Neurocomputing 72 (2009) 3771–3781

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A new learning algorithm with logarithmic performance index for complex-valued neural networks

R. Savitha [a,*], S. Suresh [b], N. Sundararajan [a], P. Saratchandran [a]

[a] School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore
[b] Korea University, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 10 September 2008
Received in revised form
10 April 2009
Accepted 2 June 2009
Communicated by K. Li
Available online 21 June 2009

Keywords:
Split complex network
Fully complex-valued networks
Multi-layer perceptron
Logarithmic performance index

ABSTRACT

In a fully complex-valued feed-forward network, the convergence of the Complex-valued Back Propagation (CBP) learning algorithm depends on the choice of the activation function, learning sample distribution, minimization criterion, initial weights and the learning rate. The minimization criteria used in the existing versions of CBP learning algorithm in the literature do not approximate the phase of complex-valued output well in function approximation problems. The phase of a complex-valued output is critical in telecommunication and reconstruction and source localization problems in medical imaging applications. In this paper, the issues related to the convergence of complex-valued neural networks are clearly enumerated using a systematic sensitivity study on existing complex-valued neural networks. In addition, we also compare the performance of different types of split complex-valued neural networks. From the observations in the sensitivity analysis, we propose a new CBP learning algorithm with logarithmic performance index for a complex-valued neural network with exponential activation function. The proposed CBP learning algorithm directly minimizes both the magnitude and phase errors and also provides better convergence characteristics. Performance of the proposed scheme is evaluated using two synthetic complex-valued function approximation problems, the complex XOR problem, and a non-minimum phase equalization problem. Also, a comparative analysis on the convergence of the existing fully complex and split complex networks is presented.

1. Introduction

For a fully complex-valued feed-forward network using a Complex-valued Back Propagation (CBP) learning algorithm, the activation function and its derivatives have to be well behaved everywhere in the complex plane. If one uses the well-known complex-valued non-linear activation functions, it is known that the activation function cannot be analytic and bounded everywhere in the complex plane [1]. Hence, developing a fully complex-valued neural network is a challenging task. In the literature, there have been three approaches to circumvent this problem.

- The first approach is to replace the complex-valued inputs and outputs with pairs of independent real-valued signals and approximate using conventional real-valued neural network. For splitting the complex-valued signal into pair of independent real-valued signal, one can use either rectangular or polar coordinate representations, and the approach is commonly referred as split complex-valued network [2]. Among various approaches in literature, split complex-valued multi-layer perceptron (split-CMLP) is quite popular [2]. The split functions overcome the problem of well behaved activation function, but it suffers from poor approximation of complex-valued functions.
- The second approach is to construct a real-valued activation function which transforms complex-valued input to real-valued output (i.e., maps $F: C^n \rightarrow R$) [3]. In this approach, the activation functions are real-valued but the connection weights between the layers are complex-valued. However, as will be shown later, the criterion used to update the parameters of the network is real-valued and does not consider phase error minimization, explicitly. So, this approach does not approximate the phase of the complex signal efficiently.
- The third approach is to define the fully complex-valued activation function in the form $f(z) = sgm(Real(z)) + j sgm(Imag(z))$, where sgm is the sigmoidal activation function. For some applications in telecommunication, the third approach yields a more efficient structure than the other two approaches, both in terms of computational complexity and performance [4,5]. But the choice of real-valued function for real and imaginary components rely on the application

* Corresponding author.
E-mail address: savi0001@ntu.edu.sg (R. Savitha).

0925-2312/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2009.06.004

addressed and identification of functions for each application is difficult.

Even though the last two approaches provide better performance in few applications, they suffer from poor convergence characteristics. In general, the initialization of the complex-valued weights, learning rate, and also learning sample population distribution affect the convergence of the neural network. Recently, in [6], a detailed sensitivity analysis for split-CMLP is presented. It is observed that, in the process of splitting the complex-valued signals into real and imaginary components, the complex-valued function approximation introduces phase distortion, i.e., the phase information of complex-valued output is not captured accurately. In many communication application problems, accurate phase estimation is very important [7]. In general, the split complex networks overcome the problem of unbounded complex functions and its derivatives, but they suffer from poor approximation of phase information.

To overcome the problem of unbounded nature of complex functions, recently, in [1], a fully complex-valued multi-layer perceptron (CMLP) using elementary transcendental functions (ETFs) has been proposed. These functions are bounded almost-everywhere and are analytic in the complex plane. The properties of CMLP have been studied and its performance has been compared with the split-CMLP networks [1]. It is worth noting here that the ETFs and their derivatives have finite number of singular points. These singular points affect the convergence of the CMLP. Also, the CMLP uses well-known mean square error performance measure for the parameters update as in the case of split-CMLP. Since the mean square error function for complex-valued outputs ($e \times \overline{e} = |e|$) does not consider the phase error directly, they suffer from phase distortion.

For CMLP, there have been many attempts for properly selecting the activation functions to avoid the singularity problem [1,8–10]. The essential properties needed for the complex-valued activation functions are given in [8]. This paper also presents a new complex phase invariant activation function, which satisfies all the essential properties for a complex-valued activation function. The new activation function maps any complex number to an open disk in the complex plane by retaining the same phase value. In [9], a new complex-valued activation function using two bounded real functions has been presented. This function will be effective when the target function has axial symmetry, like equalization for quadrature amplitude modulation. Similar to other complex-valued learning algorithms, this algorithm also suffers from convergence problem due to weight initialization, learning rates and input signals. For recurrent networks, an augmented complex-valued extended Kalman filter algorithm has been developed in [10]. This is useful for processing general complex-valued non-linear and non-stationary signals and uses the complex-valued $\tanh(\cdot)$ activation function. However, the computational cost using the complex EKF learning is more expensive than complex real time recurrent learning. So, it is understandable that the complex-valued neural networks suffer drawbacks due to choice of proper activation function and minimization criterion. The activation functions are either unbounded [1], or do not map non-linear phase [8], or application specific [9]. On the other hand, the minimization function does not consider error in phase, and hence, do not help in approximating the phase of the complex signal efficiently. Besides, the network is also sensitive to weight initializations and learning rates.

In this paper, we study systematically the various issues involved in the complex-valued neural networks, such as the selection of activation function, minimization criterion, weight initialization and the learning rate. For this purpose, we first consider two synthetic complex-valued function approximation problems. We evaluate the performance of both the split-CMLP and CMLP with ETFs on these problems. The issues relating to the existing split-complex and CMLP networks are listed. Based on the observation from the sensitivity analysis, we propose a new activation function for CMLP and a new minimization criterion for CBP learning algorithm. The proposed algorithm uses a complex-valued exponential activation function, which has singularity at infinity. Moreover, the minimization criterion, shown in Eq. (1), is usually used in literature. This criterion does not include the error in phase directly and hence, does not approximate phase accurately:

$$J = \frac{1}{2}\sum_{i=1}^{n} e \times \overline{e} \tag{1}$$

A new minimization criterion, using a logarithmic error function, that directly minimizes both the errors in magnitude and phase is proposed. The advantages of the proposed scheme are enumerated with the above problems first, and then with the complex XOR and a non-minimal phase equalization problems.

This paper is organized as follows: Section 2 summarizes the different complex-valued neural networks available in the literature along with their merits and demerits. In Section 3, we present the improved complex-valued back propagation learning algorithm. The proposed algorithm overcomes the singularity problem in the activation function and also uses a minimization criterion for better convergence. In Section 4, the performance results using two synthetic complex-valued function approximation problems, a complex XOR problem and a non-minimum phase equalization problem are presented along with a comparative evaluation of other well-known approaches in the literature. Section 5 summarizes the conclusion from this study.

2. A brief review of complex-valued neural networks

In general, the complex-valued network can be classified based on the usage of complex-valued signal in split complex networks and types of activation function in fully complex-valued networks. The typical classification of complex-valued networks is shown in Fig. 1 and they are briefly explained below along with their limitations.

