Hybrid GMDH–neural network of computational intelligence

Yevgeniy Bodyanskiy, Olena Vynokurova, Iryna Pliss

Control Systems Research Laboratory, Kharkiv National University of Radio and Electronics, Lenina av., 14, Kharkiv, Ukraine, 61166

bodya@kture.kharkov.ua, vinokurova@kture.kharkov.ua, pliss@kture.kharkov.ua

Abstract. *In the paper the hybrid GMDH–neural network architectures of computational intelligence are proposed. The first architecture is GMDH–neural network based on Q–neurons with optimal learning algorithm. The second architecture is GMDH–wavelet–neural network architecture on the adaptive compartmental wavelon with computationally simple and effective learning algorithm. The learning algorithms have both following and filtering properties and allow processing of non–stationary nonlinear signals in real time. Tuning the wavelons receptive fields, including their transformations (translation, dilation, rotation, transformation membership function form) allows to improve the network approximation properties. The experiments results were compared with conventional GMDH–neural networks based on N–adalines and have shown the advantage of the suggested approach. Proposed hybrid GMDH–neural network architectures were used in non–stationary chaotic and stochastic time series forecasting, emulation and identification tasks.*

Keywords

Computational intelligence, inductive modelling, GMDH–neural network, Q–neuron, wavelon, wavelet neural networks, emulation, forecasting.

1 Introduction

Nowadays hybrid systems of the non–stationary signals processing under uncertainty conditions are widely used in various applications. The topical problem of non–stationary dynamical process forecasting and emulation [1, 2, 3, 4, 5], can be solved via using hybrid neural network systems. At present a lot of neural network architectures and learning algorithms exist which are either structurally unwieldy or the learning algorithms have low learning rate. It all results in their low efficiency in the real time applications. Thus the main task is the network architecture selection and optimal learning algorithm synthesis, which on one the hand will allow a real time signal processing and on the other hand will ensure the possibility of systems implementation simplicity.

This paper is devoted to the synthesis of hybrid GMDH (Group Method of Data Handling) architectures based on new neuron structures and their learning algorithms. These hybrid neural networks have an increased learning rate and provide improvement of the approximation properties what allows to extend the number of neuron's inputs.

2 Q–neuron structure and learning algorithm

In some cases the nonlinear transform can be provided by linear associator via special pre-processing of input signals (functional expansion). The fig. 1 shows the quadratic neuron (Q–neuron) structure [7], implementing the transformation in form

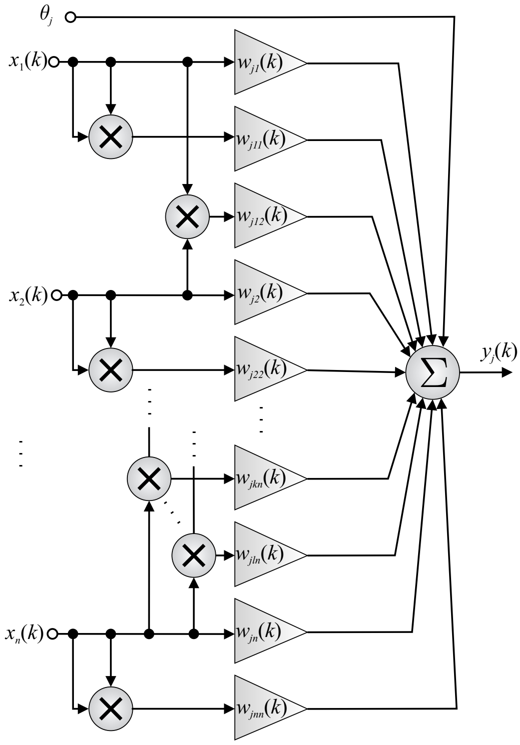


Fig. 1. Q–neuron structure.

