NEURAL NETWORK BASED MODELING OF NO_X DETECTION WITH A SENSOR – POLYMER SYSTEM

Ciprian Piuleac[1], Silvia Curteanu[1*], Gabriela Țelipan[2], Maria Cazacu[3]

[1] "Gh. Asachi" Technical University, Department of Chemical Engineering, Bd. D. Mangeron No. 71A, 700050, Iasi, Romania
[2] Research and Development National Institute for Electrical Engineering ICPE CA, Spl. Unirii 313, sect. 3, Bucharest, Romania
[3] "P. Poni" Institute of Macromolecular Chemistry, Aleea Gr.Ghica Voda 41A, Iasi, 700487, Romania

*Corresponding author: silvia_curteanu@yahoo.com

Received: 09/01/2008
Accepted after revision: 06/02/2008

Abstract: Some types of synthetic polymers have been tested in nitric oxide sensing.
Feed-forward neural networks with different types and topologies are used in mathematical modeling of the system, to predict the voltage of the sensor as function of polymer type, gas concentration and time. In this way, the efficiency of the sensor can be appreciated. The relative errors, extremely low, obtaining in the validation phase, prove the validity of the neural models. The optimization problem performed by inverse neural network modeling answers the question what are the initial conditions that lead to an imposed value for output signal of the sensor.

Key words: *neural networks, mathematical modeling, polymer-based sensors, direct and inverse modeling, nitric oxide*

INTRODUCTION

A chemical sensor provides information about its environment and consists of a physical transducer and a chemically selective layer. Sensor devices have been made from classical semiconductors, solid electrolytes, insulators, metals and catalytic materials. Since the chemical and physical properties of polymers may be tailored for particular needs, they occupy an important position as materials in various sensor devices among other materials. Although a majority of polymers are unable to conduct electricity, their insulating properties are utilized in the electronic industry. An intrinsically conducting polymer is used as a coating or encapsulating material on an electrode surface, while a non-conducting polymer is used for immobilization of specific receptor agents on the sensor device [1]. For example, polysiloxanes could be used for physical or chemical entrapment of some gas-sensing agents. In fact, the amperometric sensors fabricated by Mizutani et al. [2] for the determination of dissolved oxygen and nitric oxide are based on a permselective PDMS membrane. A hydrophobic polymer layer with porous structure is useful for the selective permeation of gases. The electrode prepared by dip coating from an emulsion of PDMS can measure a very low concentration of nitric oxide. Being permselective, the polymer coating is capable of discriminating between gases and hydrophobic species, which co-exist in the samples to be measured. Gases permeate easily through the pores to reach the electrode surface, whereas the transport of the hydrophilic compounds is strongly restricted [1]. It has been demonstrated that the presence of the complexed iron in the polymeric structure increased the sensor sensitivity for nitric oxide [3, 4]. Recently, it has been proved that the poly(aryl-ether-sulfone)s could be valuable alternative for sensors and actuators [5].

In the last decade, neural networks have attracted great interest as predictive models because they have proved to be able to approximate any continuous nonlinear functions. Neural networks have the ability to learn what happens in the process without actually modeling the physical and chemical laws that govern the system, so they are useful for modeling complex nonlinear processes where understanding is limited.

Many approaches are known in literature about efficient modeling of chemical processes using neural networks. Different types and modeling methodologies are employed: direct modeling [6], neural networks based soft sensors [7], inferential modeling [8], inverse neural network modeling [9]. Different types of neural networks are used in the chemical reaction engineering, the type of the network being correlated with the nature of the application and the chemical system: feed-forward neural networks, for stationary conditions [8], recurrent neural networks, useful for long term predictions [10], stacked neural networks [6], hybrid models, which combine phenomenological models with neural ones [11], neural networks trained with static and dynamic operating data [12]. These problems are reviewed in our previous work [13].

The present work uses the ability of neural network models to make predictions in order to substitute experiments that are time and materials consuming. A complex system that takes into account the polymer used for the sensor manufacture, the gas for testing the sensor signal and the response time is modeled. The variation of sensor voltage with time, gas concentration and type of polymer is evaluated by a neural network based methodology.

The contribution of this paper refers mainly the modeling and optimization capacities of simple topologies and simple working strategies of neural networks. Multi-layer feed forward neural networks (with one or two hidden layers) were projected and used in modeling and optimization. Likewise, the fundamental issues of designing and handling neural networks are emphasized in order to make them accessible to the researchers working in the field of chemical engineering.

MATERIALS AND METHODS

Neural networks

A neural network consists of processing units called *neurons* and information flow channels between the neurons - *interconnections*. The way in which neurons are connected to form a network represents the *neural network topology* (*architecture*). More precisely, the topology of a neural network consists of the framework of neurons together with its interconnection structure. The topology of a neural network plays a fundamental role in its functionality and performance. Most neural networks have a *layered topology*. A *multi-layer neural network* has *input, hidden* and *output layers* consisting of input, hidden and output neurons, respectively [14].
The most common neural network architecture is the *multi-layer feed-forward neural network* (often called *multi-layer perceptron*, *MLP*). The inputs of a hidden neuron are combined in a weighted sum, to which a constant term (*bias*) is added. This sum is then passed through a nonlinear transformation called *activation function*. In the multi-layer feed-forward network there are only *interlayer connections*. With each connection a *weight* is associated which is a weighted factor that reflects its importance. This weight is a scalar value, which can be positive (excitatory) or negative (inhibitory).
Reasons for the use of this kind of network are the simplicity of its theory, ease of programming and good results and because this neural network is an universal function in the sense that if topology of the network is allowed to vary freely it can take the shape of any broken curve [15].
Jordan and Elman Networks, *JEN*, extend the multilayer perceptron with context units, which are processing elements that remember past activity. Context units provide the network with the ability to extract temporal information from the data. In the Elman network, the activity of the first hidden processing elements is copied to the context units, while the Jordan network copies the output of the network. Networks which feed the input and the last hidden layer to the context units are also available. Four basic topologies, differing by the layers that feed the context units, are available for the Jordan Elman networks: 1) The default configuration feeds the context units with the input samples, providing an integrated past of the input (memory traces). 2) A second configuration creates memory traces from the first hidden layer (as proposed by Elman). 3) A third possibility is to use the past of the last hidden layer activations as input to the context units. 4) The final choice is to use the past of the output layer to create the memory traces, as proposed by Jordan [16].
Figure 1 presents the two described types of neural networks: MLP and JEN with the four configurations.

