Concurrent Blind Channel Equalization with Phase Transmittance RBF Neural Networks

Diego Vier Loss[1], Maria C. F. De Castro[1], Paulo Roberto Girardello Franco[1] & Fernando C. C. de Castro [1]

[1]Department of Electrical Engineering
Wireless Technologies Research Center - CPTW
Pontifical Catholic University of Rio Grande do Sul
Av. Ipiranga 6681
Phone: +55 (51) 33203500 sn: 4403
Zip 90619-000 – Porto Alegre - RS - BRAZIL
fabio.dagostini@pucrs.br—sirlesio.carboni@pucrs.br—cristinacastro@pucrs.br—fcastro@pucrs.br

Abstract

This paper presents a new complex valued radial basis function (RBF) neural network (NN) with phase transmittance between the input nodes and output, which makes it suitable for channel equalization on quadrature digital modulation systems. The new Phase Transmittance RBFNN (PTRBFNN) differs from the classical complex valued RBFNN in that it does not strictly rely on the Euclidean distance between the input vector and the center vectors, thus enabling the transference of phase information from input to output. In the context of blind channel equalization, results have shown that the PTRBFNN not only solves the phase uncertainty of the classical complex valued RBFNN but also presents a faster convergence rate.comes the abstract of the paper.

Keywords: Neural Network, Equalizer, Phase, Concurrent, Blind

1. Introduction

Several complex valued RBF neural network approaches has been proposed in the context of channel equalization [1]-[5]. These classical approaches share a common architecture, shown in Figure 1, and a common principle – the output of the kth Gaussian center φ_k is based on the Euclidean norm between the complex valued input vector $\underline{u}$ and the kth center vector $\underline{\Psi}_k$, as given by (1)

$$\varphi_k(\underline{u}) = \exp\left(\frac{-1}{\sigma_k^2}\left\|\underline{u} - \underline{\Psi}_k\right\|^2\right) \tag{1}$$

where $\underline{u} = \begin{bmatrix} u_0 & u_1 & \cdots & u_{L-1} \end{bmatrix}^T$ is the vector which represents the communication channel regressor and whose L complex valued components u_ℓ are such that $\mathbf{Z}\{u_\ell\} = \mathbf{Z}\{u_{\ell-1}\}z^{-1}$,i.e., the channel regressor behaves as a FIFO [6]. $\mathbf{Z}\{\cdot\}$ is the operator which returns the Z-Transform of its argument, with $\ell = 1,2,\cdots,L-1$. $\underline{\Psi}_k$ is the kth center vector $\underline{\psi} = [\psi_0 \quad \psi_1 \quad \cdots \quad \psi_{L-1}]^T$, σ_k^2 is the variance of the kth Gaussian center, $k = 0,1,\cdots,K-1$, being K the adopted number of centers. $\|\cdot\|$ is the operator which returns the Euclidean norm of its argument.

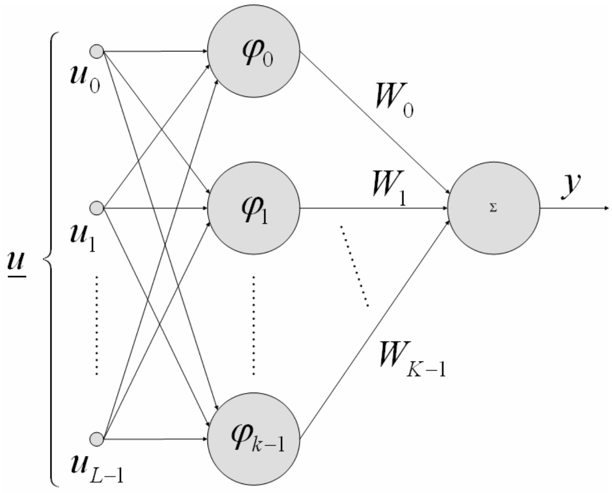


Figure 1: RBF neural network architecture.

From Figure 1 the output is

$$y = \underline{W}^T \cdot \underline{\varphi} = \sum_{k=0}^{K-1} W_k \varphi_k \tag{2}$$

where W_k is the kth complex valued component of the output synapses vector $\underline{W} = [W_0 \quad W_1 \quad \cdots \quad W_{K-1}]^T$ and φ_k is the kth real valued component of the hidden layer output vector $\underline{\varphi} = [\varphi_0 \quad \varphi_1 \quad \cdots \quad \varphi_{K-1}]^T$.

In digital communications using quadrature modulation, the information is conveyed across the channel by means of a discrete sequence $s(n)$ of complex numbers $I + jQ$, each complex number being denoted as an IQ symbol of the sequence $s(n)$ [6]. This sequence is generated by the digital transmitter at the specific symbol rate S_R of the system, which defines the time interval T between IQ symbols $s(n)$ and $s(n+1)$ as $T = 1/S_R$, being $n = 0,1,\cdots$.

The digital modulator at the transmitter maps each incoming binary word of fixed length into the respective IQ symbol. Since an IQ symbol is a complex phasor, the phase of each IQ symbol plays a crucial role in the information flow.

