Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 8, August 2015, pg.475 – 480

RESEARCH ARTICLE

Evolving Flexible Neuro-Fuzzy System for Medical Diagnostic Tasks

O. Turuta[1], I. Perova[2], A. Deineko[3]

[1] PhD., Assoc. Prof, Software Engineering Department, Kharkiv University of Radio Electronics, Ukraine
[2] PhD., Assoc. Prof, Biomedical Engineering Department, Kharkiv University of Radio Electronics, Ukraine
[3] PhD., Assoc. Prof, Control Systems Research Laboratory, Kharkiv University of Radio Electronics, Ukraine
[1] oleksii.turuta@nure.ua; [2] Rikywenok@gmail.com; [3] anastasiya.deineko@gmail.com

Abstract — In the paper architecture and training method for evolving flexible diagnostic neuro-fuzzy-system are investigated. The proposed system is simple in numeric realization and characterized by a high learning rate and flexibility, that make possible to use it in conditions of small training sets and on big data sets, coming to processing in online-mode.

Keywords — Medical Data Mining, Data Stream Mining, diagnostic neuro-fuzzy system (DNFS)

I. INTRODUCTION

Nowadays methods of Computational Intelligence [1-8] are widely used for solution of many different tasks in industry, agriculture, banking sector etc. This methods are used in medical applications too and this approach now is known as Medical Data Mining [9-11].

For various Data Mining tasks, connected with diagnostics, classification, clusterization, pattern recognition etc. nowadays methods of Computational Intelligence, firstly Soft Computing and Machine Learning are widely used.

Ones of the most effective are neuro-fuzzy systems because of its learning abilities, including self-learning, universal approximative capacities, linguistic interpretability and “transparency” of results. ANFIS and TSK-systems of different order, like approximators and extrapolators, and NEFCLASS [12] with its different modifications, oriented for classification (pattern recognition) tasks solving have the widest spread.

But there exist a broad class of tasks where these systems are not effective. Primarily, there are the tasks where training set is short, data sets are fed to processing sequentially, in the form of data stream [13] and learning of system has to realize in parallel with analysis of input information.

This situation often appears in Medical Data Mining tasks [9, 10] and complicated by the fact that data set under processing is nonstationary and dimensionality of input features space can be comparable with size of training data set. When it comes to diagnosis task, firstly, data set can have very low size in situation of rare diagnosis, and secondly, quantity of possible diagnosis (especially in situation of screening programs) can change during analysis. Naturally, that traditional diagnostic neuro-fuzzy systems like NEFCLASS can not overcome there problems.

II. FAST DIAGNOSTIC NEURO-FUZZY SYSTEM.

Let's consider architecture of diagnostic neuro-fuzzy system (DNFS) that consist of six sequentially connected layers (fig.1) [14]. Here (n×1) input vector of signals-attributes $x(k)=(x_1(k), x_2(k),...x_n(k))^T \in R^n$, where $k=1,2,...$ is current time, is fed in input layer of system. First hidden layer of system contains nh membership functions $\mu_{li}(x_i(k))$, $i=1,2,..., n$; $l=1,2,..., h$ and provides fuzzification of input feature space.

