STRUCTURE ADAPTATION OF POLYNOMIAL STOCHASTIC NEURAL NETS USING LEARNING AUTOMATA TECHNIQUE

E. Gómez-Ramírez
UNIVERSIDAD LA SALLE, Laboratorio del Centro de Investigación
Benjamín Franklin No. 47 Col. Condesa
CP 06140, México, D.F., México Tel:(5) 728 05 00 ext. 5105, FAX (5) 271-15-44
E-mail: egomez@aldebaran.ci.ulsa.mx
&
A. S. Poznyak*
CINVESTAV-IPN, Sección de Control Automático,
Av. IPN 2508 AP 14-740,
CP 07000, México D.F., México Tel: (5) 747-00-00 ext. 3229, FAX: (5) 747-70-89
E-mail: apoznyak@ctrl.cinvestav.mx

Abstract - This paper is concerned with the selection of a number of nodes in polynomial artificial neural nets containing stochastic noise perturbations in the outputs of each node. The suggested approach is based on a reinforcement learning technique. To solve this optimization problem we introduce a special performance index in such a way that the best number of nodes corresponds to the minimum point of the suggested criterion. This criterion presents a linear combination of a residual minimization functional and some "generalized variance" of the involved disturbances of random nature. A large value of the noise variance leads to a different optimal number of neurons in a neural network because of the "interference" effect. Simulation modeling results are presented to illustrate the effectiveness of the suggested approach.

Index Terms: polynomial neural nets, stochastic, learning automata, best number of nodes.

1. Introduction

Artificial Neural Networks (ANN) are a very important tool in a variety of engineering problems. Special interest to applications of ANN in control optimization and pattern recognition have been proved to be successful [5][14][15]. However a few of them discuss the problem of best selecting an ANN structure in the application to some concrete engineering problems [4]. Some of them consider upper and lower bounds for the number of nodes in the hidden layer [9]. Other papers discuss the influence of the number of nodes and hidden layers in the minimization of square error for a specific application [1][2][19]. Other papers suggest algorithms to estimate the number of nodes for different paradigms [8][3][6][7][11]. However, the number of nodes within each hidden layer is assumed to be different for different engineering applications. As it was shown in [18], in the presence of noises of stochastic nature there exists a constructive technique providing "the best selection" of the number of nodes in the used ANN.

In this study we deal with Polynomial Artificial Neural Networks (PANN) containing stochastic disturbances in the output of each node. The aim is to derive an adaptive procedure providing the best (from the approximation accuracy point of view) selection of the number of nodes for a given PANN structure using only available information containing measurable input and output signals. In [18] this problem has been solved under the assumption that complete statistical information on stochastic noises was known a priory for a single node.

To realize an optimization procedure in the absence of complete information on the model description we need to apply some sort of *"learning"* to reach finally a successful selection procedure. In the case of discrete optimization on a finite set (because of the selection from a given finite set of possible structures) in the presence of random disturbances we suggest to apply *Learning Automata Technique* which turned out to be very effective in those

* This work was supported by CONACYT Mexico.

situations [13][12] [17]. We apply the ideas presented in [20] to construct the structure adaptation process.

The paper has the following structure: Section 2 deals with the description of PANN and the statement of the problem; in section 3 we formulate the problem of the best selection of the number of terms in the approximation problem of a multi-dimensional function in the presence of noises which depended on points of measurement. Section 4 shows how the main problem for selecting the best number of nodes in PANN can be transformed into the previous approximation problem. Sections 5 describe the learning automata technique and the Bush-Mosteller procedure as a reinforcement scheme to obtain the optimal number of nodes. Section 6 explains the application of learning automata for selecting the best number of nodes in PANN. The description of the numerical algorithm is given in section 7 with some simulation results and examples. Conclusion is presented in section 8.

2. Polynomial Artificial Neural Nets with Stochastic Noises and Formulation of the Problem

We present here the basic description of the PANN model containing stochastic noises in the outputs of each node. This model is assumed to be in force throughout the paper.

$$\hat{y}_k = \phi(x_1, x_2 \quad \ldots \quad x_{1,k-1} \quad x_{2,k-1} \quad \ldots \quad x_{1,k-ndi} \quad \ldots \quad y_{k-1} \quad y_{k-2} \quad \ldots y_{k-ndo}) + \zeta_k \tag{1}$$

where:

ndi	number of delays of the input
ndo	number of delays of the output
$\phi(*)$	is a nonlinear function
ζ	is a random variable
y	is the estimated function

The corresponding block diagram of a single node is depicted in fig. 1.

