A Novel Modular Recurrent Wavelet Neural Network and Its Application to Nonlinear System Identification

Haiquan Zhao and Xiangping Zeng

Abstract To reduce the computational complexity and improve the performance of the recurrent wavelet neural network (RWNN), a novel modular recurrent neural network based on the pipelined architecture (PRWNN) with low computational complexity is presented in this paper. Its modified adaptive real-time recurrent learning (RTRL) algorithm is derived on the gradient descent approach. The PRWNN comprises a number of RWNN modules that are cascaded in a chained form and inherits the modular architectures of the pipelined recurrent neural network (PRNN) proposed by Haykin and Li. Since those modules of the PRWNN can be performed simultaneously in a pipelined parallelism fashion, it would result in a significant improvement in computational efficiency. And the performance of the PRWNN can be also further improved. Computer simulations have demonstrated that the PRWNN provides considerably better performance compared to the single RWNN model for nonlinear dynamic system identification.

Keywords Recurrent wavelet neural network · Pipelined recurrent neural network · Real-time recurrent learning · Nonlinear system identification

1 Introduction

Due to the nonlinear signal processing and learning capability by generating complex mapping between the input and the output space, artificial neural networks (ANNs) have become a powerful tool for nonlinear dynamic system identification [1]. Many research works using multilayer percetron (MLP) networks [1], radial basis function (RBF) networks [2], functional link ANNs (FLANNs) [3],

H. Zhao (✉) · X. Zeng
School of Electrical Engineering, Southwest Jiaotong University, 610031 Chengdu, China
e-mail: hqzhao@home.swjtu.edu.cn

F. Sun et al. (eds.), *Foundations and Applications of Intelligent Systems*,
Advances in Intelligent Systems and Computing 213,
DOI: 10.1007/978-3-642-37829-4_11,

and recurrent neural networks (RNNs) [4] have been reported in nonlinear dynamic system identification.

Recently, the wavelet neural network (WNN), combing the capability of ANN in learning from processes and the capability of wavelet decomposition, has received considerable interest. In [5], a WNN based on the wavelet transform theory was presented as an alternative to ANNs for approximating nonlinear functions. Research results have shown that a WNN can approximate any continuous function over a compact set and have high accuracy and fast learning ability. However, it has been proved that the NN with the recurrent architecture is superior to feedforward neural network (FNN) in identifying nonlinear dynamic system. As a recurrent network, the recurrent wavelet neural network (RWNN) [6–8], combining the properties of attractor dynamics of the RNN and good convergence performance of the WNN, can cope with time-varying input or output through its own natural temporal operation because a mother wavelet layer composed of internal feedback neurons to capture the dynamic response of a system. To further improve the performance of the RWNN, the self-recurrent wavelet neural network (SRWNN) [9] and recurrent fuzzy wavelet neural network (RFWNN) [10, 11] have been presented to deal with the problems of nonlinear dynamic system identification. Although the RWNN shows promising results, it still suffers from the heavy computational loads as the RNN.

In 1995, to reduce the computational complexity of the RNN, a computationally efficient modular nonlinear adaptive filter-based pipelined recurrent neural network (PRNN) was proposed by Haykin and Li [12]. The design of the pipelined architecture follows the important engineering principle of divide and conquer and the biological principle of NN modules. Its significant merit is relatively low computational complexity [12–20]. As a result, inspired by the pipelined architecture of the PRNN, a novel modular RWNN based on the pipelined architecture is proposed to reduce the computational complexity and improve the performance of the RWNN for nonlinear dynamic system identification in this paper.

2 A Modular RWNN Based on the Pipelined Architecture

To overcome the computational complexity problem of the RWNN, keeping the views of the pipelined architecture, a novel modular RWNN based on the pipelined architecture (PRWNN) is presented. The PRWNN, inheriting the modular architectures of the PRNN proposed by Haykin and Li, comprises a number of RWNN modules that are cascaded in a chained form. Each module is implemented by a small-scale RWNN with internal dynamics. Since those modules of the PRWNN can be performed simultaneously in a pipelined parallelism fashion, it would result in a significant improvement in computational efficiency. In addition, the nesting module of the pipelined architecture can help to circumvent the problem of vanishing gradient of the RWNN, and the performance of the PRWNN can be further improved.