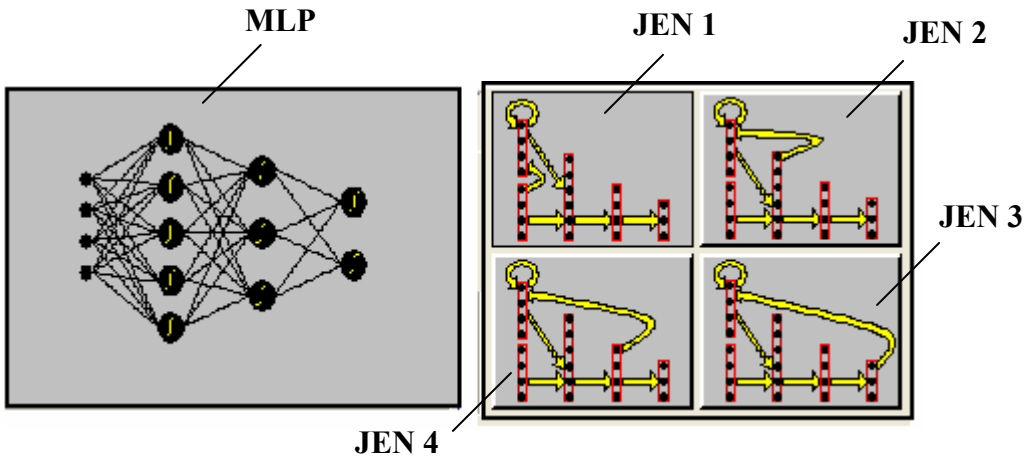

Figure 1. *Different types of neural networks: MLP and JEN*

The context unit remembers the past of its inputs using what has been called a recently gradient, i.e., the unit forgets the past with an exponential decay. This means that events that just happened are stronger than the ones that have occurred further in the past. The context unit controls the forgetting factor through the time constant. Useful values are between 0 and 1. A value of 1 is useless in the sense that all of the past is factored in. On the other extreme, a value of zero means that only the present time is factored in (i.e., there is no self-recurrent connection). The closer the value is to 1, the longer the memory depth and the slower the forgetting factor [16].

In the *training phase*, the neural network learns the behavior of the process. The *training data set* contains both input patterns and the corresponding output patterns (also called *target patterns*). Neural training leads to finding values of connection weights that minimize differences between the network outputs and the target values. The most extensively adopted algorithm for the learning phase is the *back-propagation algorithm*. The training phase is considered complete when the error of all the training patterns is less than a prespecified error criterion or a maximum number of *epochs* had been reached. If, after the entire set of training patterns was presented, the overall error is still unacceptable, the neural network would be returned to the beginning of the training patterns and the process would be repeated. [17].

The purpose of developing a neural model is to devise a network (set of formulae) that captures the essential relationships in the data. These formulae are then applied to new sets of inputs to produce corresponding outputs. This is called *generalization* and represents subsequent phase after training - *validation* or *testing phase*. Since a neural network is a nonlinear optimization process made up of a learning phase and a testing phase, the initial data set must be split into two subsets: one for training and one for testing. A learning algorithm should lead to a good fit to the training samples and, simultaneously, to a network that has a good generalization capability. A network is said to generalize well when the input-output relationship, found by the network, is correct for input/output patterns of validation data which were never used in training the network (*unseen data*).

In this paper, the characteristic voltage-time, essential to appreciate the utility of the sensor, can be predicted using neural models, at different gas concentrations.

Sensor configuration

Figure 2 shows the structure of the sensor. An alumina substrate 6 x 6 x 0.5 mm was used. On one side of the substrate was screen-printed an interdigitated electrode array using Au ink and heat treated at 950 °C for 1 h. The sensitive layer obtained by dissolving the polymer in chloroform was deposed on the substrate with provided electrode by spin coating method. The sensor contains the pads of Pd – Ag conductive ink and the conductive layer of Ag ink heat treated at 750 °C for 30 minutes.

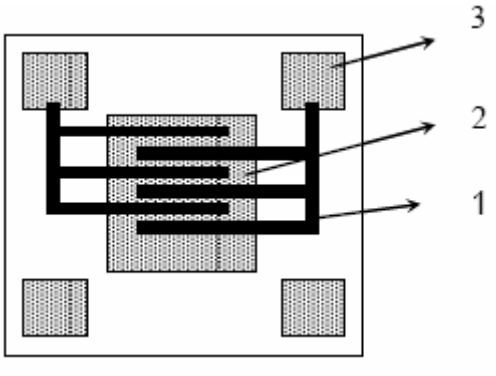

Figure 2. *Sensor configuration; 1 – electrode; 2 – sensitive layer; 3 – pad*

The detection testing of gas (NO_x) was made at different gas concentration of 50, 100, 200, 500, 100 ppm and the voltage measurements were performed with a multimeter type EXTECH LASER POINTER.

