

Rajoo Pandey

Feedforward neural network for blind equalization with PSK signals

Received: 21 January 2004 / Accepted: 6 December 2004 / Published online: 30 June 2005
© Springer-Verlag London Limited 2005

Abstract Most of the cost functions used for blind equalization are nonconvex and nonlinear functions of tap weights, when implemented using linear transversal filter structures. Therefore, a blind equalization scheme with a nonlinear structure that can form nonconvex decision regions is desirable. The efficacy of complex-valued feedforward neural networks for blind equalization of linear and nonlinear communication channels has been confirmed by many studies. In this paper we present a complex valued neural network for blind equalization with M-ary phase shift keying (PSK) signals. The complex nonlinear activation functions used in the neural network are especially defined for handling the M-ary PSK signals. The training algorithm based on constant modulus algorithm (CMA) cost function is derived. The improved performance of the proposed neural network in both, stationary and nonstationary environments, is confirmed through computer simulations.

Keywords Blind equalization · Neural networks · Complex-valued nonlinear activation functions · M-ary PSK signals · CMA cost function

1 Introduction

The adaptive channel equalization is an important task in practical implementation of efficient digital communication. The past few years have witnessed an increased interest in problems and techniques related to blind signal processing, especially blind equalization [1–10]. The classical methods of channel equalization rely on transmitting the training signal, known in advance by

the receiver. The receiver adapts the equalizer so that its output closely matches the known reference (training) signal. For time-varying situations, the training signals have to be transmitted repeatedly. Inclusion of such signals sacrifices valuable channel capacity. Therefore, to reduce the overhead of transmission of training signals, the equalization without using the training signal, i.e., blind equalization is required.

Blind equalization techniques are either based on second-order statistics (SOS), or on higher order statistics (HOS). Bussgang blind equalization techniques [11] use higher order statistics in an implicit manner, as these methods rely on optimization of some cost function. The cost functions used in blind equalization are nonconvex and nonlinear functions of tap weights, when implemented using linear FIR filter structures. A linear, finite duration impulse response (FIR) filter structure, however, has a convex decision region [12], and hence, is not adequate to optimize such cost function. Therefore, a blind equalization scheme with a nonlinear structure that can form nonconvex decision region is desirable [13].

Neural Networks, often referred to as an emerging technology, have been used in many signal processing applications, for example, filtering, parameter estimation, signal detection, system identification, signal reconstruction, signal compression, time series estimation [14–17]. Neural networks have also been applied for blind equalization, and better results, as compared to linear filtering, have been reported [1–3, 6–10, 13]. However, most of these studies are limited to real valued signals and channel models. Therefore, the development of neural network-based equalization schemes is desirable for complex-valued channel models with high level signal constellations such as M-ary phase shift keying (PSK) and quadrature amplitude modulation (QAM). One such study of blind equalization schemes is available in [13], but is limited to M-ary QAM signal only, under stationary environment.

In general, complex data can be handled in two different ways. One way is to treat the real and imaginary

R. Pandey
Department of Electronics and Communication Engineering,
National Institute of Technology, Kurukshetra, 136119, India
E-mail: rajoo_pandey@rediffmail.com
Tel.: +91-1744-238470
Fax: +91-1744-238050

parts of each complex data as two separate entities. In this case, the weights of two real-valued neural networks are updated independently. The other way is to assign complex values to the weights of neural network and update using a complex learning algorithm such as complex backpropagation algorithm (CBP). Many studies [13, 18] have shown that a complex-valued MLP yields more efficient structure than two real-valued MLPs.

The neural networks can be used to optimize any of the cost functions used for blind equalization. However, the Godard algorithm (also CMA) [19, 20] is considered to be the most successful among the HOS-based blind equalization algorithms. The Godard algorithm has many advantages when compared with other HOS-based Bussgang algorithms [12, 21]. Thus, in this paper, the complex-valued multiplayer feedforward neural networks for M-ary PSK signals are presented. The learning algorithms are based on the Godard or CMA cost functions. These blind equalization schemes yield lower mean-squared error and symbol error rate in comparison to linear FIR structures-based equalizers due to decorrelation performed by the nonlinearities of the activation functions.

The paper is organized as follows. In Sect. 2, the neural network model for M-ary PSK signals is described. The learning algorithm is presented in Sect. 3. The performance of neural network-based equalizer is described through simulation in stationary as well as in nonstationary environment, in Sect. 4. Finally, the conclusions are given in Sect. 5.

2 Neural network model

The blind equalization structure is described in Fig. 1. A signal sequence of independent and identically distributed (iid) data is transmitted through a linear channel with an impulse response $h(t)$. The output of the channel is represented, as in [12], by

$$x(t) = \sum_{k=-\infty}^{\infty} s_k h(t - kT) + v(t), \quad (1)$$

where $\{s_k\}$ represents the data sequence which is sent over the channel with symbols spaced time T apart and $v(t)$ is additive white noise.

The received signal is sampled by substituting $t = NT$ in (1)

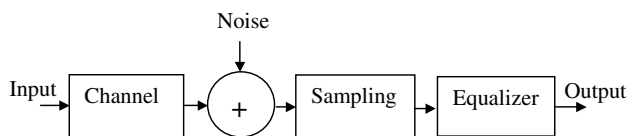


Fig. 1 Blind equalization structure in digital communication

$$x(nT) = \sum_{k=-\infty}^{\infty} s_k h[(n - k)T] + v(nT). \quad (2)$$

In simplified form, sampled signal of (2) is described as

$$x(n) = \sum_{k=0}^L s_k h_{n-k} + v(n), \quad (3)$$

where the channel is modeled as an FIR filter of length L . $x(n)$ and $v(n)$ represent the sampled channel output and the sampled noise, respectively.

The input to the equalizer is formed by N samples of channel output as

$$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-N+1)]^T. \quad (4)$$

The output of a linear FIR equalizer is expressed as

$$y(n) = \mathbf{w}^H \mathbf{x}(n), \quad (5)$$

where \mathbf{w} is an $N \times 1$ vector representing the weights of the equalizer and $y(n)$ is the output, which is obtained as a rescaled and phase-shifted version of the transmitted signal.

