

Hybrid Generalized Additive Wavelet-Neuro-Fuzzy-System and Its Adaptive Learning

Yevgeniy Bodyanskiy, Olena Vynokurova, Iryna Pliss,
Dmytro Peleshko and Yuriy Rashkevych

Abstract In the paper, a new hybrid generalized additive wavelet-neuro-fuzzy-system of computational intelligence and its learning algorithms are proposed. This system combines the advantages of neuro-fuzzy system of Takagi-Sugeno-Kang, wavelet neural networks and generalized additive models of Hastie-Tibshirani. The proposed system has universal approximation properties and learning capabilities which pertain to the neural networks and neuro-fuzzy systems; interpretability and transparency of the obtained results due to the soft computing systems; possibility of effective description of local signal and process features due to the application of systems based on wavelet transform; simplicity and speed of learning process due to generalized additive models. The proposed system can be used for solving a wide class of dynamic data mining tasks, which are connected with non-stationary, nonlinear stochastic and chaotic signals. Such a system is sufficiently simple in numerical implementation and is characterized by a high speed of learning and information processing.

Keywords Computational intelligence · Soft computing · Wavelet neuro-fuzzy networks · Hybrid intelligent systems

Y. Bodyanskiy (✉) · O. Vynokurova · I. Pliss
Kharkiv National University of Radio Electronics, 14 Nauky av.,
Kharkiv 61166, Ukraine
e-mail: yevgeniy.bodyanskiy@nure.ua

O. Vynokurova
e-mail: olena.vynokurova@nure.ua

I. Pliss
e-mail: iryna.pliss@nure.ua

D. Peleshko · Y. Rashkevych
Lviv Politechnik National University, 12 Bandera Street, Lviv 79013, Ukraine
e-mail: dpeleshko@gmail.com

Y. Rashkevych
e-mail: rashkev@polynet.lviv.ua

1 Introduction

Nowadays computational intelligence methods and especially hybrid systems of computational intelligence [1–3] are wide spread for Data Mining tasks in different areas under uncertainty, non-stationarity, nonlinearity, stochastic, chaotic conditions of the investigated objects and, first of all, in control, identification, prediction, classification, emulation etc. These systems are flexible because they combine effective approximating properties and learning abilities of artificial neural networks, transparency and interpretability of the results obtained by using neuro-fuzzy systems, the possibility of a compact description of the local features of non-stationary signals, providing wavelet neural networks and more advanced wavelet-neuro-fuzzy systems.

At the same time in the framework of such directions as Dynamic Data Mining (DDM) and Data Stream Mining (DSM) [4–6] more (if not the most) of observed systems appear either ineffective or inoperative in general. It connects with that the problems of DDM and DSM must be solved (including learning process) in on-line mode, when the data is fed to the processing sequentially, often in real time. It is clear that traditional multilayer perceptron trained based on back-propagation algorithm and requires the pre-determined training sample cannot operate in such conditions.

It is possible to implement the on-line learning process in the neural networks whose output signal depends on tuning synaptic weights linearly, for example, radial basis function networks [RBFN], normalized radial basis function networks (NRBFN) [7, 8], generalized regression neural networks (GRNN) [9] and like them neural networks. However using of architectures that based on kernel activation functions is complicated, by so-called, curse of dimensionality. Especially often such problem is appeared when using “lazy learning” [10] based on the conception “neurons on data point” [11].

Neuro-fuzzy systems have undoubted advantages over neural networks and first of all the significantly smaller number of tuning synaptic weights that allows to reduce time of learning process. Here it needs to notice the TSK-system [12] and its simplest version—Wang-Mendel system [13], ANFIS, DENFIS, SAFIS [14, 15] and etc. However, in these systems to provide the required approximating properties not only the synaptic weights but also membership functions (centres and widths) must be tuned. Furthermore, the training process of these parameters is performed using backpropagation algorithms in batch mode.

Hybrid wavelet-neuro-fuzzy systems [16], having a number of advantages, are too tedious from computational point of view, which complicates their using in real-time tasks. For solving such kind of problems, so-called, generalized additive models [17] are good. But such systems don't operate under non-stationarity, nonlinearly and chaotic conditions.