RESULTS AND DISCUSSION

In order to obtain polymer based sensors with the goal of NO_x detection, seven chemical structures were used with the following symbolic codification: 1 – polysulfone [18]; 2 – a siloxane-based polyazomethine having side-coordinated iron, Fe [19]; 3 – a siloxane polyamide having ferocenyl units within the chain, CEN21 [20]; 4 – a siloxane polyazomethine having complexing group, L3 [19]; 5 – a polymethylphenylsilane, PSMF2; 6 – a self-assembling polymer based on 4,4`-bipyridyl and siloxane-imide diacid, CH5 [21]; 7 – a self-assembling polymer based on 4,4`-bipyridyl and bis (carboxypropyl) disiloxane, CH2 [21]. All these polymers prove to be very sensitive to NO_x, providing a good signal, in a reasonable time interval.

Success in obtaining a reliable and robust network depends on the choice of process variables involved, as well as the available data set and the domain used for training purposes. The experimental data consist in some series of measurements that represent the variation of sensor output signal (voltage) with time and gas concentration. Firstly, these data are split into training and validation data sets because it is more important to evaluate the performance of the neural networks on unseen data that training data. In this way, we can appreciate the most important feature of a neural model - the generalization capability.

The neural models proposed in this paper have three inputs: type of polymer, time and gas concentration and one output – voltage. In this way, the sensor signal is correlated with the working conditions.

One major problem in the construction of neural networks is determining the network architecture, which is the number of hidden layers and the number of neurons in each hidden layer. It is generally accepted that a large number of hidden layers does not necessary improve the performance and increases the difficulties in training. Determining the number of hidden nodes depends on the nonlinearity of the problem and the error tolerance. Too many hidden nodes (an oversized network) cause the network to memorize the training set (*i.e.* over fitting) leading to poor performance of generalization. Too few hidden nodes may not achieve the required error tolerance (*i.e.* under fitting) having difficulties in representing the nonlinear processes.

Firstly, potentially good topologies must be identified. Nevertheless, no good theory or rule accompanies the neural network topology that should be used and trial-and-error method is still required. This is done by testing several topologies and comparing the prediction errors. Smaller errors indicate potentially good topologies, i.e. neural network topologies with chances to train well and to output good results [15]. The trial and error method is also applied in our work. Therefore, the number of hidden layers and units was established by training a different range of networks and selecting the one that best balanced generalization performance against network size.

The best network topology was determined based upon the mean squared errors (MSE) on the training data. Hidden neurons as well as output layer neuron use hyperbolic tangent as nonlinear activation functions. All the network weights were initialized as random numbers in the interval (–0.5, 0.5). The networks were trained using the back-propagation algorithm. Once the data have been fed into the neural networks, the weights were updated continuously based upon the back propagation learning rule. If, after the entire set of training patterns was presented, the overall error was still unacceptable, the neural network would be returned to the beginning of the training patterns and the process would be repeated. So, the training is considered terminated at the point where network error (MSE) becomes sufficiently small (less than 0.005).

The mean squared error was computed using the following formula:

$$MSE = \frac{\sum_{j=1}^{P}\sum_{i=1}^{N}(d_{ij} - y_{ij})^2}{N \cdot P} \qquad (1)$$

where P is the number of output processing elements (in this case, $P = 1$), N is the number of exemplars in the data set, y_{ij} is the network output for exemplar i at processing element j, and d_{ij} is the desired output for exemplar i at processing element j.

A special software application - *NeuroSolution* - was used in this paper in order to design and obtain predictions of neural networks. In this program, the following specifications are necessary: the network type, the input and desired output values, the stop condition of the training, the number of processing elements in hidden layers, the activation functions, the learning rule, the maximum number of epochs and some configuration parameters to display the neural model development.

Different topologies for MLP and JEN were developed as one can see in Table 1. They can be appreciating looking at MSE and correlation, *r*, (agreement between network prediction and experimental data).

Table 1. *Different topologies of MLP and JE networks*

No. crt.	Network topology	MSE	r
1	MLP(3:5:1)	0.032568	0.901849
2	MLP(3:10:1)	0.02027	0.9401
3	MLP(3:20:1)	0.014592	0.95726
4	MLP(3:12:4:1)	0.009887	0.971247
5	MLP(3:21:7:1)	0.015351	0.955204
6	**MLP(3:21:7:1)**	**0.006421**	**0.981423**
7	MLP(3:25:20:1)	0.00478	0.986343
8	MLP(3:36:12:4:1)	0.01697	0.951174
9	JEN(3:6:1)	0.027372	0.918269
10	JEN(3:10:1)	0.007963	0.976906
11	JEN(3:15:1)	0.003796	0.989062
12	**JEN(3:15:1)**	**0.002203**	**0.993665**
13	JEN(3:15:1)	0.001895	0.994553
14	JEN(3:20:1)	0.004664	0.986549
15	JEN(3:12:4:1)	0.008033	0.976704
16	JEN(3:15:1)	0.006414	0.981592

The best neural network architectures were MLP(3:21:7:1) and JEN(3:15:1), marked in grey in Table 1. These neural models supply good predictions on training data: average relative errors of 8.821% and 6.502%, respectively, and correlation between experimental data and network prediction were 0.9885 and 0.9946. Relative errors were calculated using the following formula:

$$E_r = abs\left(\frac{p_{exp} - p_{net}}{p_{exp}}\right) \cdot 100 \qquad (2)$$

where *p* represents the parameters under study (voltage), indexes *exp* and *net* signify experimental and network values.

