DE GRUYTER OPEN

Information Technology and Management Science

doi: 10.1515/itms-2014-0002

2014 / 17

Least Squares Support Vector Machine Based on Wavelet-Neuron

Yevgeniy Bodyanskiy[1], Olena Vynokurova[2], Oleksandra Kharchenko[3],
[1-3]*Kharkiv National University of Radio Electronics*

Abstract **– In this paper, a simple wavelet-neuro-system that implements learning ideas based on minimization of empirical risk and oriented on information processing in on-line mode is developed. As an elementary block of such systems, we propose using wavelet-neuron that has improved approximation properties, computational simplicity, high learning rate and capability of local feature identification in data processing. The architecture and learning algorithm for least squares wavelet support machines that are characterized by high speed of operation and possibility of learning under conditions of short training set are proposed.**

Keywords **– Adaptive wavelet function, forecasting, least squares support vector machine, non-linear non-stationary time series, wavelet-neuron.**

I. Introduction

Nowadays computational intelligence systems are widely used to solve a large class of problems associated with information processing, which is provided in different forms, from simple tables "object-property" to complex multidimensional non-stationary stochastic time series. Systems of computational intelligence, such as artificial neural networks, neuro-fuzzy systems and wavelet systems have a high processing speed, universal approximation properties, identification of local features [1]–[7].

However, the existing approaches for their training require large volumes of training data set, wherein the volume of the original training data set should be at least two orders bigger than the number of estimated parameters of such systems. Unfortunately, this situation does not always occur, especially in solving practical tasks. In many practical problems, for example, medical diagnostics, forecasting of financial indices etc., the volume of training data set is quite insufficient for constructing and training the effective system of computational intelligence.

The solving of problems in such a situation using conventional identification theory methods [8] is not effective, for which reason the method based on empirical risk minimization was proposed by V. N. Vapnik [9]–[12] and a support vector machine was designed based on this method.

Traditional support vector machine is a computational system that minimizes the empirical risk, but from viewpoint of practical implementation it is a sufficiently complex system, because it is related to the solving of nonlinear programming task at each step apart from high dimension with constraints in form of inequalities. Thus, in such a case the idea was wonderful; however, its implementation in on-line mode was not successful [2].

Therefore, at the beginning of this century, modification of this system has been proposed, called LS-SVM (least squares support vector machine) [13]. Here principal conditions were changed in such a way that it was necessary to solve a quadratic programming problem with equality constraints at each step, and it had already created the preconditions for the neural network implementation of this approach. But still, this machine was quite complicated and therefore further attempts were made to improve, firstly, speed, to simplify computing implementation and, secondly, to reduce training sample volume.

Based on fuzzy SVM [14], [15], the wavelet least squares support vector machine was proposed and compared with all previous variants – it had improved approximation properties, but, as a result, numerical implementation had become more difficult [16], [17]. The basis of LS-SVM was radial basis function network [18], where radial basis functions were replaced by multidimensional wavelet functions. For simplification of software implementation, by reducing the number of tuning parameters and increasing the speed we have proposed to use wavelet-neuron as the basic architecture [19]–[21] that has improved approximation properties and high operation speed. However, the disadvantages of wavelet-neuron are connected with its learning algorithms that do not allow tuning all parameters under a small data set.

Alternative to learning based on optimization is the learning based on memory that is based on the concept "neurons in the data points" [18]. The most typical representative of neural networks with such learning is General Regression Neural Network (GRNN); however, it solves a task of interpolation instead of approximation that essentially complicates its usage in noisy data processing.

Therefore, the development of sufficiently simple wavelet-neuro-fuzzy systems is advisable. Such systems implement learning based on empirical risk minimization and are oriented to information processing in on-line mode. Wavelet-neuron that has improved approximation and extrapolation properties can be used as a basic element of such systems.

II. Wavelet-Neuron Architecture

Let us consider the wavelet-neuron architecture, shown in Fig. 1. As seen, wavelet-neuron is quiet close to the standard n-input formal neuron, but instead of tuning synaptic weights it contains wavelet-synapses $WS_i, i = 1,2,\ldots,n$, where the tuning parameters are not only synaptic weights, but all parameters of adaptive wavelet activation functions $\varphi_{ji}(x_i(k))$ [21].

When input vector

$$x(k) = \left(x_1(k), x_2(k), \ldots, x_n(k)\right)^T$$

is fed to the wavelet-neuron input (here $k = 1,2,\ldots$ is current discrete time) its output can be written in the form