Subsequently, the transmitted IQ symbol sequence $s(n)$ is convolved with the discrete complex valued impulse response of the communication channel, impulse response which is dictated by the channel multipath scenario [6]. This multipath scenario imposes phase and amplitude distortions to $s(n)$. In addition, besides the multipath distortion, the channel adds noise to $s(n)$, so that the received sequence $r(n)$ barely resembles the original transmitted sequence $s(n)$.

At the digital receiver, the received IQ sequence $r(n)$ is stored in the FIFO buffer of the channel regressor $\underline{u}$, so that an equalizer can perform the so called channel deconvolution process [9]. Once $r(n)$ is deconvolved by the equalizer, the digital demodulator de-maps each deconvolved IQ symbol into the respective binary word of fixed length and of duration T [6].

Notice from (1) and (2) that the phase of the input $\underline{u}$ is not explicitly transmitted to the output y, since it is discarded by the Euclidean norm operator in (1). Therefore, the resulting hidden layer output vector $\underline{\varphi}$ restricts the RBFNN forward transmittance to be real valued, which inherently imposes some restrictions when the classical RBFNN is applied to blind channel equalization, since the forward transmittance does not explicitly convey any phase information.

In order to avoid the explicit phase invariance on the transmittance function of the classical RBFNN network we propose a new complex valued RBFNN – the Phase Transmittance RBFNN (PTRBFNN). Specifically, we replace the Euclidean norm based hidden-layer radial basis function by the complex valued basis function

$$\begin{aligned} \varphi_k(u) = {} & \exp\left(\frac{-1}{\mathrm{Re}\{\sigma_k^2\}} \left\| \mathrm{Re}\{\underline{u}\} - \mathrm{Re}\{\underline{\Psi}_k\} \right\|^2 \right) \\ & + j \cdot \exp\left(\frac{-1}{\mathrm{Im}\{\sigma_k^2\}} \left\| \mathrm{Im}\{\underline{u}\} - \mathrm{Im}\{\underline{\Psi}_k\} \right\|^2 \right) \end{aligned} \tag{3}$$

where $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ are the operators which respectively return the real and imaginary part of the respective arguments.

Notice that this approach turns phase-sensitive the hidden neurons input-output relationship while preserving the locality of the basis function.

Notice also that $\sigma_k^2 = \mathrm{Re}\{\sigma_k^2\} + j\,\mathrm{Im}\{\sigma_k^2\}$ does not strictly imply a complex valued physical interpretation for the variance of the kth complex valued center φ_k. $\mathrm{Re}\{\sigma_k^2\}$ is just a quadratic measure for the reach radius of the basis function $\mathrm{Re}\{\varphi_k\}$. Likewise, same interpretation apply to $\mathrm{Im}\{\sigma_k^2\}$ with respect to $\mathrm{Im}\{\varphi_k\}$.

2. THE PTRBFNN LEARNING ALGORITHM

Let $\underline{u}(n)$ be the channel regressor at the instant n, and let $d(n)$ be the desired complex value at the PTRBFNN output $y(n)$. In the context of channel equalization, $d(n)$ is the original transmitted IQ symbol training sequence whose values are stored in a look up table at the receiver as reference data for the learning process. Actually, as we shall see further, the reference IQ symbol training sequence for the RBFNN learning process is based on a blind concurrent algorithm. For now, lets just assume $d(n)$ is known at the receiver.

In this work we adopt a gradient based learning process. It consists of , for each received IQ symbol at the instant n, to adjust by means of the Delta Rule [8] the PTRBFNN free parameters $\underline{\Psi}_k$, σ_k^2 and $\underline{W}$, $k = 0,1,\cdots,K-1$, so that after a sufficient number of received IQ symbols the distance between $y(n)$ and $d(n)$ is as close as possible to zero. A quadratic measure of the distance between $y(n)$ and $d(n)$ is:

$$J(n) = \frac{1}{2}\left|d(n) - y(n)\right|^2 \tag{4}$$

Specifically, the learning process consists of to minimize the cost function given by (4) by adjusting the PTRBFNN free parameters $\underline{W}(n)$, $\underline{\Psi}_k(n)$ and $\sigma_k^2(n)$ at each instant n [8]. For any PTRBFNN free parameter $\wp$, the Delta Rule adjusts $\wp$ according to:

$$\wp(n+1) = \wp(n) - \eta\nabla J(n) \tag{5}$$

where η is the learning rate or adaptation step [7][8] and ∇J is the gradient of the cost function J with respect to the parameter $\wp$:

$$\nabla J(n) = \frac{\partial J(n)}{\partial \wp(n)} \tag{6}$$

Let W_p be the p^{th} component of vector $\underline{W} = \begin{bmatrix} W_0 & W_1 & \cdots & W_{K-1}\end{bmatrix}^T$. W_p is one of the PTRBFNN free parameters which is adjusted by (5), so that

$$W_p(n+1) = W_p(n) - \eta_W \nabla J_p^W(n) \tag{7}$$

where η_W is the adaptation step and

$$\nabla J_p^W = \frac{\partial J}{\partial W_p} = \frac{\partial J}{\partial\,\mathrm{Re}\{W_p\}} + j\frac{\partial J}{\partial\,\mathrm{Im}\{W_p\}} \tag{8}$$