In connection with that the development of hybrid system of computational intelligence is preferred. This system has to combine main advantages (the learning ability, the approximation and extrapolation properties, the identification of local

features of signals, the transparency and interpretability of wavelet neuro-fuzzy systems) with simplicity and learning rate of generalized additive models.

2 Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System Architecture

Figure 1 shows the architecture of the proposed hybrid generalized additive wavelet-neuro-fuzzy system (HGAWNFS).

This system consists of four layers of information processing; the first and second layers are similar to the layers of TSK-neuro-fuzzy system. The only difference is that the odd wavelet membership functions “Mexican Hat”, which are

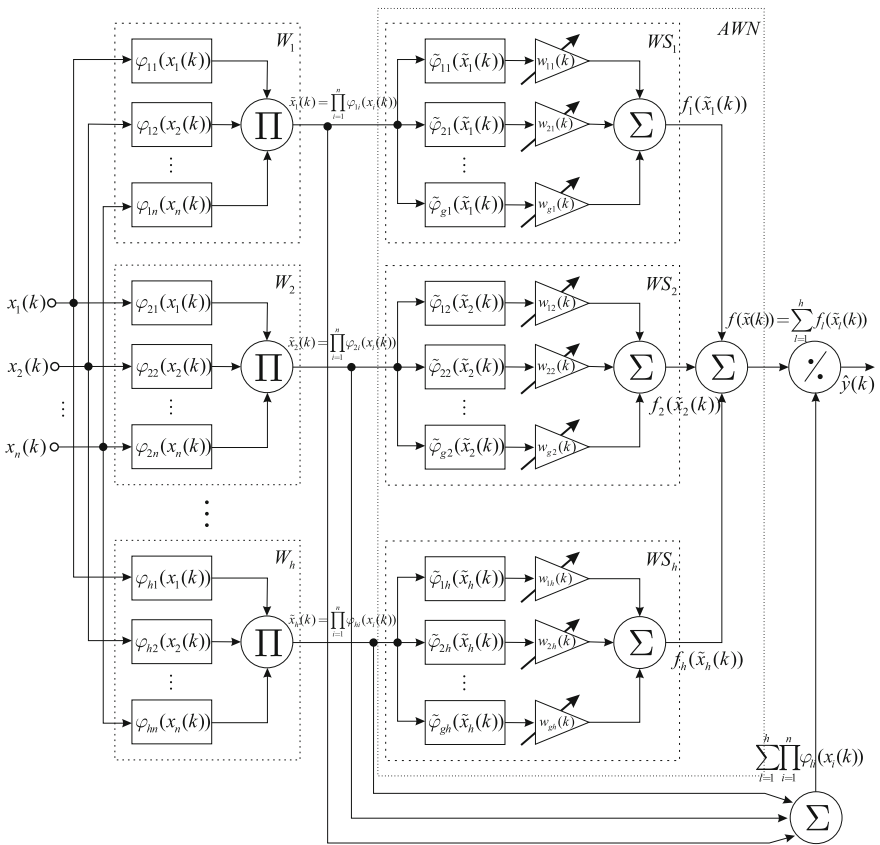


Fig. 1 Hybrid generalized additive wavelet-neuro-fuzzy system architecture

“close relative” of Gaussians, are used instead of conventional bell-shaped Gaussian membership functions in the first hidden layer

$$\varphi_{li}(x_i(k)) = (1 - \tau_{li}^2(k)) \exp(-\tau_{li}^2(k)/2) \quad (1)$$

where $x(k) = (x_1(k), \dots, x_i(k), \dots, x_n(k))^T - (n \times 1)$ is the vector of input signals, $k = 1, 2, \dots$ is a current moment of time, $\tau_{li}(k) = (x_i(k) - c_{li})\sigma_{li}^{-1}$; c_{li}, σ_{li} are the centre and width of the corresponding membership function implying that $\underline{c} \leq c_{li} \leq \bar{c}$; $\underline{\sigma} \leq \sigma_{li} \leq \bar{\sigma}$; $i = 1, 2, \dots, n$; $l = 1, 2, \dots, h$; n is the input number; h is the membership functions number.