2.1 Structure

A three-layer complex-valued feedforward network for blind equalization is shown in Fig. 2. The network has N input nodes, H hidden layer nodes and one output node. The complex-valued weight $w^{(1)}_{kl}$ denotes the synaptic weight, connecting the output of node l of input layer to the input of neuron k in the hidden layer. $w^{(2)}_k$ refers to the synaptic weight connected between neuron k of hidden layer and the output neuron.

The input of the equalizer is formed by N samples of the received signal as given by (4) and represented for convenience as

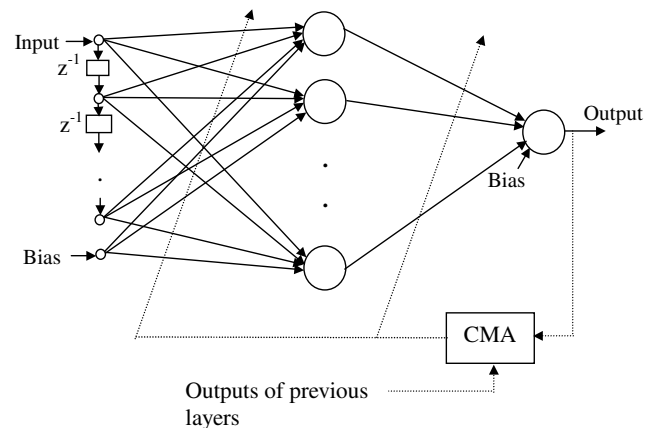


Fig. 2 Complex-valued multilayer feedforward neural network equalizer

$$\mathbf{x}(n) = [x_1(n), x_2(n), \dots, x_N(n)]^T. \quad (6)$$

The activation sum $\text{net}_k^{(1)}(n)$ and the output $u_k(n)$ of neuron k in the hidden layer are given as

$$\begin{aligned} \text{net}_k^{(1)}(n) &= \text{net}_{k,R}^{(1)}(n) + j\text{net}_{k,I}^{(1)}(n) \\ &= \sum_{l=1}^N w_{kl}^{(1)}(n)x_l(n) + \theta_k^{(1)}(n) \end{aligned} \quad (7)$$

and

$$u_k(n) = \phi^{(1)}(\text{net}_k^{(1)}(n)) \quad ; k = 1, 2, \dots, H, \quad (8)$$

where $\text{net}_{k,R}^{(1)}(n)$ and $\text{net}_{k,I}^{(1)}(n)$ are, respectively, the real and imaginary parts of the activation sum $\text{net}_k^{(1)}(n)$, at time n , and $\phi^{(1)}(\cdot)$ represents the nonlinear activation function of neurons in hidden layer and $\theta_k^{(1)}(n)$ denotes the threshold of neuron k of the hidden layer.

For the neuron of the output layer, the activation sum and the output are expressed as

$$\begin{aligned} \text{net}^{(2)}(n) &= \text{net}_R^{(2)}(n) + j\text{net}_I^{(2)}(n) \\ &= \sum_{k=1}^H w_k^{(2)}(n)u_k(n) + \theta^{(2)}(n) \end{aligned} \quad (9)$$

and

$$y(n) = \phi^{(2)}(\text{net}^{(2)}(n)), \quad (10)$$

where $y(n)$ denotes the output of the equalizer, $\text{net}_R^{(2)}(n)$ and $\text{net}_I^{(2)}(n)$ are, respectively, the real and imaginary parts of the activation sum $\text{net}^{(2)}(n)$, at time n , and $\phi^{(2)}(\cdot)$ is the activation function of the neuron in the output layer.

2.2 Activation functions for M-ary PSK signals

In the present model of complex-valued neural blind equalizer, the activation functions are defined according to the M-ary signal constellation. The choice of activation function plays an important role in the performance of the blind equalizers. For QAM signal, complex-valued activation functions are studied in [13]. However, it has been found that the choice of different activation functions for the hidden and output layers can further improve the performance of the blind equalizers [22]. Here, also for PSK signals we consider different activation functions for the nodes of hidden and output layer.

1. For the neurons of hidden layer, the activation function $\phi^{(1)}$ is described as

$$\phi^{(1)}(z) = \phi^{(1)}(z_R) + j\phi^{(1)}(z_I), \quad (11)$$

where z_R and z_I are the real and imaginary parts of the complex quantity z , and $\phi^{(1)}(\cdot)$ is a function defined by

$$\phi^{(1)}(x) = \alpha \tanh(\beta x), \quad (12)$$

while α and β are two real constants.

2. For the output layer node, the activation function is given by

$$\begin{aligned} \phi^{(2)}(z) &= f_1(|z|) \exp(jf_2(\angle z)) \\ &= f_1(|z|) \cos(f_2(\angle z)) + jf_1(|z|) \sin(f_2(\angle z)), \end{aligned} \quad (13)$$

where $|z|$ and $\angle z$ denote the modulus and the angle of a complex quantity z . The functions $f_1(\cdot)$ and $f_2(\cdot)$ are defined as

$$f_1(|z|) = a \tanh(b|z|) \quad (14)$$

and

$$f_2(\angle z) = \angle z - b \sin(m\angle z), \quad (15)$$

where b is a constant and m is the order of PSK signals. Figure 3a, b shows the plots of nonlinear activation functions defined in (12), (14) and (15).

From this figure, it can be seen that the activation functions have saturation regions around the symbol values of the PSK signal constellation shown in Fig. 6a. This multisaturation characteristic makes the network robust to noise. The complex-valued processing elements of output layers of the equalizers, defined by (14), (15), are illustrated in Fig. 4.