This non-linear function can be represented as:

$$\phi(z_1, z_2, \ldots, z_{nv}) = a_0(z_1, z_2, \ldots, z_{nv}) + a_1(z_1, z_2, \ldots, z_{nv}) + a_2(z_1, z_2, \ldots, z_{nv}) \ldots\ldots + a_{pow}(z_1, z_2, \ldots, z_{nv}) \tag{2}$$

where z_i is an input of the model, *nv* is the total number of elements in $\phi()$ description:

$$nv = ni + dvi + dvo \tag{3}$$

pow is the maximum power of the polynomial expression and $a_i(z_1,z_2,z_3)$ is a homogeneous polynomial of total degree *i*, for $i=0,\ldots,pow$, such that:

$$\begin{aligned}
&a_0(z_1, z_2, \ldots, z_{nv}) = c_0 \\
&a_1(z_1, z_2, \ldots, z_{nv}) = c_{1,1} z_1 + c_{1,2} z_2 + \ldots + c_{1,nv} z_{nv} \\
&a_2(z_1, z_2, \ldots, z_{nv}) = c_{2,1} z_1^2 + c_{2,2} z_1 z_2 + c_{2,3} z_1 z_3 + \ldots . z_1 z_{nv} + \ldots z_2^2 + \ldots z_2 z_3 \ldots \\
&a_3(z_1, z_2, \ldots, z_{nv}) = c_{3,1} z_1^3 + c_{3,2} z_1^2 z_2 + c_{3,3} z_1^2 z_3 + c_{3,4} z_1 z_2^2 + c_{3,5} z_1 z_2 z_3 + c_{3,} \\
&+ \ldots z_2^3 + .. z_2^2 z_3 + .. z_2 z_3^2 + \ldots . + c_{N_3} z_{nv}^3 \\
&a_{\text{pow}}(z_1, z_2, \ldots, z_{nv}) = c_{pow,1} z_1^{pow} + c_{pow,2} z_1^{pow-1} z_2 + \ldots + c_{pow,N_{pow}} z_{nv}^{pow}
\end{aligned} \tag{4}$$

where N_j is the number of terms of every polynomial such that $j=1,\ldots pow$. The number of terms N is equal:

$$N = \sum_{j=1}^{pow} N_j \tag{5}$$

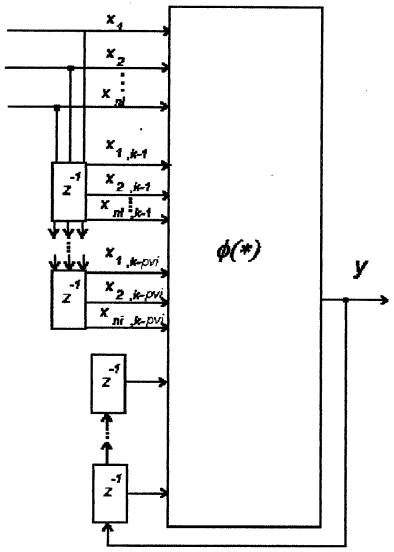

Figure 1 Scheme of PANN

Main Problem: *using the observation sequences of the input and output signals of each node, construct a learning procedure to obtain asymptotically "the best" (in some probability sense) numbers N of neurons providing the minimal approximation error.*

Next section explains the solution of this problem for the case of given basis functions used for the approximation aims.

3. Optimal Order Selection of Approximation of Multi Dimensional Functions in the Presence of Dependent Noises

The following assumptions will be in force throughout:

A1) Vector input $x_k \in R^K$ and scalar output $y_k \in R^1$ of a static reference model are connected by the following regression equation:

$$y_k = f(x_k) + \zeta_k, \quad k = 1,2...n \qquad (6)$$

where n is the number of available measurements, $\zeta_k \in R^1$ represents non-measurable disturbances of stochastic nature at the instant k and is a random variable defined on the probability space (Ω, F, P) such that:

$$E\{\zeta_k\} = 0,\ E\{\zeta_k \zeta_s\} = \sigma_k^2 \partial_{ks}$$

(here $\sigma_k^2 = \sigma_k^2(x_k)$ is the variance of the random variable ζ_k that may be dependent on the input x_k ; ∂_{ks} - Kronecker's symbol).