Several examples are presented in Figures 3 and 4 which show a comparison between two sets of data, experimental and prediction on training data.

A key issue in neural network based process modeling is the robustness or generalization capability of the developed models, *i.e.* how well the model performs on unseen data. Thus, a serious examination of the accuracy of the neural network results requires the comparison with experimental data, which were not used in the training phase (previously unseen data). The predictions of the networks on validation data are given in Table 2 and in Figures 5 and 6.

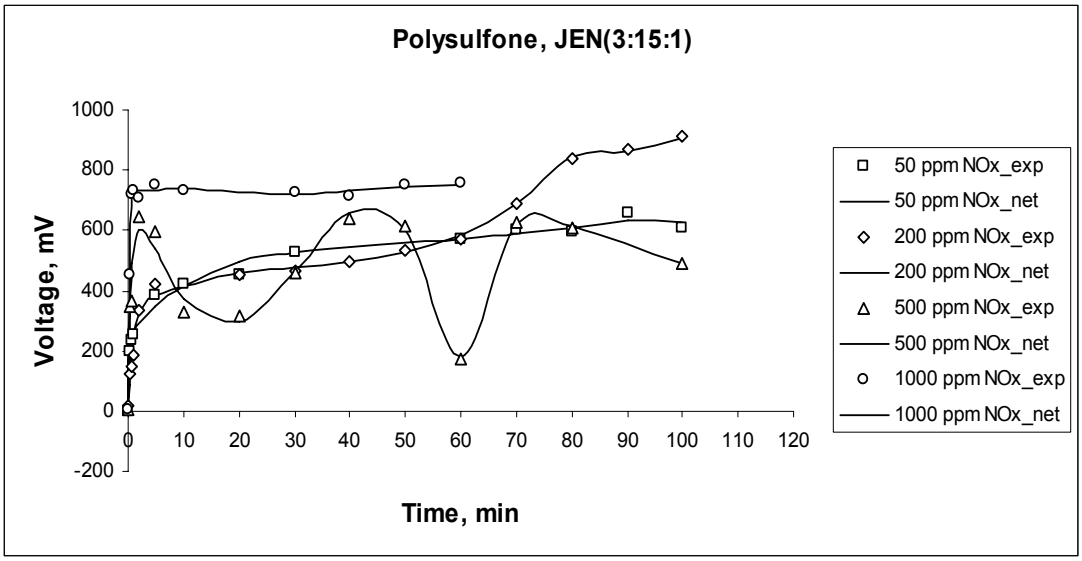

***Figure 3**. Experimental data and simulation results obtained with JEN(3:15:1) for polysulfone*

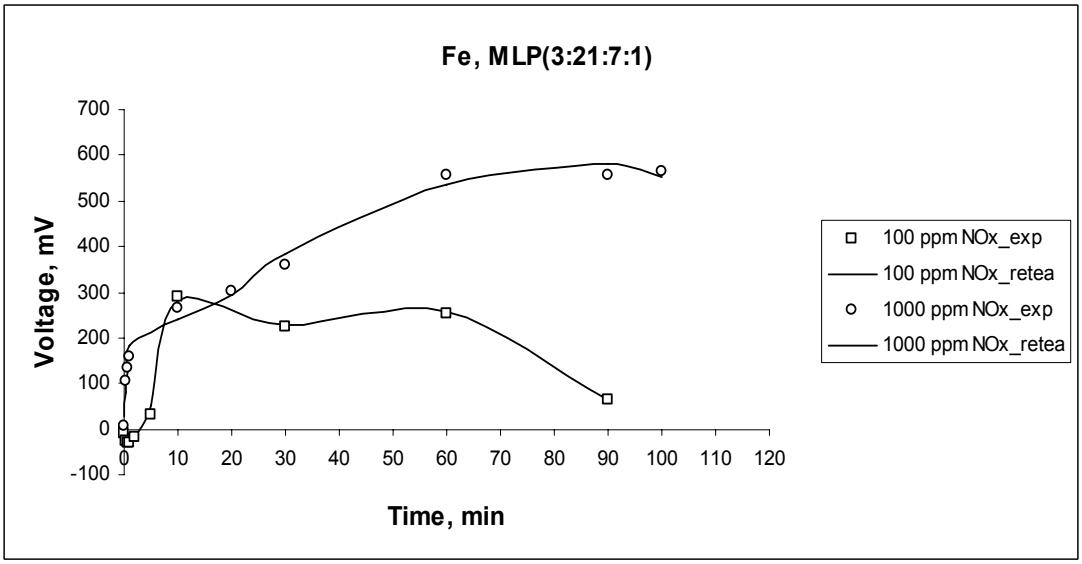

***Figure 4**. Experimental data and simulation results obtained with MLP(3:21:7:1) for Fe*

One can notice a satisfactory agreement between the two categories of data: experimental and neural network predictions. For this reason, the projected neural models can be used to make predictions under different reaction conditions, substituting the experiments that are time and material consuming.

For the studied sensors it is important to obtain a maximum signal in a shortest time, at a small gas concentration. Table 3 contains maximum values for the voltage and the corresponding time, for different type of polymers used in the construction of the sensors. Figure 7 presents comparatively the answers of different sensors; the best results are obtained with the sensors based on polysulfone and L3 at 100 ppm nitric oxide.