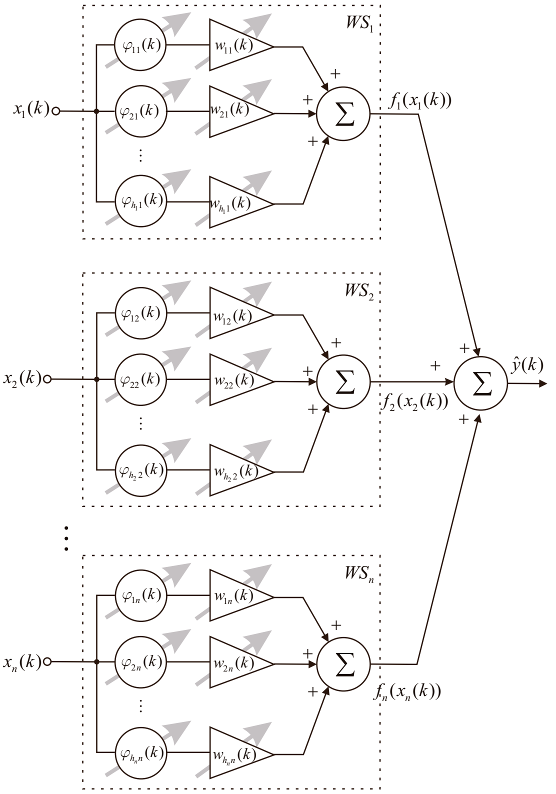

Fig. 1. Wavelet-neuron.

$$\hat{y}(k) = \sum_{i=1}^{n} f_i(x(k)) = \sum_{i=1}^{n} \sum_{j=1}^{h_i} w_{ji}(k)\varphi_{ji}(x_i(k)) \qquad (1)$$

where $w_{ji}(k)$ is synaptic weight, $\varphi_{ji}(x_i(k))$ is wavelet activation function.

The different analytical wavelet functions can be used as activation functions, but for adaptive tuning of wavelet neuron we use an adaptive wavelet activation function which was proposed in [22], [23] and has the form

$$\varphi_{ji}(x_i(k)) = \left(1 - \alpha_{ji}(k)t_{ji}^2(k)\right)\exp\left(-t_{ji}^2(k)/2\right) \qquad (2)$$

where $t_{ji}(k) = \left(x_i(k) - c_{ji}(k)\right)/\sigma_{ji}(k)$; $c_{ji}(k)$ is a center parameter of adaptive wavelet function; $\sigma_{ji}(k)$ is a width parameter of adaptive wavelet function; $\alpha_{ji}(k)$ is a shape parameter of adaptive wavelet function.

Tuning parameter α_{ji} allows changing the shape of an adaptive wavelet activation function during the training process of network, and, as a result, for $\alpha_{ji} = 0$ we obtain Gaussian function, and when $\alpha_{ji} = 1$ we obtain wavelet-function "Mexican Hat", and when $0 < \alpha_{ji} < 1$ – hybrid activation function.

Figure 2 shows the adaptive wavelet activation function with different parameters α and σ .

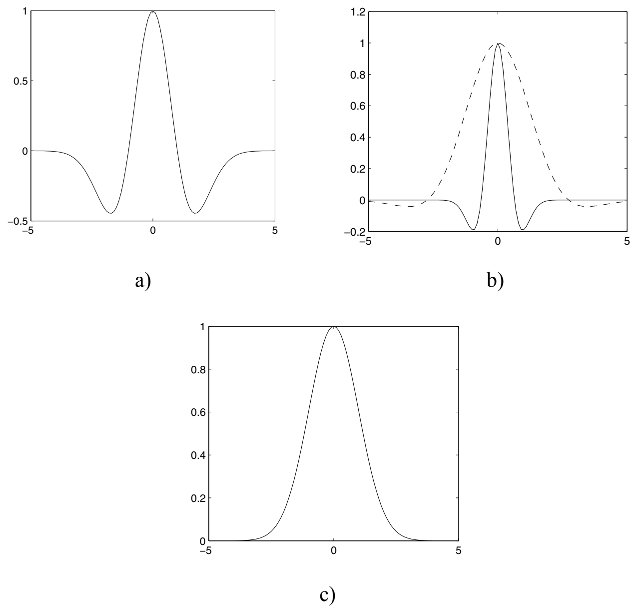

Fig. 2. Adaptive wavelet activation function: a – $\alpha = 1, \sigma = 1$; b – dashed line $\alpha = 0.3, \sigma = 1.5$; , solid line $\alpha = 0.6, \sigma = 0.5$; c - $\alpha = 0, \sigma = 1$.

The learning task is to find synaptic weights $w_{ji}(k)$, centers $c_{ji}(k)$, widths $\sigma_{ji}^{-1}(k)$ and shape parameters $\alpha_{ji}(k)$ of adaptive wavelet activation function on each k -th iteration, which optimizes the learning criterion.

III. Learning Algorithm for All Wavelet-Neuron Parameters

When the training data set is sufficiently large, as learning criteria we can use the conventional squared error function in the form

$$E(k) = \frac{1}{2}\left(y(k) - \hat{y}(k)\right)^2 = \frac{1}{2}\left(y(k) - \sum_{i=1}^{n}\sum_{j=1}^{h_i} w_{ji}(k)\varphi_{ji}(x_i(k))\right)^2 \qquad (3)$$

where $y(k)$ is a reference signal.