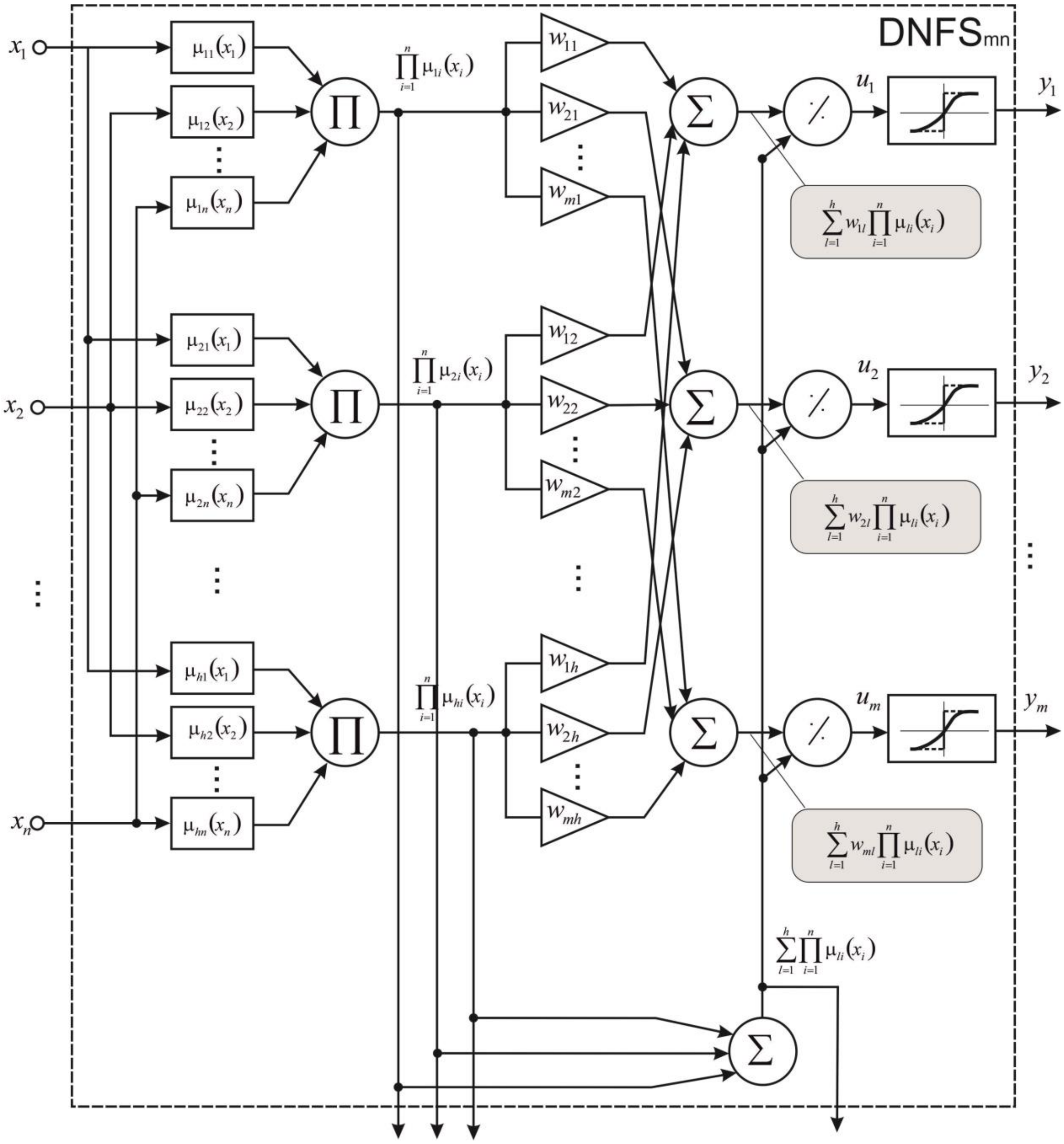

Fig.1 – Diagnostic neuro-fuzzy system DNFSmn with n inputs and m outputs

Because of this in system scatter partitioning of feature space is realized as a membership functions standard bell shape functions with unlimited supports are used. Most often they are traditional Gaussians or more exotic functions, for example, derivatives of tangent hyperbolic function.

Second hidden layer realizes aggregation of membership levels, calculated in first layer, and consist of *h* simple multipliers. Third hidden layer is a layer of synaptic weights w_{jl} ($j=1,2,..., m$ – number of possible diagnosis taken on the basis of empiric consideration) which have to be adjusted in training process. It is the most “responsible” layer of DNFS because effectiveness of whole system depends of precision and speed of training.

Common quantity of synaptic weights equals to mh. Fourth hidden layer is formed by $m+1$ adders, which realize elementary operations. In fifth hidden layer, formed by m division units, defuzzification of “gravity center” type is realized. And at last output (sixth) layer contains m nonlinear activation functions. In diagnostics task simple signum function is often used, which takes value +1 in case of true diagnosis and -1 – in other case. That’s why output signal of DNFS $y_j(k)$ can take only two values ±1.