Figure 1 describes the structure of the PRWNN, which is composed of M identical modules, and each module is designed as a decision feedback RNN with q neurons and has $q - 1$ neuron output decision feedback to its input, and the remaining neuron output (the first neuron output decision) is applied directly to the next module. In the case of the PRWNN, module M is a fully connected RNN, and a one-unit delayed signal of the M module's output is assumed to be decision feedback to the input. Information flow into and out of the modules proceeds in a synchronized fashion. Therefore, all the modules have exactly the same number of external inputs and internal decision feedback signals.

Figure 2 shows the detailed structure of module i with q neurons and p external inputs. Note that for module M, its module output decision acts as an external feedback signal to itself. In addition, all the modules of PRWNN operate similarly in that they all have exactly the same number of external inputs and feedback signals, which are properly timed. Moreover, all the modules are designed to have exactly the same $(p + q + 1)$-by-q synaptic weight matrix $W(n)$, q-by-1 weight vector $W^o(n)$ and the parameters of the wavelet function. An element $w_{k,l}(n)$ of this matrix represents the weight of the connection to the kth neuron from the lth input node. Moreover, the weight matrix W may be written as

$$W(n) = \left[w_1(n), \ldots, w_k(n), \ldots w_q(n)\right]. \tag{1}$$

where $w_k(n)$ is a $(p + q + 1)$-by-1 vector defined by

$$w_k(n) = \left[w_{1,k}(n), w_{2,k}(n), \ldots, w_{p+q+1,k}(n)\right]^T. \tag{2}$$

And the superscript T denotes transposition.

At the nth time, for the ith module, the external input signal is described by the p-by-1 vector

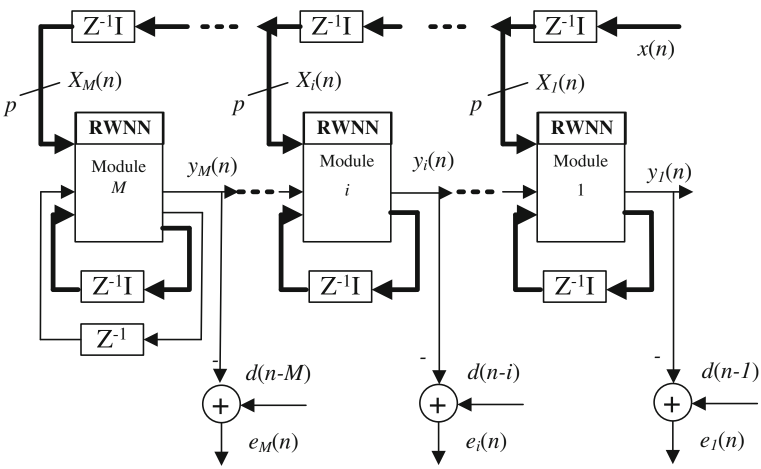


Fig. 1 A modular RWNN based on the pipelined architecture with M modules

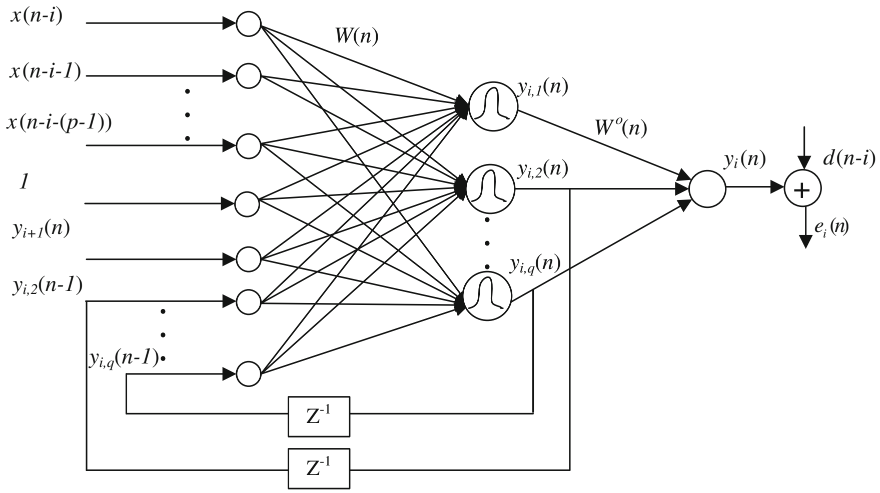


Fig. 2 Detailed architecture of module i of the PRWNN

$$X_i(n) = [x(n-i),\, x(n-(i+1)),\, \ldots,\, x(n-(i+p-1))]^T. \tag{3}$$

and is delayer by $Z^{-i}I$ at the input of the module i, where Z^{-i} denotes the delay operator i time units, and I is the $(p \times p)$-dimensional identity matrix, and p is the nonlinear adaptive equalizer order. The other input vector applied to module i is the q-by-1 decision feedback vector

