UDC 004.032.26

Y. Viktorov, Y. Bodyanskiy, A. Dolotov
Kharkiv National University of Radio Electronics, CS RL

SOLVING APPROXIMATION AND FORECASTING PROBLEMS USING DOUBLE ORTHO-NEURON

У статті розглянуто нову нетрадиційну нейромережеву архітектуру – подвійний орто-нейрон. Запропоновано алгоритм навчання на основі процедури оптимізації другого порядку. Вказано переваги розглянутої конструкції порівняно з класичними нейронними мережами. Надано результати імітаційного моделювання.

In this paper new non-conventional neural network architecture called double ortho-neuron is considered. Learning algorithm based on optimization procedure of the second order is proposed. The advantages of this construction are pointed out. Simulation results are given.

Introduction

Nowadays there are many methods and methodologies for process approximation and forecasting. Almost all of them are based on two basic approaches: heuristical and mathematical [1].

Heuristical approach is based on using of the experts knowledge in the current subject field. This is the oldest approach for process forecasting and approximation and it has following disadvantages: excessive subjectivity, complexity, labour-intensiveness etc. So these methods usually applied only for processes which cannot be formalized or their formalization requires substantial efforts.

At the present time mathematical methods are often used for solving process approximation and forecasting problems on account of objectivity of obtained information, high precision of received results (if the model has been chosen correctly) and possibility of the process automation. Artificial neural networks technique is one of the most attractive technologies within mathematical approach.

Nowadays artificial neural networks are widely applied for solving a variety of problems concerning to nonstationary, nonlinear signal processing and analysis in conditions of current and prior uncertainty. The most popular are multilayered architectures with sigmoidal, spline or bell-shaped activation functions in their nodes, such as perceptrons, radial-basis function neural networks, wavelet neural networks, neo-fuzzy systems etc [2].

But all these architectures have well known disadvantages, such as significant computational complexity during training process, lack of criteria of network architecture selection (quantity of nodes in each layer) for solving specified problem and so on.

Alternative approach for nonlinear functions approximation is using orthogonal polynomials [3]. It can be useful to reduce computational complexity and to increase accuracy of forecasting.

1. Artificial neural networks with orthogonal activation functions.

Elementary one-dimensional system described in “input-output” space of some unknown functional dependence $y(x)$ can be expressed by the following sum:

$$y(x) = w_0\varphi_0(x) + w_1\varphi_1(x) + ... + w_h\varphi_h(x) = \sum_{j=0}^{h} w_j\varphi_j(x), \tag{1}$$

where x and $y(x)$ are input and output variables of the estimated process correspondingly, $\varphi_j(x)$ – orthogonal polynomial of the j-th order ($j = 0, 1, 2,\dots, h$), which possesses the orthogonality property

$$\sum_{k=1}^{N} \varphi_j(x(k))\varphi_q(x(k)) = 0, \forall j \neq q, \tag{2}$$

j, q, – non-negative integer numbers, k = 1, 2,…,N – current discrete time or the ordinal number of an element in the sampling.

Equation (1) can be realized by the elementary scheme shown at the fig. 1 and called the ortho-synapse.

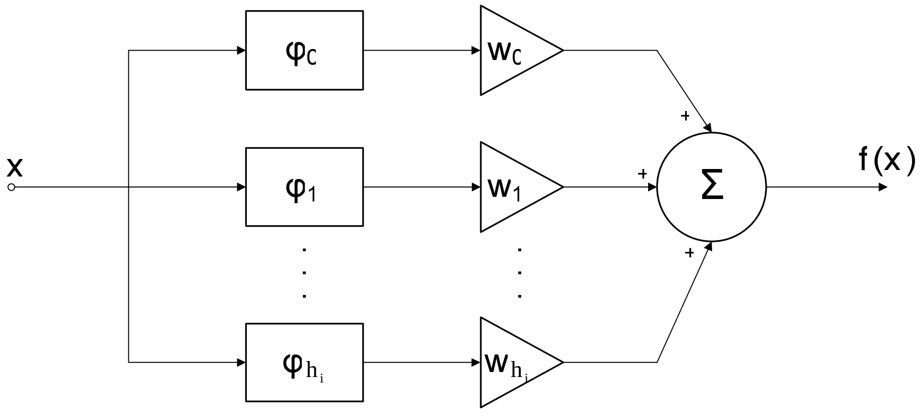


Fig 1. The ortho-synapse – OS_i

At the fig. 1 x_i is the i-th (i = 1, 2,…, n) component of the multidimensional input signal $x = (x_1, x_2,\ldots, x_n)^T$, w_{ji} (j = 1, 2,…,n) – synaptic weights which should be determined. Output signal of the ortho-synapse can be expressed in the form