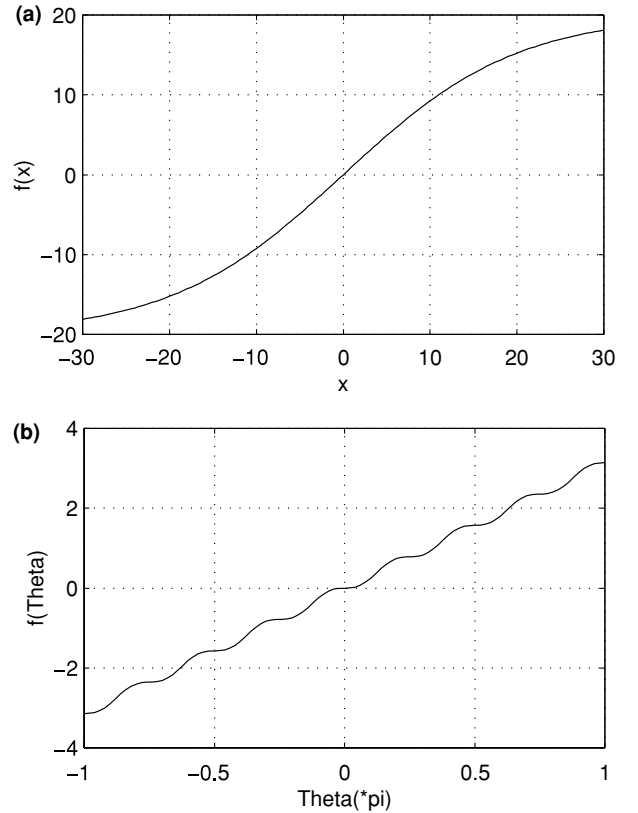


Fig. 3 a General nature of nonlinear functions ($\phi^{(1)}$ and f_1) b Plot of nonlinear function $f_2(\theta)$ for 8-PSK signal

Table 1 Multipath channel

Path gain	$3+j2$	$2-j1$	$1+j1$	$1-j0.4$	$-0.8-j0.6$
Path delay	0	$0.3T$	$0.5T$	$0.8T$	$1.2T$

The properties of a suitable complex activation function are given in [13]. However, it can be noted that it is sufficient to optimize the filter design if the gradient of the cost function exists. The gradient is defined as

$$\nabla_k J = \frac{\partial J}{\partial w_{kR}} + j \frac{\partial J}{\partial w_{kI}}; \quad k = 0, 1, 2, \dots, \quad (16)$$

where $w_{k,R}$ and $w_{k,I}$ denote the real and imaginary parts of k 'th element w_k of the vector \mathbf{w} . This gradient will exist if the activation functions of both hidden and output layers have the following first-order derivatives

$$\frac{\partial \varphi_R(z_R)}{\partial z_R}, \frac{\partial \varphi_R(z_R)}{\partial z_I}, \frac{\partial \varphi_I(z_I)}{\partial z_R} \text{ and } \frac{\partial \varphi_I(z_I)}{\partial z_I},$$

for $\varphi = \varphi^{(1)}$ and $\varphi^{(2)}$.

The activation functions defined by (12) and (13) have the following useful properties:

1. The functions are nonlinear in both z_R and z_I
2. The first-order partial derivatives mentioned above are continuous and bounded.
3. Real and imaginary parts of the complex activation functions have same dynamic range.
4. Real and imaginary parts of the complex activation functions of the output layer are saturated according to the signal constellation.

With these properties, the gradient of the CMA cost function is obtainable for M-ary PSK signal, as the required partial derivative can be easily computed w.r.t. $|z|$

3 Learning algorithm

In the task of blind equalization, the desired outputs are not available for training of the neural network. Therefore, the learning is unsupervised and is based on the minimization of a cost function. We obtain the update rules for the weights of neural networks by applying the gradient descent approach to minimize the CMA cost function. The updating rules are described as follows.

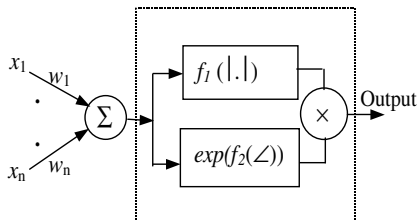


Fig. 4 The complex-valued processing element for M-ary PSK signals

- (1) For the weights connected between hidden layer and output layer:

$$w_k^{(2)}(n+1) = w_k^{(2)}(n) + \eta \delta^{(2)}(n) u_k^*(n), \quad (17)$$

where $\delta^{(2)}(n)$ is given as

$$\delta^{(2)}(n) = (|y(n)|^2 - R_2) |y(n)| \left(ab - \frac{b}{a} |y(n)|^2 \right) \frac{\text{net}^{(2)}(n)}{|\text{net}^{(2)}(n)|}. \quad (18)$$

In (18), the parameter R_2 depends on the statistical characteristics of the signal sequence, as defined in the Appendix, whereas constants a and b are chosen according to the channel outputs.

- (2) For the weights connected between input and hidden layer:

$$w_{kl}^{(1)}(n+1) = w_{kl}^{(1)}(n) + \eta \delta_k^{(1)}(n) x_l^*(n), \quad (19)$$

where $\delta_k^{(1)}(n)$ is given by

$$\delta_k^{(1)}(n) = \frac{\delta^{(2)}(n)}{\text{net}^{(2)}(n)} \left\{ \varphi^{(1)'}(\text{net}_{k,R}^{(1)}(n)) \text{Re}(w_k^{(2)}(n) \text{net}^{(2)*}(n)) - \varphi^{(1)'}(\text{net}_{k,I}^{(1)}(n)) \text{Im}(w_k^{(2)}(n) \text{net}^{(2)*}(n)) \right\}. \quad (20)$$

Here $u_k^*(n)$ and $x_l^*(n)$ denote the complex conjugate of k th and l th elements of $\mathbf{u}(n)$ and $\mathbf{x}(n)$, respectively. η is the learning rate parameter while $\phi^{(1)'}$ (.) and $\phi^{(2)'}$ (.) represent the derivatives of $\phi^{(1)}$ (.) and $\phi^{(2)}$ (.). The derivations of the update rules of (17), (18), (19), (20) are given in the Appendix.