$$r_i(n) = \left[y_{i+1,1}(n),\, \hat{r}_i(n)\right]^T, \quad i = 1, 2, \ldots, (q-1). \tag{4}$$

where $y_{i+1,1}(n)$ is the first neuron's output in the adjacent module $i+1$, vector $\hat{r}_i$ is the one-step delayed output feedback signals that originate from module i itself and is defined by

$$\hat{r}_i(n) = \left[y_{i,2}(n-1), \ldots, y_{i,q}(n-1)\right]^T. \tag{5}$$

The last module of the PRWNN, namely module M, operates as a standard fully connected RWNN. The vector r_M consists of the one-step delayed output decision signals in module M that are fed back to itself and as shown by

$$\begin{aligned} r_M(n) &= \left[y_{M,1}(n-1),\, \hat{r}_M(n)\right]^T \\ &= \left[y_{M,1}(n-1),\, y_{M,2}(n-1), \ldots, y_{M,q}(n-1)\right]^T. \end{aligned} \tag{6}$$

To accommodate a bias for each neuron, besides the $p+q$ inputs, the fixed input +1 is included. Based on the above discussion, an input vector $V_i(n)$ consisting of total $(p+q+1)$ input signals applied to module i is represented

$$V_i(n) = \left[X_i^T(n),\, 1,\, r_i^T(n)\right]^T, \quad i = 1, 2, \ldots M. \tag{7}$$

For the module i, the output $y_{i,l}(n)$ of neuron l at the nth time point is computed by passing $u_{i,l}(n)$ through a wavelet function $\varphi(\bullet)$, obtaining

$$y_{i,l}(n) = \varphi\left(\frac{u_{i,l}(n) - b_l(n)}{a_l(n)}\right). \tag{8}$$

With loss of generality, the "Gaussian-derivative" wavelet function given in [6] is used by

$$\varphi(x) = \frac{1}{\sqrt{|a_l|}}(-x)\exp\left(-\frac{x^2}{2}\right). \tag{9}$$

And the net internal activity $u_{i,l}(n)$ is given by

$$\begin{aligned} u_{i,l}(n) &= V_i^T(n)w_l(n) \\ &= \sum_{k=1}^{p+M+1} w_{k,l}(n)v_{i,k}(n) \\ &= \sum_{k=1}^{p} w_{k,l}(n)x(n-(i+k-1)) \\ &\quad + w_{p+1,l}(n) + \sum_{k=p+2}^{p+q+1} w_{k,l}(n)r_{i,k-(p+1)}(n). \end{aligned} \tag{10}$$

where

$$v_{i,k}(n) = \begin{cases} x(n-(i+k-1)), & 1 \le k \le p,\ 1 \le i \le M \\ 1, & k = p+1,\ 1 \le i \le M \\ y_{i+1,1}(n), & k = p+2,\ 1 \le i \le M-1 \\ y_{M,1}(n-1), & k = p+2,\ i = M \\ y_{i,k-(p+1)}(n-1), & p+3 \le k \le p+1+q,\ 1 \le i \le M \end{cases} \tag{11}$$

and

$$r_{i,k-(p+1)}(n) = \begin{cases} y_{i+1,1}(n), & k = p+2,\ 1 \le i \le M-1 \\ y_{M,1}(n-1), & k = p+2,\ i = M \\ y_{i,k-(p+1)}(n-1), & p+3 \le k \le p+1+q,\ 1 \le i \le M \end{cases} \tag{12}$$

Then, the output of the ith module is given as

$$y_i(n) = \sum_{j=1}^{q} w_j^o(n)y_{i,j}(n) = H^T(n)W^o(n). \tag{13}$$

Finally, the output signal computed by the PRWNN at time instant n is defined by

$$y(n) = y_1(n). \tag{14}$$

Certainly, $y_i(n)$ is interpreted as the estimate of desired signal $d(n-i)$ computed by the ith module.