It is necessary to note that using the wavelet functions instead of common bell-shaped positive membership functions gives the system more flexibility [18], and using odd wavelets for the fuzzy reasoning does not contradict the ideas of fuzzy inference, because the negative values of these functions can be interpreted as non-membership levels [19].

Thus, if the input vector $x(k)$ is fed to the system input, then in the first layer the hn levels of membership functions $\varphi_{li}(x_i(k))$ are computed and in the hidden layer h vector product blocks perform the aggregation of these memberships in the form

$$\tilde{x}_l(k) = \prod_{i=1}^n \varphi_{li}(x_i(k)). \quad (2)$$

This means the input layers transform the information similarly to the neurons of the wavelet neural networks [20, 21], which form the multidimensional activation functions providing a scatter partitioning of the input space.

At that, therefore, in the region of input space, which remote from centres $c_l = (c_{l1}, \dots, c_{li}, \dots, c_{ln})^T$ of multivariable activation functions

$$\prod_{i=1}^n (1 - \tau_{li}^2(k)) \exp(-\tau_{li}^2(k)/2), \quad (3)$$

the provided quality of approximation can be not high that is common disadvantage of all systems.

To provide the required approximation properties, the third layer of system is formed based on type-2 fuzzy wavelet neuron (T2FWN) [22, 23]. This neuron consists of two adaptive wavelet neurons (AWN) [24], whose prototype is a wavelet neuron of Yamakawa [25]. Wavelet neuron is different from the popular neo-fuzzy neuron [25] that uses the odd wavelet functions instead of the common triangular membership functions. The use of odd wavelet membership functions, which form the wavelet synapses $WS_1, \dots, WS_l, \dots, WS_h$, provides higher quality of approximation in comparison with nonlinear synapses of neo-fuzzy neurons.

In such a way the wavelet neuron performs the nonlinear mapping in the form

$$f(\tilde{x}(k)) = \sum_{l=1}^h f_l(\tilde{x}_l(k)) \quad (4)$$

where $\tilde{x}(k) = (\tilde{x}_1(k), \dots, \tilde{x}_l(k), \dots, \tilde{x}_h(k))^T$, $f(\tilde{x}(k))$ —is the scalar output of wavelet neuron.

Each wavelet synapse WS_l consists of g wavelet membership functions $\tilde{\varphi}_{jl}(\tilde{x}_l)$, $j = 1, 2, \dots, g$ (g is a wavelet membership function number in the wavelet neuron) and the same number of the tuning synaptic weights w_{jl} . Thus, the transform is implemented by each wavelet synapse WS_l in the k -th instant of time, which can be written in form

$$f_l(\tilde{x}_l(k)) = \sum_{j=1}^g w_{jl}(k-1) \tilde{\varphi}_{jl}(\tilde{x}_l(k)) \quad (5)$$

(here $w_{jl}(k-1)$ is the value of synaptic weights that are computed based on previous $k-1$ observations), and the general wavelet neuron performs the nonlinear mapping in the form

$$\tilde{f}(\tilde{x}(k)) = \sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \tilde{\varphi}_{jl}(\tilde{x}_l(k)) \quad (6)$$

i.e., in fact, this is the generalised additive model [17] that is characterised by the simplicity of computations and high approximation properties.

The output layer of system is formed by summator unit and it can be written in form

$$\sum_{l=1}^h \prod_{i=1}^n \varphi_{li}(x_i(k)) = \sum_{l=1}^h \tilde{x}_l(k) \quad (7)$$

and by division unit, which realizes the normalization for avoiding of “gaps” appearance in the parameters space.

