Neural Network, Component of Measuring Set for Error Reduction

J. Vojtko

Department of Theory of Electrical Engineering and Measurement,
Faculty of Electrical Engineering and Informatics, Technical University of Košice,
Park Komenského 3, 042 00, Slovak Republic
E-mail: Jozef.Vojtko@tuke.sk

Abstract. *The paper describes the use of neural network as a component of measuring set for error reduction (nonlinearity). The scope of interest is elastomagnetic sensor used for measuring of massive pressure force (of range about 200 kN). Nonlinearity and hysteresis errors are main limitations of elastomagnetic sensor use. The hysteresis causes ambiguity of transfer characteristic and with it related impossibility of exact conversion of output sensor voltage into measured force. The function of the neural network is to regulate the basic metrological characteristics of sensor in order to achieve the smallest deviation from an ideal transfer characteristic. The assumption of the use of neural network as data-conditioning block is non-linear dependency of sensor output from input quantity.*

Keywords: Measurement, sensor errors, error elimination, neural network

1. Introduction

The requirements of measuring set are mainly accuracy and reliability, where sensor is determining element of accuracy. The purpose of the paper is to demonstrate the use of elastomagnetic sensor for force measurement and to present neural networks as a data-conditioning block. The principle of elastomagnetic sensor is based on existence of elastomagnetic (Villari) effect. It appears in ferromagnetic material as follows - if external force affects ferromagnetic material, it is deformed. In consequence with this deformation, relative distances of atoms in crystal structure are changed. It causes a value change of exchange forces, which activates spontaneous magnetization in domains of ferromagnetic material. This fact occurs as a magnetic polarization change or magnetic induction at the identical intensity of magnetic field acting on ferromagnetic material. If this material was isotropic before acting force, it becomes anisotropic. If it was anisotropic, the anisotropy of material is changed. The magnetic properties are represented by permeability and it is changed in accordance with acting force [8].

We have been testing the elastomagnetic sensors of pressure force of range from 200 kN to 12000 kN. In this paper we are dealing with elastomagnetic sensor EMS-200kN (Fig. 1). It is connected like a transformer (see in Fig. 2). The sensor core is made of 105 sheets, the weight of sensor core is 560 g and size is 56x56x23 mm. The sheets are glued on each other and screwed together. The number of primary and secondary turns is 5 and these parallel windings are placed in four holes. Detailed description of the sensor is in [8], [9]. The inaccuracy of the relation for output sensor voltage in dependence on input force is caused by some simplifications (substitution of non-homogeneous magnetic field by homogeneous circle magnetic field, substitution of non-homogeneous mechanical tension field by the homogeneous one, ignoring mechanical tension concentration around winding hole, ignoring ferromagnetic plates contacts, which transmit pressure to sensor, ignoring defects of supply power (I_1 is not constant, it is changed by changing of sensor impedance), small differences in values B, μ (different places in ferromagnetic rolled-section).

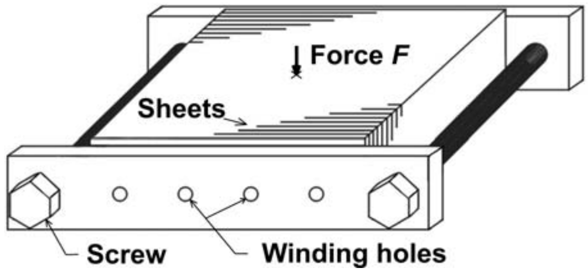


Fig. 1 Elastomagnetic sensor EMS-200kN

The main advantages of elastomagnetic sensors are high sensitivity (depending on sensor core material), sufficient output voltage and output power, very high reliability, mechanical toughness, ability of multiple force overloading, time-invariant properties (in comparison with strain gauges), and relatively simple construction. One of the sensor shortcomings is difficulty to obtain the formula of exact conversion of output sensor voltage into measured force. Other shortcomings of elastomagnetic sensors are power consumption, ambiguity of transfer characteristic and sensor errors, e.g. non-linearity and hysteresis. At present, the requirements for accuracy and reliability of sensor measuring systems are getting higher. Measuring of massive pressure forces via elastomagnetic sensor is more progressive and reliable than measuring via strain gauges.

