Estimation

In this work, it is hoped to improve the convergence characteristics and the response accuracy by reducing the processing time of the learning algorithm thus, the SGA will be employed to learn the weights of the neural network linearization model. The SGA is an improved form of the classic GA, it has same evolutionary operators, i.e., crossover and mutation which are the most important parts responsible for the performance influencing. The main cores of SGA are dividing the population into slices and duplicating good individuals. The operation of dividing will lead to implementing the optimization in multi dimensions this will speed up the process of optimization while the process of duplication will give high opportunity to good individuals to exhibit all the best traits especially when applying random crossover to it [15] and [16].

In this work, it is satisfactory to take six slices thus; dimensions of each slice can be calculated as (9).

$$Din[n \times m] = \left[\frac{(\text{populatin size})}{6} \times (V_{an} + Vb_a + W_{ba} + Wb_b) \right] \tag{9}$$

The mean square error function is a criterion for estimating the linearization model performance for temperature measurement system as in (10).

$$J = \frac{1}{N}\sum_{k=1}^{N}[(Te_J)^2 + (Te_K)^2 + (Te_T)^2 + (Te_S)^2 + (Te_R)^2 \tag{10}$$

Since the SGA maximizes its fitness function, it is necessary to map the objective function (MSE) to the fitness function by using (11) [17].

$$fitness = \frac{1}{objectivefunction + \mu} \tag{11}$$

where μ is a constant > 0 chosen to avoid division by zero.

The learning steps of the neural linearization model parameters by using SGA can be described in detail as follows [15] and [16]:

Step 1: Initialize randomly six slices with dimension [10×181], i.e. the proposed population size (number of individuals) will equal to 60.

Step 2: Calculate the fitness of each individual in each slice.

Step 3: For each slice vertically find the global maximum fitness by using (11).

Step 4: Horizontally find the optimal solution for all slices “the first slices optimal individual will be got now.
Step 5: Duplicate the individual horizontally, which sponsors with horizontal maximum fitness.
Step 6: Make a selection as in Classic GA.
Step 7: Apply arithmetic random crossover with proposed crossover probability 0.85 (this will let the duplicated individuals produce their best).
Step 8: Apply mutation as in Classic GA with proposed mutation probability of 0.01.
Step 9: Calculate the fitness vertically then find the slices for global maximum fitness.
Step 10: Horizontally find the optimal solution for all slices.
Step 11: Find the optimal global by comparing step 11 to step 4.
Step 12: Compare individual’s fitness in the current generation, with the previous one, and then pick out the best individuals to create the new population.
Step 13: Repeat Steps (6 to 12) until the stopping criterion is satisfied**.**

3.4. Dynamics Model Representation

The dynamic model of the proposed cognitive neural linearization model for the temperature measurement system can be used for prediction in one-step configuration.

Prediction means that the neural networks model and the actual system model receive the same external inputs, but the output of the actual system feed to the neural network model in order to affect the dynamic behavior of the neural networks model by the actual system model and the model predicts one step into future. The one-step prediction configuration is called a series-parallel model is shown in Fig. 6 [13] and [14].