Table 2. The validation stage for the networks JEN(3:15:1) and MLP(3:21:7:1)

				JEN(3:15:1)			MLP(3:21:7:1)		
Polymer type	**Time**	**NOx conc.**	**Voltage**	**Voltage net**	**Relative error**	**Correlation**	**Voltage net**	**Relative error**	**Correlation**
1	2	50	265	16.981	115.816	**0.97962**	289.0	9.057	**0.9954**
1	10	200	445	12.360	82.979		419.2	5.803	
1	1	500	422	7.820	155.075		450.0	6.635	
1	90	500	815	4.294	8.701		780.0	4.294	
1	20	1000	724	4.696	133.882		717.9	0.849	
2	20	100	226	1.770	104.733		257.2	13.794	
2	2	1000	200	15.500	164.808		227.4	13.713	
3	0.5	100	101.3	19.447	41.421		108.0	6.647	
3	110	100	557	15.260	63.618		515.8	7.388	
4	2	100	291	6.529	177.343		332.5	14.260	
5	5	100	210	19.048	363.520		236.9	12.831	
5	80	100	484	14.876	68.836		470.0	2.891	
6	10	100	125.0	28.000	500.660		137.1	9.717	
7	30	100	3.9		11850.166		289.0	9.057	
				Average error	**12.81**		**Average error**	**8.298**	