2. Neural network in measuring set

There are certain problems consisting of spatial and also temporal structures in the field of electrical engineering. The typical example of this is elastomagnetic sensor of massive pressure force. Output sensor voltage does not depend only on pressure force, but also on previous force loading. The transfer characteristic of such sensor is nonlinear and depends on many parameters. In addition, it has hysteresis behaviour. These matters cause impossibility of exact conversion of output sensor voltage into measured force. The total accuracy of measuring system can be significantly improved by adding a data-conditioning block (DCB). The function of this block is to regulate the basic metrological characteristics of measuring system in order to eliminate the errors. The using of exactly defined algorithm of error elimination is not suitable in this case, because hysteresis error causes ambiguity of transfer characteristic. So, exact conversion of output sensor voltage into measured force is impossible. The unconventional solution is use of the neural networks as DCB for the conversion and for the sensor errors elimination. We suggest that the performance of neural networks is better than that of other conventional techniques for the conversion. Existing solutions, which overcome the shortcomings of sensors, are based on feedforward neural networks [1], [10]. However it does not involve a time-dimension. From the neural network architectures, we decided to choose the feedforward neural networks and time delay neural networks for performance comparison.

The Fig. 2 shows a classical apparatus for measuring of large force with use of neural network as DCB. The measured characteristics of output voltage from acting force $U_2\uparrow$ (if force is increasing from 0 kN to 200 kN – characteristic upward) and $U_{2\downarrow}$ (if force is decreasing from 200 kN to 0 kN – characteristic downward) are shown in Fig. 3. U_{2lin} is linear transfer characteristic, it is calculated by the least square method. In the next subchapters, advantages, disadvantages and design of these NN architectures are described.

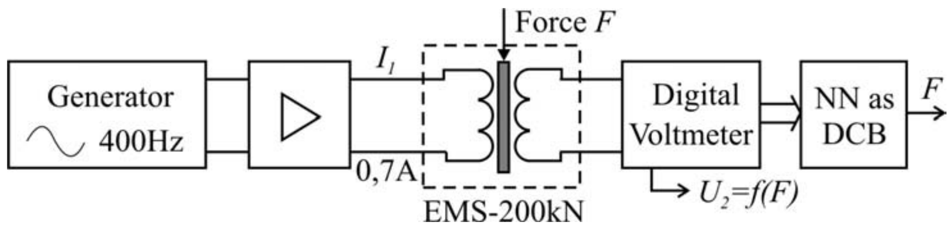


Fig. 2 Measuring apparatus with neural network

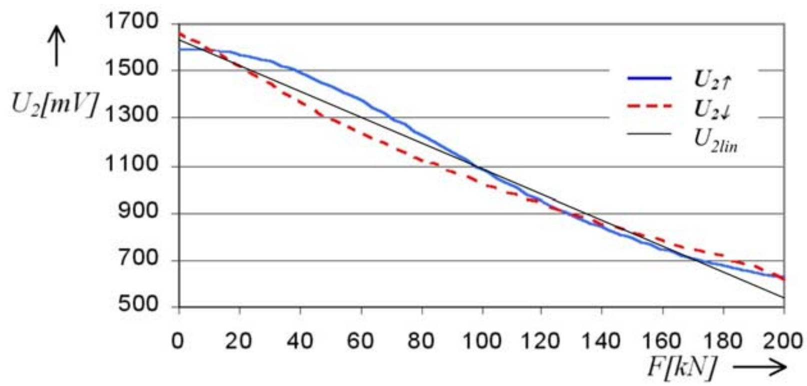


Fig. 3 Measured characteristics $U_{2\uparrow}$, $U_{2\downarrow}$ and linear transfer characteristic U_{2lin}

2.1. Feedforward neural network

The basic feedforward neural network (FFNN) performs a non-linear transformation of input data in order to approximate the output data. The number of input and output nodes is determined by the nature of the modelling problem being tackled, the input data representation and the form of the network output required. The number of hidden layer nodes is related to the complexity of the system being modelled. The interconnections within the network are between every neuron in actual layer and every neuron in the next layers. Each interconnection presents a scalar weight, which is adjusted during the training phase. The hidden layer nodes typically have sigmoid transfer functions.