A2) The approximation model is selected as follows:

$$\hat{y}_k = \hat{f}^N(x_k | c_n) \stackrel{\Delta}{=} \sum_{i=1}^{N} c_i^N \phi_i^N(x_k) \qquad (7)$$

where $\phi_i^N(x_k)$ represents a known function and c_i a weight of this function in the output of this model, N is the order of this approximation model or, in other words, the number of different terms in the approximation given above.

A3) Parameters of the model (5) are chosen based on the *Weighted Least Squares Method* (WLSM) [10]:

$$\hat{c}_n = c_n \stackrel{\Delta}{=} \arg\min_{c_n} E\{S_n^N\}$$
$$S_n^N \stackrel{\Delta}{=} \frac{1}{n}\sum_{k=1}^{n} \sigma_k^{-2} (y_k - \hat{y}_k^N)^2 \qquad (8)$$

It is easy to show (minimizing this function) that the vector $\hat{c}_n$, corresponding to the minimizing solution of the WLSM-method, can be expressed by the following formula:

$$\hat{c}_n = \Gamma_n^N \sum_{k=1}^{n} \frac{1}{\sigma_k^2} y_k \phi^N(x_k)$$
$$\Gamma_n^N = \left(\sum_{k=1}^{n} \frac{1}{\sigma_k^2} \phi^N(x_k) (\phi^N(x_k))^T \right)^{-1} \qquad (9)$$

or, in equivalent recurrent form:

$$\hat{c}_n^N = \hat{c}_{n-1}^N + \sigma_n^{-2} \Gamma_n^N \phi^N(x_n) \left[y_n - (\hat{c}_{n-1}^N)^T \phi^N(x_n) \right] \qquad (10)$$
$$\Gamma_n^N = \Gamma_{n-1}^N - \frac{\Gamma_{n-1}^N \phi^N(x_n) (\phi^N(x_n))^T \Gamma_{n-1}^N}{\sigma_n^2 + (\phi^N(x_n))^T \Gamma_{n-1}^N \phi^N(x_n)}$$

Theorem 1. [23] *Under the assumptions A1)-A3) for the WLSM-method the following relation holds:*

$$D_n^N = E\{S_n^N\} + 2\frac{N}{n} - 1 \qquad (11)$$

where S_n^N is defined by (6) and

$$D_n^N \stackrel{\Delta}{=} \frac{1}{n}\sum_{t=1}^{n} \sigma_t^{-2} E\left\{ (f(x_t) - \hat{f}^N(x_t | \hat{c}_n^N))^2 \right\} \qquad (12)$$

characterizes the mean-square accuracy of the approximation (5) after n measurements of the pairs (x_t, y_t). This criterion is a generalization of Akaike Criterion to the case of time varying noise variances [20]

4. Selecting the Best Number of Nodes in PANN: The Case of Complete Information

Next theorem shows the equivalence of the previous statement on the best order of the approximation (7) to the central problem of the optimal selection of the best number of nodes in PANN.

Let the following assumptions hold:

B1) The output of the PANN is described by the following relation:

$$y_k = \sum_{i=1}^{N} c_i \phi(z_{i,k}) + \zeta_k \ . \qquad (13)$$

where c_i is the weight associated.

B2) noise at the output $\zeta_{,k}$ of each node is a random variable defined on some probability space (Ω, F, P) such that they are stationary, independent in time and are centered with finite variance, i.e.