2.1. Split complex neural networks

Split complex networks are commonly used to avoid the singularity problems in complex-valued functions and their derivatives. The split complex function network is further divided into two types based on the nature of weights, namely, split complex networks with real-valued weights and real-valued activation functions and split complex networks with complex-valued weights and real-valued activation functions.

2.1.1. Real-valued weights and real-valued activation functions

The neural network architecture is similar to the classical multi-layer perceptron network with the back propagation algorithm. Here, the complex-valued inputs and targets are split into two real-valued quantities, based on either rectangular (real-imaginary) or polar (magnitude-phase) coordinate system. For example, two complex-valued inputs and one complex-valued output problem is converted into four real-valued inputs and two real-valued outputs problem. However, irrespective of the kind of splitting the complex value, the fully complex feed-forward

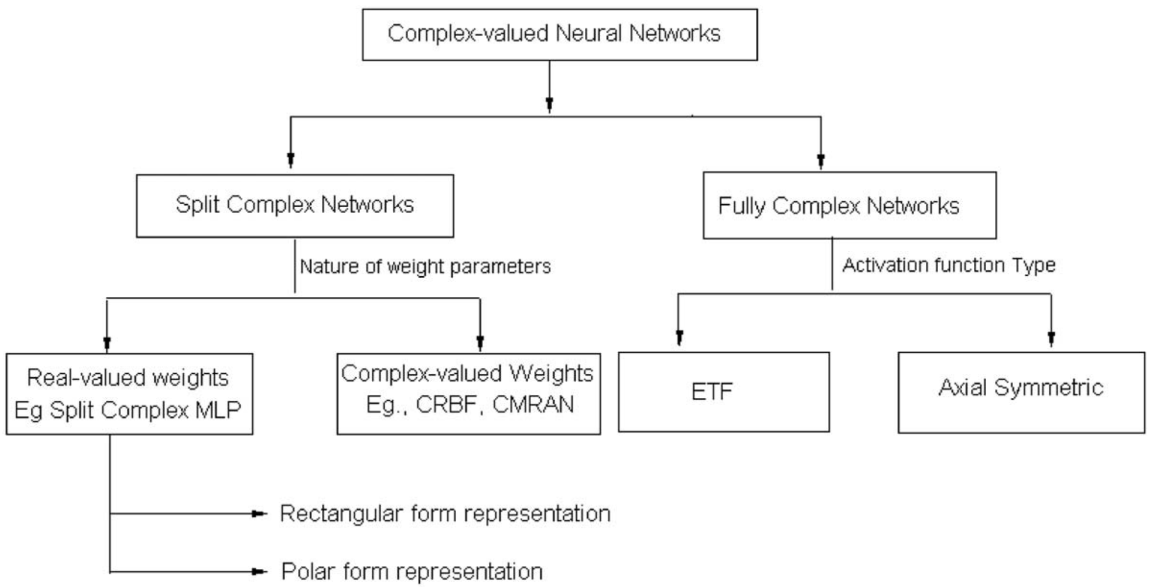

Fig. 1. Overall classification of complex-valued neural networks.

network performs better than the split complex feed-forward networks in function approximation problems. This is because, in such split complex networks, the real-valued gradient update does not reflect the true complex-valued gradient [1].

2.1.2. Complex-valued weights and real-valued activation functions

To overcome the problem of phase distortion due to the splitting the complex-valued signal, complex-valued weights with real-valued activation functions have been proposed in [3,11,12]. Here, the activation functions are selected such that the function maps the complex domain into the real domain, $f: C^n \to R$. As mentioned earlier, these networks also use real-valued activation functions to overcome the problems of singularities in complex-valued functions. In [3,11], the product of complex-valued error and its conjugate is used to estimate the weights. Such function minimizes the magnitude error and neglects the phase error, as will be shown later in Section 3. Also, the convergence of split-complex network with complex-valued weights depends on proper initialization and the choice of learning rate [6].

2.2. Fully complex-valued neural networks

The feed-forward network with a fully complex-valued activation function is capable of handling complex-valued inputs and outputs. Fully complex-valued networks use complex-valued functions as activation functions with complex-valued weights. The learning algorithm used in the fully complex-valued neural network relies on the well-defined complex-valued gradients.[1] Thus, the main issue in a fully complex-valued neural network is the proper selection of the complex-valued activation function and its derivatives. For the activation function selection, the complex-valued function should satisfy the essential properties as stated in [8] as

- $f(z)$ should be non-linear in x and y.
- $f(z)$ should be bounded.
- The partial derivatives $\partial u/\partial x, \partial u/\partial y, \partial v/\partial x, \partial v/\partial y$ exist and are bounded.
- $f(z)$ is not entire.[2]
- $u_x v_y \neq v_x u_y$ unless $u_x = v_x = 0$ and $v_y = u_y = 0$.

These conditions were, then reduced and relaxed in [1] as:

> In a bounded domain of complex plane C, a fully complex non-linear activation function $f(z)$ needs to be analytic[3] and bounded almost everywhere.

The fully complex-valued neural network is classified further based on the nature of the complex-valued activation function used.

2.2.1. Elementary transcendental activation functions

A fully complex-valued neural network with ETF has been introduced for multi-layer perceptron and its fully complex-valued back propagation weight update rule is presented in [1]. The learning algorithm presented in [1] is complex-valued version of the real-valued back propagation algorithm. For complete details on the properties of different ETFs and complex-valued back propagation, one should refer to [1]. The ETFs satisfy the properties for suitable activation functions, but they have essential, removable or isolated singularities at different locations in the complex domain [13]. For example, asinh$(\cdot)$ is a simple ETF, which has branch cut singularity along the imaginary axis. The derivative of asinh$(\cdot)$ has an isolated singularity at $\pm i$. If the learning algorithm operates in the region of singularity, the parameter update becomes unstable, which affects its convergence. Hence, the fully complex-valued algorithm is sensitive to sample population distribution, weight initialization and the choice of the learning rate. The convergence effects are different for different ETFs and they also depend on the nature of the problem.

[1] If $z = x + iy$, the complex gradient is defined as

$$\frac{\partial f}{\partial z} = \frac{1}{2}\left(\frac{\partial f}{\partial x} - j\frac{\partial f}{\partial y}\right)$$

[2] In complex analysis, an entire function, also called an integral function, is a complex-valued function that is analytic over the whole complex plane.

[3] A complex function is said to be analytic on a region R if it is complex differentiable at every point in R.

2.2.2. Axial symmetric activation function

To overcome the singularity problem in complex-valued activation function, an axial symmetric complex-valued function is introduced, to deal with QAM signals in [9]. The general form of axial-symmetric activation function is given by

$$f(z) = f(x) + if(y) \tag{2}$$

where $z \varepsilon C$, $x = real(z)$, $y = imag(z)$ and $f(\cdot)$ is any continuous function. The axial symmetric function satisfies the essential properties of an activation function to be used in the complex domain. The axial symmetric function is suitable for problems, which have symmetric targets, such as equalization in telecommunication. Even though the axial symmetric activation function does not have the singularity, it does not consider the interaction between the real and imaginary parts of a complex-valued variable. Also, selection of appropriate continuous activation function for a given problem is difficult.

2.2.3. Issue due to minimization criterion

Another important aspect in complex-valued back propagation algorithm is the selection of an appropriate minimization criterion. In most of the algorithms presented in the literature, this product of the error and its conjugate (Euclidean norm), as given below, is used as a performance measure for minimization:

$$J = \tfrac{1}{2}(e \times \overline{e}) = \tfrac{1}{2}(e_x^2 + e_y^2) \tag{3}$$

where $e = e_x + ie_y = z - \hat{z} = (x - \hat{x}) + j(y - \hat{y})$, $\overline{e}$ is the complex-conjugate of error, $\hat{z}$ is predicted output, e_x is error in real part and e_y is error in imaginary part.