$$y_j(k) = \theta_j(k) + \sum_{i=1}^{n} w_{ji}(k)x_i(k) + \sum_{p=1}^{n}\sum_{l=1}^{n} w_{jpl}x_p(k)x_l(k), \tag{1}$$

which considering the notations where $w_{j0} = \theta_j(k)$, $b_j(k) = (w_{j1}(k), w_{j2}(k), \ldots, w_{jn}(k)^T$ is a $(n \times 1)$ – vector, $C_j(k) = \{w_{jpl}(k)\}$ is a $(n\times n)$ – matrix, $x_{iv}(k) = (x_1(k), x_2(k), \ldots, x_n(k))^T$ is a $(n\times 1)$ – vector, $x(k) = (1, x_{iv}^T(k))^T$, can be rewritten in the following form

$$y_j(k) = w_{j0}(k) + b_j^T(k)x_{iv}(k) + x_{iv}^T(k)C_j(k)x_{iv}(k), \tag{2}$$

or in the more compact matrix form

$$y_j(k) = x^T(k)W_j(k)x(k), \tag{3}$$

where

$$W_j(k) = \begin{Bmatrix} w_{j0}(k) & 0.5b_j^T(k) \\ 0.5b_j^T(k) & C_j(k) \end{Bmatrix} \tag{4}$$

is a block $(n+1)\times(n+1)$ – matrix.

Essentially in this case the number of synaptic weights considerably increases, though the implementation simplicity often provides the advantage of such models, using which polynomial transformation of any required degree can be provided.

Tuning the synaptic weights matrix W_j is performed by minimization of criterion

$$E_j(k) = 1/2e_j^2(k) = 1/2(d_j(k) - x^T(k)W_j(k)x(k))^2, \tag{5}$$

by means of the gradient procedure

$$W_j(k+1) = W_j(k) + \eta(k)e_j(k)x(k)x^T(k), \tag{6}$$

where

$$e_j(k) = d_j(k) - x^T(k)W_j(k)x(k), \tag{7}$$

In order to find the value of $\eta(k)$ parameter which ensures the optimal properties of algorithm (6) we shall define the Lyapunov function in the following form [6]

$$V(k) = Tr(\widetilde{W}_j(k-1)\widetilde{W}_j^T(k-1)) - Tr(\widetilde{W}_j(k)\widetilde{W}_j^T(k)), \tag{8}$$

Maximization of this function on each step of the algorithm satisfies the requirement of maximal speed of matrix coefficients average variance $\widetilde{W}_j(k)$ decreasing, where $\widetilde{W} = W_j - W_j(k)$ is the matrix of the current values $W_j(k)$ deviation from unknown optimal values W_j, and $Tr(\bullet)$ is the trace of matrix.

Writing (6) in form

$$\widetilde{W}_j(k) = \widetilde{W}_j(k-1) + \eta(k)e_j(k)x(k)x^T(k), \tag{9}$$

and substituting (9) into (8) we can write

$$\begin{aligned} V(k) &= \eta(k)e(k)Tr(\widetilde{W}_j(k-1)x(k)x^T(k-1)) + \eta(k)e(k)Tr(x(k)x^T(k)\widetilde{W}_j(k-1)) - \\ &- \eta^2(k)e^2(k)Tr(x(k)x^T(k)x(k)x^T(k)) = 2\eta(k)x^T(k)\widetilde{W}_j^T(k-1)x(k) - \eta^2(k)e^2(k)\|x(k)\|^4. \end{aligned} \tag{10}$$

Afterwards, solving the differential equation

$$\partial V(k)/\partial\eta(k) = 0, \tag{11}$$

taking into account that

$$e_j(k) = x^T(k)\widetilde{W}_j(k-1)x(k), \tag{12}$$

we can obtain the optimal step parameter value in the form

$$\eta(k) = \|x(k)\|^{-4}, \tag{13}$$

Substitution of (13) into (6) results in the learning algorithm taking a form [7]

$$W_j(k+1) = W_j(k) + \frac{d_j(k) - x^T(k)W_j(k)x(k)}{\|x(k)\|^4}x(k)x^T(k), \tag{14}$$

what is practically an extension of the Kaczmarz–Widrow–Hoff algorithm on the Q–neuron. We shall note that using the pseudoinversion matrix function we can rewrite (14) as

$$W_j(k+1) = W_j(k) + \left(d_j(k) - x^T(k)W_j(k)x(k)\right)\left(x(k)x^T(k)\right)^+ = W_j(k) + e_j(k)\left(x(k)x^T(k)\right)^+. \tag{15}$$

3 Structure of adaptive compartmental wavelon and its learning algorithm

Let us consider the two–layers structure shown on figure 2 that coincides with the traditional radial–basis neural networks one [8].

