

The compensation of the thermocouples cold junction temperature variation errors using a neuro-fuzzy controller

Teodor Lucian Grigorie, Mihai Lungu, Petre Negrea, Dalia Baziukaitė

Abstract— The paper presents a new method to compensate errors due to temperature variation at cold junction of the thermocouples. The presented method is based on the realization of a neuro-fuzzy controller in order to get a quick error compensation model based on information related to the cold junction temperature variations. The model uses the experimentally obtained numerical values and has the advantage of fuzzy logic to process empirical signals, without using analytical mathematical models.

Keywords — Cold junction temperature variation, Error compensation, Neuro-fuzzy controller, Thermocouple.

I. INTRODUCTION

THE temperature's measurement using thermocouples are based on thermoelectric phenomenon discovered in 1821 by Seebeck [1], [2], [3]. The use of this phenomenon to measure temperature is based on the existence of certain relationships between thermo-electromotive forces (f.t.e.m.), which are established in a circuit composed of several different conductors, and the temperatures in the connection points of these conductors. If a circuit, composed by two different conductors, A and B, is considered (Fig. 1), at the heating of the junction 1 an electrical current appears in this circuit. This electrical current is called thermo-electrical, and devices that produce it are called thermocouples or thermo-elements. Electromotive forces, which arise due to different potentials of the junctions under different temperatures, are called thermo-electromotive forces [1], [2], [3].

The basic equation of the thermocouple can be written as a general equation expressing the variation of thermo-electromotive force, occurring in a circuit composed of two

different conductors (Fig. 1), as a function of the junctions' temperatures [2], [3]

$$E_{AB}(t, t_0) = e_{AB}(t) + e_{BA}(t_0). \quad (1)$$

In equation (1) the E_{AB} is a resultant electromotive force determined by the total effect, e - electromotive force subject to the temperature difference at the ends of conductors A and B; with indices A and B notes conductors between the resultant electromotive force is generated; index sequence shows how the force should be considered when changing from one electric conductor to another. It is clear that reversing the order indices one will have to change the sign before the second term

$$E_{AB}(t, t_0) = e_{AB}(t) - e_{AB}(t_0), \quad (2)$$

i.e. f.t.e.m. appearing in the circuit composed of two different conductors, whose junction points have different temperatures, is the difference between the resultant thermo-electromotive forces.

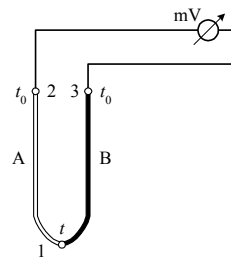


Fig. 1 Thermocouple device

Noting the electromotive force e in equation (2) with $f(t)$, one can write the general formula for the variation of the f.t.e.m. as a function of the temperatures t and t_0 of the thermocouple junctions

$$E_{AB}(t, t_0) = f(t) - f(t_0). \quad (3)$$

Assuming the temperature of one junction of the thermocouple as a constant, i.e. accepting $t_0=ct.$, and having a relationship $f(t_0)=C$, one gets

Teodor Lucian Grigorie is with the University of Craiova, Faculty of Electrical Engineering, Avionics Division, 107 Decebal Street, 200440 Craiova, Dolj, Romania, phone: +40-251-436447; fax: +40-251-436447; e-mail: lgrigorie@elth.ucv.ro, lgrigorie@yahoo.com.

Mihai Lungu is with the University of Craiova, Faculty of Electrical Engineering, Avionics Division, 107 Decebal Street, 200440 Craiova, Dolj, Romania, e-mail: Lma1312@yahoo.com.

Petre Negrea is with the University of Craiova, Faculty of Electrical Engineering, Avionics Division, 107 Decebal Street, 200440 Craiova, Dolj, Romania, e-mail: negreapetre@yahoo.com

Dalia Baziukaitė is with the Klaipėda University, Faculty of Science and Mathematics, Department of Computer Science, Herkaus Manto str. 84, LT-92294 Lithuania, email: dalia.baziukaite@ku.lt.

$$E_{AB}(t, t_0)_{t_0=const} = f(t) - C. \quad (4)$$

If the relationship expressed by equation (4) is known from a curve, a table or an experimentally determined equation, the measurement of the unknown temperature t will be reduced to the measurement of the $E_{AB}(t, t_0)$ f.t.e.m. It is assumed here that the temperature t_0 remains constant. Through the temperature t_0 deviation from the considered constant value, equation (4) loses its validity because the variation of this temperature implies the changes of the constant C [2], [3].