From Eq. (3), one can see that the performance measure minimizes only the square magnitude error and does not consider the phase quantity of the error (e) directly. Hence, the network evolved using such a performance measure does not consider the phase error directly and the network does not estimate the phase accurately, which is important for problems in the domain of telecommunication and image reconstruction. In general, for better approximation of complex-valued function (i.e., both in magnitude and phase), one should find an appropriate complex-valued activation function and a suitable minimization criterion. In the next section, we show the effect of convergence due to initialization and performance index, by conducting a sensitivity analysis on different complex-valued networks in the literature.

2.3. Sensitivity analysis

In the previous subsection, different complex-valued networks and split-complex network were presented. For a given problem, the overall performance and convergence of the complex-valued networks depends on the initial weights and learning rate. To evaluate the performance and sensitivity of different complex-valued networks, we define the mean square error in magnitude and average phase error as

$$J_{Me} = \frac{1}{N_t}\sum_{j=1}^{N_t}((e_i.\overline{e}_i)) \tag{4}$$

and

$$\Phi_e = \frac{1}{N_t}|[arg(z) - arg(\hat{z})]| \tag{5}$$

where N_t is number of samples in the training set. For a fully complex-valued network with a given activation function, the convergence and performance of the network depends significantly on the complex-valued weight initialization and learning rate. To evaluate this, systematic study has been performed here, on two synthetic complex-valued function approximation problems, by initializing the weights within a small ball of radius (r) and different learning rates for various ETFs. For each initializing values, the J_{Me} and ϕ_e are computed along with a statistical analysis on different runs.

The convergence of different activation functions is studied first. The following procedure is followed to learn the convergence of different activation functions. Initially, the performance of split complex-valued activation function is studied, presenting the training sample set over 1000 epochs. The fully complex-valued neural networks are supposed to perform better than the split complex networks. So if the J_{Me} of the network with fully complex-valued activation is less than that of the split complex-valued network after 1000 epochs, the network is said to be convergent. Otherwise, it is considered as a failure to converge. This study is conducted for a learning rate of 0.1, 0.01 and 0.001, with initial weight radius of 0.1, 0.01, 0.001 and 10^{-6}. The study was conducted with weights uniformly distributed in each of the four quadrants and in all quadrants. So, a total of 60 experiments were conducted for each activation function considered. The fully complex-valued activation functions considered for study were: (a) asinh$(\cdot)$; (b) asin$(\cdot)$; (c) atanh$(\cdot)$; (d) atan$(\cdot)$; (e) tanh$(\cdot)$ and (f) tan$(\cdot)$.

Percentage of failure (failure to converge) is measured by

$$\%failure = \frac{N_f}{N} \tag{6}$$

where N_f is number of failures at a particular learning rate (η) and N is number of experiments at that learning rate (η).

2.3.1. Synthetic complex-valued function approximation problem I

The synthetic complex-valued function to be approximated is defined as

$$f_1(z) = z_1^2 + z_2^2 \tag{7}$$

where $\underset{\sim}{z}$ is a two-dimensional complex-valued vector, z_1 and z_2 are complex-valued numbers with magnitude less than 2.5. Here, the targets are the continuous complex variables. For training the fully complex-valued multi-layer perceptron network for the synthetic complex-valued function approximation, 3000 samples (z_1 and z_2) for training and 1000 samples for testing are randomly generated and used. The network architecture used in the study is 2:25:1; 2 input neurons, 25 hidden neurons and 1 output neuron. The best number of hidden neurons is selected based on constructive and destructive procedure given in [14].

The performance of the split-complex-valued network for the synthetic complex-valued function approximation problem I is given in Table 1. The split complex network in polar form fails to minimize the phase error, whereas the polar form split complex performance index converges to 0.016. If J_{Me} is less than the performance achieved by split-complex network (i.e., J_{Me} is less than 0.016) then the algorithm is said to be convergent, otherwise it is not convergent.

The observation on performance of the six different ETFs, mentioned in Section 2.3, with different learning rates and weight initializations is given in Table 2. From Table 2, it can be seen that all the ETFs fail to converge with a higher learning rate. For higher

Table 1
Performance of split complex networks for function approximation problem I.

Type	Training		Testing	
	J_{Me}	ϕ_e	J_{Me}	ϕ_e (rad.)
Rectangular form	0.022	0.029	0.046	0.083
Polar form	0.016	0.696	0.031	0.740

Table 2
Convergence analysis results for fully complex-valued network.

ETFs	Region of failure		Percentage of failures
	Learning rate	Weight initialization	
tanh(·)	0.1	ALL	100
	0.01	II, III quadrants	33
tan(·)	0.1	ALL	100
	0.01	II and III	40
asin(·)	0.1	ALL	100
atan(·)	0.1	ALL	100
asinh(·)	0.1	ALL	100
atanh(·)	0.1	ALL	100

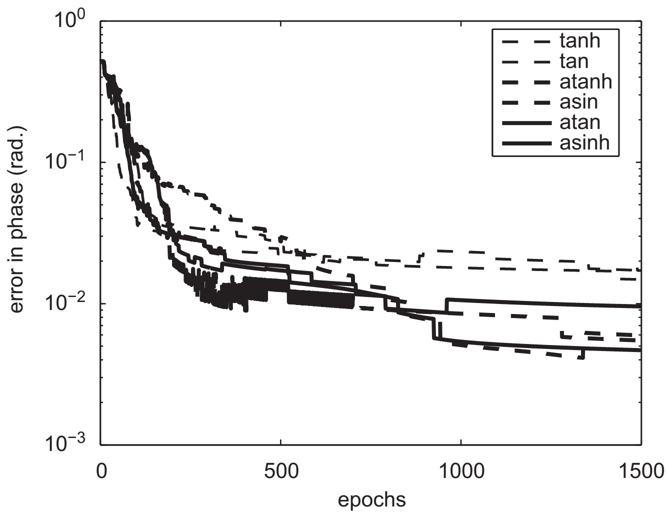

Fig. 2. Phase error (ϕ_e) in rads for different ETFs at a learning rate of 0.001 (synthetic function approximation problem I).

learning rate, it is difficult to see the effect of weight initialization. The ETFs tanh(·) and tan(·) are sensitive to weight initialization in lower learning rates. For example, the activation function tanh(·) hit the singular regions when the weights are initialized in II and III quadrants.

All the ETFs converge when the learning rates are less than 0.001. Also, they are not sensitive to weight initialization at lower learning rates. Fig. 2 shows the phase error convergence of ETFs at lower learning rates. From the figure, it can be seen that the phase error for complex-valued activation functions atanh(·) and asin(·) are oscillating considerably even at lower learning rates.

2.3.2. *Synthetic complex-valued function approximation problem II*

The function to be approximated is given

$$f_2(z) = z_3 + 10z_1z_4 + \frac{z_2^2}{z_1} \tag{8}$$

where z is a four-dimensional complex-valued vector, z_1, z_2, z_3 and z_4 are complex-valued numbers with magnitude less than 2. A training sample set with 3000 samples and testing sample set with 1000 samples were randomly generated for the study. Training was done with a network with 4 input neurons, 15 hidden neurons and 1 output neuron. The function approximation results with split complex-valued network, over 1000 epochs, are given in Table 3. It also clearly supports our argument that the split complex-valued network fails to approximate phase.

Table 3
Performance of split complex networks for function approximation problem II.

Type	Training		Testing	
	J_{Me}	ϕ_e	J_{Me}	ϕ_e (rad.)
Rectangular form	0.1933	1.9295	0.2332	1.9892
Polar form	0.2452	1.5397	0.2840	1.5846

Table 4
Convergence analysis results for fully complex-valued network.