For convenience, in the equations that follow we drop n from the notation, unless its presence is strictly necessary. From (2) and (4) the real and imaginary parts of (8) are obtained as

$$\frac{\partial J}{\partial\,\mathrm{Re}\{W_p\}} = \frac{1}{2}\frac{\partial|d-y|^2}{\partial\,\mathrm{Re}\{W_p\}} = \frac{1}{2}\frac{\partial\left|d - \sum_{k=0}^{K-1} W_k\varphi_k\right|^2}{\partial\,\mathrm{Re}\{W_p\}} = -\mathrm{Re}\{e\}\mathrm{Re}\{\varphi_p\} - \mathrm{Im}\{e\}\mathrm{Im}\{\varphi_p\} \tag{9}$$

$$\frac{\partial J}{\partial\,\mathrm{Im}\{W_p\}} = \frac{1}{2}\frac{\partial|d-y|^2}{\partial\,\mathrm{Im}\{W_p\}} = \frac{1}{2}\frac{\partial\left|d - \sum_{k=0}^{K-1} W_k\varphi_k\right|^2}{\partial\,\mathrm{Im}\{W_p\}} = -\mathrm{Im}\{e\}\mathrm{Re}\{\varphi_p\} + \mathrm{Re}\{e\}\mathrm{Im}\{\varphi_p\} \tag{10}$$

where $e = d - y$ is the learning process instantaneous error.

From (9), (10), (8), (7) and in vector representation:

$$\underline{W}(n+1)=\underline{W}(n)+\eta_W e(n)\underline{\varphi}^*(n) \qquad (11)$$

where $\{\cdot\}^*$ denotes the complex conjugate operator.

Likewise, let Ψ_{pq} be the q^{th} component of the p^{th} center vector $\underline{\Psi}_p=\begin{bmatrix}\Psi_{p0} & \Psi_{p1} & \cdots & \Psi_{pq} & \cdots & \Psi_{pL-1}\end{bmatrix}^T$, $q=0,1,\cdots,L-1$ and $p=0,1,\cdots,K-1$. Ψ_{pq} is also a PTRBFNN free parameter which is adjusted by (5), so that

$$\Psi_{pq}(n+1)=\Psi_{pq}(n)-\eta_\Psi \nabla J^{\Psi}_{pq}(n) \qquad (12)$$

where η_Ψ is the adaptation step and

$$\nabla J^{\Psi}_{pq}=\frac{\partial J}{\partial \Psi_{pq}}=\frac{\partial J}{\partial \operatorname{Re}\{\Psi_{pq}\}}+j\frac{\partial J}{\partial \operatorname{Im}\{\Psi_{pq}\}} \qquad (13)$$

From (2) and (4) the real and imaginary parts of (13) are obtained as

$$\begin{gathered}\frac{\partial J}{\partial \operatorname{Re}\{\Psi_{pq}\}}=\frac{1}{2}\frac{\partial |d-y|^2}{\partial \operatorname{Re}\{\Psi_{pq}\}}=\frac{1}{2}\frac{\partial \left|d-\sum_{i=0}^{K-1}W_i\varphi_i\right|^2}{\partial \operatorname{Re}\{\Psi_{pq}\}}= \\ -2\operatorname{Re}\{\varphi_p\}\frac{\left[\operatorname{Re}\{\underline{u}\}-\operatorname{Re}\{\underline{\Psi}_p\}\right]}{\operatorname{Re}\{\sigma_p^2\}}\left[\operatorname{Re}\{W_p\}\operatorname{Re}\{e\}+\operatorname{Im}\{W_p\}\operatorname{Im}\{e\}\right]\end{gathered} \qquad (14)$$

$$\begin{gathered}\frac{\partial J}{\partial \operatorname{Im}\{\Psi_{pq}\}}=\frac{1}{2}\frac{\partial |d-y|^2}{\partial \operatorname{Im}\{\Psi_{pq}\}}=\frac{1}{2}\frac{\partial \left|d-\sum_{i=0}^{K-1}W_i\varphi_i\right|^2}{\partial \operatorname{Im}\{\Psi_{pq}\}}= \\ -2\operatorname{Im}\{\varphi_p\}\frac{\left[\operatorname{Im}\{\underline{u}\}-\operatorname{Im}\{\underline{\Psi}_p\}\right]}{\operatorname{Im}\{\sigma_p^2\}}\left[\operatorname{Re}\{W_p\}\operatorname{Im}\{e\}-\operatorname{Im}\{W_p\}\operatorname{Re}\{e\}\right]\end{gathered} \qquad (15)$$