$$f_i(x) = \sum_{j=0}^{h_i} w_{ji} \varphi_{ji}(x_i). \tag{3}$$

Ortho-synapse has the same architecture like a nonlinear synapse of the neo-fuzzy-neuron [4-5] but provides smooth polynomial approximation instead of piecewise-linear approximation.

Using ortho-synapse (3) more complex architecture can be introduced like the ortho-neuron shown at the fig. 2.

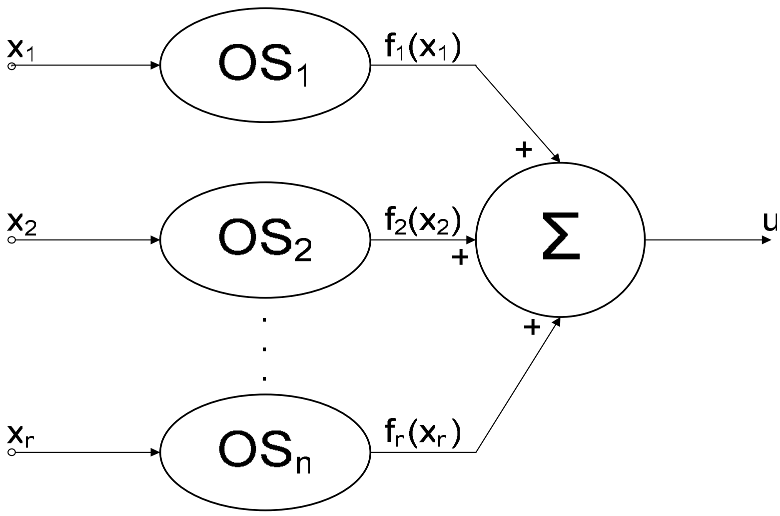


Fig 2. Ortho-neuron – On_i

Ortho-neuron which has the same architecture like a neo-fuzzy-neuron [4-5] realizes, the mapping

$$u = \sum_{i-1}^{n} f_i(x_i) = \sum_{i=1}^{n} \sum_{j=0}^{h_i} w_{ji} \varphi_{ji}(x_i), \tag{4}$$

and provides high precision approximation of nonlinear nonstationary signals and processes [6-9].Ortho-neuron has demonstrated good results quality and high rate of convergence in solving stochastic and chaotic process forecasting problem [10].

Goal of this work is improving approximation properties and forecasting quality in conditions of uncertainty about process properties and disturbances affecting on them.

2. Double ortho-neuron

Let us introduce double ortho-neuron (DON) shown at fig. 3 which has the same architecture like a double wavelet-neuron [11] but differs from it with used activation functions.

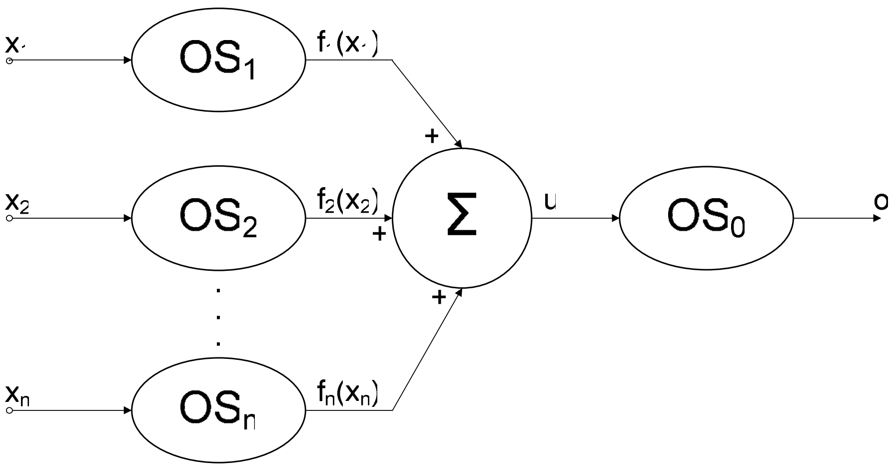


Fig 3. Doube ortho-neuron – DON

Double ortho-neuron realizes the nonlinear mapping

$$o = f_o(u) = \sum_{l=0}^{h_0} w_{lo}\varphi_{lo}(u) = \sum_{l=0}^{h_0} w_{lo}\varphi_{lo}\left(\sum_{i=1}^{n}\sum_{j=0}^{h_i} w_{ji}\varphi_{ji}(x_i)\right), \quad (5)$$