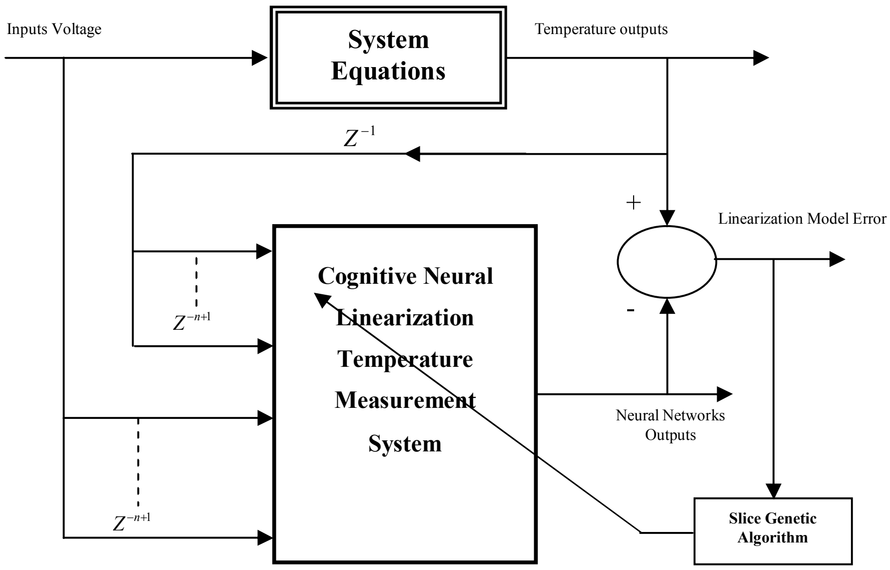

Figure 6 The series-parallel structure model.

3.5. Model Validation

The end task of linearization model for the temperature measurement system is the validation of the cognitive neural network model quality. The validation is to check the model quality by using another data set which called testing data set. The test data set should excite the system and the neural model. The validation process is performed using two different approaches; the first is the prediction error between the actual output of the system and neural network model output. The second validation can be achieved through visualization of the prediction. This visualization is given as graphic representation of the actual outputs and the predictions calculated by the neural network model.

4. Simulation Results and Laboratory Work

The cognitive neural linearization model which is proposed in section 3 for the temperature measurement system which is described in section 2 is verified by means of the LabVIEW package, the virtual instrument development platform by National Instrument. LabVIEW is a graphical programming language using icon code instead of text programming language therefore, it's simple, intuitive and the most widely applied in the measurement system [9] and [10].

The first step in the cognitive neural linearization model algorithm was attended the learning and testing thermocouple voltage sets from the measurement system, as shown in Fig. 1, for five thermocouples sensors types (J, K, T, S and R) with different temperatures within range 25 C^o to 250 C^o, as shown in Fig. 7 and 8 for learning and testing sets respectively. The inputs learning and testing patterns of the temperature are from (25 to 250)C^o and the output voltage of the thermocouples modules types (5B37-J, K, T, S and R) and the thermocouples outputs voltages types (J, K, T, S and R) are shown in Table 2. The second step in the cognitive neural linearization model algorithm was learned the multi-layer neural network which consists of the nodes of input layer, hidden layer and output layer as (10-11-5) respectively by using the learning steps of the slice genetic algorithm as the third step in the proposed algorithm which consists of six slices each slice has dimension $[10\times 181]$ with maximum population size is equal to 60 and for each slice is equal to ten and the learning pattern set is equal to 100 patterns with number of iteration is equal to 20.

Table 2. Output voltage of the thermocouples and modules.

Thermocouple Module Type	Output Voltage (volt) of the Thermocouple Module Type	Thermocouple Type	Output Voltage (mV) of the Thermocouples
5B37-J	0.1645 to 1.644	J	-3.067 to 11.017
5B37-K	0.0926 to 0.925	K	-2.483 to 7.139
5B37-T	0.3125 to 3.125	T	-1.862 to 11.777
5B37-S	0.0714 to 0.714	S	0.264 to 2.643
5B37-R	0.0714 to 0.714	R	0.298 to 2.983

Matlab programming uses to learn and test the cognitive neural linearization mode for temperature measurement system with using slice genetic algorithm in order to obtain better performance and faster convergence with the minimum number of fitness evaluation. It is very necessary to normalize the input signals of Table 2 and Fig. 7 and 8 between (-1 to +1). The signals entered to or emitted from the neural network have been normalized to lie within (-1 to +1) in order to overcome numerical problems that is involved within real values.

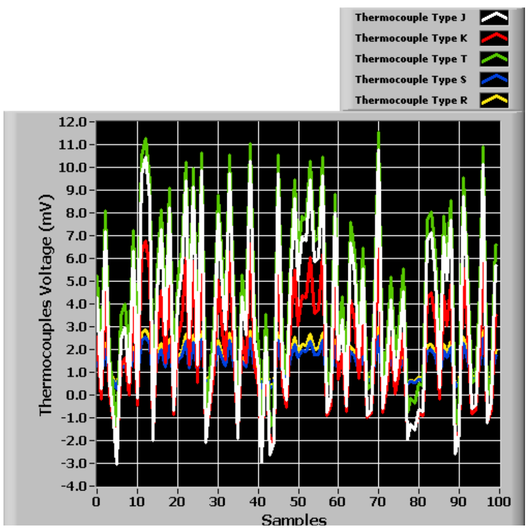

Figure 7 Learning set thermocouple voltage.