From (14), (15), (13), (12) and in vector representation:

$$\underline{\Psi}_p(n+1)=\underline{\Psi}_p(n)+\eta_\Psi\cdot\left\{\begin{aligned}&\operatorname{Re}\{\varphi_p(n)\}\frac{\left[\operatorname{Re}\{\underline{u}(n)\}-\operatorname{Re}\{\underline{\Psi}_p(n)\}\right]}{\operatorname{Re}\{\sigma_p^2(n)\}}\cdot\\&\left[W_p(n)e^*(n)+W_p^*(n)e(n)\right]-\\&\operatorname{Im}\{\varphi_p(n)\}\frac{\left[\operatorname{Im}\{\underline{u}(n)\}-\operatorname{Im}\{\underline{\Psi}_p(n)\}\right]}{\operatorname{Im}\{\sigma_p^2(n)\}}\cdot\\&\left[W_p(n)e^*(n)-W_p^*(n)e(n)\right]\end{aligned}\right\} \qquad (16)$$

The last PTRBFNN free parameter which is adjusted by (5) is the p^{th} center variance σ_p^2. Thus, from (5),

$$\sigma_p^2(n+1)=\sigma_p^2(n)-\eta_\sigma \nabla J_p^\sigma(n) \qquad (17)$$

where η_σ is the adaptation step and

$$\nabla J_p^\sigma=\frac{\partial J}{\partial \sigma_p^2}=\frac{\partial J}{\partial \operatorname{Re}\{\sigma_p^2\}}+j\frac{\partial J}{\partial \operatorname{Im}\{\sigma_p^2\}} \qquad (18)$$

From (2) and (4) the real and imaginary parts of (18) are obtained as

$$\begin{gathered}\frac{\partial J}{\partial \operatorname{Re}\{\sigma_p^2\}}=\frac{1}{2}\frac{\partial |d-y|^2}{\partial \operatorname{Re}\{\sigma_p^2\}}=\frac{1}{2}\frac{\partial \left|d-\sum_{i=0}^{K-1}W_i\varphi_i\right|^2}{\partial \operatorname{Re}\{\sigma_p^2\}}= \\ -\left|\operatorname{Re}\{\underline{u}\}-\operatorname{Re}\{\underline{\Psi}_p\}\right|^2\frac{\operatorname{Re}\{\varphi_p\}}{\left(\operatorname{Re}\{\sigma_p^2\}\right)^2}\left[\operatorname{Re}\{W_p\}\operatorname{Re}\{e\}+\operatorname{Im}\{W_p\}\operatorname{Im}\{e\}\right]\end{gathered} \qquad (19)$$

$$\begin{gathered}\frac{\partial J}{\partial \operatorname{Im}\{\sigma_p^2\}}=\frac{1}{2}\frac{\partial |d-y|^2}{\partial \operatorname{Im}\{\sigma_p^2\}}=\frac{1}{2}\frac{\partial \left|d-\sum_{i=0}^{K-1}W_i\varphi_i\right|^2}{\partial \operatorname{Im}\{\sigma_p^2\}}= \\ -\left|\operatorname{Im}\{\underline{u}\}-\operatorname{Im}\{\underline{\Psi}_p\}\right|^2\frac{\operatorname{Im}\{\varphi_p\}}{\left(\operatorname{Im}\{\sigma_p^2\}\right)^2}\left[\operatorname{Re}\{W_p\}\operatorname{Im}\{e\}-\operatorname{Im}\{W_p\}\operatorname{Re}\{e\}\right]\end{gathered} \qquad (20)$$

From (19), (20), (18) and (17):

$$\sigma_p^2(n+1)=\sigma_p^2(n)+\eta_\sigma\left\{\begin{aligned}&\left|\operatorname{Re}\{\underline{u}(n)\}-\operatorname{Re}\{\underline{\Psi}_p(n)\}\right|^2\frac{\operatorname{Re}\{\varphi_p(n)\}}{\left(\operatorname{Re}\{\sigma_p^2(n)\}\right)^2}\cdot\\&\left[\operatorname{Re}\{W_p(n)\}\operatorname{Re}\{e(n)\}+\operatorname{Im}\{W_p(n)\}\operatorname{Im}\{e(n)\}\right]+\\&j\left|\operatorname{Im}\{\underline{u}(n)\}-\operatorname{Im}\{\underline{\Psi}_p(n)\}\right|^2\frac{\operatorname{Im}\{\varphi_p(n)\}}{\left(\operatorname{Im}\{\sigma_p^2(n)\}\right)^2}\cdot\\&\left[\operatorname{Re}\{W_p(n)\}\operatorname{Im}\{e(n)\}-\operatorname{Im}\{W_p(n)\}\operatorname{Re}\{e(n)\}\right]\end{aligned}\right\} \qquad (21)$$

Summing up, the PTRBFNN gradient based learning process is given by equations (11), (16) and (21). For all these equations, the instant error $e(n)$ is a measure of the distance between the PTRBFNN output $y(n)$ and the original transmitted IQ symbol $d(n)$ whose value we assume is stored in a look up table at the receiver as reference data for the learning process:

$$e(n)=d(n)-y(n) \qquad (22)$$

3. Blind channel equalization with the PTRBFNN

Since $d(n)$ is an IQ symbol sequence which is periodically transmitted in order to perform the equalizer training, it occurs that some channel bandwidth is wasted for this purpose.