The input layer of the structure is the receptor and in current time instant k the input signal in vector form $x(k) = (x_1(k), x_2(2), \ldots, x_n(k))^T$ is fed on it. Unlike radial basis function network the hidden layer consists not of R–neurons, but by wavelons with wavelet activation function in the form

$$\varphi_j(x(k)) = \varphi_j((x(k) - c_j(k))^T Q_j^{-1}(k)(x(k) - c_j(k)), \alpha_j), j = 1, 2, \ldots, h, \tag{16}$$

in which instead of translation parameters σ_{ji} the dilation matrix Q_j is used. It moves us from using Euclidian distance to implementation of Itakura–Saito metric [9].

This results in the fact that receptive fields – wavelons hyperellipsoids can have the arbitrary orientation relatively to the coordinate axes of space X, what extends the functional properties of the adaptive compartmental wavelon.

Based on the results of the author [10], about that the wavelet–function can be used as a membership function in fuzzy systems, we can introduce the adaptive membership function based on wavelet Mexican Hat, having form

$$\varphi_j(x(k)) = (1 - \alpha_j\tau_j^2(x(k)))\exp(-\tau_j^2(x(k))/2), \tag{17}$$

where $\tau_j(x(k)) = ((x(k) - c_j(k))^T Q_j^{-1}(k)(x(k) - c_j(k)))$, α_j is turning parameter ($0 \le \alpha_j \le 1$).

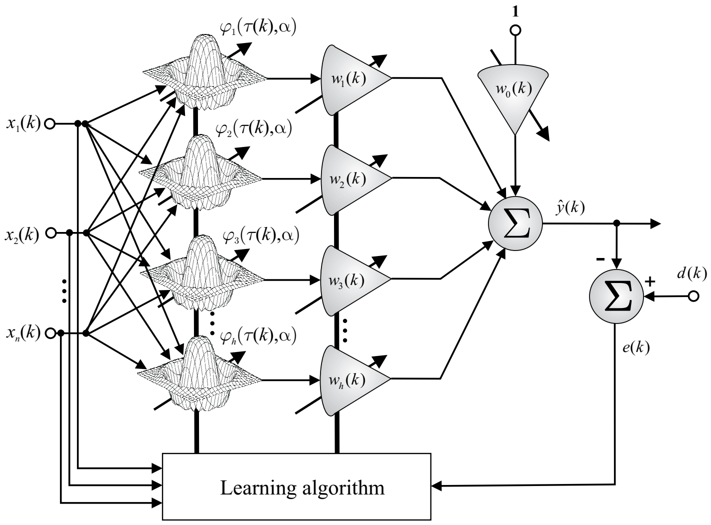


Fig. 2. Adaptive compartmental wavelon structure.

Adaptive parameter α_j allows to tune the form of membership function in process of hybrid architecture learning, thus if $\alpha_j = 0$ then we get Gauss membership function, if $\alpha_j = 1$ then we get wavelet membership function MexicanHat, and if $0 < \alpha_j < 1$ then we get hybrid membership function [11, 12].

Figure 3 shows the wavelons activation membership function (17) with arbitrary matrices Q_j and different parameters α_j.

And at last, the output layer is the adaptive linear associator with tuning synaptic weights

$$\hat{y}(k) = w_0 + \sum_{j=1}^{h} w_j \varphi((x(k) - c_j)^T Q_j^{-1}(x(k) - c_j)) = w^T \varphi(x(k)), \tag{18}$$

where $\varphi_0(x(k)) \equiv 0$, $w = (w_0, w_1, \ldots, w_h)^T$, $\varphi(x(k)) = (1, \varphi_1(x(k)), \varphi_2(x(k)), \ldots, \varphi_h(x(k)))^T$.