In such a way the output of HGAWNFS can be written in form

$$\begin{aligned} \hat{y}(k) &= \frac{\sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \tilde{\varphi}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^h \tilde{x}_l(k)} = \frac{\sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \tilde{\varphi}_{jl}(\prod_{i=1}^n \varphi_{li}(x_i(k)))}{\sum_{l=1}^h \prod_{i=1}^n \varphi_{li}(x_i(k))} \\ &= \sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \frac{\tilde{\varphi}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^h \tilde{x}_l(k)} = \sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \tilde{\psi}_{jl}(\tilde{x}(k)) = w^T(k-1) \psi(\tilde{x}(k)) \end{aligned} \quad (8)$$

where $\tilde{\psi}_{jl}(\tilde{x}(k)) = \tilde{\varphi}_{jl}(\tilde{x}_l(k)) \left(\sum_{l=1}^h \tilde{x}_l(k) \right)^{-1} = \tilde{\varphi}_{jl}(\prod_{i=1}^n \varphi_{li}(x_i(k))) \left(\sum_{l=1}^h \prod_{i=1}^n \varphi_{li}(x_i(k)) \right)^{-1}$, $w(k-1) = (w_{11}(k-1), w_{21}(k-1), \dots, w_{g1}(k-1), w_{12}(k-1), \dots,$

$$w_{jl}(k-1), \dots, w_{gh}(k-1))^T, \quad \tilde{\psi}(\tilde{x}(k)) = (\tilde{\psi}_{11}(\tilde{x}(k)), \tilde{\psi}_{21}(\tilde{x}(k)), \dots, \tilde{\psi}_{jl}(\tilde{x}(k)), \dots, \tilde{\psi}_{gh}(\tilde{x}(k)))^T.$$

3 Adaptive Learning Algorithm of Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System

The learning process of HGAWNFS is reduced in the simplest case to the synaptic weights tuning of wavelet neuron in the third hidden layer. For tuning of wavelet neuron its authors [25] used the gradient procedure which minimizes the learning criterion

$$E(k) = \frac{1}{2} (y(k) - \hat{y}(k))^2 = \frac{1}{2} e^2(k) = \frac{1}{2} \left(y(k) - \sum_{l=1}^h \sum_{j=1}^g w_{jl} \tilde{\psi}_{jl}(\tilde{x}(k)) \right)^2 \quad (9)$$

and it can be written in form

$$\begin{aligned} w_{jl}(k) &= w_{jl}(k-1) + \eta e(k) \tilde{\psi}_{jl}(\tilde{x}(k)) = w_{jl}(k-1) + \eta (y(k) - \hat{y}(k)) \tilde{\psi}_{jl}(\tilde{x}(k)) \\ &= w_{jl}(k-1) + \eta \left(y(k) - \sum_{l=1}^h \sum_{j=1}^g w_{jl}(k-1) \tilde{\psi}_{jl}(\tilde{x}(k)) \right) \tilde{\psi}_{jl}(\tilde{x}(k)) \end{aligned} \quad (10)$$

where $y(k)$ is reference signal, $e(k)$ is learning error, η is fixed learning rate parameter.

For the speed acceleration of tuning process of synaptic weights under non-stationary conditions the exponential weighted recurrent least squares method can be used in form

$$\begin{cases} w(k) = w(k-1) + \frac{P(k-1)e(k)\tilde{\psi}(\tilde{x}(k))}{\beta + \tilde{\psi}^T(\tilde{x}(k))P(k-1)\tilde{\psi}(\tilde{x}(k))}, \\ P(k) = \frac{1}{\beta} \left(P(k-1) - \frac{P(k-1)\tilde{\psi}(\tilde{x}(k))\tilde{\psi}^T(\tilde{x}(k))P(k-1)}{\beta + \tilde{\psi}^T(\tilde{x}(k))P(k-1)\tilde{\psi}(\tilde{x}(k))} \right) \end{cases} \quad (11)$$

(where $0 < \beta \leq 1$ is forgetting factor), which, therefore, can be numerical unstable for high tuning parameters number.

Under uncertain, stochastic or chaotic conditions, it is more effective to use the adaptive wavelet neuron (AWN) [26] instead of common wavelet neuron. In this case, we can tune not only synaptic weights, but the parameters of centres, widths and shapes.