ETFs	Region of failure		Percentage of failures
	Learning rate	Weight initialization	
tanh(·)	0.1	ALL	100
tan(·)	0.1	ALL	100
asin(·)	0.1	ALL	100
atan(·)	0.1	ALL	100
asinh(·)	0.1	ALL	100
atanh(·)	0.1	ALL	100

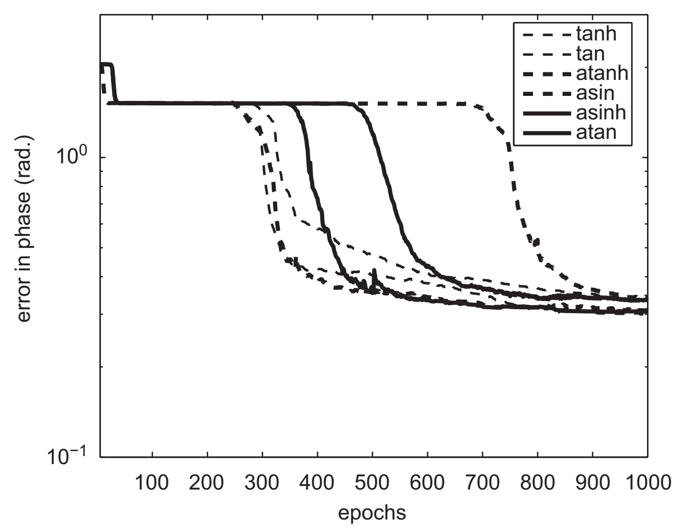

Fig. 3. Phase error (ϕ_e) in rads for different ETFs at a learning rate of 0.001 (synthetic function approximation problem II).

Convergence analysis study was done with the six activation functions with different weight initializations and learning rates. The minimum achievable J_{Me} with the split complex-valued network was 0.1933. So, if the J_{Me} of fully complex-valued network is less than 0.1933 in 1000 epochs, the network is convergent. Otherwise, it is not convergent. Table 4 gives the convergence study results of the different ETFs chosen.

From Table 4, it can be observed that the network with ETFs as activation function failed at higher learning rates, at learning rates as high as 0.1. However, they converge for lower learning rates, viz., $\eta = 0.01$ and 0.001. Fig. 3 gives the phase error convergence of ETFs at a learning rate of 0.001, over 1000 epochs. From the phase error convergence characteristic of ETFs, it is observed that tan(·), tanh(·), asin(·) and atanh(·) activation functions have poor convergence characteristic as compared to asinh(·) and atan(·). While atanh(·) and tan(·) activations showed oscillations in phase error over epochs, in the case of tanh(·) and asin(·), the phase error achievable is lesser. It can be inferred from these results that asinh(·) and atan(·) activations are a better choice in function approximation problems.

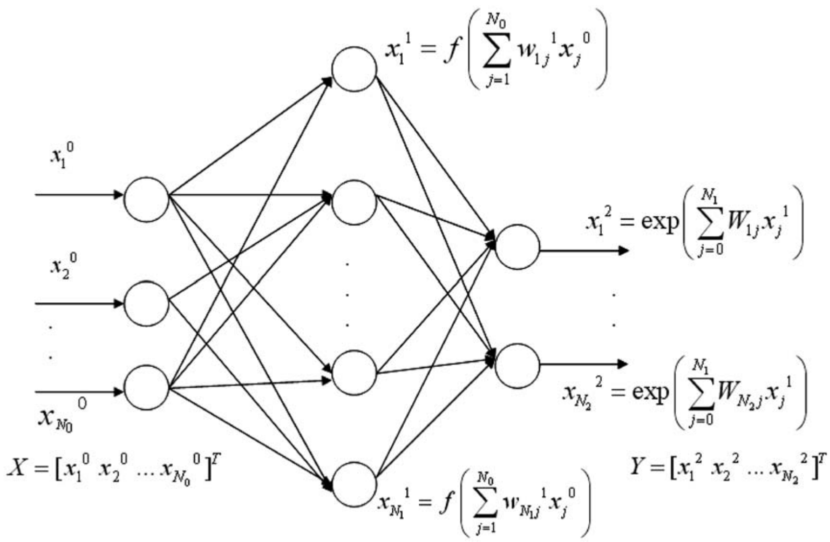

Fig. 4. Structure of the three-layered complex MLP network.

From the sensitive analysis study, the following observations emerge:

- The ETFs tanh(·) and tan(·) are sensitive to weight initialization in lower learning rates, especially, in the II and III quadrants.
- The phase error for complex-valued activation functions atanh(·) and asin(·) is oscillating considerably even at lower learning rates.
- The activation functions tan(·), tanh(·), asin(·) and atanh(·) have poor convergence characteristic as compared to asinh(·) and atan(·).
- Among the ETFs, asinh(·) and atan(·) are suitable activation functions in function approximation problems.

3. Proposal of a new learning algorithm for fully complex-valued neural network

Here, an exponential function is used as an activation function for non-linear processing of complex-valued data. The exponential function, $f(z) = \exp(z)$, is entire since $f(z) = f'(z) = \exp(z)$ in C. The complex-valued exponential function has an essential singularity at $+\infty$. By restricting the weights of the network to a small ball of radius (w_r) and the number of hidden neurons to a finite value, the bounded behavior in fully complex-valued multi-layer perceptron network can be achieved.

For illustration, a three layer ($l = 2$) complex-valued network has been considered. The structure of the three layer complex-valued network is as shown in Fig. 4. The number of neurons in the input, hidden and output layers is N_0, N_1 and N_2, respectively. Let $\mathbf{X} \in \mathrm{C}^{N_0}$ be the N_0 dimensional complex-valued input to the network. The signal of each neuron in the input layer is denoted by $x_i^0;\ i = 1, 2, \ldots, N_0$ where the superscript 0 represents the layer number and N_0 is the dimension of the input. The output be $\mathbf{Y} \in \mathrm{C}^{N_2}$, with the signal at each neuron (in the output layer) denoted by $x_j^2;\ j = 1, 2, \ldots, N_2$ where the superscript 2 represents the layer number and N_2 is the dimension of the output vector. The output of neurons in the hidden layer and output layer is computed as

$$x_i^l = f\left[\sum_{j=1}^{N_{l-1}} (w_{ij}^l x_j^{l-1})\right], \quad i = 1, 2, \ldots, N_l, \ \ l = 1, 2 \tag{9}$$

where $f : C \to C$ is a complex-valued function and w_{ij}^l is the complex-valued weight value between ith neuron in $l-1$th layer and jth neuron in lth layer.[4]

Let $T \in C^{N_2}$ be the required target complex-valued function. In literature, squared error is often used in minimization. For complex-valued signal, the squared error represents only the magnitude of error. The minimization process does not include the phase error directly. So, a criterion that includes both magnitude and phase errors becomes essential.

For example, one can suggest the following criterion Eq. (10) as a candidate for minimization:

$$J = J_{Me} + \phi_e \tag{10}$$

However, with this criterion, both the components in Eq. (10) are real and hence the error gradient will also be real. Since the original signals are complex-valued for gradient learning we need complex-valued gradients to preserve the phase information. The above criterion (Eq. (10)) will not meet this requirement.