.

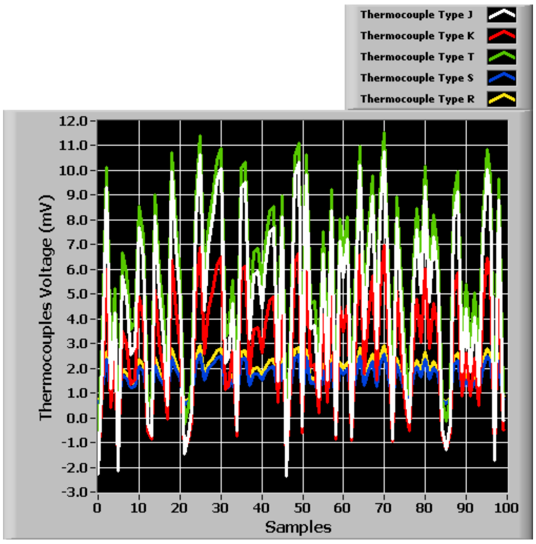

Figure 8 Testing set thermocouple voltage.

Therefore, scaling functions have to be added at the cognitive neural linearization model terminals to convert the scaled values to actual values and vice versa in order to achieve the dynamics model representation of the cognitive neural linearization model as fourth step in the proposed algorithm. After 20 iterations for learning the cognitive neural linearization model, the mean square error is equal to 0.00215 as shown in Fig. 9. Then it can be observed that the output temperature of the cognitive neural linearization model is follow the actual temperature measurement for each thermocouple, type as shown in Fig. 10.

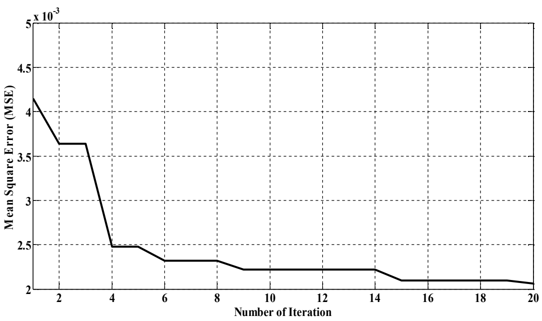

Figure 9 The mean square error for the learning set.

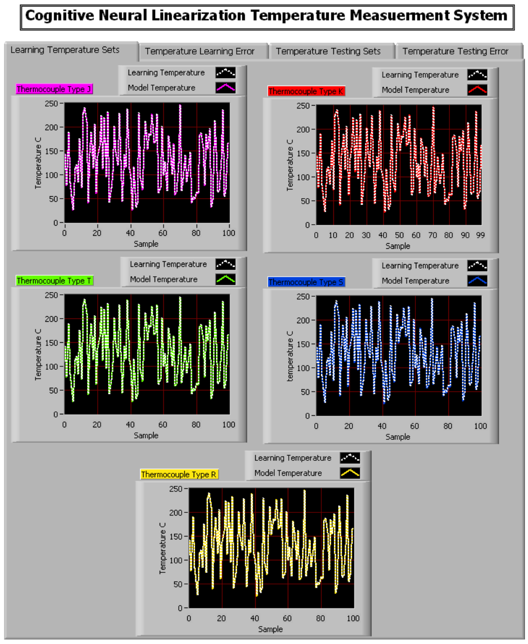

Figure 10 Model temperature output for learning set thermocouple voltage.