F.t.e.m. appearing in a thermocouple is not larger (0.01-0.06 mV/degree); however it can be measured with sufficient accuracy with a millivoltmeter or a pyrometric potentiometer. Any meter used in a thermoelectric assembly is calibrated against the reference temperature to which must be the cold junction. This temperature is indicated by the initial benchmark of the stairs (0°C, 20°C or 50°C) [1]-[3].

To perform measurement in a right way, cold junction temperature must be maintained at a constant value (the value considered to the calibration), or the measured value must be determined based on calculations by adjusting the zero position of the indicator device or by using the compensation devices, whether or not there is an extension wire.

Two approaches (methods) are commonly used for cold junction compensation [4].

The first method consists in the electronically simulation, with specialized integrated circuits, of the potential effects that result for a thermocouple wire pair between the terminals, at their measured temperature, and another junction at maintained at a reference temperature of 0 degrees. Through the measurement of the potential across the thermocouple wire pair in series with the simulated potential, followed by the linearization curve of the sum, an estimated absolute temperature will be obtained. Generally, the method is affected by two errors: cold junction temperature estimation error, and the approximation of the junction potential effects. The calibration with this method is difficult and, in the most cases, limited to the offset adjustment.

A second method supposes the independently measurement of the cold junction temperature, like in our application. In this case, the cold junction compensation can be performed by computation using look-up tables and polynomial interpolation starting from the experimentally obtained data.

Let t be the temperature of the warm junction of the thermometer and t_0 the cold junction temperature for which was calibrated the thermocouple. Suppose that cold junction temperature has suffered a change, becoming equal with t'_0 . Thus, f.t.e.m. $E_{AB}(t, t'_0)$ of the thermocouple will be different from the value determined during calibration. Results

$$E_{AB}(t, t_0) = e_{AB}(t) - e_{AB}(t_0), \quad (5)$$

$$E_{AB}(t, t'_0) = e_{AB}(t) - e_{AB}(t'_0). \quad (6)$$

So, the correction to be made is

$$E_{AB}(t, t_0) - E_{AB}(t, t'_0) = e_{AB}(t) - e_{AB}(t_0) - e_{AB}(t) + e_{AB}(t'_0), \quad (7)$$

$$E_{AB}(t, t_0) - E_{AB}(t, t'_0) = e_{AB}(t'_0) - e_{AB}(t_0). \quad (8)$$

In equation (8) the right side is equal with $E_{AB}(t'_0, t_0)$, and one obtain

$$E_{AB}(t'_0, t_0) = E_{AB}(t, t_0) - E_{AB}(t, t'_0). \quad (9)$$

So, the correction is $E_{AB}(t'_0, t_0)$, and this can be read on the thermocouple characteristic built maintaining the cold junction temperature at the value t_0 .

The purpose of the proposed controller is to get a quick error compensation model based on information related to cold junction temperature variation. The model uses numerical values experimentally obtained and has advantage of fuzzy logic to process empirical signals, without using analytical mathematical models. Fuzzy logic can emulate the decisions of human factor expert in these systems, linguistically formulated in terms of fuzzy rules in the IF-THEN form. Fuzzy logic approach can be successfully applied in complex systems, where simplified mathematical models do not exist, in highly nonlinear systems, and also in multidimensional systems [5], [6], [7].

In a fuzzy control system the input variables are usually mapped by a set of membership functions (mf) known as "fuzzy set". The mapping process is called "fuzzification". Control system makes decisions based on a set of fuzzy rules, which are invoked using the membership functions and logical values obtained from the inputs. The process is called "inference". Decisions are mapped into membership functions and truth values that control the output variables. The results are combined to give a specific response by a procedure called "defuzzification". Therefore, the development of such a model requires a set of fuzzy rules and membership functions associated with each of the inputs [5]-[7].