Introducing some denominations in the form $\varphi_i(x_i(k)) = (\varphi_{1i}(x_i(k)), \ldots, \varphi_{h_i i}(x_i(k)))^T$, $w_i(k) = (w_{1i}(k), \ldots, w_{h_i i}(k))^T$, $c_i(k) = (c_{1i}(k), \ldots, c_{h_i i}(k))^T$, $\sigma_i^{-1}(k) = (\sigma_{1i}^{-1}(k), \ldots, \sigma_{h_i i}^{-1}(k))^T$, $\alpha_i(k) = (\alpha_{1i}(k), \ldots, \alpha_{h_i i}(k))^T$, $t_i(k) = (t_{1i}(k), \ldots, t_{h_i i}(k))^T$, we can write the learning algorithm in the form

$$\begin{cases} w_i(k+1) = w_i(k) + \left(e(k)J_i^w(k)\right)/r_i^w(k), \\ c_i(k+1) = c_i(k) + \left(e(k)J_i^c(k)\right)/r_i^c(k), \\ \sigma_i^{-1}(k+1) = \sigma_i^{-1}(k) + \left(e(k)J_i^\sigma(k)\right)/r_i^\sigma(k), \\ \alpha_i(k+1) = \alpha_i(k) + \left(e(k)J_i^\alpha(k)\right)/r_i^\alpha(k), \end{cases} \qquad (4)$$

$$\begin{cases} r_i^w(k+1) = \beta r_i^w(k) + \left\| J_i^w(k) \right\|^2, \\ r_i^c(k+1) = \beta r_i^c(k) + \left\| J_i^c(k) \right\|^2, \\ r_i^\sigma(k+1) = \beta r_i^\sigma(k) + \left\| J_i^\sigma(k) \right\|^2, \\ r_i^\alpha(k+1) = \beta r_i^\alpha(k) + \left\| J_i^\alpha(k) \right\|^2 \end{cases} \tag{5}$$

where $0 \le \beta \le 1$ is a forgetting factor,
$J_i^w(k) = (J_{1i}^w(k), \dots, J_{h_i i}^w(k))^T$, $J_i^c(k) = (J_{1i}^c(k), \dots, J_{h_i i}^c(k))^T$,
$J_i^\sigma(k) = (J_{1i}^\sigma(k), \dots, J_{h_i i}^\sigma(k))^T$, $J_i^\alpha(k) = (J_{1i}^\alpha(k), \dots, J_{h_i i}^\alpha(k))^T$,
$J_{ji}^w(k) = \left(1 - \alpha_{ji}(k) t_{ji}^2(k)\right) \exp(-t_{ji}^2(k)/2)$,
$J_{ji}^c(k) = w_{ji}(k) \sigma_{ji}^{-1}(k) \left((2\alpha_{ji}(k) + 1) t_{ji}(k) - \alpha_{ji}(k) t_{ji}^3(k) \right) \times$
$\times \exp(-t_{ji}^2(k)/2)$,
$J_{ji}^\sigma(k) = w_{ji}(k)(x_i(k) - c_{ji}(k)) \left((2\alpha_{ji}(k) + 1) t_{ji}(k) - \alpha_{ji}(k) t_{ji}^3(k) \right) \times$
$\times \exp(-t_{ji}^2(k)/2)$, $J_{ji}^\alpha(k) = e(k) w_{ji}(k) t_{ji}^2(k) \exp(-t_{ji}^2(k)/2)$.

Easy to see that for $\beta = 1$ method (4), (5) has stochastic approximation properties of adaptive identification algorithm of Goodwin-Ramage-Caines [24], and for $\beta = 0$ the learning method – the popular Widrow-Hoff learning algorithm.

As can be seen, the use of the modified quasi-Newtonian learning algorithm does not complicate numerical realization of tuning all parameters of wavelet-neuron and provides an increased convergence rate.

IV. Wavelet-Neuron Learning Algorithm Based on Empirical Risk Minimization

In the case when we have a short data set, the proposed learning algorithm (4), (5) cannot tune all parameters of a network. Thus, the methods based on empirical risk minimization are more effective in this situation.

Introducing the $(nh \times 1)$ – vector of adaptive wavelet activation functions

$$\begin{aligned} \varphi(x(k)) = (&\varphi_{11}(x_1(k)), \dots, \varphi_{h1}(x_1(k)), \varphi_{12}(x_2(k)), \dots, \\ &\varphi_{h2}(x_2(k)), \dots, \varphi_{li}(x_i(k)), \dots, \varphi_{1n}(x_n(k)), \dots, \varphi_{hn}(x_n(k)))^T, \end{aligned} \tag{6}$$

and corresponding to it synaptic weights vector of wavelet-neuron

$$w = (w_{11}, \dots, w_{h1}, w_{12}, \dots, w_{h2}, \dots, w_{li}, \dots, w_{1n}, \dots, w_{hn})^T , \tag{7}$$

we can rewrite formula (1) in the form

$$\hat{y}(k) = w^T \varphi(x(k)) . \tag{8}$$

The learning of wavelet-neuron using a least squares support vector machine is connected with the optimization of criterion in the form

$$E(N) = \frac{1}{2}\|w\|^2 + \frac{\gamma}{2} \sum_{k=1}^{N} e^2(k) \tag{9}$$

with regard to constrains in the form of N linear equations

$$y(k) = w^T \varphi(x(k)) + e(k) \tag{10}$$

where $\gamma > 0$ is a regularization parameter.