Figure 11 shows the error between the desired temperature and the output temperature of the cognitive neural linearization model for learning set for each thermocouple type and the maximum temperature error range is ±3 C^{o} from full scale at (0 to 250C^{o}) therefore the error in not clear in the Fig. 10. The fifth step in proposed of the cognitive neural linearization model algorithm was tested the multi-layer neural network by the testing set in order to verify the model and clear from each of learning problems. It can be indicated that the output temperature of the cognitive neural linearization model of the temperature measurement system is closely to the desired actual temperature measurement, as shown in Fig. 12 for each thermocouple type.

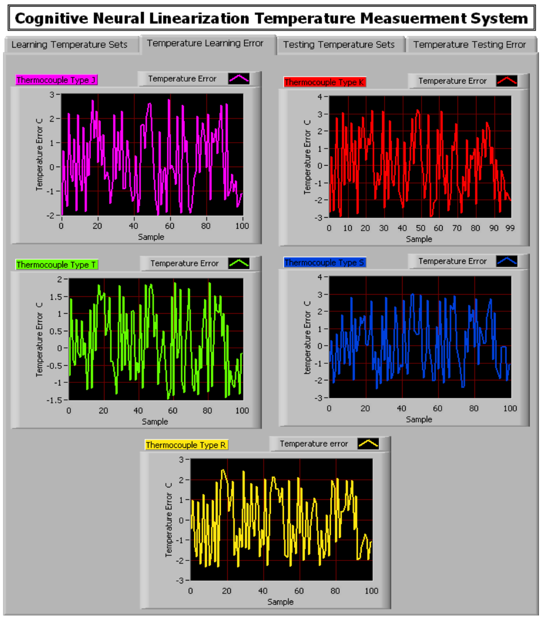

Figure 11 Temperature model error for learning set thermocouple voltage.

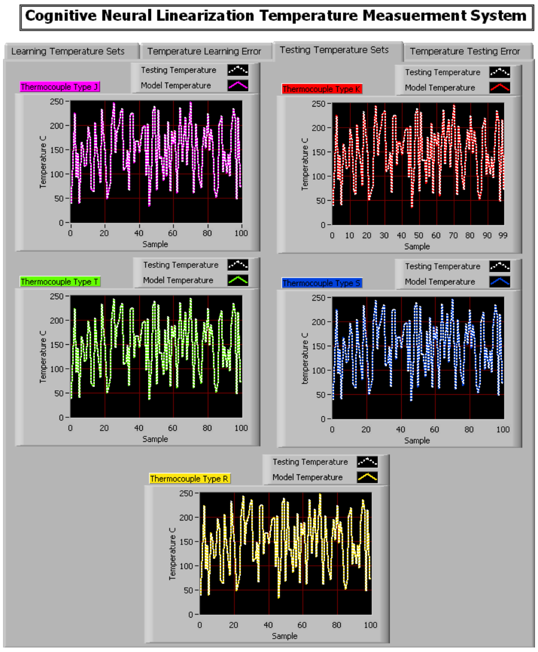

Figure 12 Model temperature output for testing set thermocouple voltage.

Figure 13 shows the error between the actual temperature and the output temperature of the cognitive neural linearization model for testing set for each thermocouple type and the maximum temperature error range is $\pm 3\,C^{o}$ from full scale at (0 to 250C^{o}) therefore the error in not clear in the Fig. 12.

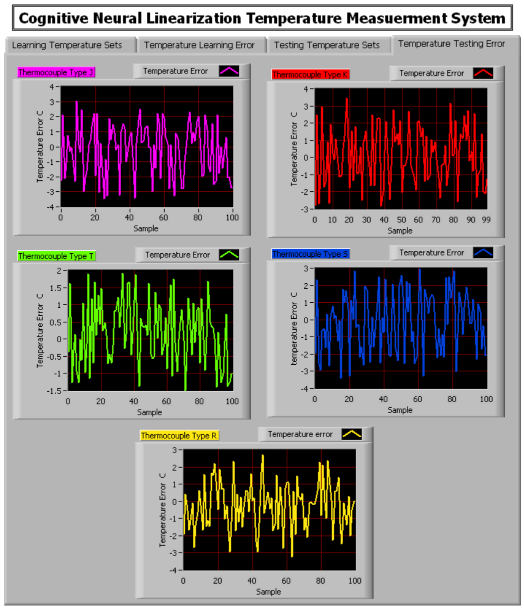

Figure 13 Temperature model error for testing set thermocouple voltage.

