ACADEMIC PRESS

Available at www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Journal of Computer and System Sciences 67 (2003) 623–632

JOURNAL OF COMPUTER AND SYSTEM SCIENCES

http://www.elsevier.com/locate/jcss

Complex-valued wavelet network

Chunguang Li,[a,*] Xiaofeng Liao,[a,b] and Juebang Yu[a]

[a] *Lab 570, Institute of Electronic Systems, College of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu 610054, People's Republic of China*

[b] *Department of Computer Science and Engineering, Chongqing University, Chongqing 400044, People's Republic of China*

Received 12 November 2001; revised 31 March 2003

Abstract

In this paper, a complex-valued wavelet network (CWN) is proposed. The network has complex inputs, outputs, connection weights, dilation and translation parameters, but the nonlinearity of the hidden nodes remains a real-valued function (real-valued wavelet function). This kind of network is able to approximate an arbitrary nonlinear function in complex multi-dimensional space, and it provides a powerful tool for nonlinear signal processing involving complex signals. A complex algorithm is derived for the training of the proposed CWN. A numerical example on nonlinear communication channel identification is presented for illustration.

Keywords: Neural network; Wavelet; Complex-valued; Identification

1. Introduction

In recent years, neural networks have been widely studied because of their outstanding capability of fitting nonlinear models. As wavelet has emerged as a new powerful tool for representing nonlinearity, a class of networks combining wavelets and neural networks has recently been investigated [5,13,15]. It has been shown that wavelet networks provide better function approximation ability than the multilayer perception (MLP) and radial basis function (RBF) networks.

In some applications, however, the inputs and outputs of a system are best described as complex-valued signals and processing is done in complex space. An example is the identification of digital communication channels with complex signal schemes such as quadrature amplitude

*Corresponding author.
E-mail address: cgli@uestc.edu.cn (C. Li).

0022-0000/03/$ - see front matter © 2003 Elsevier Science (USA). All rights reserved.
doi:10.1016/S0022-0000(03)00069-2

modulation (QAM). For complex signal processing problems, many existing neural networks cannot directly be applied. Although for certain applications it is possible to reformulate a complex signal processing problem so that a real-valued network and learning algorithm can be used to solve the problem, it is not always feasible to do so. Moreover it is preferred to preserve the concise formulation and elegant structure of complex signal.

Recently, results have appeared in the literature that generalize the well-known back-propagation (BP) algorithm for training a feed-forward neural network with complex weights [2,8,10], the complex BP algorithm has been shown to be a straight forward extension of the real-valued one. In [9], the authors extended the Real Time Recurrent Learning (RTRL) algorithm to the complex RTRL (CRTRL) and applying it to complex communication channel equalization. And the complex-valued RBF neural networks were proposed and applied to nonlinear communication channel identification and equalization in [3,4]. The complex spline neural network was presented in [14].

The advantage of using complex-valued neural network instead of a real-valued neural network counterpart fed with a pair of real values is well known [1]. In complex-valued neural networks, one of the main problems is the selecting of nodes activation function. In real case, the node activation function is usually chosen to be a continuous, bounded and nonconstant function. These conditions on the activation function are very mild and there is no problem in selecting a real function that satisfies these requirements and that is also smooth (derivative exists). In the complex case, any regular analytic function cannot be bounded unless it reduces to a constant. This is known as the Liouville's theorem. In complex case, the main constraints that the activation function should satisfy can be found in literatures [6,7].

The present study proposes a complex wavelet network, which is an extension of the real-valued wavelet network. The inputs, outputs, weights, dilation and translation parameters are all complex-valued, but the nonlinearity of the hidden nodes remains a real wavelet function. The network can be viewed as a mapping from the complex multi-dimensional input space onto the complex output space. The following of this paper is organized as follows. In Section 2, we briefly introduce the wavelet networks. We propose the CWN architecture in Section 3. And the complex-valued learning algorithm for the training of CWN is derived in Section 4. In Section 5, we evaluate the performance of the proposed CWN through applying it to the identification of a nonlinear complex communication channels. And finally in Section 6, we summarize our conclusions.