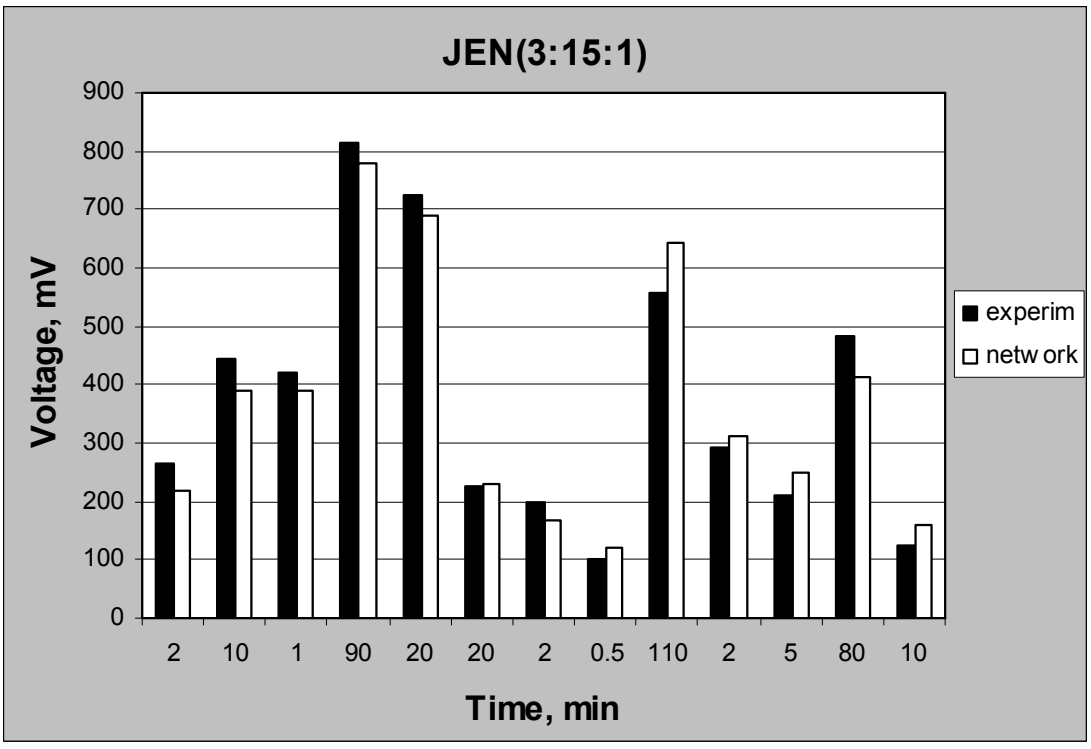

Figure 5. *Validation of JEN(3:15:1)*

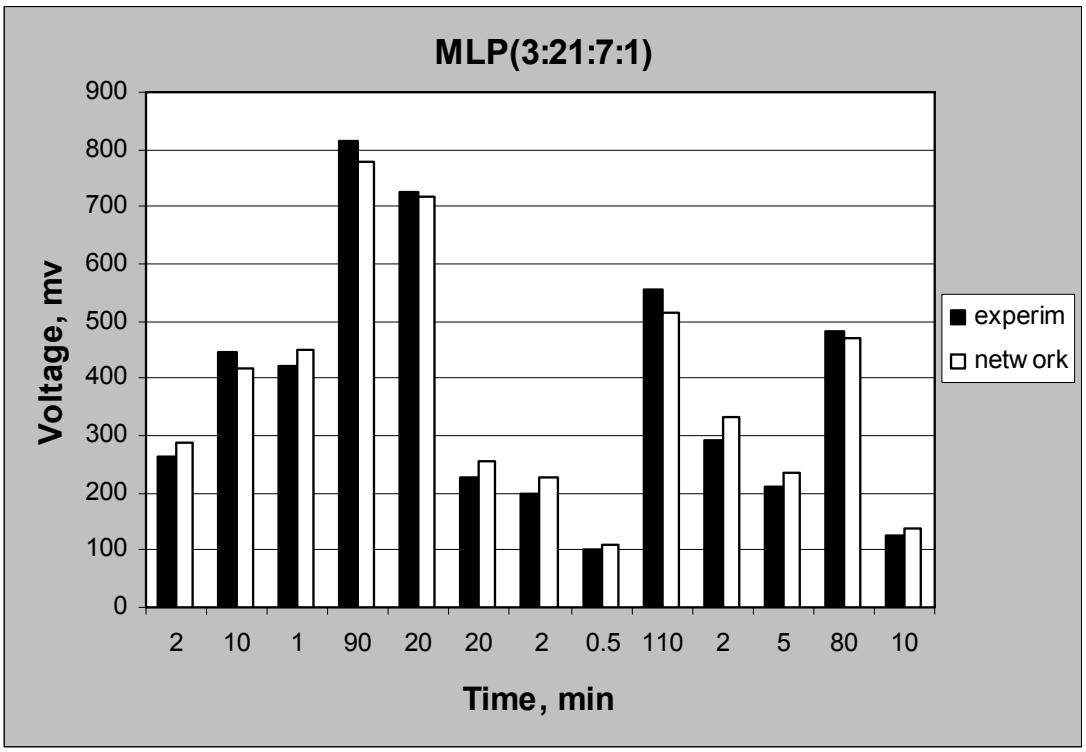

Figure 6. *Validation of MLP(3:21:7:1)*

Table 3. *Maximum values for voltage and the corresponding times*

Polymer type	Gas concentration	Time	Maximum voltage
polysulfone	50	90	655.7
polysulfone	200	100	900.4
polysulfone	500	2	641.4
polysulfone	1000	60	757.9
Fe	100	10	290.2
Fe	1000	100	567.7
CEN	100	100	554.3
L3	100	10	500.0
PSMF2	100	120	132.6
CH5	100	30	239.8
CH2	100	5	45.5

Sample Fe is L3 with Fe added; one can see in Figure 7 that L3 is more sensitive at gas detection, which means that the metal addition does not improve this property, on the contrary, it becomes worse.

The direct neural network modeling allows the estimation of the voltage for different conditions (type of polymer, time and gas concentration), even for the situations where experimental data are not available.

Supplementary information can be obtained by inverse neural modeling, which is an optimization problem for identification of initial conditions that lead to an imposed value for voltage. In our case, the optimization problem can be formulated as follows: *what is the time at which the sensor supplies a certain value for the output signal, for a specified polymer type and gas concentration ?*

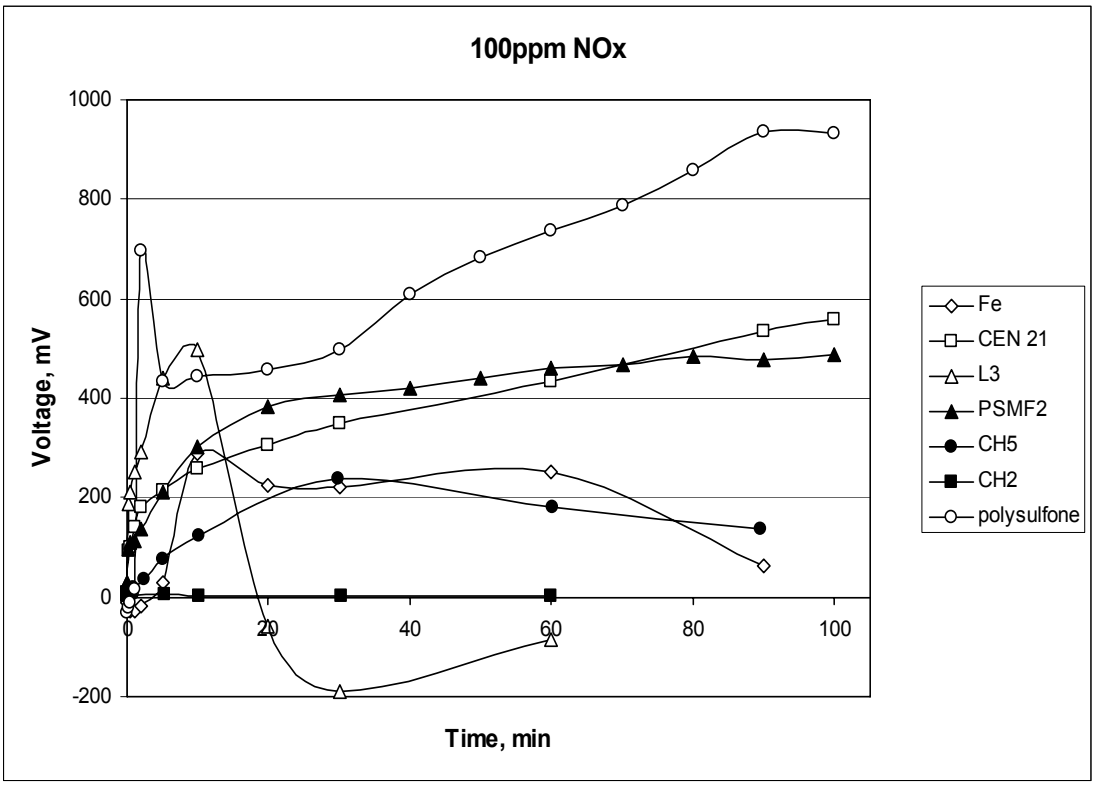

Figure 7. *The voltage variation in time for the studied sensors at 100 ppm NO_x*

Inverse neural models projected in this approach have as inputs voltage, gas concentration and type of polymer and as output, the response time. The best results are obtained with a Jordan/Elman neural network with two hidden layers, JEN(3:12:4:1) with MSE = 0.000001, r = 0.999998 şi E_p = 2.34% in the training phase.
Two examples are presented in Figures 8 and 9 for polysulfone at 200 ppm NO_x and for L3 at 100 ppm NO_x.