4 Simulation

To observe the performance of complex-valued multi-layer feedforward blind equalizer for M-ary PSK signals, three different complex channels are used. The first channel (CH-1) is the one used in [13], and its z transform is

$$\begin{aligned} H(z) = & (0.0410 + j0.0109) + (0.0495 + j0.0123)z^{-1} \\ & + (0.0672 + j0.0170)z^{-2} + (0.0919 + 0.0235)z^{-3} \\ & + (0.7920 + j0.1281)z^{-4} + (0.3960 + j0.0871)z^{-5} \\ & + (0.2715 + j0.0498)z^{-6} + (0.2291 + j0.0414)z^{-7} \\ & + (0.1287 + j0.0154)z^{-8} + (0.1032 + j0.0119)z^{-9} \end{aligned} \quad (21)$$

The second channel (CH-2) is a multipath channel whose relative values of complex path gains and path delays are given in Table 1.

The continuous time multipath channel is described as

$$c(t) = \sum_i g_i \delta(t - \tau_i), \quad (22)$$

where g_i and τ_i are the path gain and path delay of i th path, respectively. For pulse shaping, a raised cosine pulse limited to a time duration $3T$, where T is the sample period, is used with 10% roll off factor. The expression for combined channel is

$$h(t) = c(t) \oplus p(t) = \sum_i g_i p(t - \tau_i), \quad (23)$$

where $p(t)$ is the raised cosine pulse and \oplus denotes the convolution.

The discrete time channel is obtained by sampling the channel $h(t)$ at baud rate. The sampled channel and its zeros are plotted in Fig. 5a, b and c, respectively.

The structures of the complex valued multilayer feedforward networks and the linear FIR equalizer along with initializations used in the simulation are given in Table 2. As in the case of linear FIR equalizer, where the length of the equalizer is required to be greater than the channel order, the number of nodes in the input layer of neural blind equalizer should also be greater than the channel length. To determine the channel order, the algorithms given in [23, 24] can be used. The parameters of the activation functions of hidden layer neurons are chosen according to the channel output. In

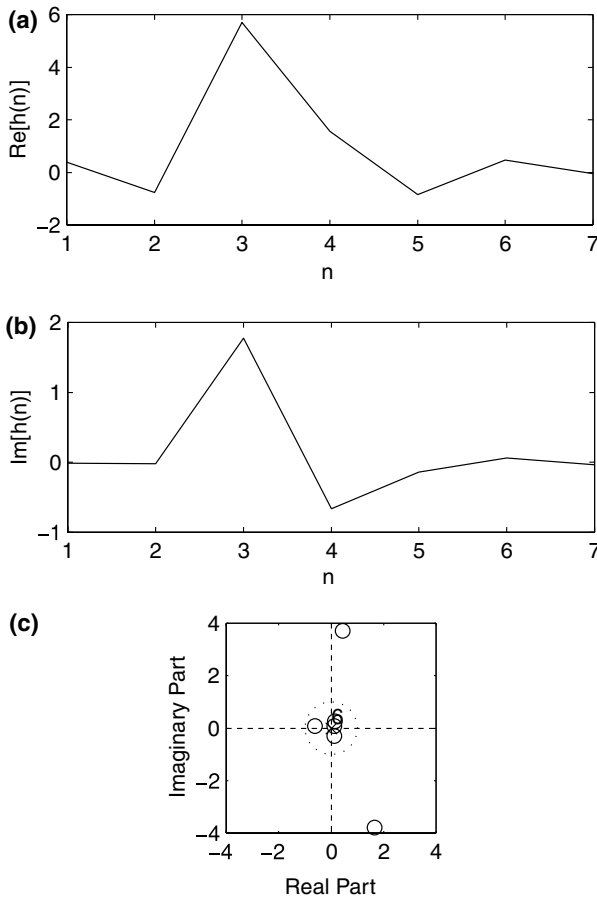


Fig. 5 Complex-valued sampled channel (CH-2). **a** The real part. **b** The imaginary part. **c** Zeros of the channel

Table 2 Structural details of the blind equalizers used in simulation ($a=2$, $b=0.5$, $\alpha=4$, $\beta=0.4$)

Type of blind equalizer	Number of nodes in the input layer	Number of nodes in hidden layer	Number of taps
NN equalizer	17	9	–
for 8-PSK signal	$w_{59,R}^{(1)} = 1$	$w_{5,R}^{(2)} = 2.5$	–
Linear FIR equalizer	–	–	25

this simulation $\eta=0.00001$. Higher value of learning rate parameter did not yield good convergence.

For the satisfactory convergence of CMA-based equalizers, the central tap of linear FIR equalizer is initialized as 1 and other taps are set to zero. The weights $w_{ij}^{(1)}$ and $w_i^{(2)}$ are initialized by small random values, close to zero, except for the real parts of the central elements of the weights, i.e., $w_{58,R}^{(1)}$ and $w_{5,R}^{(2)}$. The weight $w_{58,R}^{(1)} = 1$ while $w_{5,R}^{(2)}$ is chosen according to the channel output and is 1.5.

The output of the channel CH-1 at 20 dB SNR is shown in Fig. 6b for 8-PSK signal. Figure 6c, d show the outputs of the linear FIR and neural network equalizers, respectively.

The MSE curves for the two equalizers are shown in Fig. 7a. The MSE curves are obtained by averaging 50 independent runs. The symbol error rate performance of these blind equalizers is illustrated in Fig. 7b. The difference between symbol error rates of linear and neural network equalizers is more at higher values of SNR.