A possible approach to avoid the transmission of a training sequence is to adopt a blind channel deconvolution algorithm [9]. To this end, in this work, the training training sequence for the PTRBFNN is obtained with basis on the blind concurrent algorithm proposed by De Castro et al. [10][14].

Specifically, the concurrent arquitecture is obtained by modifying the PTRBFNN output given by (2) to the form

$$y = \sum_{k=0}^{K-1} W_k \varphi_k + \sum_{\ell=0}^{L-1} V_\ell u_\ell = \underline{W}^T \cdot \underline{\varphi} + \underline{V}^T \cdot \underline{u} \tag{23}$$

where u_ℓ is the ℓ^{th} complex valued element of the channel regressor $\underline{u}$ and V_ℓ is the ℓ^{th} complex valued component of the vector $\underline{V} = [V_0 \quad V_1 \quad \cdots \quad V_{L-1}]^T$ which is adjusted via stochastic gradient algorithm in order to minimize the Godard Cost Function [13] given by

$$J_G(n) = \frac{1}{4}\left\{\left(|y(n)|^2 - \gamma\right)^2\right\} \tag{24}$$

with γ being a dispersion constant defined as

$$\gamma = \frac{E\left\{|A|^4\right\}}{E\left\{|A|^2\right\}} \tag{25}$$

where $E\{\cdot\}$ is the expectation operator and A is the digital modulation IQ symbol set, also known as modulation alphabet [6]. For instance, for a 16-QAM modulation with a unit variance alphabet, the resulting dispersion constant from (25) is $\gamma = 1.32$. Notice that the Godard Cost Function given by (24) is phase invariant, therefore the gradient based update of vector $\underline{V}$ is insensitive to the IQ symbol sequence phase information.

Notice also from (23) that, for a given input channel regressor $\underline{u}$, the data flow goes through two paths: path I is the PTRBFNN and path II is the FIR filter whose coefficients are given by the components of vector $\underline{V}$. The two paths add together to produce the equalizer output y, thus, defining the concurrent architecture.

From (5), in order to minimize (24) with respect $\underline{V}$, and in vector representation [13]:

$$\underline{V}(n+1) = \underline{V}(n) - \eta_W \underline{\nabla} J_G(n) \tag{26}$$

where η_V is the adaptation step. From (23), (24) and (26) we have:

$$\underline{V}(n+1) = \underline{V}(n) + \eta_V y(n)\left(\gamma - |y(n)|^2\right)\underline{u}^*(n) \tag{27}$$

Similarly to the non-blind case, the PTRBFNN learning process consists of to minimize the cost function given by (4) by adjusting the PTRBFNN free parameters $\underline{W}$, $\underline{\Psi}_k$ and σ_k^2 according to (5), with $k = 0,1,\cdots,K-1$. Following the same derivation in [10], the PTRBFNN blind learning process can be described by:

$$\underline{W}(n+1) = \underline{W}(n) + \eta_W (1 - D(n)) e_b(n) \underline{\varphi}^*(n) \tag{28}$$

$$\underline{\Psi}_k(n+1) = \underline{\Psi}_k(n) + \eta_\Psi (1 - D(n)) \left\{ \begin{array}{l} \mathrm{Re}\{\varphi_k(n)\} \dfrac{\left[\mathrm{Re}\{\underline{u}(n)\} - \mathrm{Re}\{\underline{\Psi}_k(n)\}\right]}{\mathrm{Re}\{\sigma_k^2(n)\}} \cdot \\ \left[W_k(n) e_b^*(n) + W_k^*(n) e_b^*(n)\right] - \\ \mathrm{Im}\{\varphi_k(n)\} \dfrac{\left[\mathrm{Im}\{\underline{u}(n)\} - \mathrm{Im}\{\underline{\Psi}_k(n)\}\right]}{\mathrm{Im}\{\sigma_k^2(n)\}} \cdot \\ \left[W_k(n) e_b^*(n) - W_k^*(n) e_b^*(n)\right] \end{array} \right\} \tag{29}$$

$$\sigma_k^2(n+1) = \sigma_k^2(n) + \eta_\sigma (1 - D(n)) \left\{ \begin{array}{l} \left|\mathrm{Re}\{\underline{u}(n)\} - \mathrm{Re}\{\underline{\Psi}_k(n)\}\right|^2 \cdot \\ \dfrac{\mathrm{Re}\{\varphi_k(n)\}}{\left(\mathrm{Re}\{\sigma_k^2(n)\}\right)^2} \cdot \\ \begin{bmatrix} \mathrm{Re}\{W_k(n)\}\mathrm{Re}\{e_b(n)\} + \\ \mathrm{Im}\{W_k(n)\}\mathrm{Im}\{e_b(n)\} \end{bmatrix} + \\ j\left|\mathrm{Im}\{\underline{u}(n)\} - \mathrm{Im}\{\underline{\Psi}_k(n)\}\right|^2 \cdot \\ \dfrac{\mathrm{Im}\{\varphi_k(n)\}}{\left(\mathrm{Im}\{\sigma_k^2(n)\}\right)^2} \cdot \\ \begin{bmatrix} \mathrm{Re}\{W_k(n)\}\mathrm{Im}\{e_b(n)\} - \\ \mathrm{Im}\{W_k(n)\}\mathrm{Re}\{e_b(n)\} \end{bmatrix} \end{array} \right\} \tag{30}$$