$$E\{\zeta_k\} = 0, E\{\zeta_t \zeta_s\} = \sigma_\zeta^2 \partial_{ts} .$$

B3) The non-linear function can be represented like

$$\phi(z_1, z_2, ..., z_{nv}) = c_{1,1} z_1 + c_{1,2} z_2 + ... + c_{1,nv} z_{nv} + c_{2,1} z_1^2 + c_{2,2} z_1 z_2 + c_{2,3} z_1 z_3 +$$
$$....z_1 z_{nv} + ...z_2^2 + ...z_2 z_3 ... + c_{2,N_2} z_{nv}^2 + c_{3,1} z_1^3 + c_{3,2} z_1^2 z_2 + c_{3,3} z_1^2 z_3$$
$$+ c_{3,4} z_1 z_2^2 + c_{3,5} z_1 z_2 z_3 + c_{3,6} z_1 z_3^2 + ... + ...z_2^3 + ..z_2^2 z_3 + ..z_2 z_3^2$$
$$+ .... + c_{3,N_3} z_{nv}^3 .....c_{pow,1} z_1^{pow} + c_{pow,2} z_1^{pow-1} z_2 + ... + c_{pow,N_{pow}} z_{nv}^{pow} \qquad (14)$$

Then the nonlinear model (13) of the given neural node is equivalent to the static plant model (6). Then for a large enough number $k \to \infty$ "the best number" $(N_n(\omega_k))^*$ of the inputs of (node) *(i=1,2,...,N)* with probability one (almost sure) can be calculated:

The dependence of the function $D_n^N(\omega_k)$ (12) on N is depicted on fig. 2. It is clear that the curve $\tilde{S}_n^N(\omega_k)$ corresponds to the accuracy of the approximation. Curve 2 (the term $2\frac{N}{n}$) corresponds to the influence of the noises on the approximation process for different values of N; more noisy inputs produce more distortions in the measurements of the outputs of each node. The optimal

value of the number of inputs $(N_n)^*$ corresponds to the minimum value of "*the joint loss functions* " D_n^N.

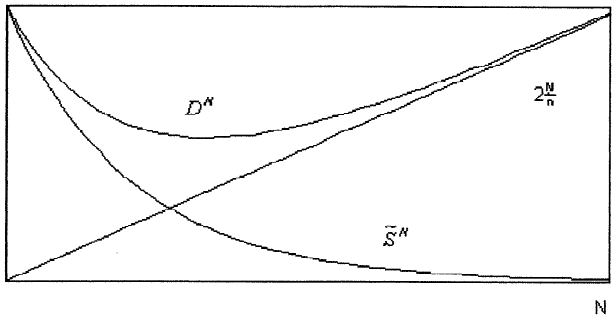

Figure 2 Function D_n^N

Notice that if the value of the variance σ^2 increases then the curve $\widetilde{S}_n^N$ moves down and the optimal number of inputs $(N_n)^*$ moves to the left.

One disadvantage of this method is that we need to estimate $E\{S_n^N\}$ and, in some cases, we have to use a lot of computing time to obtain a solution. Next section explains how to avoid this kind of problems using *learning automata technique*. It provides some sort of "adaptation" within a process of structure tuning. Learning Automata are the heart of this approach.

5. Learning Automata

A learning automaton (see, [12]) is a system

$$\Sigma = \{\Xi, X, U, P, \Re\} \tag{15}$$

where Ξ is the set of inputs (environment outputs) ξ_n at time n; $X=\{x_a(1),.....,x_a(K)\}$ is the set of states; $U=\{u(1), ...u(N)\}$ is the set of outputs; P is the set of probability vectors $p_n=p_n(1),.....,p_n(N) \in P$, the probability $p_n(i)$ corresponds to a probability to select the output $u_n=u(i)$, $i=1,....,N$ at time n; $\Re$ is the reinforcement scheme

$$\Re: \; p_n \rightarrow p_{n+1}$$

responsible for the change of this probability vector and is based on given measurements to obtain the best behavior in a given environment. Here we shall consider static automata (finite systems).

Usually, the relation between the environment outputs $\xi_n=\xi_n(u_n, \omega)$ (ω is a random factor) and the loss function Φ_n associated with a learning automaton is given by the following expression:

$$\Phi_n = \frac{1}{n}\sum_{t=1}^{n} \xi_t \tag{16}$$

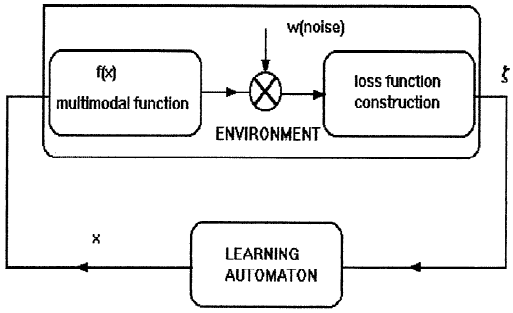

Figure 3 Learning Automata and its environment.