The crucial aspect to get a good fuzzy model is ability and experience of the designer correctly assess rules and membership functions for all inputs. Recent design method allows building of a good model using a combination of fuzzy logic and neural-network. This method offers the possibility to generate and optimize the set of fuzzy rules and membership functions parameters by training of the fuzzy inference system (FIS). The method used for numerical experimentally obtained data processing is a hybrid method, which combines a back-propagation algorithm and a least squares method. It is implemented in Matlab software already. The method is easy to use and gives good results in a very short time [5], [6], [7].

II. CONTROL IMPLEMENTATION

For the proposed study a chrome-alloy thermocouple was chosen. The values obtained from the output of the thermo-

couple can be extracted from experimental data. For this the temperature of cold junction should keep the constant value of 20 °C and the cold junction temperature is changed in the range of 50 °C÷100 °C. The disturbed values of the thermocouple output for measured temperatures in the range of 50 °C÷110 °C are calculated based on these values. Fig. 2 shows dependencies of the thermocouple output disturbed values to a real temperature and the cold junction temperature. The designed controller will provide the real value of the measured temperature t , and will use as inputs the perturbed f.t.e.m. $E_c = E_{AB}(t, t'_0)$, which is obtained directly from the thermocouple, and the real cold junction temperature t'_0 provided by an additional thermocouple [2].

Conceptually, fuzzy controllers are quite simple and use fuzzy inference systems (FIS). To build a fuzzy controller input, processing and output phases need to be accomplished. First, inputs are mapped to membership functions (mf), and a collection of logical "IF-THEN" rules is created. Second, every rule needs to be invoked for a result generation. And last, results of all rules need to be combined and translated into a specific controller output value [6].

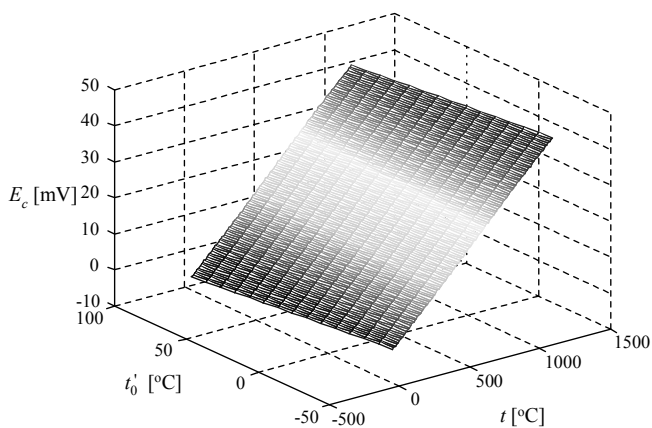


Fig. 2 The dependence of the disturbed thermocouple output by the real temperature and by the cold junction temperature

Considering the experimental numerical values, mentioned above, one can develop an empirical model based on a neuro-fuzzy network. The model can learn the behavior of the process by the use of a Fuzzy Inference System (FIS).

FIS can be easily generated with Matlab functions "genfis1" or "genfis2". The function "genfis1" generates a Sugeno type FIS with one output using the method of grid partitioning (not clustering). This FIS is used to provide initial conditions for training with ANFIS Matlab function. The function "genfis1" uses membership functions of generalized Bell-type for each input. Each rule generated by "genfis1" has an mf of linear type, with a single output. Another way to create the FIS is by using Matlab function "genfis2". This generates a Sugeno type FIS by operating domain decomposition in different regions using subtractive clustering method. For each region a linear model can describe the local process parameters. Thus, the non-linear process is locally linearized around an operating

point using the least squares method. The obtained model is valid in the surrounding region of the operating point. Fuzzy sets of membership functions are associated with each input of the model. The input domain is divided into fuzzy regions by combining fuzzy inputs. A linear model is used for each of these regions, while the overall model is obtained by using center of gravity defuzzification (Sugeno) method, which performs interpolation of local model outputs [5], [6], [7].