Optimization of criterion (9) without constrains (10) leads to expression

$$w(N) = \left(\sum_{k=1}^{N} \varphi(x(k)) \varphi^T(x(k)) + \gamma^{-1} I \right)^{-1} \sum_{k=1}^{N} \varphi(x(k)) y(k) \tag{11}$$

(where $I - (nh \times nh)$ identity matrix) that coincides with least squares ridge-estimates (biased estimates).

For taking into account constraints (10), let us introduce the Lagrange function

$$\begin{aligned} L(w, e(k), \lambda(k)) &= E(k) + \sum_{k=1}^{N} \lambda(k)(y(k) - w^T \varphi(x(k)) - e(k)) = \\ &= \frac{1}{2} w^T w + \frac{\gamma}{2} \sum_{k=1}^{N} e^2(k) + \sum_{k=1}^{N} \lambda(k)(y(k) - w^T \varphi(x(k)) - e(k)) \end{aligned} \tag{12}$$

(here $\lambda(k)$ – N indefinite Lagrange multipliers) and system of Karush-Kuhn-Tucker equations

$$\begin{cases} \nabla_w L(w, e(k), \lambda(k)) = w - \sum_{k=1}^{N} \lambda(k) \varphi(x(k)) = \vec{0}_N, \\ \dfrac{\partial L(w, e(k), \lambda(k))}{\partial e(k)} = \gamma e(k) - \lambda(k) = 0, \\ \dfrac{\partial L(w, e(k), \lambda(k))}{\partial e(k)} = y(k) - w^T \varphi(x(k)) - e(k) = 0 \end{cases} \tag{13}$$

where $\vec{0}_N - (N \times 1)$ is a vector that consists of zero elements.

Solution of system (13) can be written in the form

$$\begin{cases} w(N) = \sum_{k=1}^{N} \lambda(k) \varphi(x(k)), \\ \lambda(k) = \gamma e(k), \\ y(k) = w^T(N) \varphi(x(k)) + e(k) \end{cases} \tag{14}$$

or in the matrix form

$$(\gamma^{-1} \mathrm{I}_{NN} + \Omega_{NN}) \begin{pmatrix} \lambda(1) \\ \vdots \\ \lambda(N) \end{pmatrix} = \begin{pmatrix} y(1) \\ \vdots \\ y(N) \end{pmatrix} \tag{15}$$

(here I_{NN} – $(N \times N)$ is an identity matrix) or

$$(\gamma^{-1}\mathrm{I}_{NN}+\Omega_{NN})\Lambda_N = Y_N \qquad (16)$$

(here $\Omega_{NN} = \{\Omega_{pq} = \varphi^T(x(p))\varphi(x(q))\}$, $p = 1,2,\ldots,N$; $q = 1,2,\ldots,N$ from it follows

$$\Lambda_N = (\gamma^{-1}\mathrm{I}_{NN}+\Omega_{NN})^{-1}Y_N\,. \qquad (17)$$

Then an output signal of wavelet-neuron can be written in the form

$$\hat{y}(x) = \left(\sum_{k=1}^{N}\lambda(k)\varphi(x(k))\right)^T \varphi(x)\,. \qquad (18)$$

As can be seen from neuromathematical point of view (the neural network learning theory) and theory of support vector machines (empirical risk minimization), the proposed wavelet least squares support vector machine based on wavelet-neuron is more simple in the implementation, has high speed of operation and requires short volume of a training data set.

If the data are fed sequentially, the process of wavelet least squares support vector machine learning should be organized in on-line mode. When a pair $x(N+1), y(N+1)$ is fed to the input of wavelet-neuron, expression (18) can be written in the form

$$\hat{y}(x) = \left(\sum_{k=1}^{N}\lambda(k)\varphi(x(k)) + \lambda(N+1)\varphi(x(N+1))\right)^T \varphi(x) \qquad (19)$$

or in the matrix form

$$(\gamma^{-1}\mathrm{I}_{N+1,N+1}+\Omega_{N+1,N+1})\begin{pmatrix}\lambda(1)\\ \vdots \\ \lambda(N) \\ ---- \\ \lambda(N+1)\end{pmatrix} = \begin{pmatrix}Y(1)\\ \vdots \\ Y(N) \\ ---- \\ Y(N+1)\end{pmatrix} \qquad (20)$$

or

$$\begin{pmatrix}\Omega_{NN} & |\,\omega_{N+1} \\ ------- & \\ \omega_{N+1}^T & |\,\gamma^{-1}\end{pmatrix}\begin{pmatrix}\Lambda_N \\ ----- \\ \lambda(N+1)\end{pmatrix} = \begin{pmatrix}Y_N \\ ----- \\ y(N+1)\end{pmatrix} \qquad (21)$$

where
$\omega_{N+1} = (\mu^T(x(1))\mu(x(N+1)), \mu^T(x(2))\mu(x(N+1)),\ldots, \mu^T(x(N))\mu^T(x(N+1)))^T$.