For the synaptic weights w_j and the wavelon parameters (vectors c_j, matrices Q_j and parameters α_j) tuning we use gradient minimization of criterion

$$E(k) = \frac{1}{2}e^2(k) = \frac{1}{2}(y(k) - \hat{y}(k))^2, \tag{19}$$

so unlike in the component–wise learning considered in [3], we make some corrections in the vector–matrix form, that, firstly is less computationally expensive, and secondly it allows to optimize learning process on the operation rate.

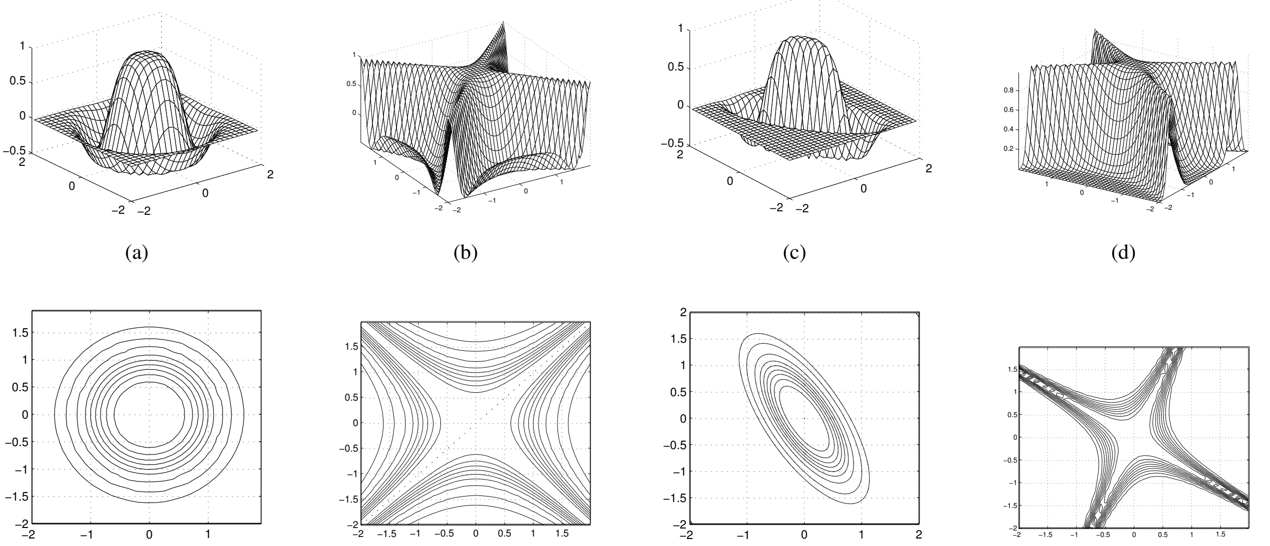
In general case the learning algorithm can be written in form

$$\begin{cases} w_j(k+1) = w_j(k) - \eta_w \left(\partial E(k)/\partial w_j\right), j = 1 \ldots, h, \\ c_j(k+1) = c_j(k) - \eta_c \nabla_{c_j} E(k), j = 1, \ldots, h, \\ Q_j^{-1}(k+1) = Q_j^{-1}(k) - \eta_Q \left\{\partial E(k)/\partial Q_j^{-1}\right\}, j = 1, \ldots, h, \\ \alpha_j(k+1) = \alpha_j(k) - \eta_\alpha \left(\partial E(k)/\partial \alpha_j\right), j = 1, \ldots, h, \end{cases} \tag{20}$$

where $\nabla_{c_j} E(k)$ is $n \times 1$–vector–gradient criterion (19) on c_j; $\left\{\partial E(k)/\partial Q_j^{-1}\right\}$ is $(n \times n)$–matrix, formed by partial derivatives $E(k)$ on components Q_j^{-1}; η_w, η_{c_j}, $\eta_{Q_j^{-1}}$ and η_{α_j} are the learning rates.