A three-layered FFNN (input, hidden and output layer) has the ability to approximate any non-linear continuous function to an arbitrary degree of precision, provided that the hidden layer contains sufficient number of nodes [4]. The problem of determining the network parameters (weights) is essentially a non-linear optimization task. The back-propagation method is the most popular training algorithm. In certain cases, multilayer FFNNs are able to learn to associate a required output to an input vector [2]. The requirement of memorization input – required output (lookup table) by FFNN is insufficient for mentioned above problem because an expectation of neural network is an intelligent response to an unknown input. It means that neural network must correctly generalize the training patterns.

2.2. Time Delay Neural Network

Time delay neural network (TDNN) comes under dynamic neural networks, which are designed to explicitly include time relationships in the input-output mappings. Time-lagged feedforward networks (TLFNs) are a special type of dynamic networks that integrate linear filter structures inside a feedforward neural network to extend the non-linear mapping capabilities of the network with a representation of time [4]. Thus, in TLFN the time representation is brought inside the learning machine. The advantage of this technique is that the learning machine can use filtering information while the disadvantage is that the learning becomes complex since the time information is also coded in. TDNN is one of the specific cases of TLFN where a tapped delay line is given in the input followed by a multilayer perceptron (MLP) as shown in the block diagram in Fig. 4. Current input (at time *t*) and *D* delayed inputs (at time *t*–1, *t*–2, ..., *t*–*D*) can be seen by the TDNN. The TDNN can be trained by using gradient descent back propagation. The ordered training patterns must be provided during training process [14].

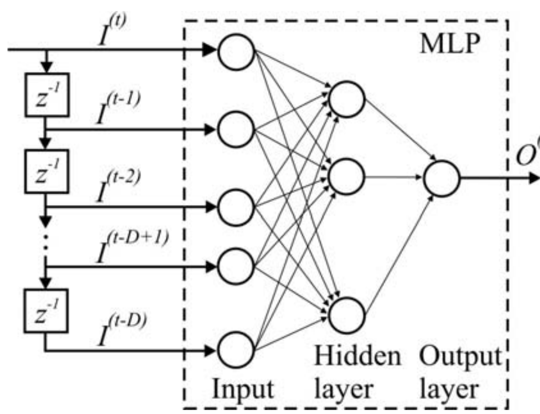


Fig. 4 Structure of Time Delay Neural Network

2.3. Design of the Neural Network

The design of the neural network was made to facilitate implementation in a distributed environment. The implementation was made in SNNS version 4.1 so that arbitrary topologies of networks could be tested and activation functions could be specified independently for every node [15]. The design of the neural network consists also of specification of number of layers (mainly hidden) and number of nodes (neurons), selection of learning algorithm and setting of proper learning parameters [3].

The learning algorithm Std_Backpropagation was used for FFNN and TimeDelayBackprop was used for TDNN. Topological_Order (for FFNN) and TimeDelay_Order (for TDNN) were used as an update functions. Randomize_Weights initialization function set up all weights and the biases with distributed random values for all NN models. The values were chosen from the interval <-1; 1>.

The learning rate can also be varied and a momentum term can be specified to filter out high frequency disturbances on the error surface that the network traverses during the learning phase. The learning rate was chosen to be low enough so that the curve got by plotting the mean square error on the test data set (and the training data set) versus the number of training epochs was a smooth curve. This meant that the mean square error of the training data set monotonically decreased over time and the mean square error of the test data set decreased monotonically till the point of overfitting, after which it increased monotonically. We experimented with various learning rates for the problem at hand and finally achieved the learning rate of 0.4. This learning rate gives rise to smooth mean square error; a higher learning rate does not lead to any learning at all. The momentum term was not experimented with in this project and it was set to 0 in all the simulations.

The problem of neural network with one hidden layer consists of global interaction of neurons. Improvement of approximation in some points causes approximation degradation in other points. In accordance with approximation, the neural network with two hidden layer can be better, because the local attributes are extracted in first hidden layer – some neurons of first hidden layer divide input area to the sub areas and other neurons of this layer learn the local attributes of these sub areas, and the global attributes are extracted in second hidden layer – the neurons of this layer combine the outputs of the first layer neurons and learn global attributes of sub areas, the second layer neurons are passive outside of this sub areas. Therefore, NNs with one and two hidden layers are discussed in the paper.