When feature vector $x(k)\in R^n$ becomes on input of system, in output of first hidden layer hn values of $\mu_{li}(x_i(k))$ are appear, in output of second hidden layer – h signals $\prod_{i=1}^{n}\mu_{li}(x_i(k))$, in output of third hidden layer – mh values $w_{jl}\prod_{i=1}^{n}\mu_{li}(x_i(k))$, output of fourth layer – $m+1$ signals: $\sum_{l=1}^{h}w_{jl}\prod_{i=1}^{n}\mu_{li}(x_i(k))$ and $\sum_{l=1}^{h}\prod_{i=1}^{n}\mu_{li}(x_i(k))$, fifth layer – $u_j(k)=\frac{\sum_{l=1}^{h}w_{jl}\prod_{i=1}^{n}\mu_{li}(x_i(k))}{\sum_{l=1}^{h}\prod_{i=1}^{n}\mu_{li}(x_i(k))}=\sum_{l=1}^{h}w_{jl}\frac{\prod_{i=1}^{n}\mu_{li}(x_i(k))}{\sum_{l=1}^{h}\prod_{i=1}^{n}\mu_{li}(x_i(k))}=\sum_{l=1}^{h}w_{jl}\varphi_l(x(k))=w_j^T\varphi(x(k))$ and sixth – m diagnostics signals $y_j(k)=sign\,u_j(k)$.

So, system under consideration is a modification of neuro-fuzzy system of Wang-Mendel [15] and intended for solving of diagnostic-classification tasks.

For providing the learning process more flexibility in sixth layer we can introduce, instead of signum functions, activation function of hyperbolic tangent type that are often used in neural networks:

$$y_j(k)=\tanh\gamma u_j(k)=\frac{1-e^{-2\gamma u_j(k)}}{1+e^{-2\gamma u_j(k)}},$$

where gain parameter γ increasing leads to approaching of function $\tanh\gamma u_j$ to $sign\ u_j$ without derivative discontinuity.

Using standard quadratic criterion of training

$$E_j(k)=\frac{1}{2}e_j^2(k)=\frac{1}{2}\left(d_j(k)-\tanh\gamma w_j^T\varphi(x(k))\right)^2=\frac{1}{2}\left(d_j(k)-\tanh\gamma u_j(k)\right)^2$$

we can write standard δ -rule of Rosenblatt’s perceptron training

$$w_j(k+1)=w_j(k)+\eta(k)e_j(k)\gamma\left(1-y_j^2(k)\right)\varphi(x(k))=w_j(k)+\eta(k)\delta_j(k)\varphi(x(k)),\qquad(1)$$

where $\eta(k)>0$ – learning rate parameter, $\delta_j(k)$ – δ -error of training for *j*-th output at *k*-th time iteration.

Using ideas of quasi-Newtonian learning [16] we can introduce optimized variation of (1) like [18]:

$$w_j(k+1)=w_j(k)+\frac{\delta_j(k)\varphi(x(k))}{\eta+\|\varphi(x(k))\|^2}$$

or in matrix form like (2):

$$W(k+1)=W(k)+\frac{\delta(k)\varphi^T(x(k))}{\eta+\|\varphi(x(k))\|^2},$$

when $\eta=0$

$$W(k+1)=W(k)+\delta(k)\varphi^{+}(x(k)),\qquad(2)$$

where $\delta(k)=(\delta_1(k),\delta_2(k),\ldots,\delta_m(k))^T$

$$\delta_j(k)=e_j(k)\gamma\left(1-y_j^2(k)\right)=\left(d_j(k)-\tanh\gamma u_j(k)\right)\gamma\left(1-\left(\tanh\gamma u_j(k)\right)^2\right).$$

To improve approximative characteristics of system under consideration we can introduce learning contour of γ -parameter which sets a form ("steepness") of activation function although usually this parameter is supposed constant. To realize this task we can use approach, proposed in [17], where gradient optimization using is proposed for adjustment of this parameter.