Parameters of Sugeno type FIS can be calculated using neural networks that are being trained with the Matlab function "ANFIS". It uses a learning algorithm to identify parameters of membership functions of the FIS with two inputs and one output. As a starting point the input-output experimental data and FIS model, generated with "genfis1" or "genfis2" functions, are used. The "ANFIS" function optimizes the parameters of membership functions for a number of training epochs determined by the user. As a consequence of this optimization the neuro-fuzzy model can reproduce better the process; the quality of reproduction is controlled using a parameter resident in the training algorithm. After training the obtained model can be used to generate the real measured temperature corresponding to the input data [6].

To train the fuzzy system, the ANFIS function uses a back-propagation algorithm for parameters associated with membership functions of inputs, and an estimation with the least squares method for the parameters associated with the membership functions of output. For the FISs generated using the function "genfis1" membership functions are generalized Bell-type, and for "genfis2" function by the Gaussian type [5].

III. CONTROLLER DESIGN AND EVALUATION THROUGH NUMERICAL SIMULATION

To generate the FIS necessary for the controller development, the Matlab function "genfis2" was used. The best training results of FIS were obtained for 100.000 epochs of training. A set of 9 rules was computed. They have a structure like: if (in1 is in1cluster,"k") and (in2 is in2cluster,"k") then (out1 is out1cluster,"k"). 9 Gaussian type membership functions for each of the two inputs were generated as well. The notation for them is as follows: in,_j"cluster,"k"; j is the number of the input (1÷2), and k is the number of the membership function (1÷9). The FIS has the structure presented in Fig. 3, while the associated controller has the structure like in Fig. 4.

To view the FIS properties, the Matlab command "anfisedit" is used. The FIS is imported as well at the interface level. The interface allows to view the following characteristics of FIS: membership functions of the first input (perturbed f.t.e.m. $E_c = E_{AB}(t, t'_0)$), membership functions of the second input (cold junction temperature), defuzzification rules, and the surface generated for the model, starting from the two inputs.

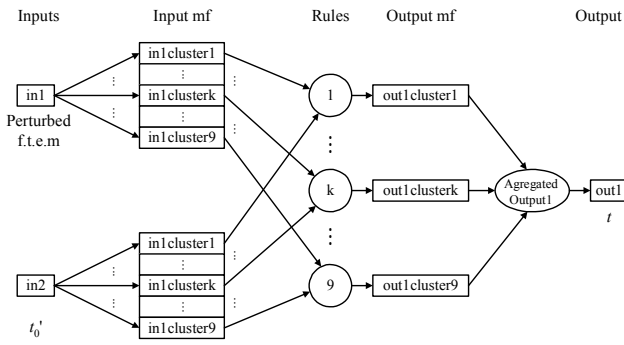


Fig. 3 FIS structure

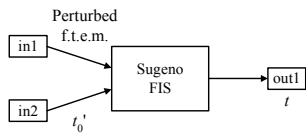


Fig. 4 Controller structure

The parameters of the inputs' membership functions before FIS training are presented in Table I. For the Gaussian type membership functions the parameters offered by Matlab are half of the dispersion $\sigma/2$ and the peak.

By calling the function "ANFIS" is triggered a neuro-fuzzy type adaptive algorithm, which identifies and modifies the parameters of the previously generated FIS membership functions. If the learning algorithm is run for 10, 100, 5.000,

10.000, 30.000 or 100.000 training epochs, then deviation between the neuro-fuzzy model and experimentally determined data, for the purposes of resident quality parameter in training algorithm, is presented in Fig. 5. The after training FIS evaluation was done with command "evalfis"; in Fig. 6 are presented the experimental data (continuous line contours) and the output of the FIS model (drawn with dashed curves) after 100.000 training epochs.

From Fig. 5 one observes a rapid decrease in deviation between the experimental data and neuro-fuzzy model, in the sense of quality parameter resident in the training algorithm, for the first 200 training epochs, followed by a slower decrease in the next 1300 training epochs, by another quick decrease between 1500÷1700 training epochs, and finally by a very slow decrease period between 1700÷100.000 training epochs range.

According to the graphs, it is sufficient that the FIS model to be trained over 100.000 training epochs, because the deviation get into an approximately constant zone with the value of $0.12 \cdot 10^{-4}$. Fig. 6 fully reflects the observations made for Fig. 5; to note the juxtaposition of the FIS model (evaluated for input data) with the experimental data even better with higher number of training epochs. Training model for more than 100.000 epochs produces a very small decrease in value of the deviation; a closer approximation of the real model by neuro-fuzzy model is accomplished if enlarged set of experimental data, describing the evolution of error, is used.