The problem to be solved in Learning Automata Theory can be formulated as follows: *Find a reinforcement scheme which generates the sequence $\{p_n\}$ and consequently $\{u_n\}$, and ensures asymptotically the minimization of the loss function (17), i.e.*

$$\limsup_{n\rightarrow\infty} \Phi_n \rightarrow \inf_{\{u_n\}} \tag{17}$$

in some probabilistic sense (fig. 3). In the next section it is explained in detail one of the reinforcement schemes used in this paper. Let us consider one of the popular reinforcement schemes developed by Bush and Mosteller (1958), which is described by the following recurrent algorithm:

$$p_{n+1} = p_n + \gamma_n\left[e(u_n) - p_n + \xi_n(e^N - Ne(u_n))/(N-1)\right] \tag{18}$$
$$p_1(i) > 0, \qquad (i = 1,.....N)$$

where: $\xi_n \in [0,1]$, $e(u_n) = (\underbrace{0,0,....,1}_{u_n},.....0,0)^T$,

$$e^N = (1,1,1,...,1,1)^T \quad \in \quad \Re^N$$

where u_n is generated according to the probability distribution p_n for the available data; and $\gamma_n \in (0,1)$ is the correction factor (adaptation gain) of this scheme.

The results concerning the convergence properties of this scheme for non-binary (continuous) and non-stationary continuous environment reactions can be found in [16] [17].

6. Optimization of the Joint Loss Function

This sections deals with the application of learning automata to the problem of multimodal function optimization based on the approach derived in [12] and [17]. and explains the relation of this problem to the neural networks synthesis.

Let $r(x)$ be some real valued function of a vector parameter $x \in X \subset \Re^M$ (X is a compact in $\Re^M$) and let that $r(x)$ be a multimodal function. We wish to find the value $x=x^*$ which minimizes $r(x)$. Let us now consider a quantification $\{X_i\}$ of the admissible region X, i.e.

$$X_i \subset X,\ \ X_i \cap X_{j\neq i} = \varnothing,\ \ i,j = 1,\ldots\ldots,N$$
$$\bigcup_{i=1}^{N} X_i = X \subset \Re^M \tag{19}$$

Let q_n be the observation of $r(x_n)$ disturbed by the noise w_n, such that

$$q_n = r(x_n) + w_n \tag{20}$$

where $x_n \in \{x(1),\ldots,x(N)\}$, $x(i) \in X_i$ are some fixed points and w_n is a random variable which characterizes the noise of the observation. The central idea associated with the use of a learning automaton is connected to the manner of constructing the inputs of this automaton. The automaton input ξ_t at time t is constructed following the next *"normalizing procedure"* [12]:

$$\xi_t = \frac{\left[s_t(i) - \min_j s_{t-1}(j)\right]_+}{\max_K \left[s_t(k) - \min_j s_{t-1}(j)\right]_+ + 1},\quad u_t = x(i) \tag{21}$$

where

$$s_t(i) = \frac{\sum_{l=1}^{n} q_l \chi(u_l = x(i))}{\sum_{l=1}^{n} \chi(u_l = x(i))},\quad i = 1,\ldots,N \tag{22}$$

and

$$[x]_+ := \begin{cases} x, & x \geq 0 \\ 0 & x < 0 \end{cases},\ \chi(u_l = u(i)) := \begin{cases} 1, & if\ u_l = u(i) \\ 0, & if\ u_l \neq u(i) \end{cases} \tag{23}$$

Let us now use the learning automaton described above to select the point $x(\alpha)$ which corresponds (in some sense) to the minimum value of the multimodal function $r(x_n)$ on the set X. This process can be organized as shown in fig 3.

To adapt this approach to our structure optimization problem we have to select $q_n = D_n^N$, $u_n = N_n$ where N_n is the number of nodes selected in a PANN model at time n. So, in our case q_n represents the joint loss function $D_{,n}^N$ and u_n the order of the model to be selected. Notice, that here we have only to select one point of x_n at each time n (each iteration) to calculate the minimal value of the function $r(x)$. Finally, generating the random sequence $u_n = N_n$ according to the probability distribution p_n, which *is* changing in time as in reinforcement scheme (18) with the normalizing procedure (21).