where w_{ji}, w_{lo} – tunable synaptic weights in the hidden and output layers correspondingly, $\varphi_{ji}, \varphi_{lo}$ – activation functions in ortho-synapses of the hidden and output layers correspondingly, h_i (i = 0, 1, 2,…, n) – dimensionality of the appropriate ortho-synapse, n – dimensionality of the input signal, $x_i(k)$ – value of i-th input signal's component at the time moment k (or for k-th training sample).

Double ortho-neuron contains $h = \sum_{i=0}^{n} h_i$ tunable parameters and, as it can be readily seen, doesn't subjected to the "curse of dimensionality".

Double ortho-neuron uses orthogonal polynomials of one variable for the activation functions. Particular system of functions can be chosen accordingly to the specificity of the solved problem. If the input data is normalized on the hypercube [-1, 1]^n, the system of Legendre polynomials orthogonal on the interval [-1, 1] with weight $\gamma(x) = 1$ can be used:

$$P_j(x_i) = 2^{-j}\sum_{p=0}^{[j/2]}(-1)^p \frac{(2j-2p)!}{p!(j-p)!(j-2p)!} x_i^{j-2p}, \quad (6)$$

where $[\bullet]$ – is the integer part of a number.

Also to simplify calculations we can exploit recurrence formula

$$P_{j+1}(x_i) = \frac{2j+1}{j+1} x_i P_j(x_i) - \frac{j}{j+1} P_{j-1}(x_i). \quad (7)$$

System of Legendre polynomials is best suited for the case when we exactly know interval of data changes before network construction. This is quite common situation as well as an opposite one. For the other case the following system of Hermite orthogonal polynomials can be used:

$$H_l(u) = l! \sum_{p=1}^{[l/2]} (-1^p) \frac{(2u)^{l-2p}}{p!(l-2p)!}. \tag{8}$$

This system is orthogonal on $(-\infty,+\infty)$ with weight function $h(u) = e^{-u^2}$ and gives us a possibility to decrease influence of the data lying far from origin. Therefore using system of Hermite orthogonal polynomials in the ortho-synapses of double ortho-neuron's output layer is necessary, because of it allows to avoid signals' normalization in the hidden layer during training process.

3. Training of the double ortho-neuron.

Training of double ortho-neuron consists in adjusting of weight coefficients that are situated inside ortho-synapses. The sum of the squared errors is be used as the learning criterion:

$$E(k) = \frac{1}{2}(d(k) - o(k))^2 = \frac{1}{2}e(k)^2, \tag{9}$$

where k = 1,2,… – discrete time or ordinal number of sample, *d(k)* – value of the reference signal at time moment *k* (for *k*-th training sample), *o(k)* – value of the output signal.

For the adjusting output layer's weight coefficients gradient algorithm can be used

$$w_{lo}(k+1) = w_{lo}(k) + \eta_o(k)e(k)\varphi_{lo}(u(k)), \tag{10}$$

or in vector notation:

$$w_o(k+1) = w_o(k) + \eta_o(k)e(k)\varphi_o(u(k)), \tag{11}$$

where $\eta_0(k)$ – learning rate parameter , $w_o(k) = (w_{0o}, w_{1o}, ..., w_{h_0 o})^T$ – $(h_0 \times 1)$ vector, $\varphi_o(k) = (\varphi_{0o}(u(k)), \varphi_{1o}(u(k)), ..., \varphi_{h_0 o}(u(k)))^T$ – $(h_0 \times 1)$ vector, $u(k) = \sum_{i=1}^{n}\sum_{j=0}^{h_i} w_{ji}\varphi_{ji}(x_i(k))$.

Also Kaczmarz-Widrow-Hoff adaptive algorithm [12, 13] can be used for this purpose as well:

$$w_o(k+1) = w_o(k) + \frac{d(k) - o(k)}{\|\varphi_o(k)\|^2}\varphi_o(k) \tag{12}$$

and because of signals' orthogonality on the weights' w_{lo} inputs algorithm (12) has maximally possible rate of convergence [14].