2. Wavelet networks

Wavelet is a new powerful tool for representing nonlinearity. A function $f(x)$ can be represented by the superposition of daughters $\psi_{a,b}(x)$ of a mother wavelet $\psi(x)$. Where $\psi_{a,b}(x)$ can be expressed as

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \tag{1}$$

$a \in R_+$ and $b \in R$ are, respectively, called dilation and translation parameters.

The continuous wavelet transform of $f(x)$ is defined as

$$w(a,b) = \int_{-\infty}^{\infty} f(x)\overline{\psi_{a,b}(x)}\,dx \tag{2}$$

and the function $f(x)$ can be reconstructed by the inverse wavelet transform

$$f(x) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} w(a,b)\psi_{a,b}(x)\,\frac{da\,db}{a^2}. \tag{3}$$

The continuous wavelet transform and its inverse transform are not directly implementable on digital computers. When the inverse wavelet transform (3) is discretized, $f(x)$ has the following approximative wavelet-based representation form:

$$\hat{f}(x) \approx \sum_{k=1}^{K} w_k \psi\left(\frac{x-b_k}{a_k}\right), \tag{4}$$

where the w_k, b_k, and a_k are weight coefficients, translations and dilations for each daughter wavelet. This approximation can be expressed as the neural network of Fig. 1, which contains wavelet nonlinearities in the artificial neurons rather than the standard sigmoidal nonlinearities.

The wavelet network (4) can only be used to approximate single-input-single-output (SISO) functions. It can be extended to M-D case. The simplest scheme is to use the M-D wavelets constructed from the tensor products of 1-D wavelets [15]. However, the complexity of each wavelet increases when the dimension increases. As a result, a rather complex network architecture is inevitable.

The theoretical result proved in [16] pointed out that if $w(x)$ is a measurable function and $w(x) \in L^q[a,b]$ $(1 \leqslant q \leqslant \infty)$ for every $a, b \in R$, $a < b$ and the set of finite linear combinations $\sum h_i w(\xi_i x + \eta_i)$, where $h_i, \xi_i, \eta_i \in R$, are dense in any $L^q[a,b]$, then the set of finite linear combinations $\sum d_i w(z_i x + \theta_i)$ are dense in $L^q(K)$, where $d_i, \theta_i \in R$, $z_i, x \in R^q$ and K is a compact set in R^q. Combining the result with the above discussion on (4), we can use a model of neural network with N inputs as suggested in [5].

$$\hat{f}(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \psi\left(\frac{\sum_{n=1}^{N} c_{kn} x_n - b_k}{a_k}\right), \tag{5}$$

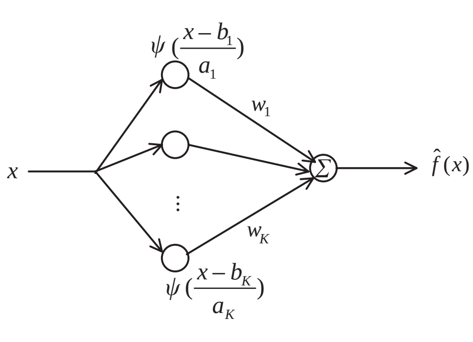

Fig. 1. Wavelet neural network architecture.

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_N] \in R^N$ and c_{kn} are the connection weight from the nth input to the kth wavelet neuron. Thus, only 1-D wavelet bases are used in the wavelet neural network (5) for any dimension case. The network architecture is simplified and simple learning method can be used to search for optimal weights.

3. The complex wavelet network architecture

Since any multi-input-multi-output (MIMO) system can be decomposed into multi-input-single-output (MISO) systems, we only consider the MISO CWN as depicted in Fig. 2 in this paper.