The basis of adaptive wavelet neuron is the adaptive wavelet activation function that was proposed in [22] and can be written in form

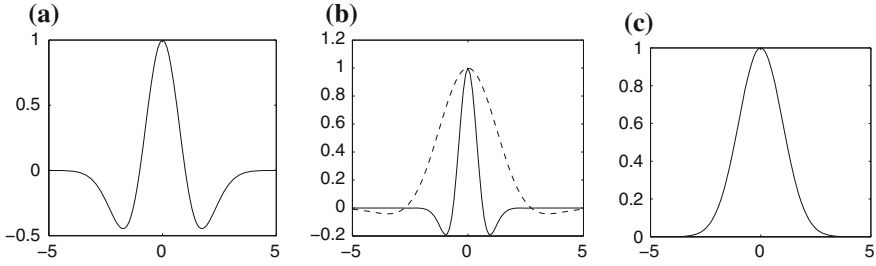


Fig. 2 Adaptive wavelet activation function: **a** $\alpha = 1, \sigma = 1$; **b** dashed line $\alpha = 0.3, \sigma = 1.5$; solid line $\alpha = 0.6, \sigma = 0.5$; **c** $\alpha = 0, \sigma = 1$

$$\tilde{\varphi}_{jl}(\tilde{x}_l(k)) = (1 - \alpha_{jl}(k)\tau_{jl}^2(k)) \exp(-\tau_{jl}^2(k)/2) \quad (12)$$

where $0 \leq \alpha_{jl} \leq 1$ is the shape parameter of adaptive wavelet function, if $\alpha_{jl} = 0$ it is conventional Gaussian, if $\alpha_{jl} = 1$ it is the wavelet ‘‘Mexican Hat’’, and if $0 < \alpha_{jl} < 1$ it is some hybrid activation-membership function (see Fig. 3).

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

Basically to tune the centers, widths and shapes parameters we can use optimization of the learning criterion (9) by the gradient procedure (10), calculated the partial derivative on c_{jl}, σ_{jl}^{-1} and α_{jl} , but for the increasing speed of learning process we can use the one-step modification of Levenberg-Marquardt algorithm [27] for tuning all-parameters of each wavelet synapse simultaneously.

By introducing in the consideration $(g \times 1)$ -vectors $w_l(k) = (w_{1l}(k), w_{2l}(k), \dots, w_{gl}(k))^T$, $\tilde{\psi}_l(\tilde{x}_l(k)) = (\tilde{\psi}_{1l}(\tilde{x}_l(k)), \tilde{\psi}_{2l}(\tilde{x}_l(k)), \dots, \tilde{\psi}_{gl}(\tilde{x}_l(k)))^T$, $c_l(k) = (c_{1l}(k), c_{2l}(k), \dots, c_{gl}(k))^T$, $\sigma_l^{-1}(k) = (\sigma_{1l}^{-1}(k), \sigma_{2l}^{-1}(k), \dots, \sigma_{gl}^{-1}(k))^T$, $\alpha_l(k) = (\alpha_{1l}(k), \alpha_{2l}(k), \dots, \alpha_{gl}(k))^T$, $\tau_l(k) = (\tau_{1l}(k), \tau_{2l}(k), \dots, \tau_{gl}(k))^T$, we can write the learning algorithm in form

$$\left\{ \begin{array}{l} w_l(k) = w_l(k-1) + \frac{e(k)\tilde{\psi}_l(\tilde{x}_l(k))}{\eta^w + \|\tilde{\psi}_l(\tilde{x}_l(k))\|^2} = w_l(k-1) + \frac{e(k)\tilde{\psi}_l(\tilde{x}_l(k))}{\eta^w}, \\ c_l(k) = c_l(k-1) + \frac{e(k)\tilde{\psi}_l^c(\tilde{x}_l(k))}{\eta^c + \|\tilde{\psi}_l^c(\tilde{x}_l(k))\|^2} = c_l(k-1) + \frac{e(k)\tilde{\psi}_l^c(\tilde{x}_l(k))}{\eta^c}, \\ \sigma_l^{-1}(k) = \sigma_l^{-1}(k-1) + \frac{e(k)\tilde{\psi}_l^\sigma(\tilde{x}_l(k))}{\eta^\sigma + \|\tilde{\psi}_l^\sigma(\tilde{x}_l(k))\|^2} = \sigma_l^{-1}(k-1) + \frac{e(k)\tilde{\psi}_l^\sigma(\tilde{x}_l(k))}{\eta^\sigma}, \\ \alpha_l(k) = \alpha_l(k-1) + \frac{e(k)\tilde{\psi}_l^\alpha(\tilde{x}_l(k))}{\eta^\alpha + \|\tilde{\psi}_l^\alpha(\tilde{x}_l(k))\|^2} = \alpha_l(k-1) + \frac{e(k)\tilde{\psi}_l^\alpha(\tilde{x}_l(k))}{\eta^\alpha} \end{array} \right. \quad (13)$$