When we deal with noisy signal it will be reasonable bring in some additional smoothing. Following procedure can be used for this purpose [15]

$$\begin{cases} w_o(k+1) = w_o(k) + r_o^{-1}(k)(d(k) - o(k))\varphi_o(k), \\ r_o(k+1) = \alpha r_o(k) + \|\varphi_o(k)\|^2, \end{cases} \tag{13}$$

where $0 \le \alpha \le 1$ – forgetting factor. When $\alpha = 0$ procedure (13) coincides with algorithm (12) and when $\alpha = 1$ we obtain stochastic approximation learning rule.

At last weight coefficients of the output ortho-synapse can be adjusted with the standard least square method, minimizing learning criterion specified on entire training set

$$E^N = \sum_{k=1}^{N} E(k) = \frac{1}{2}\sum_{k=1}^{N} e^2(k), \tag{14}$$

in the form

$$w_o(N) = \left(\sum_{k=1}^{N}\varphi_o(k)\varphi_o^T(k)\right)^{-1}\sum_{k=1}^{N}\varphi_o(k)d(k) = P(N)\sum_{k=1}^{N}\varphi_o(k)d(k). \tag{15}$$

Expression (14) also can be represented in recursive form:

$$\begin{cases} w_0(k+1) = w_0(k) + P(k+1)(d(k) - o(k))\varphi_0(k), \\ P(k+1) = P(k) - \dfrac{P(k)\varphi_0(k+1)\varphi_0^T(k+1)P(k)}{1 + \varphi_0^T(k+1)P(k)\varphi_0(k+1)}. \end{cases} \quad (16)$$

By act of condition (2) matrixes *P(N)*, *P(k)*, *P(k+1)* are diagonal (or then to the same) and therefore learning process will obtain numerical stability and high rate of convergence.

Synaptic weights of the hidden layer are included nonlinearly into the description of the double ortho-neuron. For adjusting of ortho-synapses of the hidden layer it is necessary to use algorithms based on the error backpropagation procedures.

Let us define the following criterion function

$$E(k) = \frac{1}{2}\left(d(k) - f_o(\sum_{i=}^{n} \sum_{j=0}^{h_i} w_{ji}\varphi_{ji}(x_i(k))) \right)^2 \quad (17)$$

and introduce can use gradient algorithm for adjusting synaptic weights of the hidden layer in the form

$$w_{ji}(k+1) = w_{ji}(k) + \eta_i(k)e(k)f_o'(k)\varphi_{ji}(x_i(k)) = w_{ji}(k) + \eta_i(k)\delta(k)\varphi_{ji}(x_i(k)), \quad (18)$$

or in vector notation:

$$w_{ji}(k+1) = w_{ji}(k) + \eta_i(k)\delta(k)\varphi_{ji}(x_i(k)) \quad (19)$$

where $\delta(k) = e(k)f'_o(k)$ – δ -error, accepted at the multilayered neural networks learning theory, $w_i(k) = (w_{0i}(k), w_{1i}(k),..., w_{h_i i}(k))^T$ – $(h_i \times 1)$ vectors, $\varphi_i(k) = (\varphi_{0i}(x_i(k)), \varphi_{1i}(x_i(k)),..., \varphi_{h_i i}(x_i(k)))^T$ – $(h_i \times 1)$ vectors.

However disadvantages of gradient algorithm are well known, so we propose the modification of the Levenberg-Marquardt procedure [16, 17] for tuning ortho-synapses of the hidden layer which possesses both filtering and tracking properties:

$$\begin{cases} w_i(k+1) = w_i(k) + r_i^{-1}(k)\delta(k)\varphi_i(k), \\ r_i(k+1) = \alpha r_i(k) + \|\varphi_i(k+1)\|^2, 0 \le \alpha \le 1. \end{cases} \quad (20)$$

Because of the orthogonality or orthonormality of the activation functions learning procedure will retain numerical stable. Also using systems of orthogonal polynomials as activation functions allows to speed up learning process and decrease time required for adjusting weight coefficients.

4. Simulation results.

For signal forecasting and approximation neuropredictor of stochastic and chaotic processes based on double ortho-neuron shown on fig. 4 can be used.