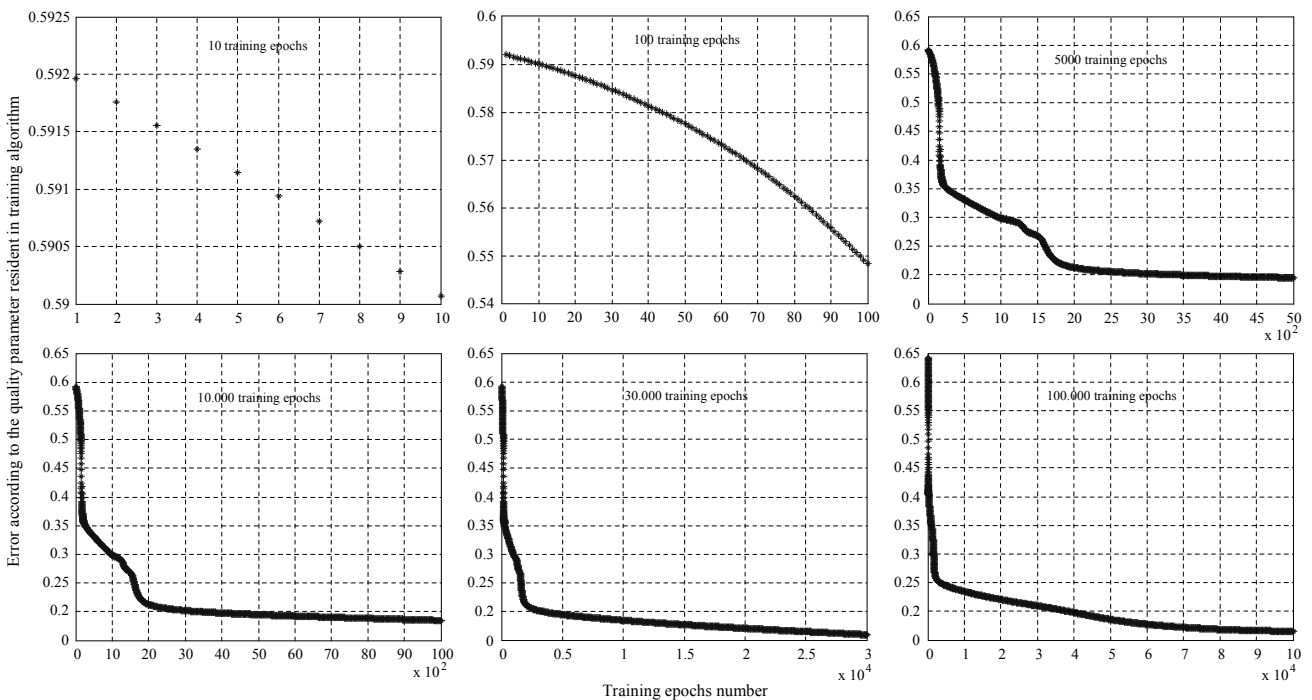


Fig. 5 Deviation between the neuro-fuzzy model and the experimental data, as a function of training epochs