3 Training Algorithm for the PRWNN

According to the learning algorithms of the PRNN, adaptive learning algorithm of the PRWNN is derived by the real-time recurrent learning (RTRL) rule in the following subsection.

The overall cost function for the PRWNN is defined by

$$E(n) = \sum_{i=1}^{M} \varepsilon^{i-1} e_i^2(n). \tag{15}$$

where ε is an exponential forgetting factor that lies in the range of $0 < \varepsilon \leq 1$, the inverse of ε^{i-1} is a measure of the memory of the PRWNN. And the corresponding error $e_i(n)$ of the ith module is given by

$$e_i(n) = d(n-i) - y_i(n). \tag{16}$$

After every module of the PRWNN finishes its calculations, $e_1(n)$, $e_2(n)$, ... and $e_M(n)$ error signals are obtained. Thus, adjustments to the synaptic weight matrix $W(n)$ and $W^o(n)$ of each module are made to minimize $E(n)$ in accordance with the RTRL algorithm.

According to the approach in [12], the change to klth element of the weight matrix $W(n)$ is

$$\Delta w_{k,l}(n) = \frac{\eta_1}{2} \frac{\partial E(n)}{\partial w_{k,l}(n)}, \quad 1 \leq l \leq q,\ 1 \leq k \leq p+2+q. \tag{17}$$

Then, the element of weight matrix $W(n)$ is updated as

$$w_{k,l}(n+1) = w_{k,l}(n) + \Delta w_{k,l}(n) = w_{k,l}(n) - \frac{\eta_2}{2} \frac{\partial E(n)}{\partial w_{k,l}(n)}. \tag{18}$$

Similarly, the parameters $a_l(n)$ and $b_l(n+1)$ are updated by, respectively

$$a_l(n+1) = a_l(n) - \frac{\eta_3}{2} \frac{\partial E(n)}{\partial a_l(n)}. \tag{19}$$

$$b_l(n+1) = b_l(n) - \frac{\eta_4}{2} \frac{\partial E(n)}{\partial b_l(n)}. \tag{20}$$

Furthermore, according to the RTRL rule, the recursive equations of (18, 19) and (20) can be obtained by

$$\begin{aligned} w_{k,l}(n+1) =& w_{k,l}(n) + \eta_2 \sum_{i=1}^{M} \varepsilon^{i-1} e_i(n) \left\{ \sum_{j=1}^{q} w_j^o(n) \frac{\varphi'(\mathrm{net}_{i,j}(n))}{a_j(n)} \left[\frac{\partial y_{i+1,1}(n)}{\partial w_{k,l}(n)} w_{p+2,l}(n) \right. \right. \\ & \left. \left. + \sum_{m=2}^{q} w_{m+p+1,j}(n) \frac{\partial y_{i,m}(n-1)}{\partial w_{k,l}(n)} + \delta_{k,j} v_{i,l}(n) \right] \right\}. \end{aligned} \tag{21}$$

$$\begin{aligned} b_l(n+1) =& b_l(n) - \eta_3 \sum_{i=1}^{M} \varepsilon^{i-1} e_i(n) \left\{ \sum_{j=1}^{q} w_j^o(n) \frac{\varphi'(\mathrm{net}_{i,j}(n))}{a_j(n)} \left[w_{p+2,l}(n) \pi\_w_{k,l}^{i+1,1}(n) \right. \right. \\ & \left. \left. + \sum_{m=2}^{q} w_{m+p+1,j}(n) \frac{\partial y_{i,m}(n-1)}{\partial w_{k,l}(n)} + \delta_{k,j} v_{i,l}(n) \right\} \right]. \end{aligned} \tag{22}$$

$$a_l(n+1) = a_l(n) + \eta_4 \sum_{i=1}^{M} \varepsilon^{i-1} e_i(n) \left\{ \sum_{j=1}^{q} w_j^o(n) \varphi'(\mathrm{net}_{i,j}(n)) \frac{\partial \left(\frac{u_{i,j}(n) - b_j(n)}{a_j(n)} \right)}{\partial a_l(n)} \right\}. \tag{23}$$

In addition, by using the gradient rule, the weight $W^o(n)$ of the PRWNN is updated as

$$W^o(n+1) = W^o(n) + \eta_1 \sum_{i=1}^{M} \varepsilon^{i-1} e_i(n) H_i(n). \tag{24}$$

where $\eta_i (i = 1, 2, 3, 4)$ is learning rate and controls the convergence performance of the PRWNN.