For the adaptive membership function (17) we can write

$$\begin{cases} \partial E(k)/\partial w_j = e(k)\left(1 - \alpha_j \tau_j^2(x(k))\right) \exp(-\tau_j^2(x(k))/2) = e(k) J_{w_j}(k), \\ \nabla_{c_j} E(k) = 2e(k) w_j(k) \left(\alpha_j \tau_j^3(x(k)) - (2\alpha_j + 1)\tau_j(x(k))\right) \exp(-\tau_j^2(x(k))/2) Q_j^{-1}(x(k) - c_j(k)) = \\ \qquad = e(k) J_{c_j}(k), \\ \left\{\partial E(k)/\partial Q_j^{-1}\right\} = e(k) w_j(k) \left(\alpha_j \tau_j^3(x(k)) - (2\alpha_j + 1)\tau_j(x(k))\right) \exp(-\tau_j^2(x(k))/2)(x(k) - c_j(k)) \cdot \\ \qquad \cdot (x(k) - c_j(k))^T = -e(k) J_{Q_j^{-1}}(k), \\ \partial E(k)/\partial \alpha_j = -e(k) w_j(k) \tau_j^2(x(k)) \exp(-\tau_j^2(x(k))/2) = e(k) J_{\alpha_j}(k), \end{cases} \tag{21}$$



(a) (b) (c) (d)

(e) $Q_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \alpha_1 = 1$ (f) $Q_2 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \alpha_2 = 0.5$ (g) $Q_3 = \begin{pmatrix} 4 & 3 \\ 1 & 2 \end{pmatrix}, \alpha_3 = 1$ (h) $Q_4 = \begin{pmatrix} 4 & 3 \\ 1 & -2 \end{pmatrix}, \alpha_4 = 0$

Fig. 3. Wavelons activation membership function with arbitrary matrices Q_j and adaptive parameters α_j

where $\tau_j(x(k)) = ((x(k) - c_j(k))^T Q_j^{-1}(x(k) - c_j(k)))$.

Increasing of the learning rate can be achieved by using more complex procedures than gradient ones, such as Hartley or Marquardt procedures and using the inverse matrices lemma and after applying simple transformations we obtain the effective parameters learning algorithm in the form

$$w_j(k+1) = w_j(k) + \lambda_w \left((e(k) J_w(k)) / (\eta_w + J_w^2(k)) \right), \tag{22}$$

$$c_j(k+1) = c_j(k) - \lambda_c \left((e(k) J_c(k)) / \eta_c + \|J_c(k)\|^2) \right), \tag{23}$$

$$\alpha_j(k+1) = \alpha_j(k) - \lambda_\alpha \left((e(k) J_\alpha(k)) / (\eta_\alpha + J_\alpha^2(k)) \right), \tag{24}$$

accurate within the descriptions and coinciding with optimal (for $\lambda_w = \lambda_c = \lambda_\alpha = 1$, $\eta_w = \eta_c = \eta_\alpha = 0$) one–step Kaczmarz algorithm.

For the turning of matrices Q_j^{-1} we can use the matrix modification of algorithm (23) in form

$$Q_j^{-1}(k+1) = Q_j^{-1}(k) + \lambda_Q \left((e(k) J_Q(k)) / \left(\eta_Q + Tr\left(J_Q^T(k) J_Q(k) \right) \right) \right), \tag{25}$$

where λ_Q is a positive dampening parameter and η_Q is a momentum term parameter.

It is known, that one–step algorithms such as Kaczmarz one, have rapid response, but they don't have filtering properties, i.e. they are not operated well in the conditions of intensive disturbance and noise. In order to provide the learning algorithm with smoothing properties, we can introduce next learning algorithm [11, 12]:

$$\begin{cases} w_j(k+1) = w_j(k) + \lambda_w \left(e(k) J_w(k) / \eta_w(k) \right), \quad \eta_w(k+1) = \gamma_w \eta_w(k) + J_w^2(k+1), \\ c_j(k+1) = c_j(k) - \lambda_c \left(e(k) J_c(k) / \eta_c(k) \right), \quad \eta_c(k+1) = \gamma_c \eta_c(k) + \|J_c(k+1)\|^2, \\ Q_j^{-1}(k+1) = Q_j^{-1}(k) + \lambda_Q \left(e(k) J_Q(k) / \eta_Q(k) \right), \quad \eta_Q(k+1) = \gamma_Q \eta_Q(k) + Tr\left(J_Q^T(k) J_Q(k) \right), \\ \alpha_j(k+1) = \alpha_j(k) + \lambda_\alpha \left(e(k) J_\alpha(k) / \eta_\alpha(k) \right), \quad \eta_\alpha(k+1) = \gamma_\alpha \eta_\alpha(k) + J_\alpha^2(k+1), \end{cases} \tag{26}$$

(here $0 \leq \{\lambda_w, \lambda_c, \lambda_Q, \lambda_\alpha\} \leq 1$ are parameters of weighted out–dated information). This procedure is nonlinear hybrid of Kaczmarz–Widrow–Hoff and Goodwin–Ramadge–Caines algorithms and has following and filtering properties.

Proposed learning algorithm has advantages of high operation rate and approximation accuracy as compared to conventional ones at the expense of inverse matrix absence which become more complex under big number of the network parameters what results in "curse of dimensionality" problem.

4 Hybrid GMDH architectures based on proposed neuron models

The proposed above neuron structures can be used as a structural block of neural network architecture. In this paper we propose to synthesize modified GMDH-neural network where Q–neurons and adaptive compartmental wavelon with the input number $n = 2$ are used as the nonlinear synapses, and also the identical architecture with extended input number in node $n = 3$. The learning of modified GMDH–neural network is performed by the traditional method [1, 2]. The selection of input vectors for the next layer is implemented using residual variance. We continue the process of the layers number increase till the necessary accuracy is obtained. The nodes number in the layer is determined by the quantity of inputs combination number C_n^2.

5 Results of experiments

The efficiency of the proposed hybrid GMDH–neural network based on Q–neuron was confirmed by solving the dynamical object [13] emulation task described by the equation

$$y(k+1) = 0.3y(k) + 0.6y(k-1) + f(u(k)). \qquad (27)$$

The neural network learning was performed based on the sample set generated based on equation (27) with the control signals $f(u(k)) = 0.6 \sin u(k) + 0.3 \sin 3u(k) + 0.1 \sin 5u(k)$ and $u(k) = sin(2\pi k/250)$ for $k = 1 \ldots 1500$. After 1500 steps the learning process was stopped. The dynamic object 27 with the same control signals for $k = 1501 \ldots 2500$ and $f(u(k)) = u^3(k) + 0.3u^2(k) - 0.4u(k)$, and $u(k) = sin(2\pi k/250) + sin(2\pi k/250)$ for $k = 2501 \ldots 4000$ was used as the test data for emulation.

The GMDH–neural network had the number of inputs equal to $n_{GMDH} = 6$. Every Q–neuron was learned in the batch mode for 10 epochs. The initial Q–neuron parameters values were given as equal to 0. The mean square error (MSE) was used as emulation quality criterion. Figures 4 illustrates the emulation results of the dynamic object (27). The two curves representing the actual values (dashed line) and emulation results values (solid line) are almost indistinguishable.

The table 1 shows the comparative analysis of the emulation process based on expanded GMDH–neural network on Q–neurons setting number of node inputs $n = 2$ and $n = 3$ correspondingly with the results of the standard GMDH–neural network.

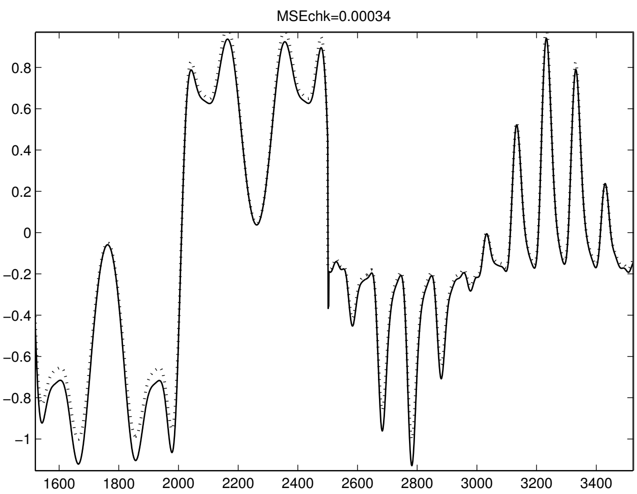


Fig. 4. The emulation results fragment of dynamical object (27).