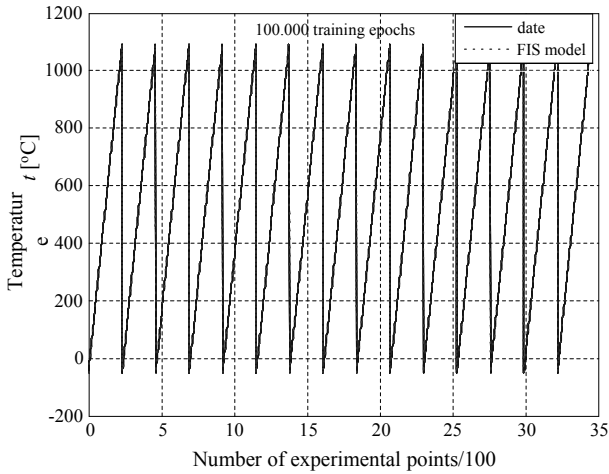


Fig. 6 Evaluation of the trained FIS

TABLE I
PARAMETERS OF THE INPUTS' MEMBERSHIP FUNCTIONS BEFORE TRAINING

	Perturbed f.t.e.m. [mV]		Cold junction temperature [°C]		
	$\sigma/2$	Peak	$\sigma/2$	Peak	
mf1	9.224	15.14	mf1	24.75	20
mf2	9.224	34.93	mf2	24.75	0
mf3	9.224	30.44	mf3	24.75	70
mf4	9.224	3.29	mf4	24.75	60
mf5	9.224	8.07	mf5	24.75	-30
mf6	9.224	25.08	mf6	24.75	-40
mf7	9.224	16.05	mf7	24.75	80
mf8	9.224	43.62	mf8	24.75	-40
mf9	9.224	42.8	mf9	24.75	50

Trained model properties presented in Fig. 7 are: membership function of the first input, membership function of

the second input, defuzzification rules, and the surface generated for the model starting from the two inputs.

The parameters of the inputs' membership functions after FIS training over 100.000 training epochs are presented in Table II.

A comparative analysis of the FIS characteristics and parameters of membership functions before and after training has shown that there is a redistribution of membership functions in the domain and their shapes change while dispersion is modified. As we said earlier, for the Gaussian type membership functions the parameters are the dispersion and the peak. According to the parameters numerical values in Table I, the FIS generation with the Matlab function "genfis2" from the experimental data gives the same dispersion value for all membership functions of an input. Also, for each input of the FIS, different values of the Gaussian functions peaks are automatically generated, in order to approximate the general model with local linear models using least squares method.

To optimize membership functions parameters (the dispersion and the peak), the training of the FIS model with "ANFIS" function with hybrid algorithm was used. As a consequence of the training, the surface showing the dependence between the FIS's inputs and output (Fig. 7), better approaches to the surface showing the dependence between the same parameters acquired from the experimental data set (Fig. 2).

Evaluation of the controller for experimental data as inputs, after different training epochs, lead to deviations presented in Table III between the neuro-fuzzy model and experimental data; deviations reflect the corrected temperature. Analysis of data in Table III shows a positive trend of the deviations, in order to reduce their absolute values, both positive and negative peak deviations, and the absolute mean deviations.