The running experiments determined that the larger number of nodes in the neural network was not the optimal case and that there were networks with a certain number of nodes that were better suited for the task than other networks. The reason for this is over-generalization when there are too few nodes and under-generalization when there are too many nodes. That is, when there are too few nodes, the network is unable to do enough processing and has enough information to detect trends and predict good output. However, when there are too many nodes, the network acts like a lookup table and stores specific variations in output – the network is unable to generalize input data [13].

The number of neurons in hidden layers is stated by analysis of NN in wide range. Topology 1-15-1 was the final topology for NN with one hidden layer and topology 1-8-6-1 was the final topology for NN with two hidden layers. Moreover, in comparison with FFNN topologies, the TDNN topologies were time delayed in input.

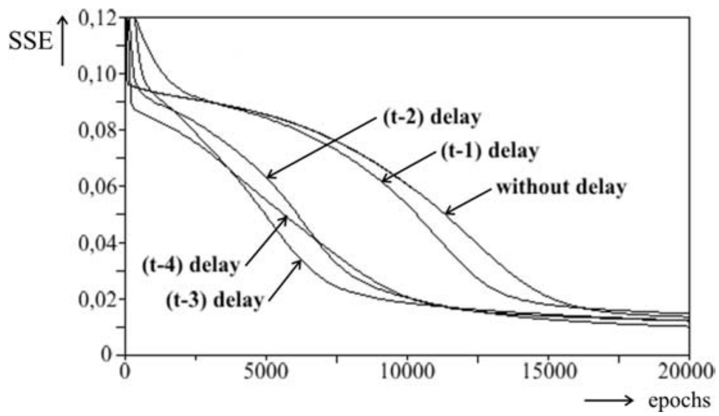
In training phase, two training sets (with 102 and 42 patterns) and two validation sets (with 81 and 21 patterns) were created from measured values. For testing were used three extra test sets with 40 patterns each. For training sets with 102 and 42 patterns, respectively, obtained results were equal. In case of FFNN, it does not matter whether input-output patterns are ordering or they are chosen randomly. However in case of TDNN, input-output patterns must be ordering in accordance with measured data. Finding the best FFNN and TDNN models for sensor errors elimination is described in the following chapter.

3. Evaluation of experiments

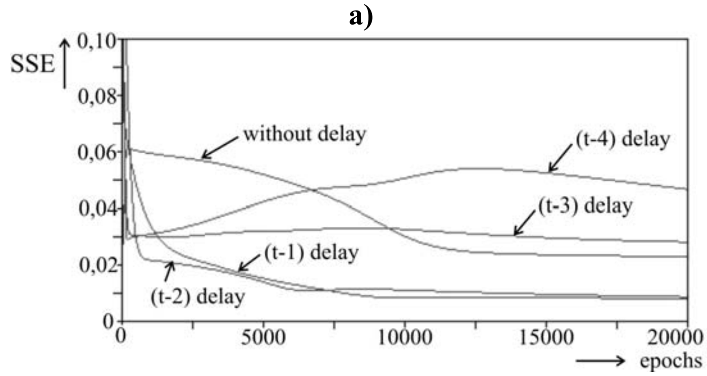
A sum-squared error (SSE) is used to compare a performance of neural network models. The SSE is defined as a squared difference between a target output and an actual output of the neural network:

$$SSE = \sum_{p} \left(t_p - y_p \right)^2 \tag{1}$$

Where t_p is the target output for pattern number p and y_p is the actual output for pattern p. The all types of TDNN are compared with a basic model without any time delays (FFNN).
One of the problems faced in training neural networks is overfitting the data used for training. Overfitting is a critical problem in most all-standard NN architectures. Furthermore, NNs and other AI machine learning models are prone to overfitting [7]. One of the solutions is early stopping. The stopping criterion is also another issue to consider in preventing overfitting [11], [12]. Hence, for this problem during training, validation set is used instead of training data set. After a few epochs the network is tested with the validation data. The training is stopped as soon as the error of validation set increases rapidly higher than the last time it was checked [11]. Next figures show error curves during training and show that chosen number of training epochs is sufficient for convergence these processes toward a finite value.