We have applied proposed neuropredictor for the forecasting of a chaotic process defined by the Mackey-Glass equation [18]

$$y'(t) = \frac{0{,}2t(t-\tau)}{1 + y^{10}(t-\tau)} - 0{,}1y(t). \quad (21)$$

Signal was quantized with step 0,1. We took a fragment containing 500 points for training set. Our goal was forecasting signal value on six steps forward using its prehistory (its values at time moments *k*, (*k*-6), (*k*-12) and (*k*-18)). Testing set contained 9500 element of the sequence – signal values from 501 to 1000.

For estimation of received result we used normalized mean square error (22).

$$NRMSE(k, N) = \frac{\sum_{q=1}^{N} e^2(k+q)}{N\sigma}, \quad (22)$$

where σ – mean square deviation of the predictable process on the training set.

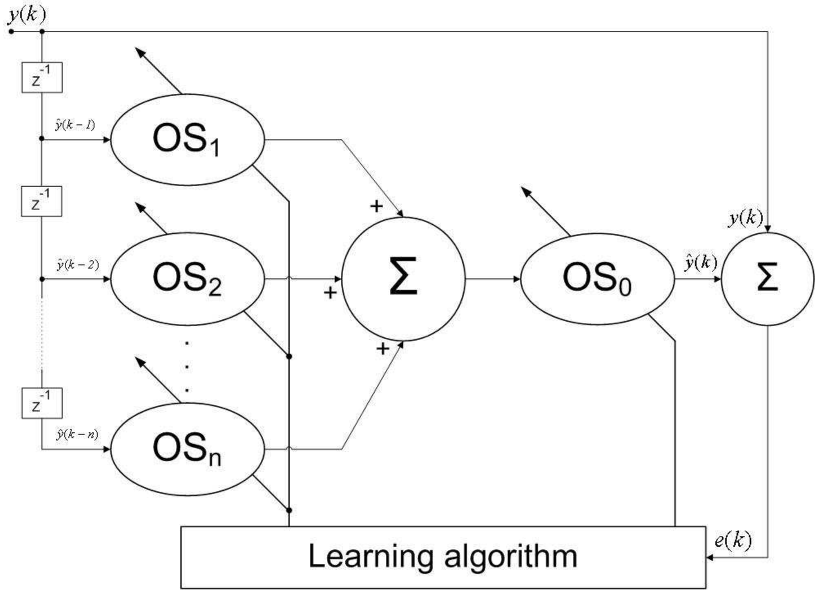


Fig 4. Neuropredictor based on DON

Following systems of orthogonal polynomials were used as activation functions: Tschebyshev 1-st type orthogonal polynomials (T1), Tschebyshev 2-nd type orthogonal polynomials (T2), Hermite orthogonal polynomials (H1), Hermite orthonormalized polynomials (H2) and Laguerre orthogonal polynomials (L).Quantity of activation functions $h_i = h_0 = h$ in ortho-synapses varied from 2 to 9. Forecasting errors are given in the table.