For the multipath channel CH-2, the MSE curves for the linear equalizer and the NN equalizer are given in

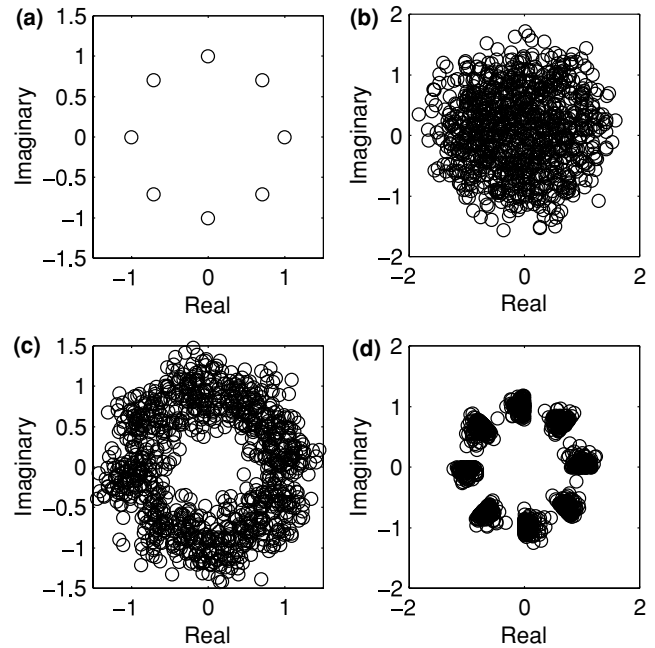


Fig. 6 **a** 8-PSK signal constellation. **b** Output of the channel CH-1 at 20 dB SNR. **c** Output of the linear equalizer for the channel CH-1. **d** Output of the neural network equalizer for the channel CH-1

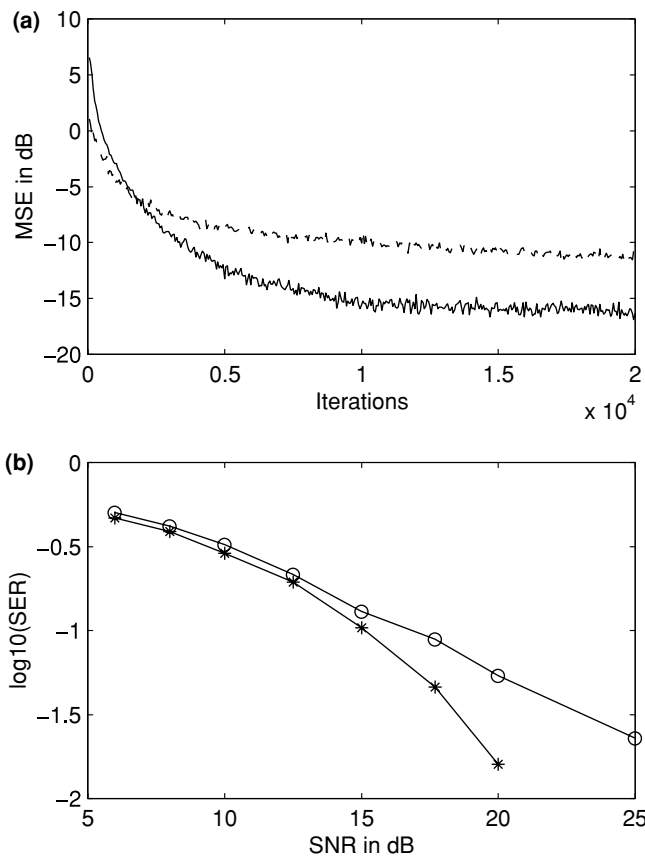


Fig. 7 Performance of linear and neural network equalizers for channel CH-1. **a** MSE curves: (solid line) neural network equalizer; (dotted line) linear equalizer. **b** SER curves: (line with circle) linear filter; (line with asterisk) neural network equalizer

Fig. 8a and the corresponding symbol error rate curves are plotted in Fig. 8b.

It can be observed that in comparison with linear FIR equalizer, the NN equalizers achieve lower MSE and symbol error rate for stationary channels CH-1 and CH-2. The MSE of NN equalizer is less than the MSE of linear FIR equalizer by about 4 dB in the case of channel CH-1, and by about 2 dB for channel CH-2.

The performance of an adaptive system in nonstationary environment depends upon the tracking ability of the training algorithm that is employed [12]. However, in order to compare the performances of linear and neural blind equalizers, both trained by the same stochastic gradient method in nonstationary environment, the simulation of a nonstationary channel is presented here.

The nonstationary channel (CH-3) used for the simulation is shown in Fig. 9a. This channel incorporates both a sudden change and a gradual change in the environment. There is a fixed zero at $z_1 = 0.5$. After 3,000 iterations another zero which is a mobile zero, appears as given below:

$$z_2(n) = 1.6 \exp\left(\frac{j2\pi}{3}\right) + 0.2 \exp(j\pi(n - 3000) 10^{-4}). \quad (24)$$

The channel suddenly changes after $n = 3,000$ and becomes a continuously varying medium. Figure 9b shows 1,000 samples of the output of this channel after $n = 5,000$ at 20 dB SNR.

For 8-PSK signal, the MSE plots of linear FIR and neural blind equalizers are shown in Fig. 10a. The MSE plots are obtained after correcting the phase shift of output symbols of the two equalizers. The symbol error rate curves shown in Fig. 10b are obtained by considering the outputs of the two equalizers after 10,000 iterations, without stopping the training. Again, the neural network equalizer gives lower MSE and symbol error rate as compared to the linear FIR filter.