Therefore, we propose a new performance index, which minimizes both the magnitude and phase errors directly and includes a fully complex-valued error gradient. The new performance index is defined as

$$J_{new} = \frac{1}{2}\left[\log\left[\frac{T}{Y}\right]\overline{\log\left[\frac{T}{Y}\right]}\right], \quad \mathbf{Y} = [x_1^2, x_2^2, \ldots, x_{N_2}^2]^T \tag{11}$$

where $\overline{\log[T/Y]}$ is the complex-conjugate of $\log[T/Y]$. This is different from Eq. (3) in that Eq. (3) tends to minimize only the magnitude, while Eq. (11) minimizes both the error in magnitude and phase. The above equation can be written as

$$J_{new} = \frac{1}{2}\left[\log\left[\frac{\|T\|}{\|Y\|}\right]^2 + [\arg(T) - \arg(Y)]^2\right] \tag{12}$$

Constants k_1 and k_2 can be included as weighting factors for the magnitude error and phase error, respectively, to accelerate the convergence of the magnitude/phase component of the output. With the introduction of constants k_1 and k_2, Eq. (12)

[4] This can be easily extended with multiple hidden layers for multi-layer networks.

becomes

$$J_{new} = \frac{1}{2}\left[k_1 \log\left[\frac{\|T\|}{\|Y\|}\right]^2 + k_2[\arg(T) - \arg(Y)]^2\right] \tag{13}$$

From the above equation, one can see that the performance index (J_{new}) approaches zero when Y approaches T. Based on the new performance index J_{new}, we can derive weight update rule and is given by

$$w^l_{ij} = w^l_{ij} + \eta\delta^l_i \overline{X}^{l-1}_j \tag{14}$$

where

$$\delta^2_i = k_1 \log\left[abs\left[\frac{T_i}{Y_i}\right]\right]\delta^1_i = \overline{X}^1_j \sum_{j=1}^{N^2} \overline{w}^2_{ij}\delta^2_j$$

Here, the output neuron uses an exponential activation function. The targets are all scaled into first and fourth quadrants. k_1 and k_2 are weights applied to the log magnitude error and phase error, respectively.

In the next section, the results on synthetic complex-valued function approximation problem are presented. The performance of the new activation function with the ETFs is compared first. Finally, the performance of the activation functions with the new minimization criterion and the old minimization criterion are compared.

4. Performance study of the proposed algorithm

4.1. Synthetic complex-valued function approximation problem I

4.1.1. Sensitivity study of fully complex-valued network

The comparison of the failure analysis results for the synthetic complex-valued function approximation problem I is presented in Table 5. From Table 5, it can be seen that the exponential activation function performance was good even at a higher learning rate of 0.1, while the ETFs chosen for study failed at higher learning rates.

4.1.2. Performance study of activation functions

Comparison of the performance of the network with different activation functions, for the synthetic complex-valued function approximation problem I defined in Eq. (7), is tabulated in Tables 6 and 7. Training was done with a sample set of randomly generated 3000 samples, presented over 5000 epochs. The performances of the network with different activation functions were studied for different learning rates viz., $1\times10^{-3}, 3\times10^{-3}, 5\times10^{-3}, 8\times10^{-3}$ and 1×10^{-2} and different weight initialization radius, viz., $1\times10^{-2}, 1\times10^{-3}, 1\times10^{-6}$, with the old minimization criterion. The learning rate and the initial weight radius at which the activation function showed minimal mean square error is chosen as the best learning rate and best radius of initialization. The asinh($\cdot$) activation function showed best performance for a learning rate of 0.005 and initialization of 10^{-3}, while the best performance of atan($\cdot$) activation was seen at a learning rate of 0.008, with the weights initialized in a ball of radius 10^{-6}. The exp($\cdot$) activation approximated the function best (with least magnitude error and phase error) at $\eta = 0.003$ with the weights initialized in a small ball of radius 10^{-6}. From the results, it can be seen that the fully complex-valued network developed using proposed exponential activation function has better approximation and generalization performance than the ETFs, even at a lower learning rate. Even though the ETFs are constructed using the basic exp($\cdot$) function, the presence of singular point inside the complex domain of interest affects the performance considerably.

Table 5
Failure analysis comparison of ETFs and exponential activation function.

ETFs	Region of failure		Percentage of failures
	Learning rate	Weight initialization	
asinh($\cdot$)	0.1	ALL	100
atan($\cdot$)	0.1	ALL	100
exp($\cdot$)	0.1	II, III	6

Table 6
Performance comparison of the ETFs and exponential network for J_{Me} with the new criterion.

Activation type	k_1	k_2	Training J_{Me}		Testing J_{Me}	
			Old crit.	New crit.	Old crit.	New crit.
asinh($\cdot$)	12	2	0.004	0.015	0.006	0.054
atan($\cdot$)	10	2	0.002	0.02045	0.067	0.0246
exp($\cdot$)	25	2	0.002	0.002	0.002	0.0032

Table 7
Performance comparison of the ETFs and exponential network for ϕ_e with the new criterion.

Activation type	k_1	k_2	Training ϕ_e (rad.)		Testing ϕ_e (rad.)	
			Old crit.	New crit.	Old crit.	New crit.
asinh($\cdot$)	12	2	0.007	0.003	0.0106	0.006
atan($\cdot$)	10	2	0.002	0.00113	0.018	0.00106
exp($\cdot$)	25	2	0.004	8×10^{-4}	0.002	0.0015

4.1.3. Performance study of minimization functions

Next, we study the performance of the network using the logarithmic criterion for the synthetic complex-valued function approximation problem I, in the complex-valued back propagation algorithm. The behavior of the fully complex-valued network with the three activation functions, viz., asinh($\cdot$), atan($\cdot$), and exponential activation functions with new logarithmic criterion is presented in Tables 6 and 7.

Results shown in Tables 6 and 7 were obtained using $\eta = 5\times10^{-3}$ and $w_r = 1\times10^{-6}$ for the new criterion. As seen, the phase approximation performance of the CBP has improved with the new logarithmic criterion. But the magnitude approximation performance is seen to be similar to the old minimization criterion. This can be further improved with proper choice of k_1 and k_2. However, considering the activation functions asinh($\cdot$) and atan($\cdot$), increasing k_1 and k_2 further resulted in the network hitting the singular point, thereby, making the network unstable.

4.2. Synthetic complex-valued function approximation problem II

4.2.1. Sensitivity study of fully complex-valued network

The sensitivity study explained in Section 2.3 was performed on the synthetic complex-valued function approximation problem II. It was observed that all the activations, asinh($\cdot$), atan($\cdot$), and exp($\cdot$) failed at a learning rate of 0.1. This could be due to the presence of a $1/z$ term in the function that is being approximated. However, at lower learning rates, i.e., at $\eta = 0.01$ and 0.001, the convergence was good. In Fig. 5, the phase error convergence of

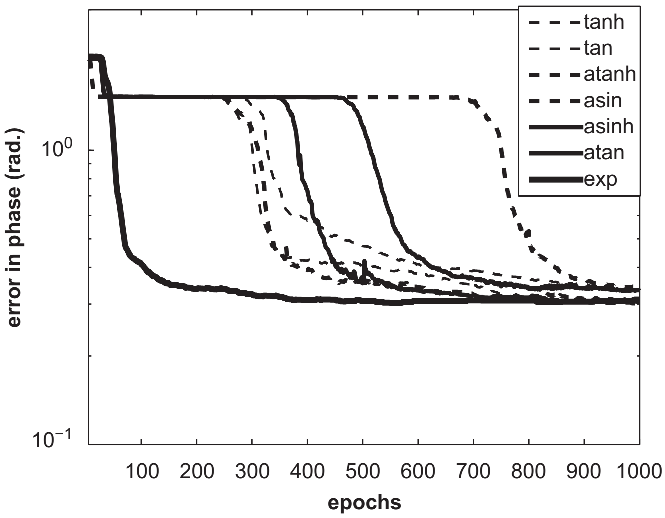

Fig. 5. Phase error (ϕ_e) in rads for different ETFs at a learning rate of 0.001 (synthetic function approximation problem II).

Table 8
Performance comparison of the ETFs and exponential network for J_{Me} with the new criterion.