In this case we use two layers of learning automaton to avoid convergence problems.

Suggested algorithm for every layer consists of the following steps:

1. Generating the control signal u_n using the given distribution p_n and a given number of observations.
2. Calculating the realization of the functions D_n^N : only for the point $N_n = u_n$.
3. Return to step 2 until one of the probabilities tends to 1.

In this situation during this adaptation process we select often and often the number of nodes N_t which corresponds to the minimal point of the performance index D_n^N to be minimized. The probability that tends to 1 when $t \to \infty$ corresponds to the value $\left(N_n^{ij}\right)^*$ which represents the argument that minimizes the loss function D_n^N, e.g. $\left(N_n^{ij}\right)^*$ is *the optimal number of nodes.*

7. Illustration Example

For the numerical example we select the lorenz equations (Fig. 4) to be approximated by PANN. With ndo=3 and σ_ζ=1 Fig. 5 shows the evolution of the component p_n which corresponds to the best number of nodes. The simulation results show that after 500 steps of a learning procedure, the reinforcement scheme (19) selects only the control action $u_n = 17$ (See Histogram for the control action). (Remember that the automaton is of two layers and u_n is the value of the first interval more the result of the second interval). It means that the *learning process* is practically finished.

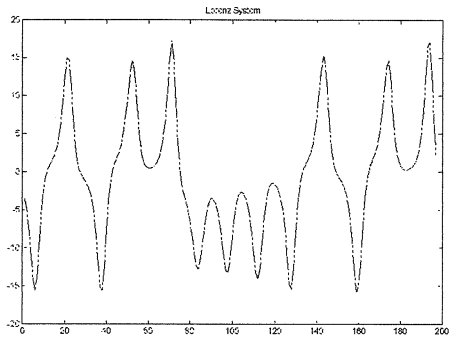

Figure 4 Original function *f(i).*

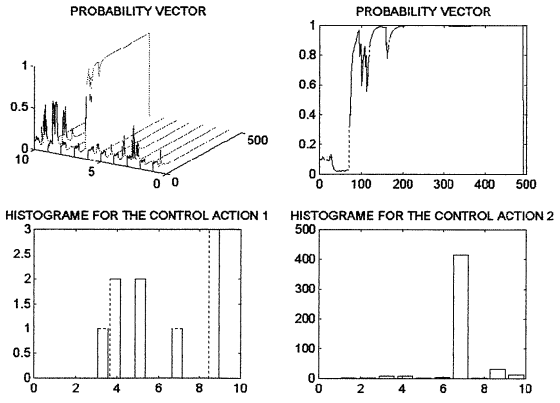

Figure 5 Probability and Criterion using σ_ζ=1

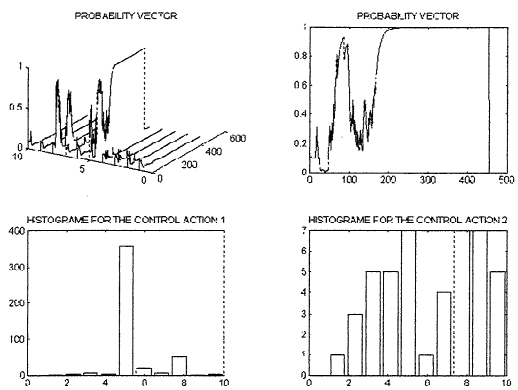

Figure 6 Probability and Criterion using σ_ζ=10.

The curve at fig. 5, corresponding to the loss function evaluation Φ_n with ξ_k defined by (21)-(22), shows its decrement. Then we change the value of σ_ζ to demonstrate that the best number of nodes change with the disturbances increment. As it is shown in fig. 6 for σ_ζ=10 the control action u_n is 5. In the case when the noise variance increases, the best number of nodes is less than in the previous case.

8. Conclusion

A learning system has been proposed to obtain the optimal number of nodes in ANN using loss function criterion (of the Akaike type) and Bush-Mosteller reinforcement scheme.