where $\tilde{\psi}_l^c(\tilde{x}_l(k)) = 2w_l(k-1)\sigma_l^{-1}(k-1)((2\alpha_l(k-1)+1)\tau_l(\tilde{x}_l(k)) - \alpha_l(k-1)\tau_l^3(\tilde{x}_l(k))) \times \exp(-\tau_l^2(\tilde{x}_l(k))/2)$; $\tilde{\psi}_l^\sigma(\tilde{x}_l(k)) = w_l(k-1)(\tilde{x}_l(k) - c_l(k-1))(\alpha_l(k-1)\tau_l^3(\tilde{x}_l(k)) - (2\alpha_l(k-1)+1)\tau_l(\tilde{x}_l(k))) \exp(-\tau_l^2(\tilde{x}_l(k))/2)$; $\tilde{\psi}_l^z(\tilde{x}_l(k)) = -w_l(k-1)\tau_l^2(\tilde{x}_l(k)) \times \exp(-\tau_l^2(\tilde{x}_l(k))/2)$; $\tau_l^2(\tilde{x}_l(k)) = \sigma_l^{-1}(k) \odot \sigma_l^{-1}(k) \odot (\tilde{x}_l(k)I_l - c_l(k-1)) \odot (\tilde{x}_l(k)I_l - c_l(k-1))$, \odot is direct product symbol, I_l is $(l \times 1)$ —the unit vector, $\eta^w, \eta^c, \eta^\sigma, \eta^z$ are nonnegative momentum terms.

For increasing of filtering properties of learning procedure the denominators in recurrent equation system (13) can be modified in such way

$$\begin{cases} \eta^w(k) = \beta\eta^w(k-1) + \|\tilde{\psi}_l(\tilde{x}_l(k))\|^2, \\ \eta^c(k) = \beta\eta^c(k-1) + \|\tilde{\psi}_l^c(\tilde{x}_l(k))\|^2, \\ \eta^\sigma(k) = \beta\eta^\sigma(k-1) + \|\tilde{\psi}_l^\sigma(\tilde{x}_l(k))\|^2, \\ \eta^z(k) = \beta\eta^z(k-1) + \|\tilde{\psi}_l^z(\tilde{x}_l(k))\|^2 \end{cases} \quad (14)$$

where $0 \leq \beta \leq 1$ have the same sense that in the algorithm (11).

4 Robust Learning Algorithm of Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System

Although the square criterion allows to obtain the optimal evaluation when the processed signal and disturbances have Gaussian distribution, but when the distribution has, so-called, “heavy tails” (for example, Laplace and Cauchy distribution etc.) the evaluation which based on quadratic criterion can be inadequate. In this case the robust methods with M-criterion are more effective [28].

Introducing for the consideration the modified Welsh robust identification criterion in the form

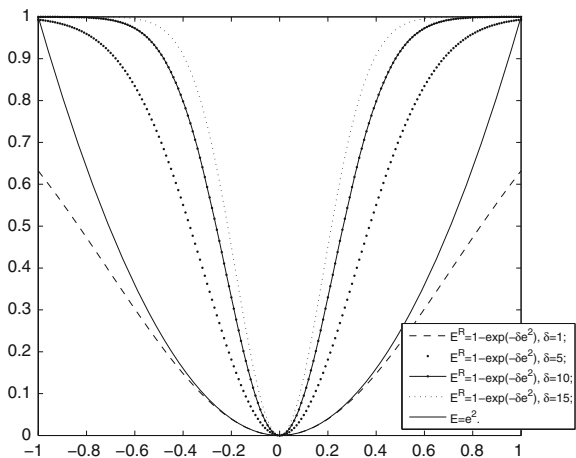
$$E_R(k) = 1 - \exp(-\delta e^2(k)) \quad (15)$$

where $e(k)$ is the learning error, δ is positive parameter, which set from empirical reasoning and defined the size of nonsensitivity zone for the outliers.