4 Simulations

To evaluate the performance of the PRWNN, nonlinear dynamic system identification application is carried out in this subsection.

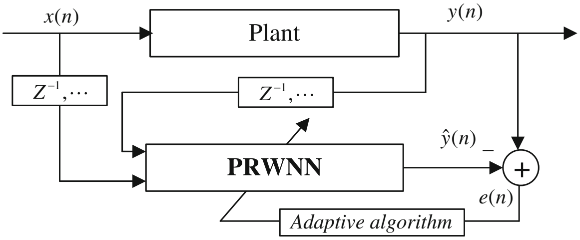
Figure 3 depicted the identification scheme of a nonlinear dynamic system based on the PRWNN filter. The plant is described by the following difference equation [6]

$$\hat{y}(n+1) = f[\hat{y}(n), \hat{y}(n-1), \hat{y}(n-2), x(n), x(n-1)]. \tag{25}$$

where the function $f(\cdot)$ is defined by

$$f[x_1, x_2, x_3, x_4, x_5] = \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}. \tag{26}$$

Fig. 3 Identification scheme of a nonlinear dynamic system



During the test phase, the following test signal is used to test the performance of the PRWNN models:

$$x(n) = \begin{cases} \sin(2\pi n/250) & 1 \le n \le 250 \\ 0.8\sin(2\pi n/250) + 0.2\sin(2\pi n/25) & < 250 n \le 600 \end{cases} \quad (27)$$

Fig. 4 Idenfication of the nonlinear dynamic plant with the test signal. **a** PRWNN. **b** RWNN

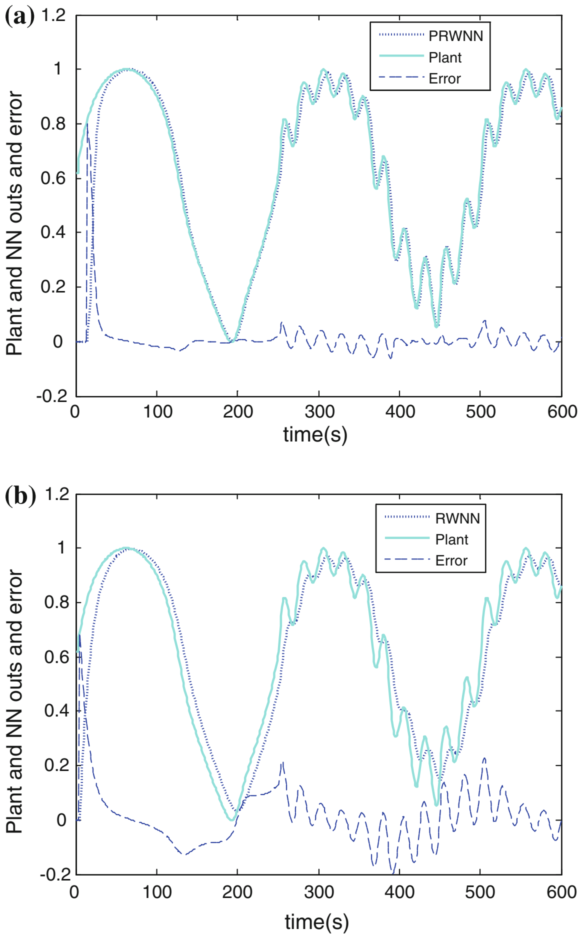


Figure 4 shows the actual neural network's output and error of the nonlinear dynamic plant for the test signal with the RWNN and PRWNN model. It is obviously observed that the proposed PRWNN shows much better performance than the conventional RWNN in this nonlinear dynamic system identification problem. This result is reasonable due to the fact that the pipelined architecture of the PRWNN helps to enhance nonlinear processing capability and improve the performance. Moreover, the computational complexity of the PRWNN is much lower than that of the RWNN.