5 Conclusions

In this paper, a complex-valued feedforward neural network, with complex activation functions having multisaturation characteristics, is applied for the blind equalization of complex communication channels with M-ary PSK signals. The learning rules of the complex-valued weights of the networks are based on the constant modulus algorithm (CMA). Comparison with linear FIR equalizers shows that the proposed neural equalizer is

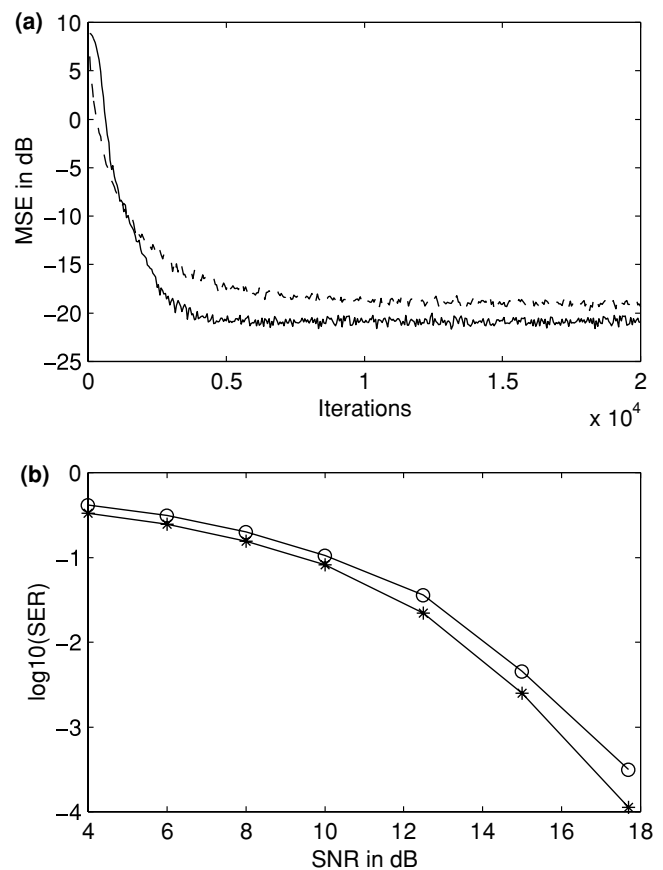


Fig. 8 Performance of linear and neural network equalizers for channel CH-2. **a** MSE curves: (solid line) neural network equalizer; (dotted line) linear equalizer. **b** SER curves: (line with circle) linear filter; (line with asterisk) neural network equalizer

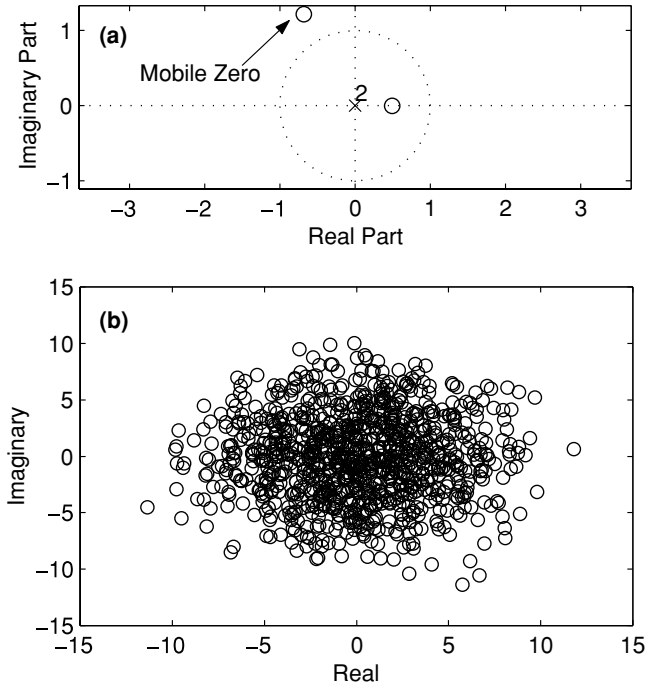


Fig. 9 **a** Zeros of the nonstationary channel; **b** output of the nonstationary channel at SNR = 20 dB

able to deliver better performance in terms of lower MSE and symbol error rate. The performance of these neural equalizers is also examined in nonstationary environment. The plots of MSE computed after correcting the phase shift of output symbols show that neural equalizers maintain lower MSE as compared to linear equalizers in nonstationary environment as well. The superior performance of the equalizer based on the neural network is attributed to its ability to form nonconvex decision regions and the decorrelation performed by the nonlinearities present in the node of the output layer. Since this nonlinear function used in the output node has been selected according to the signal constellation, this also makes the equalizer robust to noise. However, the improvement in the performance is obtained at the cost of increased computational complexity.

Appendix

The CMA cost function is expressed as

$$J(n) = \frac{1}{4} E \left[\left(|y(n)|^2 - R_2 \right)^2 \right], \quad (25)$$

where

$$R_2 = \frac{E[|s(n)|^4]}{E[|s(n)|^2]}.$$

Using the gradient descent technique, the weights of the neural network can be updated as

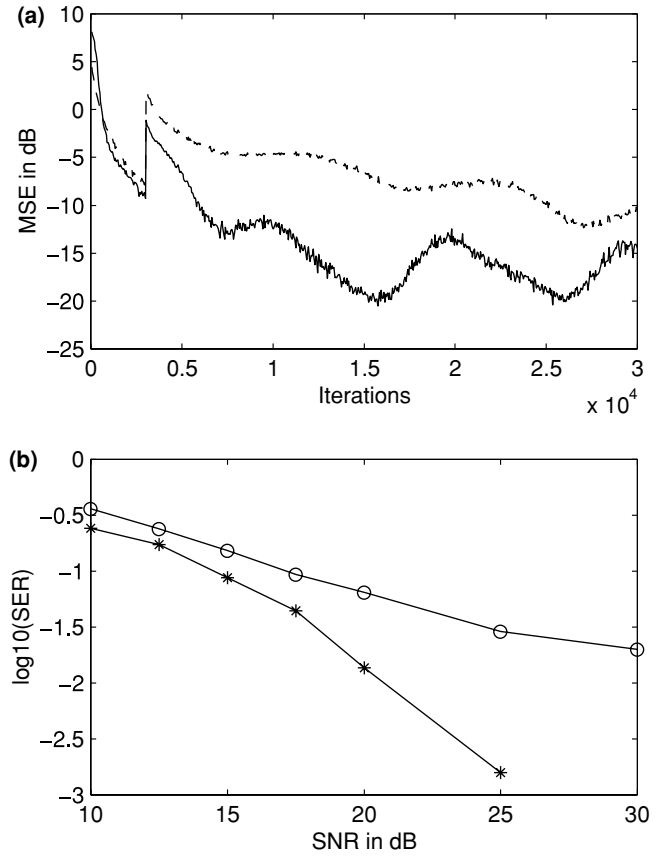


Fig. 10 Performance of linear and neural network equalizer under nonstationary channel CH-3. **a** MSE curves (solid line): neural network equalizer; (dotted line) linear equalizer. **b** SER after 10,000 iterations: (line with circle) linear equalizer; (line with asterisk) neural network equalizer