Denote the number of input and wavelet neuron as N and K, respectively. As usual, a complex quantity is defined as

$$y = \mathrm{Re}(y) + j\,\mathrm{Im}(y) = y_\mathrm{R} + jy_\mathrm{I}, \tag{6}$$

where the subscript R and I denote the real and imaginary part, respectively, and $j = \sqrt{-1}$ denote the imaginary unit vector, the asterisk denotes complex conjugate. The input $x_i,\ i = 1, \ldots, N$, the output y, the connected weight c_{ij} (connected from the jth input node to the ith wavelet neuron), w_i (connected from the ith wavelet neuron to the output node), and the total input to the ith wavelet neuron s_i are all complex-valued. From Fig. 2, we can obtain

$$s_i = \boldsymbol{c}_i^H \boldsymbol{x}, \tag{7}$$

where $\boldsymbol{c}_i = [c_{i1}, c_{i2}, \ldots, c_{iN}]^T,\ \boldsymbol{x} = [x_1, x_2, \ldots, x_N]^T$, the superscript H denotes Hermitian transposition. The outputs of hidden nodes are given by

$$h_i = \psi_i(s_i) = \psi\left(\left(\frac{s_i - b_i}{a_i}\right)^H \left(\frac{s_i - b_i}{a_i}\right)\right) \triangleq \psi(z_i), \tag{8}$$

where $z_i = (\frac{s_i - b_i}{a_i})^H (\frac{s_i - b_i}{a_i}),\ b_i$ and a_i are complex dilation and translation parameters, respectively. $\psi(\cdot)$ is a real wavelet function and is referred to as the nonlinearity of the hidden nodes. The selection of the real wavelet function is the same as in the case of real wavelet networks. Two typical examples are the Harr wavelet function

$$\psi(x^2) = (1 - x^2)e^{-\frac{1}{2}x^2} \tag{9}$$

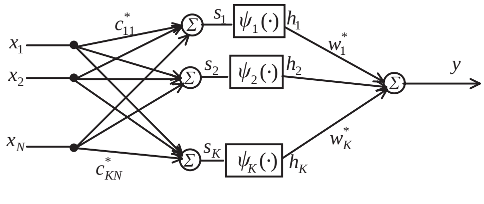

Fig. 2. MISO complex wavelet network architecture.

and the so-called "difference of Gaussian" wavelet function

$$\psi(x^2) = e^{-\frac{1}{2}x^2} - \frac{1}{2}e^{-\frac{1}{8}x^2}; \tag{10}$$

the output node is a complex linear combiner defined by

$$y = \boldsymbol{w}^H \boldsymbol{h}, \tag{11}$$

where $\boldsymbol{w} = [w_1, \ldots, w_K]^T$, $\boldsymbol{h} = [h_1, \ldots, h_K]^T$.

4. The complex-valued backpropagation of CWN

The backpropagation (BP) algorithm [11] is one of the most popular learning algorithms in neural networks. This algorithm performs an approximation to the minimization achieved by the method of steepest descent.

The error signal $e(n)$ required for adaptation is defined as the difference between the desired response $d(n)$ and the actual output of the CWN $y(n)$:

$$e(n) = d(n) - y(n); \tag{12}$$

we can define the objective function as

$$E(n) = |e(n)|^2 = e(n)e^*(n). \tag{13}$$

The BP algorithm minimizes the objective function $E(n)$ by recursively adjusting the parameters $\{w_i, c_{ij}, b_i, a_i\}$ based on the gradient search technique. Thus, finding the gradient vector of $E(n)$ is the main idea of deriving the BP algorithm. We first find the partial derivative of $E(n)$ with respect to the weights w_i, and then extend to all parameters. We define the gradient vector $\nabla_w E(n)$ as the derivative of the cost function $E(n)$ with respect to the real and imaginary parts of the weight vector w as shown by

$$\nabla_w E(n) = \frac{\partial E(n)}{\partial w_{\mathrm{R}}} + j\frac{\partial E(n)}{\partial w_{\mathrm{I}}}. \tag{14}$$