5 Conclusion

In this paper, we proposed a nonlinear adaptive filter with a pipelined RWNN to reduce the computational burden of the RWNN. The network model consists of a number of modules-based RWNN that are interconnected in a chained form and inherits the major characteristics (low computational complexity) of the pipelined architecture. The parameter update rules of the PRWNN are derived according to the modified RTRL algorithm. The performance of the proposed PRWNN has been assessed for nonlinear dynamic system identification and compared with that of the RWNN model. Simulation results show that the proposed PRWNN with lower computational complexity can outperform the single RWNN model.

Acknowledgments This work was partially supported by the National Science Foundation of People's Republic of China (grant no: 61271340 and 61071183), Sichuan Provincial Youth Science and Technology Fund (grant no. 2012JQ0046), and Fundamental Research Funds for the Central Universities under grant SWJTU12CX026.

References

1. Narendra KS, Parthasarathy K (1990) Identification and control of dynamical systems using neural networks. IEEE Trans Neural Networks 1(1):4–27
2. Park J, Sandberg IW (1991) Universal approximation using radial basis function networks. Neural Comput 3:246–257
3. Patra JC, Pal RN, Chatterji BN, Panda G (1999) Identification of nonlinear dynamic systems using functional link artificial neural network. IEEE Trans Syst Man Cybern B 29(2):254–262
4. Wu YL, Song Q, Liu S (2008) A normalized adaptive training of recurrent neural networks with augmented error gradient. IEEE Trans Neural Networks 19(2):351–356
5. Zhang Q, Benviste A (1992) Wavelet networks. IEEE Trans Neural Networks 3(6):889–898
6. Rao SS, Kumthekar B (1993) Recurrent wavelet networks. In: Proceedings of the 1993 IEEE-SP Workshop IEEE, vol III, pp 3143–3147
7. Zhao F, Hu L, Li Z (2009) Nonlinear system identification based on recurrent wavelet neural network. In: ISNN 2009, vol 56, pp 517–525
8. Lu CH (2009) Design and application of stable predictive controller using recurrent wavelet neural networks. IEEE Trans Ind Electron 56(9):3733–3742

9. Yoo SJ, Choi YH, Park JB (2006) Generalized predictive control based on self-recurrent wavelet neural network for stable path tracking of mobile robots: adaptive learning rates approach. IEEE Trans Circuits Syst I Regul Pap 53(6):1381–1394
10. Abiyev RH, Kaynak O (2008) Fuzzy wavelet neural networks for identification and control of dynamic plants-a novel structure and a comparative study. IEEE Trans Ind Electron 55(8):3133–3140
11. Lin C, Chin C (2004) Prediction and identification using wavelet-based recurrent fuzzy neural networks. IEEE Trans Syst Man Cybern B Cybern 34(5):2144–2154
12. Haykin S, Li L (1995) Nonlinear adaptive prediction of nonstationary signals. IEEE Trans Signal Process 43(2):526–535
13. Mandic DP, Chambers JA (2000) On the choice of parameters of the cost function in nested modular RNNs. IEEE Trans Neural Networks 11(2):315–322
14. Zhao HQ, Zhang JS (2009) Nonlinear dynamic system identification using pipelined functional link artificial recurrent neural network. Neurcomputing 72:3046–3054
15. Stavrakoudis DG, Theocharis JB (2007) Pipelined recurrent fuzzy neural networks for nonlinear adaptive speech prediction. IEEE Trans Syst Man Cybern B Cybern 37(5):1305–1320
16. Zhao HQ, Zeng XQ, He ZY, Jin WD, Li TR (2012) Adaptive extended pipelined second-order Volterra filter for nonlinear active noise controller. IEEE Trans Audio Speech Lang Process 20(4):1394–1399
17. Zhao HQ, Zeng XQ, He ZY (2011) Low-complexity nonlinear adaptive filter based on a pipelined bilinear recurrent neural network. IEEE Trans Neural Networks 22(9):1494–1507
18. Zhao HQ, Zhang JS (2010) Pipelined Chebyshev functional link artificial recurrent neural network for nonlinear adaptive filter. IEEE Trans Syst Man Cybern B Cybern 40(1):162–172
19. Zhao HQ, Zeng XQ, Zhang JS, Li TR (2010) Nonlinear adaptive equalizer using a pipelined decision feedback recurrent neural network in communication systems. IEEE Trans Commun 58(8):2193–2198
20. Zhao HQ, Zhang JS (2009) A novel adaptive nonlinear filter-based pipelined feedforward second-order Volterra architecture. IEEE Trans Signal Process 57(1):237–246