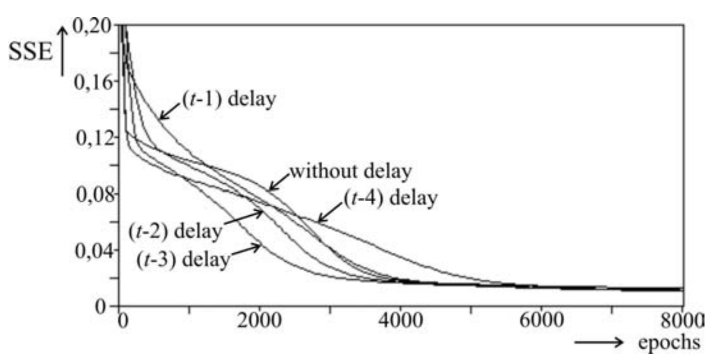


a)

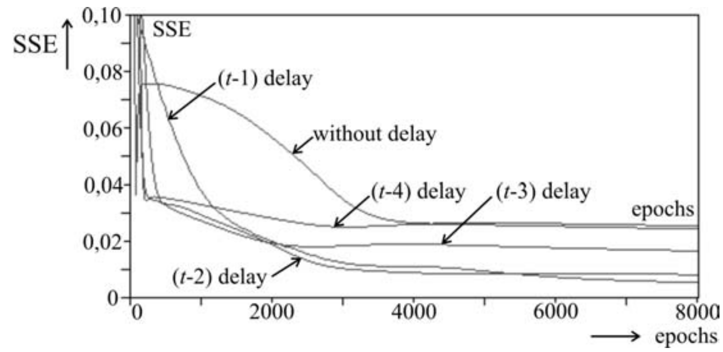


b)

Fig. 5 Training process – NN models with one hidden layer: a) error curves of training sets, b) error curves of validation sets



a)



b)

Fig. 6 Training process – NN models with two hidden layers: a) error curves of training sets, b) error curves of validation sets

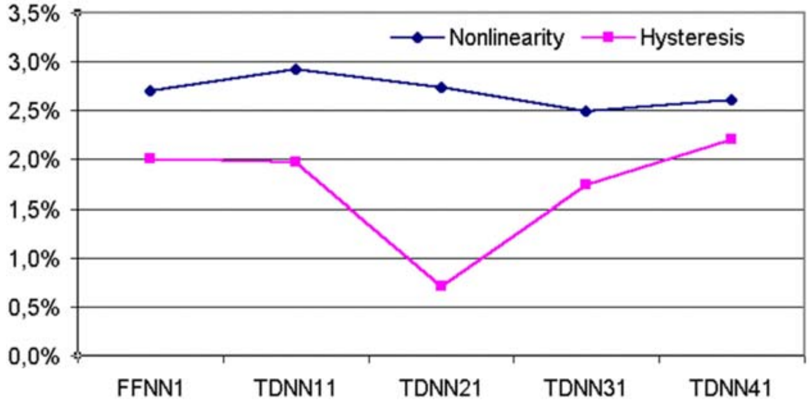
The error curves acquired during training of NN are showed in Fig. 5 (for NN with one hidden layer) and Fig. 6 (for NN with two hidden layers). There are the curves for all the models (FFNN – without delay, TDNNs with (*t*-1), (*t*-2), (*t*-3), (*t*-4) delay of input). Furthermore, the error curves of training sets are compared. As we can see, all the NN models in Fig. 5a needed about 20000 training epochs for convergence toward a finite small SSE value. However, the NN models in Fig. 6a needed less than 8000 training epochs for the convergence. The error curves of validation sets are showed in Fig. 5b and Fig. 6b. The convergence is not guaranteed in one specific case – TDNN with one hidden layer and with (*t*-4) delay in input. Achieved SSE values are showed in Tab. 1. These conclusions make us believe that the NN models with two hidden layers are faster in training patterns.

After this analysis, three test sets were used for errors determination of NN models. The inaccuracy, measured error, nonlinearity and hysteresis were computed by standard IEC 60770 [5], see in Tab. 1.