Writing j-th output of system in the form:

$$y_j(k)=\tanh\gamma_j u_j(k)=\frac{1-e^{-2\gamma_j u_j(k)}}{1+e^{-2\gamma_j u_j(k)}},$$

and learning criterion–

$$E_j(k)=\frac{1}{2}\left(d_j(k)-\tanh\gamma_j w_j^T\varphi(x(k))\right)^2=\frac{1}{2}\left(d_j(k)-\tanh\gamma_j u_j(k)\right)^2,$$

we can instead traditional δ -learning rule (1) to write modified algorithm

$$\begin{cases} w_j(k+1)=w_j(k)+\eta_w(k)e_j(k)\gamma_j(k)\left(1-y_j^2(k)\right)\varphi(x(k))= \\ =w_j(k)+\eta_w(k)\gamma_j(k)\delta_j(k)\varphi(x(k)), \\ \gamma_j(k+1)=\gamma_j(k)+\eta_\gamma(k)e_j(k)\left(1-y_j^2(k)\right)u_j(k)= \\ =\gamma_j(k)+\eta_\gamma(k)\delta_j(k)u_j(k)=\gamma_j(k)+\eta_\gamma(k)\delta_j(k)w_j^T(k)\varphi(x(k)). \end{cases} \tag{3}$$

Introducing compartmental vector of turned parameters $\tilde{w}_j^T(k)=\left(w_j^T(k),\gamma_j(k)\right)$ and signals $\varphi_j(x(k))=\left(\gamma_j(k)\varphi^T(x(k)),u_j(k)\right)$ it's easy to introduce the learning algorithm of all parameters of output perceptrons in the form

$$\tilde{w}_j^T(k+1)=w_j(k)+\eta(k)\delta_j(k)\varphi_j(x(k))$$

or its optimized modification

$$\begin{cases} \tilde{w}_j^T(k+1)=\tilde{w}_j(k)+\dfrac{\delta_j(k)\varphi_j(x(k))}{\eta+\left\|\varphi_j(x(k))\right\|^2}, \\ \tilde{w}_j^T(k+1)=\tilde{w}_j^T(k)+\delta_j(k)\varphi_j^+(x(k)). \end{cases} \tag{4}$$

During the learning, in each output of our system value $\gamma_j(k)$ can be different. That's why separately adjustment of output perceptrons using (4) is more preferable than matrix procedure (3).

III. EVOLVING DIAGNOSTIC NEURO-FUZZY SYSTEM

Diagnostic system under consideration is designed to be used in condition, when quantity of diagnostic features *n* and diagnosis quantity *m* is fixed, that is natural for neural networks and neuro-fuzzy-systems, whose architecture is set a priori during synthesis.

In real medical tasks during training new diagnosis can appear, those was not previously involved. To enlarge quantity of possible diagnosis we can use ideas of evolving systems of computational intelligence [19, 20], that can tune their parameters and architecture. Architecture of evolving system $DNFS_{m+1,n}$ with *n* inputs and *m*+1 outputs is shown in Figure 2.

It is based on $DNFS_{mn}$ system, shown on Fig.1,neuro-fuzzy-element NFE was added, containing *h* synaptic weights $w_{m+1,l}$, one adder (summation block), one divider and activation function $\tanh\gamma w_{m+1}^T\varphi(x(k))$.