As the simulation results have shown, the proposed approach provides more high emulation accuracy as compared to such architecture with common number of node inputs and also to the standard GMDH–neural network.

The second experiment was made on real electroencephalogram (EEG) signal in the deep artificial hypothermia based on GMDH–neural network with adaptive compartmental wavelon. The data of EEG signal was obtained from the exper-

Tab. 1. The results of dynamical object emulation.

Neural network/ Learning algorithm	Number of layers in GMDH–neural network	Number of input in neuron	MSE
Modified extension GMDH–neural network based on Q–neuron / Proposed learning algorithm (15)	2	3	0.000349
Modified GMDH–neural network based on Q–neuron / Proposed learning algorithm (15)	2	2	0.000676
GMDH–neural network [14]	2	2	0.0032

iment carried on the Vistar line rat–male with mass 180–200 gr. in the winter period. The signal quantization frequency was 40 Hz. The hypometabolic state in the rat was invoked based on Andzhusa–Bakhmeteva–Dzhaja method. The data was obtained on the base of joint research under scientific collaboration with Institute of cryobiology and cryomedicine of Academia Medical Science. The GMDH–neural network had the input number equal $n_{GMDH} = 5$ and the values $x(k-4), x(k-3), x(k-2), x(k-1), x(k)$ were used to forecast $x(k+1)$. Adaptive compartmental wavelon was trained in bath mode with procedure (26) for 1000 iterations (1000 training samples for $k = 1, \ldots, 1000$). Parameters of the learning algorithm were taken as $\gamma_c = \gamma_Q = \gamma_\alpha = 0.99$ and $\lambda_c = \lambda_Q = \lambda_\alpha = 0.99$. Initial values were $\eta_c = \eta_Q = \eta_\alpha = 10000$ and adaptive vector parameters $\alpha = (\alpha_1, \ldots, \alpha_h) = 0.1$. After 1000 iterations the training process was stopped, and the next 500 points for $k = 1001, \ldots, 1500$ we used as the testing data set to compute forecast. Initial values of synaptic weights were taken as equal to 0. As the quality criterion of forecasting was used root mean square error (RMSE). Figure 5 shows the results of EEG signal forecasting. The two curves, representing the actual (dot line) and forecasting (solid line) values, are almost indistinguishable.

Thus as it can be seen from experimental results the proposed GMDH–neural network with adaptive compartmental wavelon with the learning algorithm (26) having the same number of adjustable parameters ensures the best quality of forecast and high learning rate in comparison with conventional architecture with different mode of the learning algorithm. Table 2 shows the results of forecasting EEG signal.

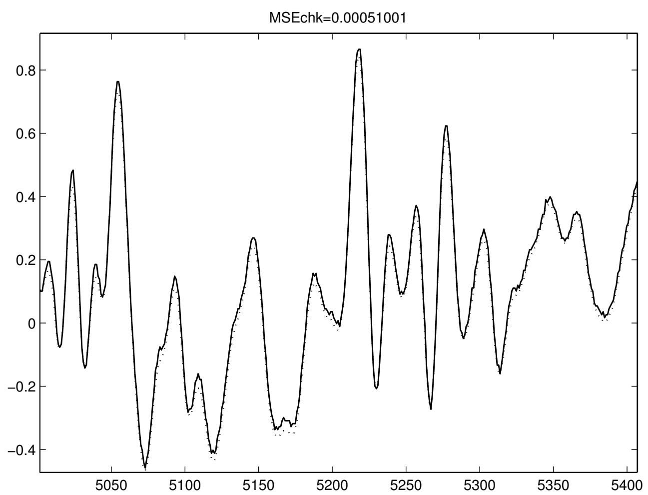


Fig. 5. Forecasting of EEG signal fragment using adaptive compartmental wavelon.