$$w_k^{(2)}(n+1) = w_k^{(2)}(n) - \eta \nabla_{w_k^{(2)}} J(n) \quad (26)$$

and

$$w_{kl}^{(1)}(n+1) = w_{kl}^{(1)}(n) - \eta \nabla_{w_{kl}^{(1)}} J(n), \quad (27)$$

where η is the learning rate parameter and the terms $\nabla_{w_k^{(2)}} J(n)$ and $\nabla_{w_{kl}^{(1)}} J(n)$ represent the gradients of the cost function $J(n)$ defined by (25) with respect to the weights $w_k^{(2)}$ and $w_{kl}^{(1)}$, respectively.

Since the activation function of the output layer neuron for M-ary PSK signal is defined in terms of modulus and angle of the activation sum, the gradient of the CMA cost function with respect to the output layer weight $w_k^{(2)}(n)$ is expressed as

$$\begin{aligned} \nabla_{w_k^{(2)}} J(n) &= (|y(n)|^2 - R_2) |y(n)| \left(ab - \frac{b}{a} |y(n)|^2 \right) \\ &\quad \times \frac{\partial \text{net}^{(2)}(n)}{\partial w_k^{(2)}(n)}. \end{aligned} \quad (28)$$

To obtain an expression for the partial derivative of (28), we use the relationship

$$|\text{net}^{(2)}(n)|^2 = \left(\text{net}_{\text{R}}^{(2)}(n)\right)^2 + \left(\text{net}_{\text{I}}^{(2)}(n)\right)^2. \quad (29)$$

On differentiating (29) with respect to $w_k^{(2)}$, we get

$$\begin{aligned} \frac{\partial |\text{net}^{(2)}(n)|}{\partial w_k^{(2)}(n)} &= \frac{1}{|\text{net}^{(2)}(n)|} \left[\text{net}_{\text{R}}^{(2)}(n) \frac{\partial \text{net}_{\text{R}}^{(2)}(n)}{\partial w_k^{(2)}(n)} + \text{net}_{\text{I}}^{(2)}(n) \frac{\partial \text{net}_{\text{I}}^{(2)}(n)}{\partial w_k^{(2)}(n)} \right] \\ &= \frac{1}{|\text{net}^{(2)}(n)|} \left[\text{net}_{\text{R}}^{(2)}(n) \left(\varphi^{(1)}(\text{net}_{k,\text{R}}^{(1)}(n)) - j\varphi^{(1)}(\text{net}_{k,\text{I}}^{(1)}(n)) \right) + \text{net}_{\text{I}}^{(2)}(n) \left(\varphi^{(1)}(\text{net}_{k,\text{I}}^{(1)}(n)) + j\varphi^{(1)}(\text{net}_{k,\text{R}}^{(1)}(n)) \right) \right] \\ &= \frac{\varphi^{(1)*}(\text{net}_k^{(1)}(n))\text{net}^{(2)}(n)}{|\text{net}^{(2)}(n)|} \end{aligned} \quad (30)$$

On substituting (30) in (28), the expression for the gradient becomes

$$\nabla_{w_k^{(2)}} J(n) = \delta^{(2)}(n) u_k^*(n) \quad (31)$$

where

$$\begin{aligned} \delta^{(2)}(n) &= \left[\left(|y(n)|^2 - R_2 \right) |y(n)| \left(ab - \frac{b}{a} |y(n)|^2 \right) \right] \\ &\quad \times \frac{\text{net}^{(2)}(n)}{|\text{net}^{(2)}(n)|}. \end{aligned} \quad (32)$$

Substitution of (31) in (26) along with (32) gives the update equation (17) with (18) for M-ary PSK signal.

In order to obtain the update equation for the weights $\{w_{kl}^{(1)}\}$, we need the following gradient

$$\begin{aligned} \nabla_{w_{kl}^{(1)}} J(n) &= \left(|y(n)|^2 - R_2 \right) |y(n)| \\ &\quad \times \left(ab - \frac{b}{a} |y(n)|^2 \right) \frac{\partial |\text{net}^{(2)}(n)|}{\partial w_{kl}^{(1)}(n)}. \end{aligned} \quad (33)$$

The partial derivative terms in (33) can be obtained by using (29)

$$\begin{aligned} \frac{\partial |\text{net}^{(2)}(n)|}{\partial w_{kl}^{(1)}(n)} &= \frac{1}{|\text{net}^{(2)}(n)|} \left[\text{net}_{\text{R}}^{(2)}(n) \frac{\partial \text{net}_{\text{R}}^{(2)}(n)}{\partial w_{kl}^{(1)}(n)} \right. \\ &\quad \left. + \text{net}_{\text{I}}^{(2)}(n) \frac{\partial \text{net}_{\text{I}}^{(2)}(n)}{\partial w_{kl}^{(1)}(n)} \right], \end{aligned} \quad (34)$$

where

$$\begin{aligned} \frac{\partial \text{net}_{\text{R}}^{(2)}(n)}{\partial w_{kl}^{(1)}(n)} &= w_{k,\text{R}}^{(2)}(n) \varphi^{(1)'}(\text{net}_{k,\text{R}}^{(1)}(n)) (x_{l,\text{R}}(n) - jx_{l,\text{I}}(n)) \\ &\quad - w_{k,\text{I}}^{(2)}(n) \varphi^{(1)'}(\text{net}_{k,\text{I}}^{(1)}(n)) (x_{l,\text{I}}(n) + jx_{l,\text{R}}(n)) \end{aligned} \quad (35)$$

and

$$\begin{aligned} \frac{\partial \text{net}_{\text{I}}^{(2)}(n)}{\partial w_{kl}^{(1)}(n)} &= w_{k,\text{R}}^{(2)}(n) \varphi^{(1)'}(\text{net}_{k,\text{I}}^{(1)}(n)) (x_{l,\text{I}}(n) + jx_{l,\text{R}}(n)) \\ &\quad + w_{k,\text{I}}^{(2)}(n) \varphi^{(1)'}(\text{net}_{k,\text{R}}^{(1)}(n)) (x_{l,\text{R}}(n) - jx_{l,\text{I}}(n)) \end{aligned} \quad (36)$$