Using (21) we can write

$$\Lambda_{N+1} = \begin{pmatrix}\Lambda_N \\ ----- \\ \lambda(N+1)\end{pmatrix} = \begin{pmatrix}\Omega_{NN} & |\,\omega_{N+1} \\ ------- & \\ \omega_{N+1}^T & |\,\gamma^{-1}\end{pmatrix}^{-1}\begin{pmatrix}Y_N \\ ----- \\ y(N+1)\end{pmatrix}. \qquad (22)$$

Using the Frobenius formula for a block matrix in the form [25]

$$M = \begin{pmatrix}A & | & B \\ - & - & - \\ C & | & D\end{pmatrix}, |D| \neq 0,$$

$$M^{-1} = \begin{pmatrix}A & | & B \\ - & - & - \\ C & | & D\end{pmatrix}^{-1} =$$

$$= \begin{pmatrix}K^{-1} & | & -K^{-1}BD^{-1} \\ ------ & - & ----------- \\ -D^{-1}CK^{-1} & | & D^{-1}+D^{-1}CK^{-1}BD^{-1}\end{pmatrix},$$

$$K = A - BD^{-1}C\,,$$

we can write

$$K = \Omega_{NN} - \omega_{N+1}\gamma\omega_{N+1}^T,\;\; K^{-1} = (\Omega_{NN} - \gamma\omega_{N+1}\omega_{N+1}^T)^{-1}\,.$$

Further, it is easy to compute $(N+1)$ –th Lagrange multiplier using an expression in the form

$$\lambda(N+1) = -\gamma\omega_{N+1}^T K^{-1}Y_N + \gamma(1+\gamma\omega_{N+1}^T K^{-1}\omega_{N+1})y(N+1)\,. \qquad (23)$$

Further, using the Sherman-Morisson formula for matrix inversion we can rewrite the learning algorithm in the final form

$$\begin{cases} K^{-1} = \Omega_{NN}^{-1} + \dfrac{\Omega_{NN}^{-1}\omega_{N+1}\omega_{N+1}^T\Omega_{NN}^{-1}}{1-\omega_{N+1}^T\Omega_{NN}^{-1}\omega_{N+1}}, \\ \lambda(N+1) = 1 + \gamma\omega_{N+1}^T K^{-1}(\omega_{N+1} - Y_N). \end{cases} \qquad (24)$$

V. Result of Simulation

To demonstrate the effectiveness of the proposed wavelet least squares support vector machine, the practical problem of time series forecasting, which describes the average monthly temperature in Kharkiv, Ukraine, was solved [26]. Time series consisted of 24 points and, thus, a training sample contained 16 points and 8 points were taken as a testing sample.

Values $x(k-2), x(k-1), x(k)$ were taken as prehistory for the forecasting $x(k+1)$ value. Initial value of shape parameter of the adaptive wavelet activation function was taken $\alpha = 1$. As forecasting quality criterion we used a mean squared error (MSE).

Fig. 3 shows the results of time series forecasting based on wavelet-neuron with different learning algorithms. As can be seen in Fig. 3a, the curves of the actual values (a dashed curve) and forecast ones (a solid curve) are close enough. Fig. 3b shows the results of forecasting using the wavelet-neuron and gradient learning method with a constant step, and Fig. 3c shows the results of wavelet-neuron and the proposed learning algorithm (4) and (5).

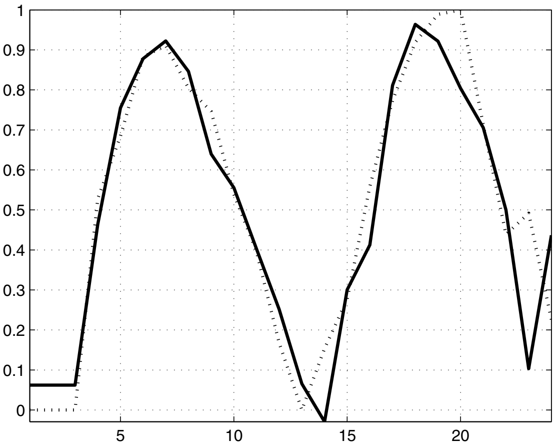

Fig. 3. (a) Result of wavelet-neuron with the proposed learning algorithm based on SVM-criterion.