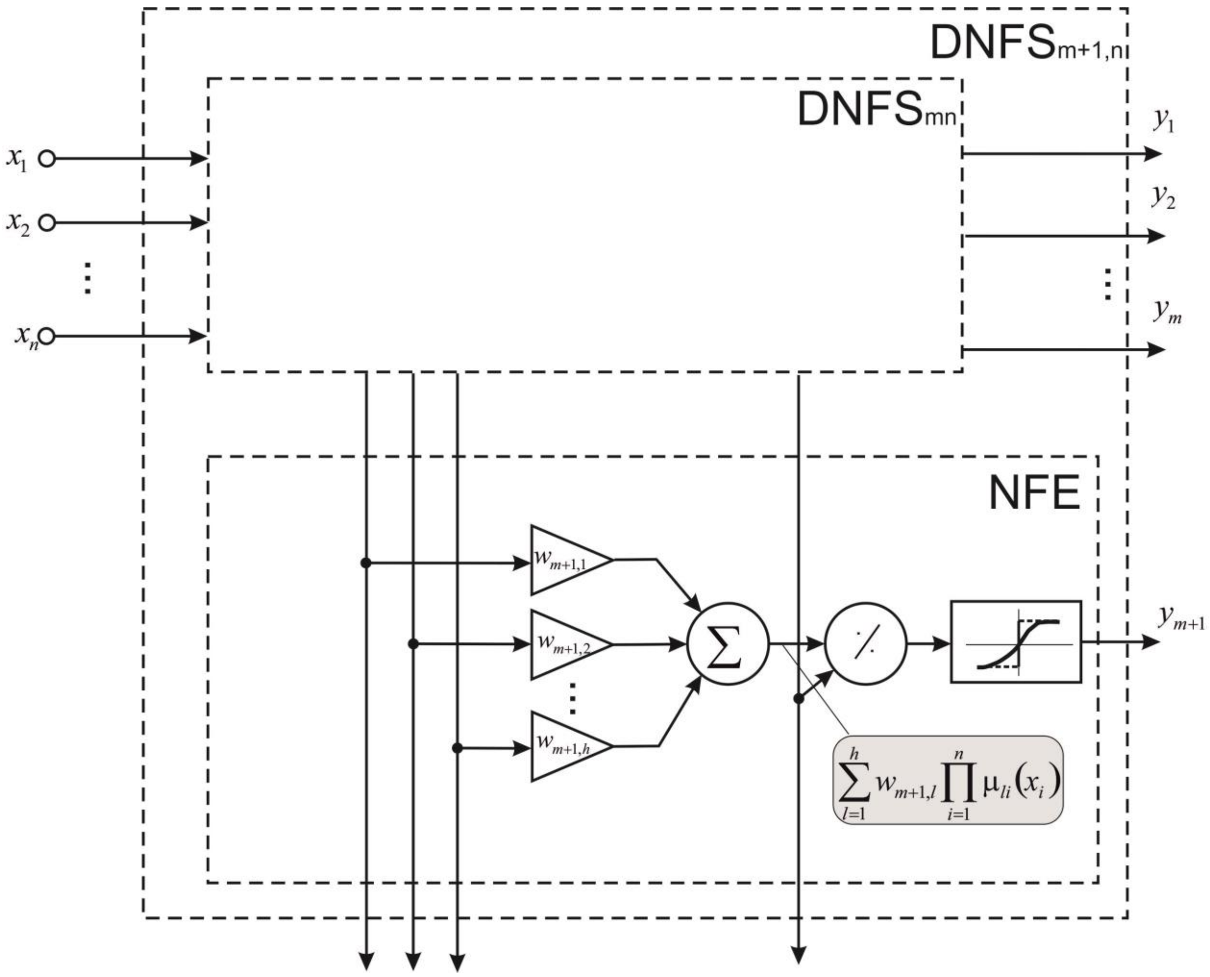

Fig.2 – Evolving diagnostic neuro-fuzzy system with n inputs and m+1 outputs (DNFSm+1,n)

Rearranging training algorithm (3) for $DNFS_{mn}$ in the form

$$W^m(k+1) = W^m(k) + \delta^m(k)\varphi^+(x(k)),$$

we can introduce algorithm for $DNFS_{m+1,n}$:

$$W^{m+1}(k+1) = \begin{pmatrix} W^m(k+1) \\ ------- \\ w_{m+1}^T(k+1) \end{pmatrix} = \begin{pmatrix} W^m(k) \\ ---- \\ w_{m+1}^T(k) \end{pmatrix} + \begin{pmatrix} \delta^m(k) \\ ---- \\ \delta_{m+1}(k) \end{pmatrix} \varphi^+(x(k)).$$

Easy to see, that including of new NFE blocks in extended diagnostic system does not change original $DNFS_{mn}$ training.

In case when together with synaptic weights w_{m+1} a setting of γ_{m+1} parameter occurs, NFE-element of evolving system is turned using second recurrent expression (4)

$$\tilde{w}_{m+1}^T(k+1) = \tilde{w}_{m+1}^T(k) + \delta_{m+1}(k)\varphi_{m+1}^+(x(k)).$$

Using of turned activation function provides to system under consideration more flexibility and allows to improve its diagnostic capabilities.

IV. Conclusion

In this paper architecture and training method for evolving flexible diagnostic neuro-fuzzy-system are proposed. This system is designed for broad class of Data Stream Mining tasks solving, especially Medical Data Mining ones in online mode in situations of unknown quantity of possible diagnosis, that can change during training-diagnostics processes. Proposed system is simple in numeric realization and characterized by a high learning rate and flexibility, that make possible to use it in conditions of small training sets and on big data sets, coming to processing in online-mode.