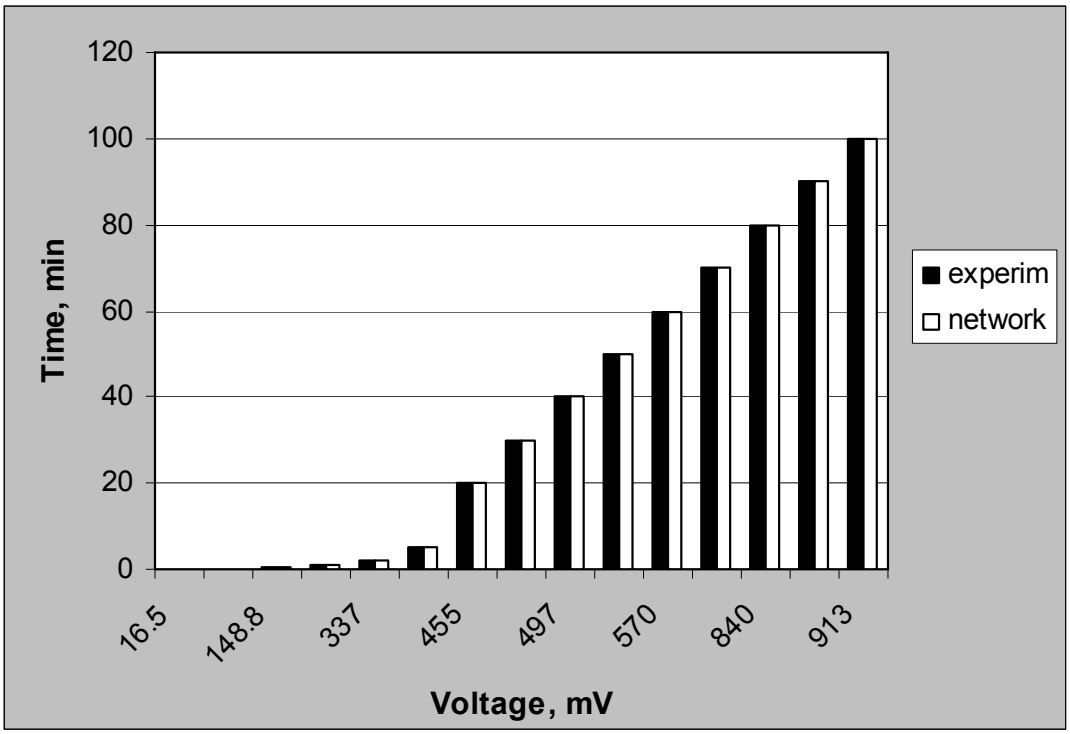

***Figure 8**. Time for obtaining certain values of voltage for polysulfone at 200 ppm NO_x*

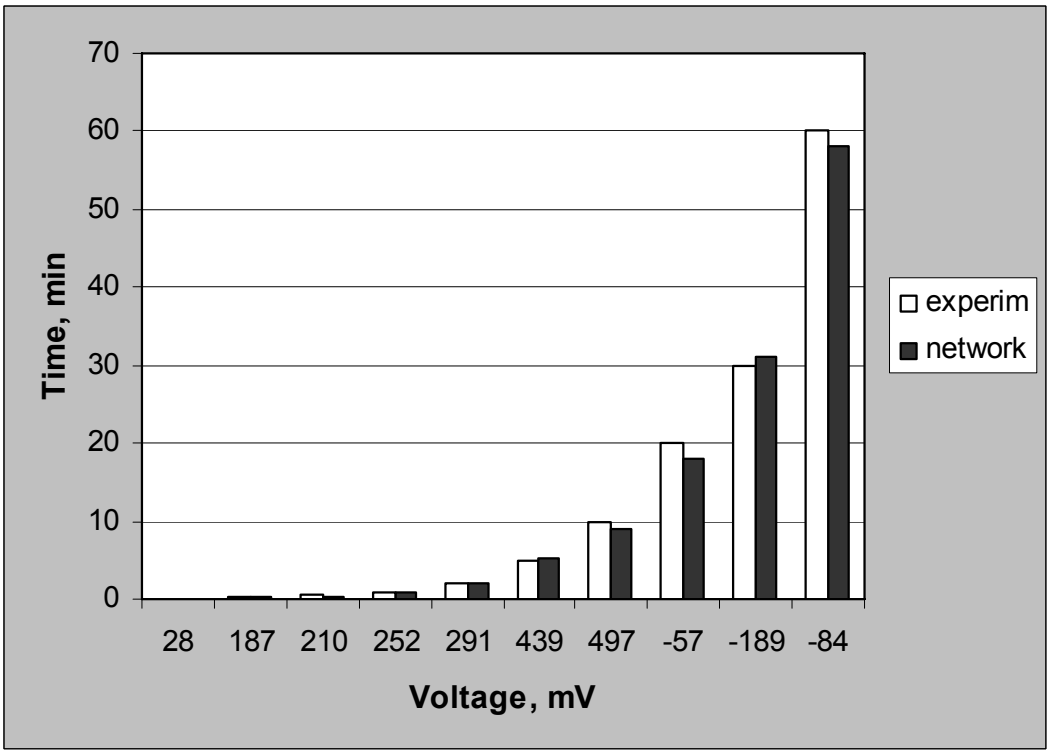

***Figure 9**. Time for obtaining certain values of voltage for L3 at 100 ppm NO_x*

CONCLUSIONS

The tested polymers, with various structures, have been proved to be good sensing elements in a sensor for NO_x detection.
The neural networks are adequate and recommended instruments for modeling nonlinear processes. The wide use of neural networks is based on their ability of learning from a set of numerical data (corresponding to input and output desired variables), using an adjustment algorithm for the network's parameters. Once trained, the network will be able to generate predictions faster than any other type of model.
Simple topologies and types of neural networks are designed for direct modeling of the characteristic voltage – time for the sensor, taking into account type of polymer and gas concentration. Accurate predictions of the neural models prove that the behavior of the system is well described.
Good models, relatively easy to develop and use, are, also, inverse neural networks which allow to appreciate the time intreval for obtaining a certain output signal of the sensor. In addition, as an advantage of this particular optimization procedure, one can mention the fact that is not necessary a mathematical model and a solving method like for classical optimizations.

REFERENCES

1. Adhikari, B., Majumdar, S.: Polymers in sensor applications, *Progress in Polym. Sci.*, **2004**, **29**, 699-766;
2. Mizutani, F., Yabuki, S., Sawaguchi, T., Hirata, Y., Sato, Y., Iijima, S.: Use of a siloxane polymer for the preparation of amperometric sensors: O_2 and NO sensors and enzyme sensors, *Sensors and Actuators B*, **2001**, **76**, 489-93;
3. Melo Jr., J.P., Biazzotto, J.C., Brunello, C.A., Graeff, C.F.O.: Solid state nitric oxide sensor prepared by sol–gel entrapment of iron-diethyldithiocarbamate in a siloxane matrix, *J. Non-Crystall. Solids*, **2004**, **348**, 235-239;
4. Fujii, S., Yoshimura, T., Kamada, H.: Nitric oxide trapping iron(III) complexes with dithiocarbamate derivatives, *Chem. Lett.*, **1996**, **25**, 785;
5. http://www.mii.vt.edu/Thrust3.htm;
6. Tian, Y., Zhang, J., Morris, J.: Modeling and Optimal Control of a Batch Polymerization Reactor Using a Hybrid Stacked Recurrent Neural Network Model, *Ind. Eng. Chem. Res.*, **2001**, **40**, 4525-4535;