where

$$D(n) = \begin{cases} 0 \text{ if } Q\{y(n)\} = Q\{\tilde{y}(n)\} \\ 1 \text{ if } Q\{y(n)\} \neq Q\{\tilde{y}(n)\} \end{cases} \tag{31}$$

being $\tilde{y}(n)$ the value of output y after the update of vector $\underline{V}$ by means of (27):

$$\tilde{y}(n) = \underline{W}^T(n) \cdot \underline{\varphi}(n) + \underline{V}^T(n+1) \cdot \underline{u}(n) \tag{32}$$

and

$$e_b(n) = Q\{y(n)\} - y(n) \tag{33}$$

being $Q\{\cdot\}$ the operator which returns the reference constellation IQ symbol s_m with the smallest Euclidean distance to the argument:

$$\begin{aligned} &Q\{\cdot\} = \arg\min_{s_m} \left|\{\cdot\} - s_m\right|^2 \\ &s_m \in A \\ &m = 0,1,\cdots,M-1 \end{aligned} \tag{34}$$

where M is the cardinality of the IQ symbol set A.

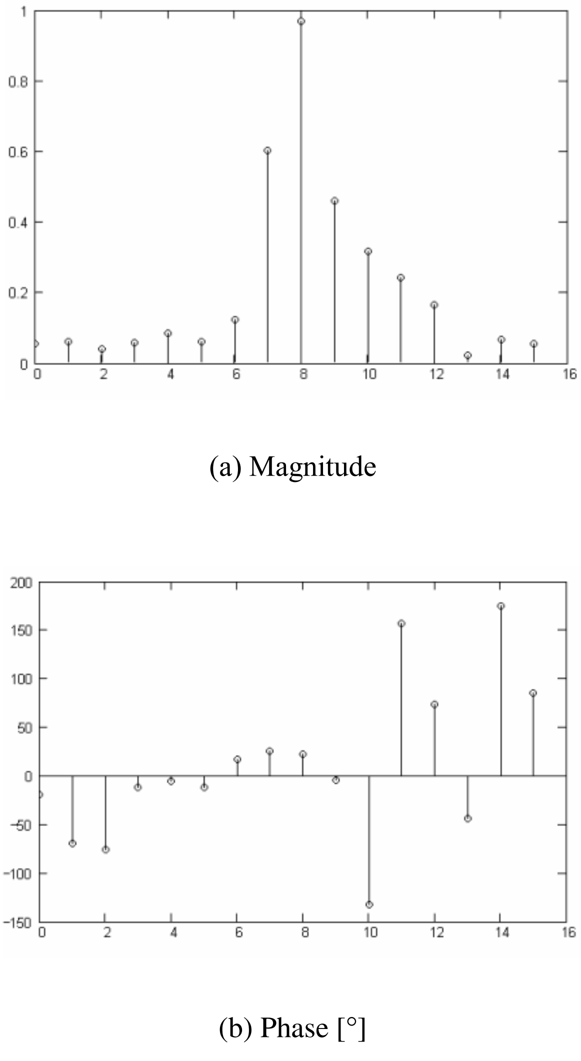
Notice that with this approach no training sequence is necessary for the PTRBFNN learning process.

4. SIMULATION RESULTS

In this section we evaluate the PTRBFNN performance in comparison with the classical complex valued RBFNN whose basis function is given by (1). We adopt a 16-QAM digital modulation with a unit variance alphabet A for the transmitted IQ symbol sequence $s(n)$, thus $M = 16$ [6]. The samples of source sequence $s(n)$ are randomly drawn from alphabet A such that $s(n)$ is statistically independent with uniform distribution.

In order to reduce the noise enhancement we use a fractionally-spaced equalizer [6][11], i.e., the channel regressor $\underline{u}$ stores $T/2$–spaced IQ samples from the communication channel. Aiming to evaluate the PTRBFNN performance in a realistic situation, we adopt the channel models available in the Signal Processing Information Base (SPIB) [12] at http://spib.rice.edu/. The SPIB microwave channel models are available at http://spib.rice.edu/spib/microwave.html. These models are the impulse response of several practical microwave channels obtained from field measurements using a high fractionally-spaced sampling rate (tens of megabauds per second). This allows the database user to decimate the impulse response sequence according to the particular requirements of the study, without losing significant information. The great majority of works on the subject of channel equalization seldom use more than some tens of samples for the channel model. In this work we will decimate the original SPIB microwave channel impulse response to a channel length $L_C = 16$. In order to not alter the channel root locus we use frequency domain decimation.