By substituting (13) into (14), we can get

$$\nabla_w E(n) = e(n)\frac{\partial e^*(n)}{\partial w_{\mathrm{R}}} + e^*(n)\frac{\partial e(n)}{\partial w_{\mathrm{R}}} + je(n)\frac{\partial e^*(n)}{\partial w_{\mathrm{I}}} + je^*(n)\frac{\partial e(n)}{\partial w_{\mathrm{I}}}. \tag{15}$$

Differentiating $e(n)$ in Eq. (12) with respect to the real and imaginary parts of w, we can obtain the following four partial derivatives:

$$\frac{\partial e(n)}{\partial w_{\mathrm{R}}} = -h(n), \quad \frac{\partial e^*(n)}{\partial w_{\mathrm{R}}} = -h^*(n), \quad \frac{\partial e(n)}{\partial w_{\mathrm{I}}} = jh(n), \quad \frac{\partial e^*(n)}{\partial w_{\mathrm{I}}} = -jh^*(n). \tag{16}$$

By substituting Eq. (16) into (15), we can obtain the following result:

$$\nabla_w E(n) = -2h(n)e^*(n). \tag{17}$$

Thus the update equation of $w(n)$ can be expressed as

$$w(n+1) = w(n) - \frac{1}{2}\mu_1 \nabla_w E(n) = w(n) + \mu_1 h(n) e^*(n). \tag{18}$$

where μ_1 is the step size.

Similarly, we have

$$\begin{aligned}\frac{\partial E(n)}{\partial c_{i\mathrm{R}}} &= \frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\left[\frac{\partial z_i(n)}{\partial s_{i\mathrm{R}}}\frac{\partial s_{i\mathrm{R}}}{\partial c_{i\mathrm{R}}} + \frac{\partial z_i(n)}{\partial s_{i\mathrm{I}}}\frac{\partial s_{i\mathrm{I}}}{\partial c_{i\mathrm{R}}}\right]\\ &= -2\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)[2(s_{i\mathrm{R}} - b_{i\mathrm{R}})x_\mathrm{R} + 2(s_{i\mathrm{I}} - b_{i\mathrm{I}})x_\mathrm{I}]/(a_i^* a_i)\end{aligned} \tag{19}$$

and

$$\begin{aligned}\frac{\partial E(n)}{\partial c_{i\mathrm{I}}} &= \frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\left[\frac{\partial z_i(n)}{\partial s_{i\mathrm{R}}}\frac{\partial s_{i\mathrm{R}}}{\partial c_{i\mathrm{I}}} + \frac{\partial z_i(n)}{\partial s_{i\mathrm{I}}}\frac{\partial s_{i\mathrm{I}}}{\partial c_{i\mathrm{I}}}\right]\\ &= -2\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)[2(s_{i\mathrm{R}} - b_{i\mathrm{R}})x_\mathrm{I} + 2(s_{i\mathrm{I}} - b_{i\mathrm{I}})(-x_\mathrm{R})]/(a_i^* a_i),\end{aligned} \tag{20}$$

where $\psi'(x) = d\psi(x)/dx$, thus we can obtain

$$\begin{aligned}c_i(n+1) &= c_i(n) - \frac{1}{4}\mu_2 \nabla_{c_i} E(n)\\ &= c_i(n) + \mu_2\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)(s_i(n) - b_i(n))^* x(n)/[a_i^*(n)a_i(n)].\end{aligned} \tag{21}$$