Activation type	k_1	k_2	Training J_{Me}		Testing J_{Me}	
			Old crit.	New crit.	Old crit.	New crit.
asinh(·)	1	1	0.031	0.0218	0.0573	0.04304
atan(·)	1	1	0.0292	0.02213	0.0566	0.0415
exp(·)	5	10	0.0279	0.02016	0.0539	0.0377

Table 9
Performance comparison of the ETFs and exponential network for ϕ_e with the new criterion.

Activation type	k_1	k_2	Training ϕ_e (rad.)		Testing ϕ_e (rad.)	
			Old crit.	New crit.	Old crit.	New crit.
asinh(·)	1	1	0.3047	0.00949	0.2779	0.02173
atan(·)	1	1	0.2882	0.00999	0.2778	0.02196
exp(·)	5	10	0.2785	0.00803	0.2596	0.0199

the exponential activation function and the ETFs at a learning rate of 0.001 is given. It can be readily seen from the figure that exponential activation function shows smooth error convergence characteristics than the ETFs.

4.2.2. *Performance study of activation functions*

In Tables 8 and 9, the performance of the network with different activation functions, for the synthetic complex-valued function approximation II defined in Eq. (8), is compared. Training was done with a sample set of randomly generated 3000 samples, presented over 1000 epochs. The performance of the network with different activation function was studied for different learning rates viz., 1×10^{-3}, 3×10^{-3}, 5×10^{-3} and 8×10^{-3} and for an initial weight radius of 1×10^{-6}, with the old minimization criterion. The learning rate and the initial weight radius at which the activation function showed minimal mean square error is chosen as the best learning rate and initializations. The asinh(·), atan(·) and exp(·) activation functions showed best performance at a learning rate (η) of 0.001, 0.001 and 0.005, respectively. From the results, it can be seen that the fully complex-valued network developed using proposed exponential activation function has better generalization performance, as well as better magnitude/phase approximations, than the ETFs. Again, though the ETFs are constructed using the basic exp(·) function, the presence of singular point inside the complex domain of interest affects the performance considerably. Fig. 5 gives the plot of phase error comparison between ETFs and exp(·) activation function for the complex-valued synthetic function approximation problem II, using the old minimization criterion.

4.2.3. *Performance study of minimization functions*

Next, we study the performance of the network using the logarithmic criterion for the synthetic complex-valued function approximation problem II, in the complex-valued back propagation algorithm. The behavior of the fully complex-valued network with the three activation functions, viz., asinh(·), atan(·), exp(·) with new logarithmic criterion is presented in Tables 8 and 9.

Tables 5–9 clearly enumerate the advantages of the proposed algorithm over the existing complex MLP network and training algorithm, on synthetic complex-valued function approximation problems. Next, the proposed algorithm was applied for the benchmark problem-complex XOR problem and in non-minimal phase equalization for 4-QAM channel equalization problem.

4.3. *Complex XOR problem*

The complex-valued XOR is a commonly used problem in literature to study the convergence of the network [15,16] The XOR problem in complex domain has been defined as:

- The real part of the output is unity if the first input is equal to the second input else it is zero.
- The imaginary part of the output is unity if the second input is equal to unity else it is zero.

From the definition of the CXOR gate, it can be seen that the target could be one of the following set: $0, 0+i, 1, 1+i$. A 2–5–1 network (2 input neurons, 5 hidden neurons and 1 output neuron) was used for the study. Initially, the sensitivity of different activation functions to various learning rates and weight initializations were analyzed. A training set with 3000 randomly chosen samples was generated and used for this study.

4.3.1. *Sensitivity study*

The comparison of the failure analysis results for the CXOR problem is presented in Table 10. The sensitivity analysis study described in Section 2.3 is performed, with the activation functions chosen among the ETFs, compared against the proposed exponential activation function. It is observable from Table 10 that the exponential activation function does not fail even for a learning rate of $\eta=0.1$. However, the ETFs chosen, viz., asinh(·) and atan(·) activation functions failed at learning rates, as high as 0.1. However, all the three activations did converge at lower learning rates i.e., at 10^{-2} and 10^{-3}.

Table 10
Failure analysis comparison of ETFs and exponential activation function.

ETFs	Region of failure		Percentage of failures
	Learning rate	Weight initialization	
asinh(·)	0.1	ALL	100
atan(·)	0.1	ALL	100
exp(·)	NA	NA	NIL

4.3.2. Performance comparison of activation functions

Tables 11 and 12 show the best performance training/testing results, for the CXOR problem. The training set consists of 3000 randomly generated samples. The best learning rate (η) is chosen as the learning rate at which the network converged faster. Convergence, in this context, is defined as achieving a training mean square error of 10^{-3}. Tables 11 and 12 give the learning statistics over 50 successive runs. The weights were initialized to 1×10^{-6} for all the functions. The performance of the network with different activation function was studied for different learning rates viz., $1 \times 10^{-3}, 3 \times 10^{-3}, 5 \times 10^{-3}, 8 \times 10^{-3}$ and 1×10^{-2}. The learning rate at which the activation function showed minimal mean square error is chosen as the best learning rate for that activation function. When the network was trained with the old minimization criterion, the atan($\cdot$) and asinh($\cdot$) activation functions showed best performance at a learning of 0.001, while the performance was the best for the exp($\cdot$) activation function at a learning rate of 0.008. It can be observed from Tables 11 and 12 that the proposed exponential activation function has better training performance with the old minimization criterion, in terms of number of epochs required for convergence, than the ETFs chosen.

4.3.3. Performance comparison of minimization criteria

Results in Tables 11 and 12 were obtained using $w_r = 1 \times 10^{-6}$ for both the criteria, with $k_1 = 1$; $k_2 = 1.5$. It is clearly seen that, using the new criterion, the network converges faster than with the old criterion. In other words, the network with the new criterion required less number of epochs than the network with the old criterion to achieve a mean square error of 1×10^{-3}. Among the activations chosen, exponential activation function outperforms the ETFs, with the new criterion also.

4.4. Non-minimum phase equalization

A complex non-minimum phase channel model introduced by Cha and Kassam [17,18] is used to evaluate the performance of CBP equalizer. The equalization model is of order three with non-linear distortion for 4-QAM signaling

$$z_n = o_n + 0.1o_n^2 + 0.05o_n^3 + v_n, v_n\aleph(0, 0.01) \tag{15}$$

$$o_n = w_1 s_n + w_2 s_{n-1} + w_3 s_{n-2} \tag{16}$$

where $w_i = 0.34 - 0.27i, w_2 = 0.87 + 0.43i, w_3 = 0.34 - 0.21i, \aleph(0, 0.01)$ means the white Gaussian noise (of the non-minimum phase channel) with mean 0 and variance 0.01. The equalizer input dimension is chosen as 3. As usually done in equalization problems, a decision delay τ is introduced in the equalizer so that at time n, the equalizer estimates the input symbol $s_{n-\tau}$ rather than s_n and we set $\tau = 1$. Four-QAM symbol sequence s_n is passed through the channel and the real and imaginary parts of the symbol are valued from the set. A total of 3000 samples for training and 1000 samples for testing were randomly generated, and the performance of the CBP network for QAM equalization was studied with this randomly generated sequence. The architecture of the network used in the study is: 3–15–1; 3 input neurons, 15 hidden neurons and 1 output neuron.

4.4.1. Sensitivity study

The sensitivity analysis study, explained in Section 2.3, was done for the QAM equalization problem, for different ETFs and exponential activation function. The study was done over different weight initializations and learning rates mentioned earlier. The comparison of the failure analysis results for the 4-QAM equalization problem is presented in Table 13. It is observable from Table 13 that the exponential activation function does not fail even at higher learning rates, as high as $\eta = 0.1$. However, the ETFs chosen, viz., asinh($\cdot$) and atan($\cdot$) activation functions failed at higher learning rates.

4.4.2. Performance comparison of activation Functions

Tables 14 and 15 give the comparison of activation functions for the non-minimum phase equalization problem. A training

Table 11
Training statistics comparison of the ETFs and exponential network with the new criterion for $J_{Me} = 1 \times 10^{-03}$.