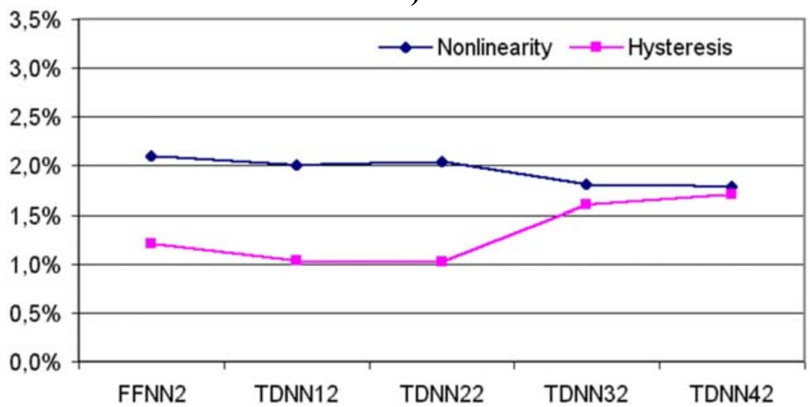
Tab. 1 Comparison of neural network models

Input delay	No. of hidden layers	SSE of		Inaccuracy	Measured error	Non-linearity	Hysteresis	Model title
		Tr. set	Val. Set					
without	1 layer	0,01193	0,02305	(-3,52; 3,62)	3,28	2,70	2,00	FFNN1
	2 layers	0,00524	0,02040	(-2,55; 2,08)	2,28	2,10	1,21	FFNN2
(t-1)	1 layer	0,01117	0,00802	(-3,90; 3,43)	3,17	2,92	1,97	TDNN11
	2 layers	0,00514	0,00316	(-2,70; 2,03)	2,10	2,00	1,03	TDNN12
(t-2)	1 layer	0,01109	0,00846	(-3,76; 3,43)	2,93	2,74	0,71	TDNN21
	2 layers	0,00451	0,00628	(-2,99; 1,71)	2,30	2,04	1,02	TDNN22
(t-3)	1 layer	0,01090	0,02793	(-3,69; 3,44)	2,87	2,49	1,74	TDNN31
	2 layers	0,00430	0,01217	(-2,72; 1,71)	2,08	1,81	1,60	TDNN32
(t-4)	1 layer	0,01107	0,04644	(-3,76; 3,20)	3,00	2,61	2,20	TDNN41
	2 layers	0,00585	0,01974	(-2,77; 1,79)	2,13	1,78	1,70	TDNN42

The disadvantages of elastomagnetic sensor are its relatively high hysteresis and nonlinearity. In addition, the hysteresis causes ambiguity of transfer characteristic and with it related impossibility of exact conversion of output sensor voltage into measured force. The solution of this problem is finding an ideal transfer characteristic. This characteristic is a straight line calculated from measured data sets [6] by the least square method – U_{2lin} (see Fig. 3). The sensor errors in this case are: inaccuracy (-5,20%; 6,97%), measured error 6,67%, nonlinearity 6,67% and hysteresis 1,30%. The comparison of NN models according to nonlinearity and hysteresis is showed in Fig. 7.



a)



b)

Fig. 7 Comparison of NN models: a) with one hidden layer, b) with two hidden layers

As we can see, NN models are able to significantly reduce the nonlinearity and NN with two hidden layers are more effective in this reduction. This assumption (mentioned above, Sec. 2.3) is confirmed experimentally. However, hysteresis reduction is not so significant. The NN models with one hidden layer (except NN with (t-2) delay) are not suitable for hysteresis reduction. The NN models TDNN22 and TDNN32 reduce the hysteresis in a small scale (from 1,30% to approximately 1%). Others NN models are not able to reduce the hysteresis.

4. Conclusion

The standard evaluation of pressure force measurements by elastomagnetic sensor led to sizable errors, nonlinearity over 5%. It was due to impossibility of exact conversion of output sensor voltage into measured force. Therefore an ideal transfer characteristic was gained from measured values. This characteristic could not represent well the real dependency $U_2=f(F)$ of the sensor and conversion $F=f^{-1}(U_2)$. Neural networks are one potential solution to apply to this task. Implementation of neural networks into measuring set solves the problem with mentioned conversion and reduces the sensor errors; nonlinearity is reduced from approx. 6% to approx. 2%. Problems of neural networks are the lack of defining rules to help construct a network given a problem - there are many factors to take into consideration: the learning algorithm, architecture, number of neurons per layer, number of layers, data representation and much more. Our research showed that the more available solution appears to use neural networks with temporal dependence of output from input data in comparison with using of

FFNN. Time delay neural networks have been successful but require many training examples before a satisfactory level of accuracy is obtained. The best results of sensor errors reduction are obtained by using of TDNN with two hidden layers and with (t-2) and (t-3) delay in input.