The partial derivative with respect to the translation parameters b_i is as follows:

$$\begin{aligned}\nabla_{b_i} E(n) &= \frac{\partial E(n)}{\partial b_{i\mathrm{R}}} + j\frac{\partial E(n)}{\partial b_{i\mathrm{I}}} = \frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\frac{\partial z_i}{\partial b_{i\mathrm{R}}} + j\frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\frac{\partial z_i}{\partial b_{i\mathrm{I}}}\\ &= 4\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_{i\mathrm{R}} - b_{i\mathrm{R}})}{a_i^* a_i} + 4j\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_{i\mathrm{I}} - b_{i\mathrm{I}})}{a_i^* a_i}\\ &= 4\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_i - b_i)}{a_i^* a_i}.\end{aligned} \tag{22}$$

Therefore

$$\begin{aligned}b_i(n+1) &= b_i(n) - \frac{1}{4}\mu_3 \nabla_{b_i} E(n)\\ &= b_i(n) - \mu_3\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)(s_i(n) - b_i(n))/[a_i^*(n)a_i(n)].\end{aligned} \tag{23}$$

And

$$\begin{aligned}\nabla_{a_i} E(n) &= \frac{\partial E(n)}{\partial a_{i\mathrm{R}}} + j\frac{\partial E(n)}{\partial a_{i\mathrm{I}}} = \frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\frac{\partial z_i}{\partial a_{i\mathrm{R}}} + j\frac{\partial E(n)}{\partial h_i(n)}\frac{\partial h_i(n)}{\partial z_i(n)}\frac{\partial z_i}{\partial a_{i\mathrm{I}}}\\ &= 4\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_i(n) - b_i(n))^*(s_i(n) - b_i(n))a_{i\mathrm{R}}}{(a_i^* a_i)^2}\\ &\quad + 4j\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_i(n) - b_i(n))^*(s_i(n) - b_i(n))a_{i\mathrm{I}}}{(a_i^* a_i)^2}\\ &= 4\,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\frac{(s_i(n) - b_i(n))^*(s_i(n) - b_i(n))}{(a_i^* a_i)^2}a_i.\end{aligned} \tag{24}$$

Hence

$$a_i(n+1) = a_i(n) - \frac{1}{4}\mu_4 \nabla_{a_i} E(n)$$
$$= a_i(n) - \mu_4 \,\mathrm{Re}[w_i(n)e(n)]\psi'(z_i)\,\frac{(s_i(n)-b_i(n))^*(s_i(n)-b_i(n))}{(a_i^*(n)a_i(n))^2}\,a_i(n). \tag{25}$$

By selecting the Harr wavelet as the neuron activation function, we can know from (8) and (9)

$$\psi(z_i) = (1-z_i)e^{-\frac{1}{2}z_i} \tag{26}$$

and

$$\psi'(z_i) = \left(\frac{1}{2}z_i - \frac{3}{2}\right)e^{-\frac{1}{2}z_i}. \tag{27}$$

Substituting Eq. (27) into Eqs. (21), (23), (25), we can obtain the parameters adjusting formula.

5. Application to the identification of a nonlinear channel model

The approximation capabilities of the CWN and the efficiency of the complex learning algorithms are illustrated using an example of modeling a complex-valued nonlinear communication channels. The scheme of this nonlinear channel is shown in Fig. 3. In current study, the 4-QAM signaling scheme is considered, i.e. the constellation of $x(t)$ is shown by

$$x(t) = x_{\mathrm{R}}(t) + jx_{\mathrm{I}}(t) = \begin{cases} x^{(1)} = 1 + j, \\ x^{(2)} = -1 + j, \\ x^{(3)} = 1 - j, \\ x^{(4)} = -1 - j. \end{cases} \tag{28}$$

The transmitted signals are first passed through a linear FIR filter with transfer function $H(z)$ defined by

$$H(z) = (1.0119 - j0.7589) + (-0.3796 + j0.5059)z^{-1}. \tag{29}$$

$H(z)$ has a zero $z_0 = 0.4801 - j0.1399$ in the Z-plane. Then signals are passed through a nonlinear element defined by

$$v(t) = \frac{2u(t)}{1+|u(t)|^2}\exp\left(j\frac{\pi}{3}\frac{|u(t)|^2}{1+|u(t)|^2}\right). \tag{30}$$

This static nonlinearity is used to represent the nonlinear high power amplifier in the transmitter, which is described in literature [12].