Figure 3 shows the comparison the robust identification criterion with different values of parameter δ and least squares criterion.

Providing the sequence of the same transformation we can write the robust learning algorithm in form

Fig. 3 Robust identification criterion (6) with different values of parameter δ



$$\begin{cases} w_l(k) = w_l(k-1) + \lambda_w \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l(\tilde{x}_l(k)) / \eta^w \right), \\ \eta^w(k) = \beta \eta^w(k-1) + \|\tilde{\psi}_l(\tilde{x}_l(k))\|^2, \\ c_l(k) = c_l(k-1) + \lambda_c \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^c(\tilde{x}_l(k)) / \eta^c \right), \\ \eta^c(k) = \beta \eta^c(k-1) + \|\tilde{\psi}_l^c(\tilde{x}_l(k))\|^2, \end{cases} \quad (16)$$

$$\begin{cases} \sigma_l^{-1}(k) = \sigma_l^{-1}(k-1) + \lambda_\sigma \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^\sigma(\tilde{x}_l(k)) / \eta^\sigma \right), \\ \eta^\sigma(k) = \beta \eta^\sigma(k-1) + \|\tilde{\psi}_l^\sigma(\tilde{x}_l(k))\|^2, \\ \alpha_l(k) = \alpha_l(k-1) + \lambda_\alpha \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^\alpha(\tilde{x}_l(k)) / \eta^\alpha \right), \\ \eta^\alpha(k) = \beta \eta^\alpha(k-1) + \|\tilde{\psi}_l^\alpha(\tilde{x}_l(k))\|^2. \end{cases} \quad (17)$$

5 Conclusions

In this paper, the hybrid generalised additive wavelet-neuro-fuzzy system and its learning algorithms have been proposed. This system combines advantages of neuro-fuzzy system of Takagi-Sugeno-Kang, wavelet neural networks and generalised additive models of Hastie-Tibshirani.

The proposed hybrid system is characterised by computation simplicity, improving approximation and extrapolation properties as well as high speed of learning process. Hybrid generalized additive wavelet-neuro-fuzzy system can be used to solve a wide class of tasks in Dynamic Data Mining and Data Stream Mining, which are related to on-line processing (prediction, emulation,

segmentation, on-line fault detection etc.) of non-stationary stochastic and chaotic signals corrupted by disturbances. The computational experiments are confirmed to effectiveness of developed approach.