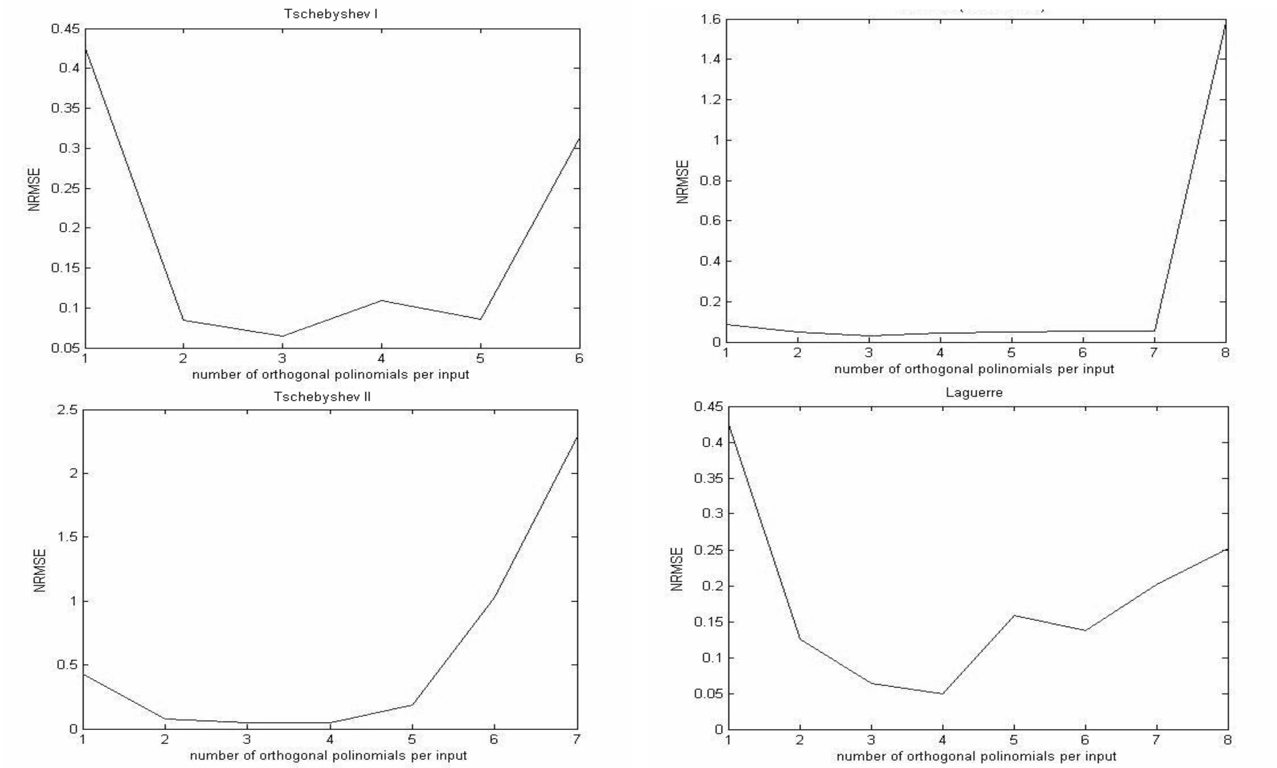
Doubel ortho-neuron forecasting errors

h	System of polynomials				
	T1	T2	H1	H2	L
2	0.425	0.425	0.425	0.086	0.425
3	0.084	0.078	0.127	0.05	0.126
4	0.064	0.045	0.031	0.032	0.063
5	0.109	0.050	0.039	0.044	0.049
6	0.085	0.19	0.209	0.051	0.159
7	0.314	1.027	0.199	0.052	0.138
8	13.62	2.29	0.246	0.056	0.202
9	80.93	1131	0.26	1.583	0.252

Dynamic of the error variation process with increasing number of activations functions in ortho-synapses is shown on fig. 5.

To solve Mackey-Glass time-series forecasting problem only three-four activation functions in each ortho-synapse are necessary. So we have only 20 – 25 adjustable parameters – much lesser then if we use conventional artificial neural networks. Therefore time required for adjusting weight coefficients will be lesser too.

As can be readily seen the best result has been received when systems of Hermite polynomials (both orthogonal and orthonormalized) were used as activation functions. This fact can be explained by the mean of these two systems have interval of orthogonality $[-\infty, \infty]$ and output from the hidden layer signal is not normalized on interval [-1, 1]. So if we use another system of polynomials (which have [-1,1] interval of orthogonality) situation when we obtain signal which not belongs to this interval at the output layer's input may occur and lead to decreasing of the forecasting quality.



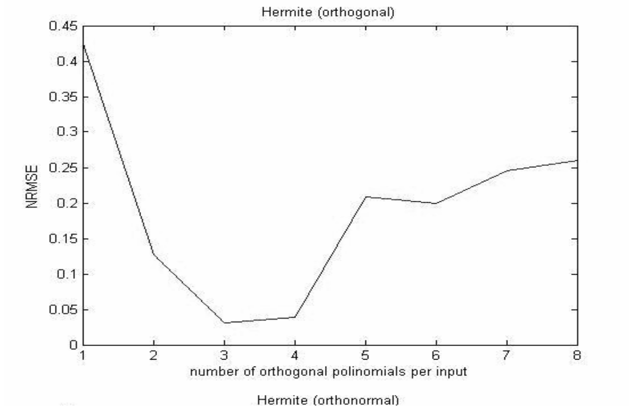


Fig 5. Double ortho-neuron forecasting errors

Conclusion

Double ortho-neuron provides quite good approximation quality using much lesser adjusting parameters then conventional artificial neural networks. When we deal with noisy or missed data we can configure input ortho-synapses in the hidden layer and adjust them uniquely for each input component of the signal. Acting in such manner we can assure more or less generalization level for each component of the input vector and achieve better quality of approximation or forecasting.