One of the channel models available from SPIB is the microwave channel M7, whose impulse response is shown in Figure 2. This channel rotates the received IQ symbol constellation when deconvolved via CMA equalizer [13], which is recognized as one of the most widely used algorithms in blind channel equalization practice. Thus this channel model can be used as a benchmark for the PTRBFNN performance determination with respect to the phase sensitiveness.



(a) Magnitude

(b) Phase [°]

Figure 2: SPIB channel M7 impulse response.

Besides the multipath effects represented by the channel impulse response in Figure 2, we assume an additive white Gaussian noise with Signal to Noise Ratio (SNR) of 35dB at the receiver input. The non-linear transmittance of the receiver RF front-end when operating near the upper limit of its dynamic range is also taken into account – we introduce 5% of 2nd and 3rd order harmonic distortion so that the received sequence $r(n)$ is submitted to the non-linear relationship $r_o(n) = r(n) + \alpha_2 r^2(n) + \alpha_3 r^3(n)$, $\alpha_2 = \alpha_3 = 0.05$, where $r_o(n)$ is the IQ sequence after the receiver RF front-end.

The simulation conditions are such both equalizers – the PTRBFNN equalizer and the classical complex valued RBFNN equalizer – operate in the blind concurrent architecture dictated by (23). Both equalizers share the same architectural and training parameters: $L = 16$, $K = 5$, $\eta_V = 1\times10^{-3}$, $\eta_W = 0.9$, $\eta_\Psi = 0.5$, $\eta_\sigma = 5.0$. Both equalizers also share a common initialization procedure: $\underline{V}(0) = [0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1+j0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$, $\underline{W}(0) = \underline{0}$ and all the L components of the k^{th} center initial vector $\underline{\Psi}_k(0)$ are randomly drawn from the 16-QAM alphabet A. σ_k^2 is initialized with half the maximum Euclidean distance between all vectors $\underline{\Psi}_k(0)$ for the classical RBFNN. For the PTRBFNN, $\mathrm{Re}\{\sigma_k^2\}$ and $\mathrm{Im}\{\sigma_k^2\}$ are initialized with half the maximum Euclidean distance between all respective vectors $\mathrm{Re}\{\underline{\Psi}_k(0)\}$ and $\mathrm{Im}\{\underline{\Psi}_k(0)\}$, with $k = 0,1,\cdots,K-1$.

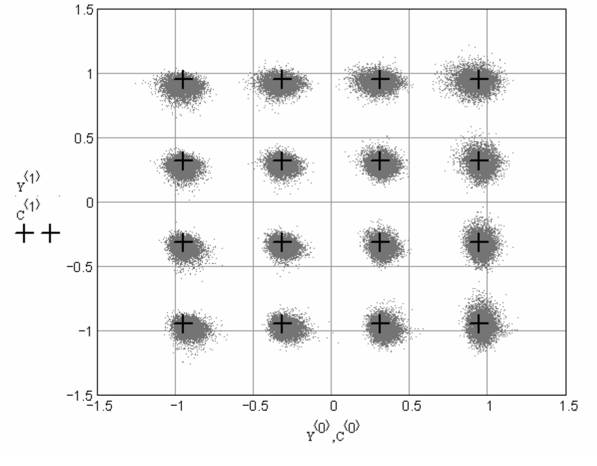


Figure 3: PTRBFNN equalizer output y after convergence.

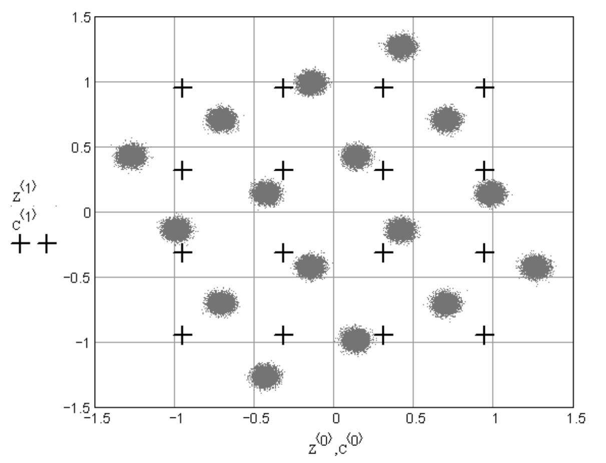


Figure 4: Classical complex valued RBFNN equalizer output y after convergence.

Figures 3 and 4 respectively show the resulting digital modulation constellations for output y of the PTRBFNN equalizer and for the output y of the classical complex valued RBFNN equalizer, after the equalizer convergence and for the previously given simulation conditions. Convergence is considered to be achieved when the output mean square error $MSE = E\{e^2(n)\}$ has decreased to a small enough value $MSE < 0.076$ [10]. In these figures, the characters '+' represent the source reference alphabet IQ symbols. Notice that the rotation of the symbols in Figure 4 with respect to the reference constellation stems from the fact that the RBFNN is not able to solve SPIB channel M7 phase distortion. Comparing Figures 3 and 4 its clear that both the RBFNN and the PTRBFNN were able to reduce the output samples dispersion, the difference being that the RBFNN output is rotated with respect to the reference constellation. This difference in terms of rotation evidences the influence of the phase transmittance on the PTRBFNN learning process, which effectively takes into account the channel phase rotation.