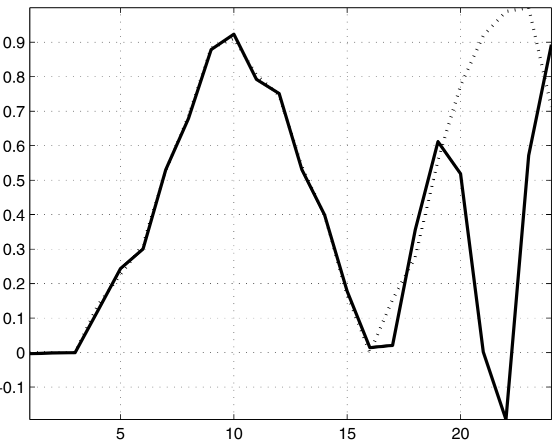

Fig. 3. (b) Results of wavelet-neuron with a gradient learning algorithm with a constant step.

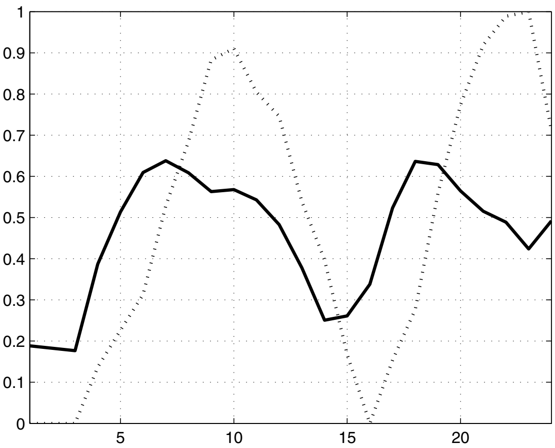

Fig. 3. (c) Results of wavelet-neuron with a learning algorithm of all-parameters (4), (5).

Table I provides the comparison of results of time series forecasting using the wavelet-neuron with different approaches to learning.

TABLE I

THE RESULT COMPARISON OF TIME SERIES FORECASTING

Neural network / Learning algorithm	Num. of input / Num. of activation function	MSEtrn	MSEchk
Wavelet-neuron / Proposed learning algorithm based on SVM-criterion	3/8 (on-line)	0.0063	**0.0306**
Wavelet-neuron/ Gradient learning algorithm with a constant step	3/8 (10 epoch)	**0.000093**	0.3186
Wavelet-neuron/ Proposed learning algorithm of all-parameters (4), (5)	3/8 (10 epoch)	0.0374	0.0572

As it can be seen, the wavelet neuron with a gradient learning algorithm shows the best result on a training set, but such a system has the worst prediction abilities. The wavelet-neuron with learning algorithm (4), (5) is not able to train all parameters of the system because of a small data set. Thus, as can be seen from experimental results, the proposed approach provides the best quality of forecasting in comparison with similar approaches due to a special learning algorithm that is able to process information in both off-line and on-line modes.

VI. CONCLUSION

The wavelet least squares support vector machine based on wavelet-neuron was introduced and investigated. The proposed wavelet least squares support machine has such advantages as computational simplicity due to the wavelet-neuron architecture, small number of tuning parameters, high speed operation thanks to the use of the second order learning algorithms and the possibility of on-line information processing.

REFERENCES

[1] K.-L. Du and M. N. S. Swamy, *Neural Networks and Statistical Learning* London: Springer-Verlag, 2014. http://dx.doi.org/10.1007/978-1-4471-5571-3

[2] S. Haykin, *Neural Networks. A Comprehensive Foundation* Upper Saddle River, N. J.: Prentice Hall, 1999.

[3] J.-S. R. Jang, “ANFIS: Adaptive-network-based fuzzy inference systems,” *IEEE Trans. on Syst. Man. and Cybern.*, vol. 23, issue 3, pp. 665–685, 1993. http://dx.doi.org/10.1109/21.256541

[4] J.-S. R. Jang, C. T. Sun, and E. Mizutani. *Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence*, N. J.: Prentice Hall, 1997.

[5] O. Nelles, *Nonlinear System Identification*, Berlin: Springer, 2001. http://dx.doi.org/10.1007/978-3-662-04323-3

[6] E. Uchino and T. Yamakawa, “Soft computing based signal prediction, restoration and filtering,” in *Intelligent Hybrid Systems: Fuzzy Logic, Neural Networks and Genetic Algorithms*, Da Ruan Eds., Boston: Kluwer Academic Publisher, 1997, pp. 331–349.

[7] Ye. Bodyanskiy and O. Vynokurova, ”Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification,” Information Science, n. 220, pp.170–179, 2013.

[8] L. Ljung, *System Identification: Theory for the User*, N.Y.: Prentice-Hall, 1999.