The research analysis presented in this paper clarified the following properties of ANN:

- if physical devices which realize a given ANN are "non ideal" (produce any distortions at inputs or outputs), then there exists an optimal number of nodes in Artificial Neural Networks, which can be detected by the learning process suggested here;

9. References

[1] Albrecht, T.; Matz, G. & Hildermann, J. "An Intelligent gas sensor system for the identification of hazardous airborne compounds using an array of semiconductor gas sensors and Kohonen Feature Map Neural Networks. *Intelligent Systems Engineering*, 5-9 September, 1994. Conference Publication No. 395, 1994

[2] Alfonzetti, Coco, Cavalieri. & Malgeri., Automatic Mesh Generation by the Let-It-Grow Neural Network. *IEEE Trans. on Magn.*, Vol. 32 No.3 May 1996.

[3] Barron, A. R., Universal Approximation in Bounds for Superpositions of a Sigmoidal Function. *IEEE Trans. on Information Theory*. Vol. 39 No. 3 May, 1993

[4] Bebis, G & Georgiopoulos. "Feed-forward neural netowrks". *IEEE Potentials*, October/November, 1994

[5] Bhat, N. and T. J. Mc-Avoy. "Use of Neural Nets for Dynamic Modeling and Control of Chemical Process Systems". *Computers Chem.Eng.*, No.14, pp.573-583, 1990.

[6] Chang, T. & Abdel-Ghaffar, K., A Universal Neural Net with Guaranteed Convergence to Zero System Error. *IEEE Transactions on signal processing*, Vol. 40, no. 12, December 1992.

[7] Govind, G. & Ramamoorthy, P., An Adaptive-Topology Neural Architecture and Algorithm, for Nonlinear System Identification. *IEEE ICNN*, 28 march-1 april, San Francisco Ca, USA, 1993

[8] Huang, Q. & Liu, R. a Neural Computation for Canonical Representations of nonlinear Functions, *IEEE International Symposium on Circuits and Systems*, New Orleans, USA, 1-3 May, 1990

[9] Huang, S.C. and Y.F. Huang, "Bounds on the number of hidden neurons in multilayer perceptrons", *IEEE Trans. Neural Networks*, Vol. 2 pp. 47-55,1991.

[10] Ljung & Soderstrom, 1983. Theory and Practice of Recursive Identification. MIT Press, Cambridge, MA.

[11] Marchesi, M.; Orlandi, G. Piazza, F & Uncini A., Neural Networks with self-adaptive topology, *IEEE International symposium on Circuits and Systems*, Singapore,11-14 June 1991

[12] Najim, K. & Poznyak, A.S., Learning Automata: Theory and Applications (Oxford, U.K.: Pergamon Elsevier Sciences). 1994,

[13] Narendra & Thathachar, M. A. L., 1989, Learning Automata- An Introduction (Englewood Cliffs, NJ:Prentice Hall)

[14] Narendra, K. S. and K. Parthasarathy. "Identification and Control of Dynamical Systems Using Neural Networks". *IEEE Trans. Neural Networks*, No.1, pp.4-27, 1990.

[15] Poznyak, A. S, K. Najim and M. Chtourou. Use of recursive stochastic algorithm for neural networks synthesis. *Appl. Math. Modelling*, vol.17, August, pp. 444 - 448, 1993.

[16] Poznyak, A. S., K. Najim & E. Ikonen. "Adaptive Selection of the optimal order of linear regression models using learning automata". *Inter. Journal of Systems Science*, 1996, volume 27, number 1, pages 151-159.

[17] Poznyak, A. S., K. Najim & M. Chtourou. "Learning Automata with continuous inputs and their application for multimodal functions optimization". *Inter. Journal of Systems Science*, 1996, vol. 27, number 1, pages 87-95.

[18] Poznyak, A.S. & Gómez-Ramírez, E., 1994, How to select the number of nodes in artificial neural networks. AMCA/IEEE Int. Workshop on Neural Networks Applied to Control and Image Processing, Mexico City.

[19] Syam, M. M., A Neural Expert System for Diagnosing Eye diseases. *Proceedings of the Tenth Conference on Artificial Intelligence for Applications*, San Antonio TX, USA, 1-4 March, 1994.

[20] Verulava, Yu. Sh. and B. T. Polyak. "Selecting the order of a regression model". *Avtomatika i Telemekhanika*, No.11, pp.113 - 129,1988.