Activation type	Learning rate (η)	Average number of epochs for convergence	
		Old crit.	New crit.
asinh($\cdot$)	1×10^{-03}	1864	790
atan($\cdot$)	1×10^{-03}	4598	670
exp($\cdot$)	8×10^{-03}	831	449

Table 12
Performance comparison of the ETFs and exponential network for ϕ_e with the new criterion.

Activation type	Learning rate (η)	Training ϕ_e (rad.) at the end of training	
		Old crit.	New crit.
asinh($\cdot$)	1×10^{-03}	0.074	1×10^{-4}
atan($\cdot$)	1×10^{-03}	0.04052	3.1×10^{-4}
exp($\cdot$)	8×10^{-03}	0.04	9.5×10^{-4}

Table 13
Failure analysis comparison of ETFs and exponential activation function.

ETFs	Region of failure		Percentage of failures
	Learning rate	Weight initialization	
asinh($\cdot$)	0.1	ALL	100
	0.01	I Quadrant	6
atan($\cdot$)	0.1	ALL	100
exp($\cdot$)	NA	NA	NIL

Table 14
Performance comparison of the ETFs and exponential network for J_{Me} with the new criterion.

Activation type	Training J_{Me}		Testing J_{Me}	
	Old crit.	New crit.	Old crit.	New crit.
asinh($\cdot$)	0.2153	0.1635	0.7198	0.8968
atan($\cdot$)	0.4187	0.1179	0.6126	0.395
exp($\cdot$)	0.2870	0.228	0.4809	0.2278

Table 15
Performance comparison of the ETFs and exponential network for ϕ_e with the new criterion.

Activation type	Training ϕ_e (rad.)		Testing ϕ_e (rad.)	
	Old crit.	New crit.	Old crit.	New crit.
asinh($\cdot$)	0.1129	0.1028	0.5429	0.2457
atan($\cdot$)	0.2534	0.1019	0.5781	0.234
exp($\cdot$)	0.283	0.1986	0.3079	0.1976

set of 5000 randomly generated samples at 20 dB SNR was used to train the network over 1000 epochs. It was observed that the exponential activation function performed better in classifying the symbols than the asinh(·) and atan(·) activation functions, when the network was trained with the old minimization criterion. This is clearly evident from Tables 14 and 15.

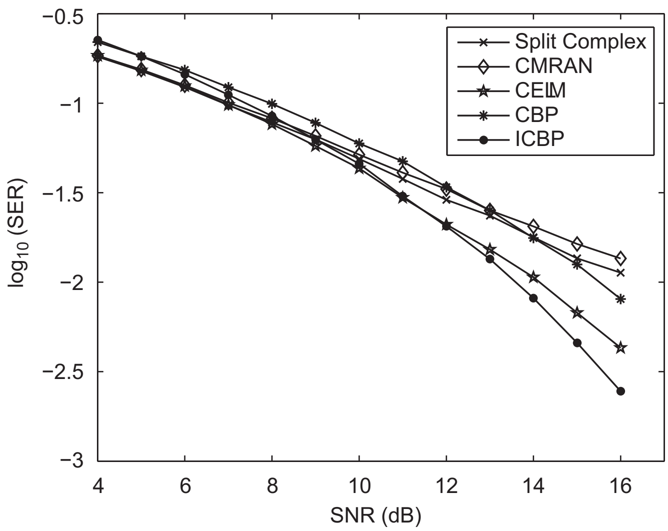

Fig. 6. SER performance of the equalizers.

4.4.3. Performance comparison of minimization criteria

Tables 14 and 15 show the performance comparison of the minimization criterion in the non-minimum phase equalization. It is clearly seen that the network with the new improved minimization criterion performs better than the existing one. The phase error has reduced considerably in the network with the new minimization criterion.

Fig. 6 gives the plot of SNR vs BER (bit error rate) for various SNR. The network was trained with data at SNR of 20 dB. Test sets with 1×10^5 samples at various SNRs were used for the error probability evaluation, for both the networks. It is clearly visible that the network with the proposed activation function and minimization criterion has improved the symbol classification performance of the network. Fig. 7 gives the symbol classification performance of the equalizers using the different neural network algorithm. It can be inferred from the figure that the proposed algorithm performs better symbol classification compared to the other algorithms considered.

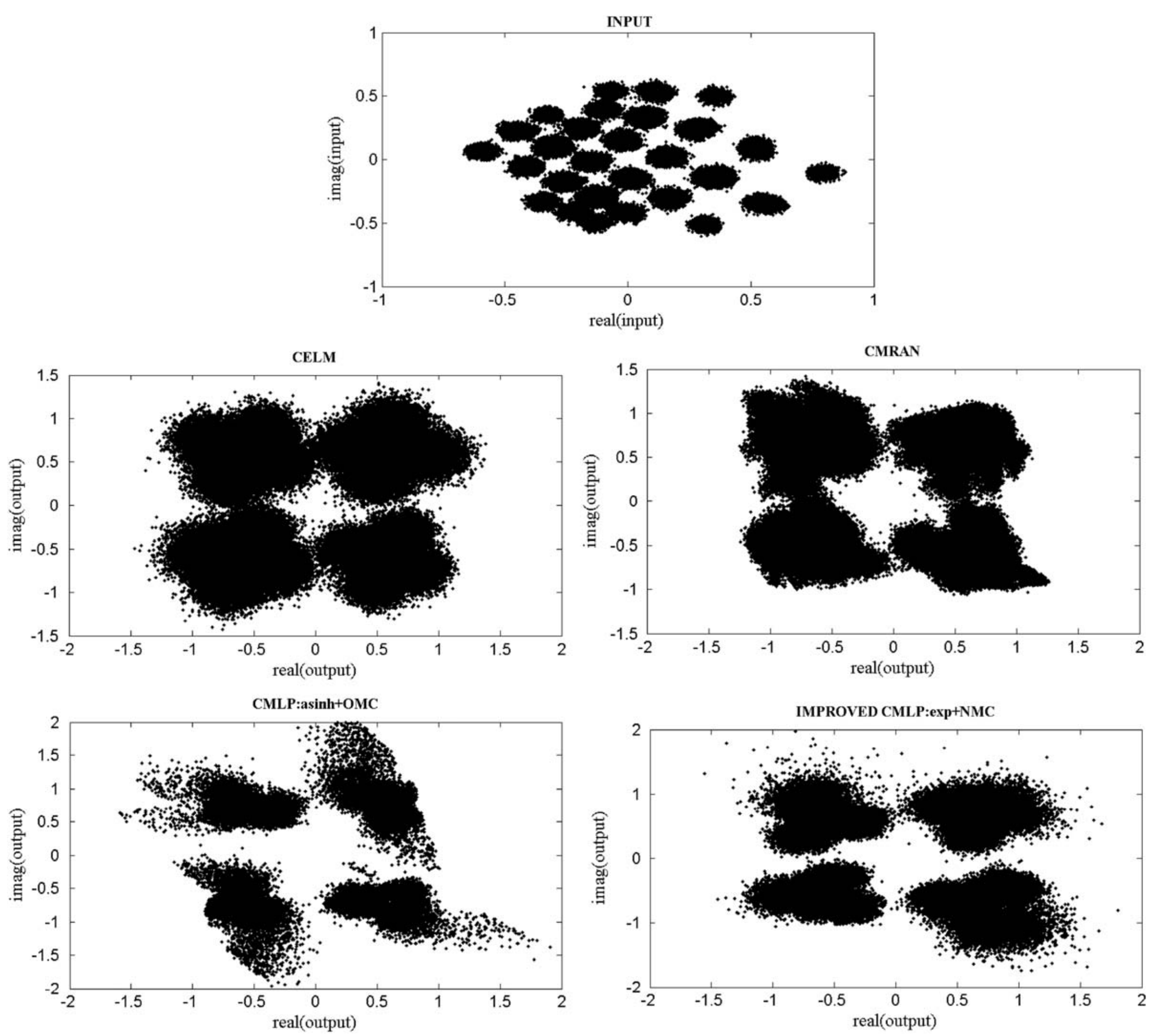

Fig. 7. Eye diagram of symbol classification performance of different equalizers.