REFERENCES

[1] Jang J.-S.R. *Neuro-Fuzzy and Soft Computing* / J.-S.R. Jang, C.-T. Sun, E. Mizutani. – Prentice Hall, Upper Saddle River, NJ. – 1997. – 640 p.

[2] Friedman J. *The Elements of Statistical Learning*. *Data Mining, Inference and Prediction* / Friedman J., Hastie T., Tibshirani R. – Berlin: Springer, 2003. – 552p.

[3] Osowski S. *Sieci neuronowe do przetwarzania informacji*. – Warszawa: Oficina Wydawnicza Politechniki Warszawskiej, 2006. – 422s.

[4] L. Ruthowski, *Computational Intelligence. Methods and Techniques*, Berlin-Heidelberg: Springer-Verlag, 2008. – 514 p.

[5] Mumford C.L., Jain L.C. *Computational Intelligence. Collaboration, Fusion and Emergence*. – Berlin: Springer-Verlag, 2009. – 725p.

[6] Kruse R., Borgelt C., Klawonn F., Moewes C., Steinbrecher M., Held P. *Computational Intelligence. Methodological Introduction*. – Berlin: Springer-Verlag, 2013. – 488p.

[7] Kroll A. *Computational Intelligence. Eine Einführung in Probleme, Methoden und technische Anwendungen*. – Munchen: Oldenbourg Verlag, 2013. – 428 S.

[8] K.-L. Du, M.N.S. Swami, *Neural Networks and Statistical Learning*, London: Springer-Verlag, 2014. – 824 p.

[9] Rizzo R. *Computational Intelligence Methods for Bioinformatics and Biostatistics* / Rizzo R. - In Lecture Notes in Bioinformatics (7th International Meeting, CIBIB 2010, Palermo, Italy, September 16-18, 2010). – Springer, 2011. – 301 p.

[10] Kountchev R. *Advances in Intelligent Analysis of Medical Data and Decision Support Systems* (Studies in Computational Intelligence) / Kountchev R. and etc - Springer-Verlag, Berlin Heidelberg. – 2013. – 246 p.

[11] Mulesa P., Perova I. *Fuzzy Spacial Extrapolation Method Using Manhattan Metrics for Tasks of Medical Data Mining* // Computer Science and Information Technologies CSIT’2015. – Lviv, Ukraine. – 2015. – P.104-106.

[12] Nauck D., Kruse R. *NEFCLASS - a neuro-fuzzy approach for the classification of data*/Eds. K.M. George e.a. Proc. of the 1995 ACM Symposium on Applied Computing, “Applied Computing 1995”Nashville, 1995. – P. 461-465.

[13] Bifet A. *Adaptive Stream Mining. Pattern Learning and Mining from Evolving Data Streams*. – Amsterdam: IOS Press, 2010. – 224p.

[14] BodyanskiyYe.V., VynokurovaO.A., Mulesa P.P., Perova I.G. *Diagnostic neuro-fuzzy system and its adaptive learning algorithm in the medical and biological data mining tasks*. – System technologies. N 2(91) - Dnipropetrovsk, 2014. – P.125 – 135.

[15] Wang L.X. *Fuzzy basis functions, universal approximation, and orthogonal least squares learning* / L.X. Wang, J.M. Mendel // IEEE Trans. on Neural Network. – 1992. – 3. – P. 807-814.

[16] Shepherd A.J. *Second-Order Methods for Neural Networks* / A.J. Shepherd. – London: Springer-Verlag. – 1997. – 145 p.

[17] Kruschke J.K., Mevellan J.R. B*enefits of gain: speeded learning and minimal layers backpropagation networks* // IEEE Trans. on Systems, Man, and Cybernetics. – 1991. – 21. – P. 273-280.]

[18] Otto P., Bodyanskiy Ye., Kolodyazhniy V. *A new learning algorithm for forecasting neural-fuzzy network* // Integrated Computer-Aided Engineering. – 2003. – 10. – №4. – P. 399–409.

[19] Kasabov N. *Evolving Connectionist Systems: Methods and Applications in Bioinformatics, Brain Study and Intelligent Machines.* – London: Springer-Verlag, 2003. – 307p.

[20] Lughofer E. Evolving Fuzzy Systems – *Methodologies, Advanced Concept and Applications*. – Berlin-Heidelberg: Springer-Verlag, 2011. – 454p.