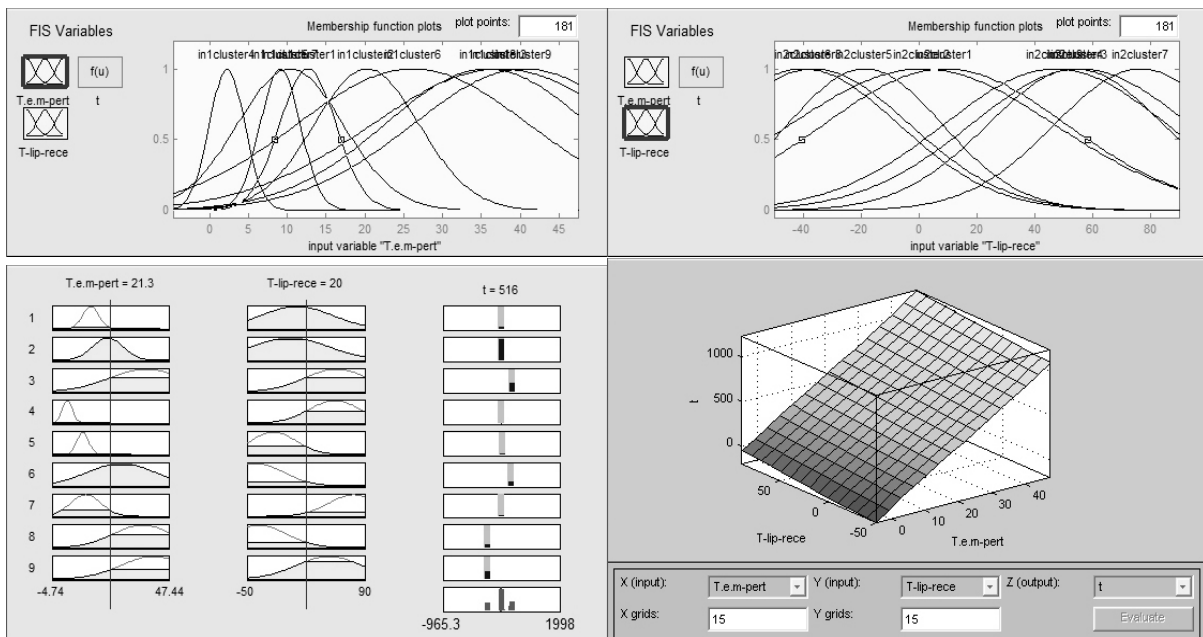


Fig. 7 Properties of the trained model

TABLE II
PARAMETERS OF THE INPUTS' MEMBERSHIP FUNCTIONS AFTER TRAINING

	Perturbed f.t.e.m. [mV]		Cold junction temperature [°C]		
	$\sigma/2$	Peak	$\sigma/2$	Peak	
mf1	3.636	12.62	mf1	41.96	9.111
mf2	6.786	20.21	mf2	45.62	1.25
mf3	16.47	37.37	mf3	37.9	55.91
mf4	2.292	222	mf4	30.45	54.27
mf5	2.594	9.059	mf5	26.87	-18.61
mf6	14.99	26.15	mf6	33.25	-40.05
mf7	6.83	10.35	mf7	29.82	76.74
mf8	13.22	35.72	mf8	33.29	-36.34
mf9	14.42	40.52	mf9	38.61	47.3

TABLE III
DEVIATIONS BETWEEN FUZZY CONTROLLER OUTPUT AND EXPERIMENTAL DATA REFLECTING CORRECTED VALUES OF THE TEMPERATURE

Training epochs	Absolute mean deviation [°C]	Maximum positive deviation [°C]	Maximum negative deviation [°C]
10	0.44187	1.8936	-2.8235
100	0.3892	1.5008	-2.7416
5000	0.1363	0.8658	-1.5820
10.000	0.1310	0.8161	-1.4879
15.000	0.1273	0.7675	-1.4065
30.000	0.1185	0.6600	-1.1630
100.000	0.0922	0.3243	-0.5017

The values obtained after 10 training epochs were 0.44187 °C for the absolute mean deviation, 1.8936 °C for the maximum positive deviation, and -2.8235 °C for the maximum

negative deviation. After 100.000 training epochs the deviation were seriously reduced, the acquired values being 0.0922 °C for the absolute mean deviation, 0.3243 °C for the maximum positive deviation, and -0.5017 °C for the maximum negative deviation. So, using this method the indication error for the temperature is maximum 0.5017 °C for a chrome-alloy thermocouple calibrated at 20 °C that measure temperatures in the range -50 °C ÷ 1100 °C, and is affected by a cold junction temperature variation in the range -50 °C ÷ 100 °C. For the previous conditions, if no compensation method is used, the maximum indication error is 79 °C.

REFERENCES

- [1] I. Aron, *Aparate de bord pentru aeronave*. Editura Tehnica, Bucuresti, 1984
- [2] P. Vezeanu, S. Patrascu, *Masurarea temperaturii in tehnica*. Ed. Tehnică, București, 1968
- [3] R. Lungu, T.L. Grigorie, "A calculation method of the compensation bridge of cold junctions temperature variation at the thermocouples", in *Proceedings of the XIIth International Symposium on Electrical Apparatus and Technologies SIELA 2001*, Plovdiv, Bulgaria, 31 may - 1 june, 2001, pp. 188-196.
- [4] M. Scervini, *Thermocouple Operating Principle*, University Of Cambridge, 31 August, 2009
- [5] *** Matlab - Fuzzy Logic and Neural Network Toolboxes - Help
- [6] T.L. Grigorie, *Sisteme de navigatie inertiala strap-down*. Studii de optimizare. Editura Sitech, Craiova, 2007
- [7] T.L. Grigorie, R. Botez, "The bias temperature dependence estimation and compensation for an accelerometer by use of the neuro-fuzzy techniques", *Transactions of the Canadian Society of the Mechanical Engineering (CSME)*, Vol. 32(3), 2008, pp. 383-400