7. Araújo, P.H.H., Sayer, C., de la Cal, J.C., Asua, J.M., Lima, E.L., Pinto, J.C.: Utilization of neural networks as soft sensors to monitor emulsion polymerization reactions (average particle diameter and conversion). *II ENPROMER,* Florianópolis, **1999**;
8. Fernandes, F.A.N., Lona, L.M.F., Penlidis, A.: Inverse modeling applications in emulsion polymerization of vinyl acetate, *Chem. Eng. Sci.*, **2004**, **59**, 3159-3167;
9. Zhang, J.: Developing robust non-linear models through bootstrap aggregated neural networks, *Neurocomputing*, **1999**, **25**, 93-113;
10. Xiong, Z., Zhang, J.:A batch-to-batch iterative optimal control strategy based on recurrent neural network models, *J. Proc. Control*, **2005**, **15**, 11-21;

11. Hinchliffe, M., Montague, G., Willis, M., Burke, A.: Hybrid approach to modeling an industrial polyethylene process, *AIChE J*., **2003**, **49**, 3127-3137;
12. Zhang, J.: Developing Robust Neural Network Models by Using Both Dynamic and Static Process Operating Data, *Ind. Eng. Chem. Res.*, **2001**, **40**, 234-241.
13. Curteanu, S., Popa, A.M.: Applications of neural networks in polymerization reaction engineering, *Rev. Roum. Chim.*, **2004**, **49**, 3-23;
14. Fiesler, E., Beale, R., Fiesler, E., Beale, R.: *Handbook of Neural Computation*, IOP Publshing LTd. and Oxford University Press, Oxford, **1997**;
15. Fernandes, F.A.N., Lona, L.M.F.: Neural network applications in polymerization processes, *Brazilian Journal of Chemical Engineering*, **2005**, **22**, 323-330;
16. Principe, J.C., Euliano, N.R., Lefebvre , W.C.: *Neural and Adaptive Systems: Fundamentals Through Simulations*, John Wiley & Sons Inc., **2000**;
17. Dhib, R., Hyson, W.: Neural Network Identification of Styrene Free Radical Polymerization, *Polym. Reaction Eng*., **2002**, **10**, 101-113;
18. Telipan, G., Ignat, M., Cozan, V.: Use of the polysulfone polymer in NOx detection, *J. Optoelectr. Adv. Mater.*, **2006**, **8**, 582-584;
19. Telipan, G., Cazacu, M., Ignat, M., Vlad, A., Marcu, M.: Siloxane-based polyazomethine with coordinated iron used in NOx detection, *Joint International Conference Materials for Electrical Engineering*, 15-16 June, **2006**, The 5th Edition MmdE 2006 and 3th Edition of IEEE ROMSC 2006;
20. Cazacu, M., Vlad, A., Marcu, M., Racles, C., Airinei, A., Munteanu, G.: New Organometallic Polymers by Polycondensation of Ferrocene and Siloxane Derivatives, *Macromolecules*, **2006**, **39**, 3786-3793;
21. Telipan, G., Ignat, M., Racles, C., Cazacu, M.: The synthesis, characterization of organo-siloxane supramolecular polymer and their gas sensing properties, *Surf. Interf. Analys*. – in press.