5. Conclusions

The convergence of the fully complex-valued back propagation learning algorithm in a fully complex-valued feed-forward network depends on the choice of activation function, learning rate and initial weight radius. The issues in the complex-valued neural networks have been enumerated with two synthetic complex-valued function approximation problems. To overcome these issues, a new scheme with exponential activation function and logarithmic minimization criterion has been proposed. From the various studies conducted on the above problems, it is seen that the exponential activation function has superior convergence and generalization characteristics compared to the ETFs chosen for study. Further, their performance is less affected by the weight initialization and the learning rate parameter. It was also inferred that the split complex network does not perform better than the fully complex-valued network, and among the ETFs, the network with asinh$(\cdot)$ and atan$(\cdot)$ as an activation function performed better than the other ETFs. Besides these findings, it is also observed that the proposed logarithmic minimization criterion improves the approximation ability of the complex-valued back propagation network, for both discrete-valued and continuous-valued targets. Performance comparisons of the various algorithms using the two synthetic complex-valued function approximation problems, the complex-valued XOR problem and non-minimum phase equalization problem showed that the network with the proposed learning algorithm performs better than the other algorithms.

References

[1] T. Kim, T. Adali, Fully-complex multilayer perceptron network for nonlinear signal processing, Journal of VLSI Signal Processing 32 (1–2) (2002) 29–43.
[2] H. Leung, S. Haykin, Complex backpropagation algorithm, IEEE Transactions on Signal Processing 39 (9) (1991) 2101–2104.
[3] D. Jianping, N. Sundararajan, P. Saratchandran, Complex-valued minimal resource allocation network for nonlinear signal processing, International Journal of Neural Systems 10 (2) (2000) 95–106.
[4] N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, A comparison between real and complex-valued neural networks in communication applications, in: Proceedings of the International Neural Networks Conference, Helsinki, Finland, June 1991, pp. 1177–1180.
[5] N. Benvenuto, M. Marchesi, F. Piazza, A. Uncini, Nonlinear satellite radio links equalized using blind neural networks, in: Proceedings of the International Conference ASSP, Toronto, Canada, April 1991, pp. 1521–1524.
[6] S.S. Yang, C.L. Ho, S. Siu, Sensitivity analysis of the split-complex valued multilayer perceptron due to the errors of the i.i.d. inputs and weights, IEEE Transactions on Neural Networks 18 (5) (2007) 1280–1293.
[7] D.V. Loss, M.C.F. DeCastro, P.R.G. Franco, F.C.C. DeCastro, Phase transmittance RBF neural networks, Electronics Letters 43 (16) (2007) 882–884.
[8] G.M. Georgios, C. Koutsougeras, Complex domain backpropagation, IEEE Transactions on Circuit and Systems II 39 (5) (1992) 330–333.
[9] C. You, D. Hong, Nonlinear blind equalization schemes using complex valued multilayer feedforward neural networks, IEEE Transactions on Neural Networks 9 (6) (1998) 1442–1455.
[10] S.L. Goh, D.P. Mandic, An augmented extended Kalman filter algorithm for complex-valued recurrent neural networks, Neural Computation 19 (4) (2007) 1039–1055.
[11] N. Benvenuto, F. Piazza, On the complex backpropagation algorithm, IEEE Transactions on Signal Processing 40 (4) (1992) 967–969.
[12] J.P. Deng, N. Sundararajan, P. Saratchandran, Communication channel equalization using complex-valued minimal radial basis function neural networks, IEEE Transactions on Neural Networks 13 (3) (2002) 687–696.
[13] T. Kim, T. Adali, Approximation by fully complex multilayer perceptrons, Neural Computation 15 (7) (2003) 1641–1666.
[14] S. Suresh, S.N. Omkar, V. Mani, T.N. Guru Prakash, Lift coefficient prediction at high angle of attack using recurrent neural network, Aerospace Science and Technology 7 (8) (2003) 595–602.
[15] P.K.K. Anita, S. Gangal, D.S. Chauhan, Performance evaluation of complex-valued neural networks using various error functions, in: Proceedings of the World Academy of Science Engineering and Technology, vol. 23, August 2007, pp. 27–32.
[16] T. Nitta, An extension of the back-propagation algorithm to complex numbers, Neural Networks 10 (8) (1997) 1391–1415.
[17] I. Cha, S.A. Kassam, Channel equalization using adaptive complex radial basis function networks, IEEE Journal on Selected Areas in Communications 13 (1) (1995) 122–131.
[18] M.-B. Li, G.-B. Huang, P. Saratchandran, N. Sundararajan, Fully complex extreme learning machines, Neurocomputing 68 (2005) 306–314.

R. Savitha received the B.E. degree in electrical and electronics engineering from Manonmaniam Sundaranar University in 2000, and M.E. degree in control and instrumentation engineering from College of Engineering Guindy, Anna University, India. She was a lecturer in MSEC, India, between 2002 and 2005. From 2005 to 2006, she worked in the Machine Vision Division of Premier Evolvics, India. From 2006, she is a student at Nanyang Technological University, Singapore.

Sundaram Suresh received the B.E. degree in electrical and electronics engineering from Bharathiyar University in 1999, and M.E. (2001) and Ph.D. (2005) degrees in aerospace engineering from Indian Institute of Science, India. He was post-doctoral researcher in School of Electrical Engineering, Nanyang Technological University from 2005 to 2007. From 2007 to 2008, he was in INRIA-Sophia Antipolis, France, as ERCIM research fellow. He was in Korea University for a short period as a visiting faculty in Industrial Engineering. From 2009 onwards, he has been with Indian Institute of Technology—Delhi, where he is presently an Assistant Professor in Department of Electrical Engineering. His research interest includes flight control, unmanned aerial vehicle design, machine learning, optimization and computer vision.

Narasimhan Sundararajan received the B.E. in electrical engineering with First Class Honors from the University of Madras in 1966, M.Tech from the Indian Institute of Technology, Madras in 1968 and Ph.D. in electrical engineering from the University of Illinois, Urbana-Champaign in 1971.

From 1971, he worked in various capacities in the Indian Space Research Organization (ISRO) and from February 1991 is working as a professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He was also a NRC Research Associate at NASA—Ames in 1974 and a senior research associate at NASA Langley in 1981–1986. His research interests are in the areas of aerospace control, neural networks and has published more than 200 papers and four books. He has served as an associate editor for a number of journals and has been a program committee member in a number of international conferences.

Dr. Sundararajan is a Fellow of IEEE, an Associate Fellow of AIAA and also a Fellow of the Institution of Engineers (IES), Singapore.

P. Saratchandran obtained his Ph.D. in the area of control engineering from Oxford University, Oxford, UK. He is an associate professor with the Nanyang Technological University, Singapore. He has more than 150 publications in refereed journals and conferences and has authored four books titled "Fully tuned RBF networks for flight control" (Kluwer Academic 2002), "Radial basis function neural networks with sequential Learning" (World Scientific Publishing Co., 1999) "Parallel architectures for artificial neural networks" (IEEE Computer Society Press, 1998) and "Parallel implementations of backpropagation neural networks" (World Scientific Publishing Co., 1996). He is an editor for the journal "Neural Parallel and Scientific Computations". His interests are in neural networks, bioinformatics and adaptive control. He is listed in the Macquarie Who's Who in the world and in the Leaders in the World of the International Biographics Centre, Cambridge, UK.