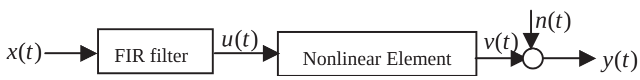

Fig. 3. A nonlinear channel model.

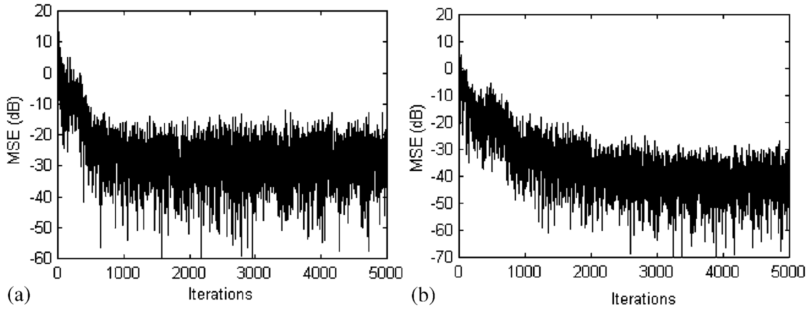

Fig. 4. MSE versus learning iterations, left for MLP and right for CWN.

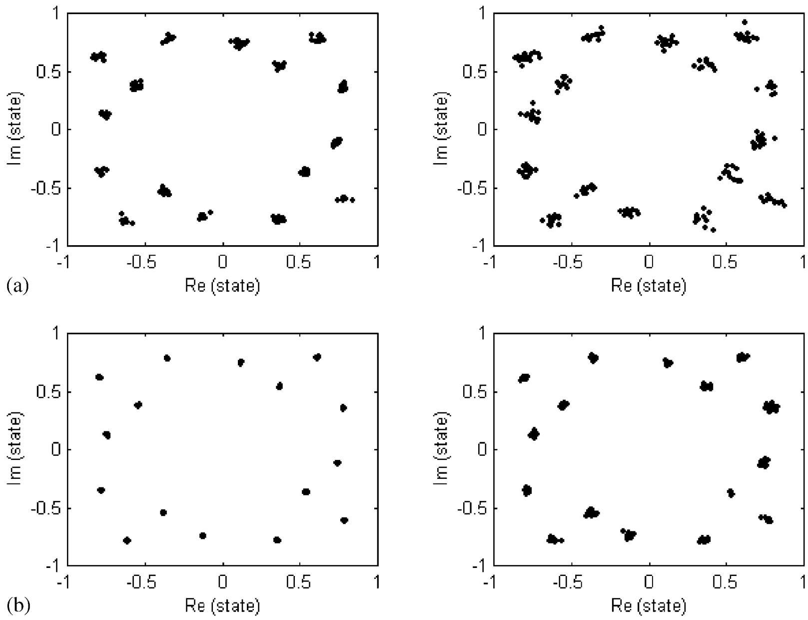

Fig. 5. State constellation of channel output for the identifiers, (left) no additive noise, (right) SNR = 25 dB. (a) MLP, (b) CWN.

For the purpose of comparison, the tests are between the following two complex-valued neural network identifiers.

(1) Complex-valued standard 3-layer multilayer perception (MLP) composed of 2 inputs, 10 hidden and 1 output neurons.
(2) CWN composed of 2 inputs, 10 hidden and 1 output neurons.

The networks input vector was defined by $\boldsymbol{x}(t) = [x(t), x(t-1)]^T$. The learning rates μ_i $(i = 1, 2, 3, 4)$ of CWN are firstly set as 0.1, and the learning rates is divided by 2 every 500 iterations. So were the learning rates of MLP. The learning curves with respect to the learning iterations are shown in Fig. 4. As we can see from this figure that the CWN has good convergence behaviors than that of the MLP.