Figure 5 shows the convergence rate $MSE(n)$ for both equalizers. Notice that the RBFNN equalizer minimizes the channel dispersion, but it is not able to achieve a low steady-state MSE due to the phase rotation of its output y. On the other hand, the PTRBFNN not only minimizes the channel dispersion but also corrects the channel phase rotation after approximately 1.4×10^5 IQ symbols are received from the channel.

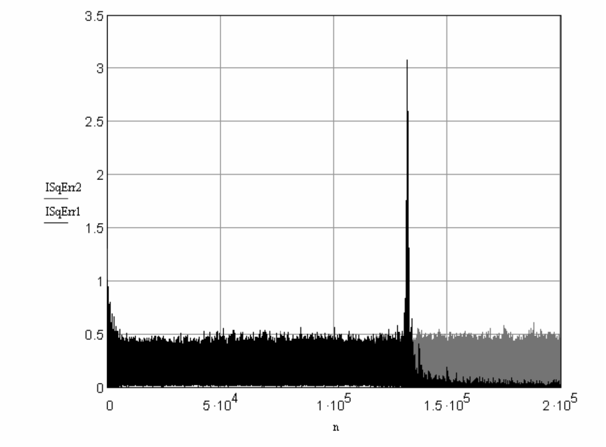


Figure 5: $MSE(n)$ for the PTRBFNN equalizer (black curve) and the classical complex valued RBFNN equalizer (gray curve).

5. CONCLUSION

The proposed complex valued radial basis function (RBF) neural network with phase transmittance (PTRBFNN) outperforms the classical complex valued RBF neural network (RBFNN) for those channels which impose drastic phase rotation to the received symbol sequence. The PTRBFNN not only presents a higher convergence rate than the RBFNN but also exhibits a lower steady state MSE, thus minimizing the phase invariance problem inherent to the RBFNN hidden-layer when applied to blind channel equalization. This is an important PTRBFNN feature, since adaptive joint equalization and carrier phase recovery is desirable for wireless mobile operation.

Although we have applied the new PTRBFNN to solve a specific blind channel equalization problem, its usefulness extends to any situation for which the forward transmittance of the hidden layer neurons must include phase information, such as is the case for spectral analysis in signal processing or near-field analysis in electromagnetics.

REFERENCES

[1] S. Chen, S. McLaughlin and B. Mulgrew: Complex-Valued Radial Basis Function Network, Part I: Network Architecture and Learning Algorithms. *Signal Processing*. vol. 35, pages 19-31,1994.

[2] S. Chen, S. McLaughlin and B. Mulgrew: Complex-Valued Radial Basis Function Network, Part II: Application to Digital Communications Channel Equalisation. *Signal Processing*. vol. 36, pages 175-188, 1994.

[3] Cha I. and Kassam S.: Channel Equalization Using Adaptative Complex Radial Basis Function Networks. *IEEE Journal on Selected Areas in Communications*. vol.13, no. 1, pages.122-131, 1995.

[4] Gan Q., Saratchandran P., Sundararajan N., and Subramanian K. R.: A Complex Valued Radial Basis Function Network for Equalization of Fast Time Varying Channels. *IEEE Transactions on Neural Networks*. vol. 10, no. 4, 1999.

[5] Jianping D., Sundararajan N., and Saratchandran P.: Communication Channel Equalization Using Complex-Valued Minimal Radial Basis Function Neural Networks. *IEEE Transactions on Neural Networks*, vol. 13, no. 3, 2002.

[6] J. G. Proakis. Digital Communications, 3rd ed., McGraw-Hill, 1995.

[7] S. Haykin. Adaptative Filter Theory. 3rd ed., Prentice Hall, Upper Saddle River, New Jersey, 1996.

[8] S. Haykin. Neural Networks. 2nd ed., Prentice Hall, Upper Saddle River, New Jersey, 1999.

[9] S. Haykin. Blind Deconvolution. Prentice-Hall, 1994.

[10] Fernando C.C. De Castro, M. Cristina F. De Castro, Dalton S. Arantes. Concurrent Blind Deconvolution for Channel Equalization. *IEEE International Conference On Communications ICC2001*. page. 366-371, Helsinki, Finland , June 2001.

[11] Gitling R.D. and Weinstein S.B.: Fractionally-Spaced Equalization: An Improved Digital Transversal Equalizer. *Bell Systems Technical Journal,* vol. 60, 1981.

[12] SPIB – Signal Processing Information Base. http://spib.rice.edu/spib/microwave.html, http://spib.rice.edu/spib/cable.html

[13] Godard D.N.: Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems. *IEEE Transactions on Communications*. vol. COM-28, no.11, November 1980.

[14] Chen S. and Chang E.S.: Fractionally spaced blind equalization with low-complexity concurrent constant modulus algorithm and soft decision-directed scheme. *International Journal of Adaptive Control and Signal Processing*. 19(6) page. 471-484, 2005.