Double ortho-neuron is an enough simple and compact architecture, not affected by the “curse of dimensionality”. This architecture provides high precision of nonlinear nonstationary signal approximation and forecasting. An apparent advantage is easier implementation and lower computational complexity as compared to the conventional artificial neural networks.

1.. Чуев Ю.В., Михайлов Ю.Б., Кузьмин В.И. Прогнозирование количественных характеристик процессов. – Москва, 1975. – 400 с. 2. J.S. R. Jang, C.T. Sun, E. Mizutani. Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence. – NJ: Prentice Hall, 1997. – 614p. 3. Гроп Л. Методы идентификации систем. – Москва: Мир, 1979. – 302 с. 4. Yamakawa T., *Uchino E., Miki T., Kusanagi H. A neo fuzzy neuron and its applications to system identification and prediction of the system behavior // Proc. 2-nd Int.Conf. on Fuzzy Logic and Neural Networks "IIZUKA-92". – Iizuka, Japan, 1992. – P. 477 – 483. 5. Miki T., Yamakawa T. Analog implementation of neo-fuzzy neuron and its on-board learning / Ed. by N.E. Mastorakis "Computational Intelligence and Applications". – Piraeus: WSES Press, 1999. – P.144 – 149. 6. Bodyanskiy Ye., Kolodyazhniy V., Slipchenko O. Structural and synaptic adaptation in the artificial neural networks with orthogonal activation functions // Sci. Proc. of Riga Technical University. Comp. Sci., Int. Technology and Management Sci. – 2004. – №20. – P.69 – 76. 7. Bodyanskiy Ye., Kolodyazhniy V., Slipchenko O. Artificial neural network with orthogonal activation functions for dynamic system identification / Ed. by O. Sawodny, P. Scharff "Synergies between Information Processing and Automation". – Aachen: Shaker Verlag, 2004. – P.122 – 127. 8. Bodyanskiy Ye., Pliss I., Slipchenko O. Growing neural networks based on orthogonal activation functions // Proc. XII-th Int. Conf. "Knowledge – Dialog – Solution". – Varna, 2006. – P. 84 – 89. 9. Bodyanskiy Ye., Slipchenko O. Ontogenic neural networks using orthogonal activation functions // Prace. Naukowe Akademii Ekonomicznej we Wroclawiu. – 2006. – №21. – P.13 – 20. 10. Бодянский Е.В., Колодяжный В.В., Слипченко А.Н. Нейропредиктор с ортогональными функциями активации // Автоматизація виробничих процесів. – 2005. – №1(20). – С.123 – 127. 11. Бодянский Е.В., Винокурова Е.А., Ламонова Н.С. Двойной вэйвлет-нейрон и алгоритм его обучения // Радиоэлектроника. Информатика. Управление. – 2006. – №2. – С.85 – 91. 12. Widrow B., Hoff Jr. M.E. Adaptive switching circuits // 1960 URE WESCON Convention Record. – N.Y.: IRE, 1960. – Part 4. – P.96 – 104. 13. Kaczmarz S. Approximate solution of systems of linear equations // Int. J. Control. – 1993. – 53. – P. 1269 – 1271. 14. Райбман Н.С., Чадеев В.М. Построение моделей процессов производства. – Москва: Энергия, 1975. – 376с. 15. Бодянский Е.В., Плисс И.П., Соловьева Т.В. Многошаговые оптимальные упредители многомерных нестационарных стохастических процессов // Доклады АН УССР – 1986. – Сер.А. - №12. – С.47 – 49. 16. Bodyanskiy Ye., Kolodyazhniy V., Stephan A. An adaptive learning algorithm for a neuro-fuzzy network / Ed. by B. Reusch. "Computational Intelligence. Theory and Applications." – Berlin – Heidelberg – New York: Springer, 2001. – P. 68 – 75. 17. Otto P., Bodyanskiy Ye, Kolodyazhniy V. A new learning algorithm for a forecasting neuro-fuzzy network // Integrated Computer-Aided Engineering. – Amsterdam: IOS Press, 2003 – 10. – №4. – P.399 – 409. 18. Mackey M.C., Glass L. Oscillation and chaos in physiological control systems // Science. – 1977. – 197. – P. 238 – 289.*