6 Conclusion

In the given paper modified GMDH–neural network architecture has been proposed. The Q–neurons and adaptive compartmental wavelon were used as the nodes. The computationally simple and effective Q–neuron learning algorithm in the matrix form which allows processing of non–stationary nonlinear signals and sequences under significantly uncertainty

Tab. 2. The results of EEG time series forecasting.

Neural network/ Learning algorithm	**Number of layers in GMDH–neural network**	**Number of input in neuron**	**RMSE**
GMDH–neural network with adaptive compartmental wavelon / Proposed learning algorithm (26) (All parameters were trained)	2	2	0.00113
GMDH–neural network with adaptive compartmental wavelon / Gradient learning algorithm (The weights and receptive field were trained)	2	2	0.0194
GMDH–neural network [14]	2	2	0.0297

condition was considered. The highly effective adaptive compartmental wavelon all–parameters learning algorithm was proposed. The replacement of standard GMDH neural network node and number of node inputs expansion allow to improve the network approximating properties. The experimental simulation based on the different signals kind was carried out, and its results have confirmed the advantages of the proposed approach.

References

[1] Stepashko V. S.: Analyzing the Criteria Effectiveness for Structure Identification of Forecasting Models. *P*roblems of Control and Informatics, 28: 3-4, 1994.

[2] Ivakhnenko A. G., Madala H.R.: Inductive learning algorithms for complex systems modeling. *L*ondon, Tokyo: CRC Press, 1994.

[3] Bishop C. M.: Neural Networks for Pattern Recognition. *O*xford: Clarendon Press, 1995.

[4] Bodyanskiy Ye., Pliss I., Vynokurova O.: Adaptive wavelet-neuro-fuzzy network in the forecasting and emulation tasks. *I*nt. Journal on Information Theory and Applications, 15(1): 47-55, 2008.

[5] Bodyanskiy Ye., Vynokurova O.: Hybrid radial-basis neuro-fuzzy wavelon in the non-stationary sequences forecasting problems *P*roc. 2nd Int. Conf. on Inductive Modelling, 144-147, 2008.

[6] Carotenuto L., Raiconi G.: On the minimization of quadratic functions with bilinear constraints via augmented Lagrangians. *J*ournal of Optimization Theory and Applications, 55: 23-36, 1987.

[7] Bodyanskiy Ye., Vynokurova O., Pliss I.: Hybrid neural network architecture on Q-neurons and its learning algorithms. *P*roc. Int. Conf. Intellectual systems for decision making and problems of computational intelligence, 2: 235-239, 2009. (in russian)

[8] Jang J. S. R., Sun C. T.: Functional equivalence between radial basis function networks and fuzzy inference systems. *I*EEE Trans. on Neural Networks, 4: 156-159, 1993.

[9] Itakura F.: Maximum prediction residual principle applied to speech recognition. *I*EEE Trans. on Acoustics, Speech and Signal Processing, 23: 67-72, 1975.

[10] Mitaim S., Kosko B.: What is the best shape for a fuzzy set in function approximation? *I*n proceedings of the 5th IEEE Int. Conf on Fuzzy Systems ”Fuzz-96”, 2: 1237-1213, 1996.

[11] Bodyanskiy Ye., Vynokurova O., Yegorova E.: Radial-basis-fuzzy-wavelet-neural network with adaptive activation-membership function. *I*nt. Journal on Artificial Intelligence and Machine Learning, 8: 9-15, 2008.

[12] Bodyanskiy Ye., Vynokurova O.: Adaptive wavelon and its learning algorithm. *I*nt. Journal Control Systems and Computers, 1: 47-53, 2009. (in russian)

[13] Narendra K. S., Parthasarathy K.: Identification and control of dynamical systems using neural networks. *I*EEE Trans. on Neural Networks, 1: 4-26, 1990.

[14] Jekabsons G. A.: A software tool for performing regression modelling using various modelling methods. *[*http://www.cs.rtu.lv/jekabsons/].