[9] C. Cortes and V. Vapnik, “Support vector networks,” Machine Learning, n. 20, pp.273–297, 1995. http://dx.doi.org/10.1007/BF00994018

[10] V.N. Vapnik and A.Ya. Chervonenkis, *Pattern Recognition Theory (Statistical Learning Problems)*, M.: Nauka, 1974. (in Russian)

[11] V. N. Vapnik and A. Ya. Chervonenkis, *Empirical Data Dependencies Restoration*, M.: Nauka, 1979. (in Russian)
[12] V. N. Vapnik, *The Nature of Statistical Learning Theory*, N. Y.: Springer, 1995. http://dx.doi.org/10.1007/978-1-4757-2440-0
[13] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*, Singapore: World Scientific, 2002.
[14] S. Abe and D. Tsujinishi, "Fuzzy Least Squares Support Vector Machines for multiclass problems," Neural Networks, n. 16, pp. 785–792, 2003.
[15] Ch.-F. Lin and Sh.-D. Wang, "Fuzzy Support Vector Machines," IEEE Trans. on Neural Networks, n. 13, pp. 646–671, 2002.
[16] S.M. Pandhiani and A.B. Shabri, "Time Series Forecasting Using Wavelet-Least Squares Support Vector Machines and Wavelet Regression Models for Monthly Stream Flow Data," Open Journal of Statistics, n. 3, pp. 183–194, 2013.
[17] D. Kumar, R. K. Tripathy and A. Acharya, "Least squares support vector machine based multiclass classification of EEG signals," WSEAS Transactions on Signal Processing, vol. 10, pp. 86–94, 2014.
[18] D. Zahirniak, R. Chapman, S. K. Rogers, B. W. Suter, M. Kabrisky and V. Pyati, "Pattern recognition using radial basis function network," in *Application of Artificial Intelligence Conf.*, Dayton, OH, 1990, pp. 249–260.
[19] T. Yamakawa, "A novel nonlinear synapse neuron model guaranteeing a global minimum – Wavelet neuron," in *Proc. 28th IEEE Int. Symp. On Multiple-Valued Logic*, Fukuoka, Japan, IEEE Comp. Soc., 1998, pp. 335–336.
[20] Ye. Bodyanskiy, O. Vynokurova, and O. Kharchenko, "Hybrid cascade neural network based on wavelet-neuron," *Information Theories and Application*, vol. 18, no. 4, pp. 335–343, 2011.
[21] Ye. Bodyanskiy, N. Lamonova, I. Pliss, and O. Vynokurova, "An adaptive learning algorithm for a wavelet neural network," *Expert Systems*, vol. 22, no. 5, pp. 235–240, 2005. http://dx.doi.org/10.1111/j.1468-0394.2005.00314.x
[22] Ye. Bodyanskiy, I. Pliss, and O. Vynokurova, "Flexible neo-fuzzy neuron and neuro-fuzzy network for monitoring of time series properties," *Scientific J. of Riga Technical University. Information Technology and Management Science*, vol. 16, pp. 47–52, 2013.
[23] Ye. Bodyanskiy, O. Vynokurova, E. Yegorova, "Radial-basis-fuzzy-wavelet-neural network with adaptive activation-membership function" *Int. J. on Artificial Intelligence and Machine Learning*, no. 8 (II), pp. 9–15, 2008.
[24] G. C. Goodwin, P. J. Ramadge, and R. E. Caines, "A globally convergent adaptive predictor," *Automatica*, vol. 17, no. 1, pp. 135–140, 1981. http://dx.doi.org/10.1016/0005-1098(81)90089-3
[25] F. R. Gantmacher, *The Theory of Matrices* AMS, Chelsea Publishing: Reprinted by American Mathematical Society, 2000.
[26] Data Market – the open portal to thousands of datasets [Online]. Available: http://datamarket.com/en/data/set/1loo/#!ds=1loo!1n6s=2qi.2ql.2qn&display=line&title=Average+monthly+temperatures+across+the+world+(1701-2011)&s=8gd

Yevgeniy Bodyanskiy. In 1971 he graduated with honor from Kharkiv National University of Radio Electronics. In 1980 he defended the Ph.D. Thesis. In 1984 he took an academic title of Senior Researcher. In 1990 he defended the Doctor Thesis (Dr.habil.sci.ing.). In 1994 he took an academic title of Professor. Since 1974 he has been working at Kharkiv National University of Radio Electronics. In 1974-1976 he was a Researcher; in the period of 1977-1983, he was a Senior Researcher; in 1986-1991 he was a Scientific Head of Control Systems Research Laboratory; in 1991-1992 he was a Fellow Researcher. Since 1992 he has been a Professor of the Artificial Intelligence Department KhNURE, Scientific Head of Control Systems Research Laboratory KhNURE. He has more than 600 scientific publications, including 40 inventions and 12 monographs. Research interests include hybrid systems of computational intelligence: adaptive, neuro-, wavelet-, neo-fuzzy-, real-time systems, including problems connected with control, identification, forecasting, clustering, diagnostics, fault detection in technical, economical, medical and ecological objects.
E-mail: bodya@kture.kharkov.ua.

Olena Vynokurova. In 2002 she graduated with honor from Kharkiv National University of Radio Electronics. In the period of 2002-2005, she undertook the postgraduate study in the Artificial Intelligence Department. In 2005 she defended the Ph.D. Thesis. In 2007 she took an academic title of Senior Researcher. In 2012 she defended the Doctor Thesis (Dr.habil.sci.ing.). Since 2002 she has been working at Kharkiv National University of Radio Electronics. Since 2014 she has been a Chief Researcher of the Control Systems Research Laboratory and since 2013 she has been a Professor of IT Security Department at Kharkiv National University of Radio Electronics. She has more than 120 scientific publications, including 2 monographs. Research interests include evolving hybrid systems of computational intelligence: wavelet neural networks, hybrid wavelet neuro-fuzzy systems, identification, forecasting, clustering, diagnostics, fault detection in technical, economical, medical and ecological objects.
E-mail: vinokurova@kture.kharkov.ua.