5. Conclusions

The cognitive neural linearization model for the temperature measurement system based on the proposed five steps algorithm with slice genetic technique for learning the multi-layer perceptron neural network has been presented in this work. Laboratory work via LabVIEW package and simulation results via Matlab package demonstrate the cognitive neural network approach acting as a preciously linearization model for the five different types of the thermocouples which have nonlinear behavior. The proposed learning algorithm for the cognitive neural linearization model has the capability as follows:

Fast and stable tuning weights of the neural network with a minimum number of fitness evaluations as compared with [5] and [6]; Reducing the output oscillation compare with [5]; Minimizing the error between the actual temperature output and neural linearization temperature output compare with [5]; Overcoming the problem of the over learning in the linearization model.

References

[1] R.P. Areny and J.G. Webster, Sensors and Signal Conditioning. Wiley, New York, 2001.

[2] L. Michalski, K. Eckersdorf, J. Kucharski and J. Mcghee. Temperature Measurement. Wiley Interscience, 2001.

[3] K. Danisman, I. Dalkiran and F.V. Celebi. Design of a high precision temperature measurement system based on artificial neural network for different thermocouple types. Measurement, vol. 39 (2006), pp. 695–700.

[4] Y. Chong, C. Quek and P. Loh A novel neuro-cognitive approach to modeling traffic control and flow based on fuzzy neural techniques. Expert Systems with Applications, vol. 36, (2009) pp. 4788–4803.

[5] N. J. Medrano, R. H. Alonso and B. M. Brio. A Thermocouple Model Based on Neural Networks**.** Proceedings of the 6th International Work-Conference on Artificial and Natural Neural Networks: Bio-inspired Applications of Connectionism-Part II. (2001), pp. 531-538.

[6] A. Al-Araji. Design of a Neural Networks Linearization for Temperature Measurement System Based on Different Thermocouples Sensors Types. Eng. and Tech. Journal. Vol. 27, no. 8, (2009), pp. 1622-1639.

[7] A. Al-Araji, Design of a cognitive neural predictive controller for mobile robot, PhD thesis, Brunel University, United Kingdom, (2012).

[8] Y. Lei, Z. Xu, L. Zhang and L. Meng. An environmentally-flexible measurement system for ammunition temperature detection in the powder chamber with high accuracy. Measurement, vol. 45, (2012), pp. 2021–2028.

[9] J. Song. Air condition Control system of Weaving Workshop Based On LabVIEW. Physics Procedia, vol. 24, (2012), pp. 541-545.

[10] Z. Minglea, Y. Jintiana, J. Guoguanga and L. Ganga. System on Temperature Control of Hollow Fiber Spinning Machine Based on LabVIEW. Procedia Engineering, vol. 29 (2012), pp. 558-562.

[11] Text manual from analog device company. www.analog.com/static/imported-files/data_sheets/5B37.pdf . Accessed 1 Aug. 2014

[12] Omatu, M. Khalid, and R. Yusof, Neuro - control and its applications. London: Springer-Velag, 1995.

[13] Nells, "Nonlinear system identification, Springer – Verlag Berlin Heidelberg, 2001.

[14] M. Zurada, Introduction to Artificial Neural Systems (Jaico Publishing House, Pws Pub Co. 1992**).**

[15] K. E. Dagher, Design of an auto-tuning PID controller for systems based on slice genetic algorithm. Iraqi Journal of Computer, Communication, Control Systems Eng. 13(3), 1-9 (2013).

[16] A. Al-Araji, Development of kinematic path-tracking controller design for real mobile robot via back-stepping slice genetic robust algorithm technique. Arabian Journal for Science and Engineering. Vol. 39, no. 12, (2014), pp. 8825-8835.

[17] A. Al-Araji, Genetic algorithm and Elman neural network used for tuning the parameters of the PID neural controller based model reference. Iraqi Journal of Computer, Communication, Control Systems Eng. vol. 5, no.1, (2005), pp. 111-126.