After learning, a forward phase test is performed. The state constellation of channel output is depicted in Fig. 5, (a) for the MLP and (b) for the CWN (the test data consists 200 points). Two different cases are reported: case 1 (left) without additive noise and case 2 (right) with SNR = 25 dB. It is worth noting that the output constellation when using CWN is less distorted than in the MLP case.

6. Conclusions

In this paper, we propose a complex-valued wavelet network, whose inputs, outputs and weights are all complex-valued, and the nonlinear activation function remains real-valued. The backpropagation learning algorithm for training the complex-valued wavelet network was derived. The performance of the proposed CWN is illustrated with application to the identification of complex-valued nonlinear communication channel model. The simulation results demonstrated that the CWN has powerful approximation ability. In signal processing and communications where the inputs, outputs and transfer functions of a system are modeled in the complex domain, the proposed complex wavelet networks provide a useful tool for such cases.

Acknowledgments

This work was supported partially by the National Natural Science Foundations of China under Grant 60271019, and partially by the Youth Science and Technology Foundation of UESTC under Grant YF020207.

References

[1] N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, A comparison between real and complex-valued neural networks in communication applications, Proceedings of the International Conference on Neural Networks, Espoo, Finland, June, 1991.
[2] N. Benvenuto, F. Piazza, On the complex back propagation algorithm, IEEE Trans. Signal Process. 40 (1992) 967–969.
[3] S. Chen, S. Mclaughlim, B. Mulgrew, Complex-valued radial based function network, Part I: network architecture and learning algorithms, Signal Process. 35 (1994) 19–31.
[4] S. Chen, S. Mclaughlim, B. Mulgrew, Complex-valued radial based function network, Part II: application to digital communications channel equalization, Signal Process. 36 (1994) 175–188.
[5] Y. Fang, T.W.S. Chow, Orthogonal wavelet neural networks applying to identification of Wiener model, IEEE Trans. CAS-I 47 (2000) 591–593.
[6] N. Georgin, C. Koutsougeras, Complex domain backpropagation, IEEE Trans. CAS 39 (1992) 330–334.
[7] S. Haykin, Adaptive Filter Theory, 3rd Edition, Prentice–Hall, Englewood Cliffs, NJ, 1996.
[8] G. Kechriotis, E.S. Manolakos, Training fully recurrent neural networks with complex weights, IEEE Trans. CAS-II 41 (3) (1994) 235–238.
[9] H. Leung, S. Haykin, The complex back propagation algorithm, IEEE Trans. Signal Process. 39 (1991) 2101–2104.

[10] G.R. Little, S.C. Gustafson, R.A. Senn, Generalization of the back propagation neural network learning algorithms to permit complex weights, Appl. Opt. 29 (11) (1990) 1591–1592.
[11] D.E. Rumelhart, J.L. McClelland, Learning internal representation by error propagation, in: Parallel Distributed Processing, Vol. I, MIT Press, Cambridge, MA, 1986.
[12] A.A.M. Saleh, Frequency-independent and frequency-dependent nonlinear models of TWT amplifiers, IEEE Trans. Comm. 29 (1981) 1715–1720.
[13] H.H. Szu, B. Telfer, S. Kadambe, Neural network adaptive wavelets for signal representation and classification, Opt. Eng. 31 (1992) 1907–1916.
[14] A. Uncini, L. Vecci, P. Campolucci, F. Piazza, Complex-valued neural networks with adaptive spline activation function for digital radio links nonlinear equalization, IEEE Trans. Signal Process. 47 (1999) 505–514.
[15] Q. Zhang, A. Benveniste, Wavelet networks, IEEE Trans. Neural Networks 3 (1992) 889–898.
[16] T. Chen, H. Chen, Approximation of continuous functional by neural networks with application to dynamic system, IEEE Trans. Neural Networks 4 (1993) 910–918.