Now the substitution of (35) and (36) in (34) and some simplification lead to

$$\begin{aligned} \frac{\partial |\text{net}^{(2)}(n)|}{\partial w_{kl}^{(1)}(n)} &= \frac{x_l^*(n)}{|\text{net}^{(2)}(n)|} \left(\varphi^{(1)'}(\text{net}_{k,\text{R}}^{(1)}(n)) \text{Re}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] \right. \\ &\quad \left. - j\varphi^{(1)'}(\text{net}_{k,\text{I}}^{(1)}(n)) \text{Im}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] \right). \end{aligned} \quad (37)$$

Finally, by substituting (37) in (33), we get

$$\begin{aligned} \nabla_{w_{kl}^{(1)}} J(n) &= \delta^{(2)}(n) x_l^*(n) \left(\varphi^{(1)'}(\text{net}_{k,\text{R}}^{(1)}(n)) \right. \\ &\quad \left. \text{Re}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] - j\varphi^{(1)'}(\text{net}_{k,\text{I}}^{(1)}(n)) \right. \\ &\quad \left. \text{Im}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] \right) \\ &= \delta_k^{(1)}(n) x_l^*(n) \end{aligned} \quad (38)$$

where

$$\begin{aligned} \delta_k^{(1)}(n) &= \frac{\delta^{(2)}(n)}{\text{net}^{(2)}(n)} \left(\varphi^{(1)'}(\text{net}_{k,\text{R}}^{(1)}(n)) \text{Re}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] \right. \\ &\quad \left. - j\varphi^{(1)'}(\text{net}_{k,\text{I}}^{(1)}(n)) \text{Im}[w_k^{(2)}(n) \text{net}^{(2)*}(n)] \right). \end{aligned}$$

Using (38) and (27), we get the update rule of (19) for M-ary PSK signal.

References

1. Amari SI, Cichocki A (1998) Adaptive blind signal processing—Neural network approaches. *Proc IEEE* 86(10):2026–2048
2. Chen S, Gibson GJ, Cowan CFN, Grant PM (1990) Adaptive equalization of finite nonlinear channels using multilayer perceptrons. *Signal Process* 20:107–109
3. Chow TWS, Fang Y (2001) Neural blind deconvolution of MIMO noisy channels. *IEEE T Circuits Syst-I* 48(1):116–120
4. Lin H, Amin M (2003) A dual mode technique for improved blind equalization for QAM signals. *IEEE Signal Process Lett* 10(2):29–31
5. Thirion MN, Moreau E (2002) Generalized criterion for blind multivariate signal equalization. *IEEE Signal Process Lett* 9(2):72–74
6. Destro Filho JB, Favier G, Travassos Romano JM (1996) Neural networks for blind equalization. *Proc IEEE Int Conf Globcom* 1:196–200
7. Fang Y, Chow TWS, Ng KT (1999) Linear neural network based blind equalization. *Signal Process* 76(1):37–42

8. Feng CC, Chi CY (1999) Performance of cumulant based inverse filters for blind deconvolution. *IEEE Trans Signal Proces* 47(7):1922–1935
9. Choi S, Cichocki A (1998) Cascade neural networks for multichannel blind deconvolution. *Electron Lett* 34(12):1186–1187
10. Kechriotis G, Zervas E, Manolakos ES (1994) Using recurrent neural networks for adaptive communication channel equalization. *IEEE T Neural Networ* 5(2):267–278
11. Haykin S (1994) *Blind deconvolution*. Prentice-Hall, Englewood Cliffs
12. Haykin S (1996) *Adaptive filter theory*, 3rd edn. Prentice Hall, Upper Saddle River
13. You C, Hong D (1998) Nonlinear blind equalization schemes using complex valued multilayer feedforward neural networks. *IEEE T Neural Networ* 9(6):1442–1455
14. Haykin S (1994) *Neural networks: a comprehensive foundation*. Prentice Hall, Upper Saddle River
15. Haykin S (1996) Neural networks expand SP's horizons. *IEEE Signal Process Mag*, pp 24–49
16. Luo FL, Unbehauen R (1997) *Applied neural networks for signal processing*. Cambridge University Press, UK
17. Cichocki A, Unbehauen R (1994) *Neural networks for optimization and signal processing*. Wiley, Chichester
18. Benvenuto N, Piazza F (1992) On the complex backpropagation algorithm. *IEEE T Signal Proces* 40:967–969
19. Johnson CR et al (1998) Blind equalization using the constant modulus criterion : a review. *Proc IEEE* 86(10):1927–1950
20. Schniter P, Johnson CR Jr (1999) Dithered signed error CMA: robust, computationally efficient blind adaptive equalization. *IEEE T Signal Proces* 47(6):1592–1603
21. Jalon NK (1992) Joint blind equalization, carrier recovery and timing recovery for high order QAM constellations. *IEEE T Signal Proces* 40:1383–1398
22. Pandey R (2001) *Blind equalization and signal separation using neural networks*. PhD. thesis, I.I.T. Roorkee, India
23. Liavas AP, Regalia PA, Delmas JP (1999) Blind channel approximation: effective channel order determination. *IEEE T Signal Proces* 47(12):3336–3344
24. Vesin JM, Gruter R (1999) Model selection using a simplex reproduction genetic algorithm. *Signal Proces* 78:321–327