Oleksandra Kharchenko. She is a Master Student at Kharkiv National University of Radio Electronics. Her major field of research is hybrid neuro-fuzzy systems for data mining problems.
She has 10 scientific publications. Research interests include evolving hybrid systems of computational intelligence: wavelet neural networks, hybrid wavelet neuro-fuzzy systems, forecasting, clustering in technical, economical, medical and ecological objects.
E-mail: kharchenko.alexandra@gmail.com.

Jevgeņijs Bodjanskis, Jeļena Vinokurova, Aleksandra Harčenko. Uz *wavelet* neironiem balstītā minimālo kvadrātu atbalsta vektoru mašīna
Atbalsta vektoru mašīnas (SVM), kuru arhitektūra sakrīt ar RBFN un GRNN, sinoptiskie svari tiek noteikti, risinot nelineārās programmēšanas uzdevumu, bet aktivācijas funkciju centri tiek noteikti pēc principa „neironi datu punktos", kas pēc savas būtības ir dažādu neironu tīklu savdabīgs hibrīds, kuru apmācība bāzējas uz optimizāciju un atmiņu. Neskatoties uz vairākām SVM-tīklu priekšrocībām, no skaitļošanas viedokļa to apmācība ir pietiekami sarežģīts process, jo tas ir saistīts ar lielu dimensiju nelineāro programmēšanu. Tāpēc, kā SVM alternatīva, tika piedāvātas mazāko kvadrātu atbalsta vektoru mašīnas (LS-SVM), kuru apmācība vienkāršojas līdz lineāro vienādojumu sistēmu risināšanai, ko ir daudz vienkāršāk izskaitļot un var realizēt on-line režīmā. Klasiskais SVM *wavelet* analogs ir atbalsta vektoru *wavelet* mašīna (WSVM), kurā daudzdimensiju aktivācijas kodolu funkcijas ir aizvietotas ar adaptīvām viendimensijas *wavelet* funkcijām. Neskatoties uz to, ka WSVM piemīt vairāk iespēju, salīdzinot ar klasiskajām SVM, to aprēķinu apmācība ir saistīta ar pietiekami sarežģītu procedūru realizāciju, kas, loģiski, ierobežo to iespējas pielietošanai reāla laika uzdevumu risināšanai. Sakarā ar to kļūst aktuāla pietiekami vienkāršo *wavelet* neironu sistēmu izstrāde, kas balstās uz empīriskā riska minimizēšanu un orientējas uz informācijas apstrādi on-line režīmā. Kā šādu sistēmu bāzes elementu mēs pieņemam *wavelet* neironus, kam raksturīgas augstas aproksimācijas īpašības, vienkāršība, ātra apmācība un iespēja atklāt datos paslēptās saites.

Евгений Бодянский, Елена Винокурова, Александра Харченко. Машина опорных векторов наименьших квадратов на основе вэйвлет-нейрона
Своеобразным гибридом различных нейронных сетей, обучение которых основывается как на оптимизации, так и на памяти, являются машины опорных векторов (SVM), архитектура которых совпадает с RBFN и GRNN, синаптические веса определяются в результате решения задачи нелинейного программирования, а центры активационных функций устанавливаются по принципу «нейроны в точках данных». И хотя SVM-сети обладают целым рядом несомненным преимуществ, их обучение с вычислительной точки зрения представляется достаточно трудоемким, поскольку связано с решением задач нелинейного программирования высокой размерности. В связи с этим в качестве альтернативы SVM были предложены машины опорных векторов наименьших квадратов (LS-SVM), обучение которых сводится к решению систем линейных уравнений, что гораздо проще с вычислительной точки зрения и может быть реализовано в онлайн режиме. Вэйвлет-аналогом традиционной SVM является вэйвлет машина опорных векторов (WSVM), в которой многомерные ядерные функции активации заменены одномерными адаптивными вэйвлет-функциями. И хотя WSVM обладает большими возможностями по сравнению с традиционными SVM, их обучение с вычислительной точки зрения связано с реализацией достаточно сложных процедур, что, естественно, ограничивает их возможности для решения задач реального времени. В связи с этим представляется целесообразной разработка достаточно простых вэйвлет-нейро-систем, реализующих идеи обучения, основанного на минимизации эмпирического риска и ориентированных на обработку информации в online-режиме. В качестве базового элемента таких систем нами принят вэйвлет-нейрон, характеризующийся высокими аппроксимирующими свойствами, простотой, высокой скоростью обучения и возможностью выявлять скрытые зависимости в обрабатываемых данных.