References

1. Rutkowski, L.: Computational Intelligence: Methods and Techniques. Springer, Berlin (2008). <http://www.springer.com/us/book/9783540762874>
2. Du, K.-L., Swamy, M.N.S.: Neural Networks and Statistical Learning. Springer, London (2014). <http://www.springer.com/us/book/9781447155706>
3. Mumford, C.L., Jain, L.C.: Computational Intelligence Collaboration, Fusion and Emergence. Springer, Berlin (2009). <http://www.springer.com/us/book/9783642017988>
4. Lughofer, E.: Evolving Fuzzy Systems—Methodologies, Advanced Concepts and Applications. Springer (2011). <http://www.springer.com/us/book/9783642180866>
5. Aggarwal, C.: Data Streams: Models and Algorithms (Advances in Database Systems). Springer (2007). <http://www.springer.com/us/book/9780387287591>
6. Bifet, A.: Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams. IOS Press, Amsterdam (2010). <http://www.iospress.nl/book/adaptive-stream-mining-pattern-learning-and-mining-from-evolving-data-streams/>
7. Sunil, E.V.T., Yung, CSh: Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems. *IEEE Trans. Neural Netw.* **5**, 594–603 (1994)
8. Bugmann, G.: Normalized Gaussian radial basis function networks. *Neurocomputing* **20**(1–3), 97–110 (1998)
9. Specht, D.F.: A general regression neural network. *IEEE Trans. Neural Networks* **2**(6), 568–576 (1991)
10. Nelles O.: Nonlinear System Identification. Springer, Berlin (2001). <http://www.springer.com/jp/book/9783540673699>
11. Zahirniak, D., Chapman, R., Rogers, S., Suter, B., Kabritsky, M., Piati, V.: Pattern recognition using radial basis function network. In: 6th Annual Aerospace Application of Artificial Intelligence Conference, pp. 249–260, Dayton, OH (1990)
12. Jang, J.-S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Prentice Hall, NJ (1997). <http://www.pearsonhighered.com/educator/product/NeuroFuzzy-and-Soft-Computing-A-Computational-Approach-to-Learning-and-Machine-Intelligence/9780132610667.page>
13. Mendel, J.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Upper-Saddle River, NJ (2001). <http://www.pearsonhighered.com/educator/product/Uncertain-RuleBased-Fuzzy-Logic-Systems-Introduction-and-New-Directions/9780130409690.page>
14. Kasabov, N.K., Qun, S.: DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Trans. Fuzzy Syst.* **10**(2), 144–154 (2002)
15. Rong, H.J., Sundararajan, N., Huang, G.-B., Saratchandran, P.: Sequential Adaptive Fuzzy Inference System (SAFIS) for nonlinear system identification and prediction. *Fuzzy Sets Syst.* **157**(9), 1260–1275 (2006)
16. Abiyev, R., Kaynak, O.: Fuzzy wavelet neural networks for identification and control of dynamic plants—a novel structure and a comparative study. *IEEE Trans. Ind. Electron.* **2**(55), 3133–3140 (2008)
17. Hastie, T., Tibshirani, R.: Generalized Additive Models. Chapman and Hall/CRC (1990)
18. Takagi, H., Hayashi, I.: NN-driven fuzzy reasoning. *Int. J. Approx. Reason.* **5**(3), 191–212 (1991)

19. Mitaim, S., Kosko, B.: What is the best shape for a fuzzy set in function approximation? In: Proceedings of the 5th IEEE International Conference on Fuzzy Systems "Fuzz-96", vol. 2, pp. 1237–1213 (1996)
20. Alexandridis, A.K., Zapanis, A.D.: Wavelet Neural Networks: With Applications in Financial Engineering, Chaos, and Classification. Wiley (2014). <http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118592522.html>
21. Bodyanskiy, Y., Lamonova, N., Pliss, I., Vynokurova, O.: An adaptive learning algorithm for a wavelet neural network. *Expert Syst.* **22**(5), 235–240 (2005)
22. Bodyanskiy, Y., Vynokurova, O.: Hybrid type-2 wavelet-neuro-fuzzy network for businesses process prediction. *Bus. Inform.* **21**, 9–21 (2011)
23. Bodyanskiy, Y., Pliss I., Vynokurova, O.: Type-2 fuzzy-wavelet-neuron for solving data mining problems. In: Proceedings of the East West Fuzzy Colloquium 2012, 19th Zittau Fuzzy Colloquium. Zittau/Goerlitz: HS, pp. 96–103 (2012)
24. Bodyanskiy, Y., Kharchenko, O., Vynokurova O.: Least squares support vector machine based on wavelet-neuron. *Inform. Technol. Manag. Sci.* **7**, 19–24 (2014)
25. Yamakawa, T., Uchino, E., Miki, T., Kusanagi, H.: A neo-fuzzy neuron and its applications to system identification and prediction of the system behaviour. In: Proceedings of the 2nd International Conference on Fuzzy Logic and Neural Networks "IIZUKA-92", pp. 477–483, Iizuka, Japan (1992)
26. Bodyanskiy, Y., Vynokurova, O., Kharchenko, O.: Least squares support vector machine based on wavelet-neuron. *Inform. Technol. Manag. Sci.* **17**, 19–24 (2014)
27. Wang, L.: Adaptive Fuzzy Systems and Control. Design and Stability Analysis. Prentice Hall, New Jersey (1994)
28. Cichocki, A., Unbehauen, R.: Neural Networks for Optimization and Signal Processing. Teubner, Stuttgart (1993)