Acknowledgement

The research paper has been prepared by the support of the Slovak Grant Agency as project No.1/0376/2003 and of the Institutional project of Faculty of Electrical Engineering and Informatics, Technical University of Košice, No. 4433.

REFERENCES

[1] ARPAIA, P. – DAPONTE, P. – GRIMALDI, D. – MICHAELI, L. "ANN-Based Error Reduction for Experimentally Modeled Sensors", In: *IEEE Trans. on Instrumentation and Measurement,* vol. 51, no. 1, 2002, pp. 23-30.

[2] HECHT-NIELSEN, R. Kolmogorov's Mapping Neural Network Existence Theorem. In: *1st IEEE International Conference on Neural Networks*, San Diego, CA, vol. 3, 1987, pp. 11-14.

[3] HAECKER, J. – RUDOLPH, S. On Neural Network Topology Design for Nonlinear Control. In: *Proceedings SPIE Aerosense 2001 – Conference On Applications and Science of Computational Intelligence IV.* Orlando, Florida, 2001.

[4] HAYKIN, S. Neural Networks: A Comprehensive Foundation, 2nd Edition, Prentice Hall PTR. ISBN 0-13-273350-1, 1998, p. 837.

[5] IEC 60770, Transmitters for Use in Industrial-process Control Systems - Part 1: Methods for Performance Evaluation, 1999-02, Part 2: Methods for Inspection and Routine Testing, 2003-01.

[6] IEC 61298-2, Process Measurement and Control Devices - General Methods and Procedures for Evaluating Performance - Part 2: Tests under Reference Conditions, 1995-07.

[7] LAWRENCE, S. – GILES, C.L. – TSOI, A.C. Lessons in Neural Network Training: Training May be Harder than Expected. In: *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, AAAI-97, Menlo Park, California: AAAI Press, 1997, pp. 540-545.

[8] MOJŽIŠ, M. et al. Properties of 200 kN Force Sensor, In: *Journal of Electrical Engineering,* vol. 50, no. 3-4, 1999, pp. 105-108.

[9] MOJŽIŠ, M. – ORENDÁČ, M. – VOJTKO, J. Pressure Force Sensor. In: *Proceedings ISC'2001 - II. Internal Scientific Conference of the Faculty of Electrical Engineering and Informatics*, Košice, Slovak Republic, 2001, pp. 19-20, ISBN 80-88964-84-9.

[10] PANDA, G. – DAS, D.P. – MISHRA, S.K. – PANDA, S.K. – MEHER, S. – MAHAPATRA K.K. Development of an Efficient Intelligent Pressure Sensor Using RBF based ANN Technique

[11] PRECHELT, L. Early Stopping -- but when? In: *Neural Networks: Tricks of the Trade*, Lecture Notes in Computer Science 1524, Springer Verlag, Heidelberg, 1998, pp 55-69.

[12] SARLE, W. S. Neural Network FAQ, part 1 of 7: Introduction. Periodic posting to the Usenet Newsgroup comp.ai.neural-nets, 2004 from World Wide Web: ftp://ftp.sas.com/pub/neural/FAQ.html

[13] SHANMUGASUNDARAM, J. – NAGENDRA PRASAD, M. V. – VADHAVKAR, S. – GUPTA, A. Use of Recurrent Neural Networks for Strategic Data Mining of Sales Information, In: *Information Management Resources Association (IRMA) International Conference*, May 1999.

[14] WEIBEL, A. – HANAZAWA, T. – HINTON, G. – SHIKANO, K. – LANG, K. Phenomena Recognition Using Time-delay Neural Networks. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37, 1989, pp. 328-339.

[15]ZELL, A. et al. "SNNS User Manual, version 4.1", University of Stuttgart, Institute for Parallel and Distributed High Performance Systems; University of Tübingen, Since 1989.

BIOGRAPHY

Jozef VOJTKO (MSc.) was born in Michalovce, in 1976. He graduated from Department of Electronics and Multimedia Telecommunications, Faculty of Electrical Engineering and Informatics at the Technical University of Košice in 1999. He works as an Assistant Professor at the Department of Theory of Electrical Engineering and Measurement, TU FEI in Košice. His interests are the error elimination of sensors and use of neural networks in measurement.