# A NEURAL PREDICTOR FOR BLIND EQUALIZATION OF DIGITAL COMMUNICATION SYSTEMS: IS IT PLAUSIBLE?


Charles Casimiro Cavalcante*
Universidade Federal do Ceará
Fortaleza CE, Brazil
charlesc@dee.ufc.br

Jugurta Rosa Montalvão Filho.*
Universidade Tiradentes
Aracaju SE, Brazil
Jugurta.Montalvao@int-evry.fr

Bernadette Dorizzi
*Inst. Nat. des Télécom.
Evry, France
Bernadette.Dorizzi@int-evry.fr

João Cesar Moura Mota
Universidade Federal do Ceará
Fortaleza CE, Brazil
mota@dee.ufc.br



**Abstract. In digital channel equalization, self-learning techniques are used in the cases where a training period is not available. Considering the transmitted sequence as composed of independent random variables, the equalization task can be done by means of prediction. In this work we propose Artificial Neural Networks (ANN), instead of a linear prediction device, in order to obtain a better performance and analyse its performance and applicability. Linear and nonlinear prediction concepts are revisited and a new self-organized algorithm is proposed to update the first layer in the nonlinear predictor whose aim is to avoid local minimum points in the applied cost function. The second layer is updated by using a classical supervised algorithm based on prediction error. Simulation results are presented which illustrate the performance of this technique.**


## INTRODUCTION

Equalization of digital communication channels is usually done by using a transmitted sequence also known to the receiver during a preamble period. Figure 1 depicts a simplified digital communication system, where $\mathbf{a}(n)$ is the transmitted sequence, $\mathbf{b}(n)$ is the noise sequence and $\hat{a}(n-d)$ is the estimated symbol after a delay $d$.

Self-learning (blind) equalizers are used in order to provide the correct identification of transmitted symbols when one does not have a training period or when it is not practical to use such a strategy, as in digital TV

0-7803-6278-0/00/$10.00 (C) 2000 IEEE

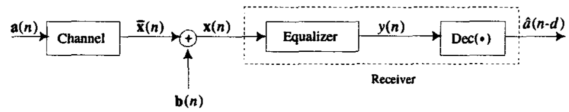



Figure 1: Digital Communication System.

broadcasting and multipoint networks where training has to be redone whenever one single receiver is inserted in the system. Another example is mobile communication systems, where due to multipath fading, the received signal may be so low that the receiver does not synchronize adequately.

Some classical strategies for blind equalization are the following related algorithms: Direct Decision (DD), Sato, Godard [3], Benveniste-Goursat [1] and Shalvi-Weinstein [6].

Considering the transmitted symbols to be uncorrelated, it is possible to deal with the blind equalization problem by means of prediction [5]. In this context, it seems that the pioneer work is that of Macchi and Hachicha, in 1986, who used a linear filter as a prediction error filter. The symbol with the desired information is recovered, in this case, by elimination of the existing redundancy in the time sequence formed by the channel outputs.

In classical implementations for minimum phase channels, the prediction filter is linear and has a finite impulse response which is adapted to minimize the prediction squared error. This, indeed, is equivalent to a whitening process over the received time sequence and the white sequence obtained in the prediction error filter output could be the same as that of the transmitted symbols when we use an Automatic Gain Control (AGC) [5]. The prediction error sequence will be i.i.d. if the transmitted sequence $\{\mathbf{a}(n)\}$ is also i.i.d. and the noise is negligeable.

Nevertheless, in communication systems, a crucial point limits the benefits from the linear prediction: if the channel is nonminimum phase, the original transmitted sequence cannot be recovered from a whitened error sequence. In other words, the original transmitted sequence cannot be recovered as result of the intrinsic linear mapping of past samples on the current estimated one. In this case, it is necessary include in the linear mapping future samples. A delay could also be introduced to compensate for noncausality condition, but equalization is still not guaranteed. Nonetheless, it is quite easy to show that, in most cases, the ideal mapping is nonlinear (see example in Section 2).

Therefore, in this work, we propose a nonlinear structure based on Artificial Neural Networks, with one input and two layers, as a prediction device with a criterion based on $2^{nd}$ order statistics. Moreover, in order to improve the adaptive solution, we divided the learning task in to two stages. First, a new self-organized learning algorithm is proposed to adapt the first layer then the second layer connections are updated by means of a classical supervised algorithm (supervised with respect to the prediction error, but blind with respect to the transmitted symbols).

In Section 2, we explore the prediction concepts. Section 3 is dedicated to the new proposed self-organized algorithm. In Section 4, some simulation results are presented to illustrate the performance of this new strategy and, in the last section, conclusions are presented.

## PREDICTION CONCEPTS

In digital communication systems, the implicit goal of applying prediction is to remove the temporal redundancies from the received signal, which can be used in blind equalization. The representation of a prediction-based equalizer is shown in Figure 2, where $\mathbf{x}(n)$ is the noisy channel output sequence,

$\bar{\mathbf{x}}(n)$ noiseless channel output sequence, $\hat{x}(n)$ is the predicted signal, $e(n)$ is the prediction error, $P$ is a prediction filter and $g$ is an AGC.

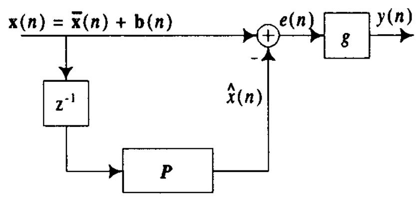



Figure 2: Prediction-Based Equalizer.

The channel is modeled as a linear filter with finite impulse response (FIR) and its transfer function is represented by

$$F(z) = \sum_{i=0}^{N-1} f_i z^{-i} \tag{1}$$

where $f_i$ are the channel coefficients and $N$ is the channel length. We also can represent the channel model in a vectorial form: $\mathbf{f} = [\ f_0\ f_1\ f_2\ \cdots\ f_{N-1}\ ]^T$. Then, the system model will be: $\mathbf{a}(n) = [\ a(n)\ a(n-1)\ \cdots\ a(n-N+1)\ ]^T$, $\mathbf{x}(n) = [\ x(n)\ x(n-1)\ \cdots\ x(n-N-M+1)\ ]^T$ and $\mathbf{b}(n) = [\ b(n)\ b(n-1)\ \cdots\ b(n-N-M+1)\ ]^T$ where $M$ is the order of the equalizer.

Therefore, the noiseless channel outputs, which we call *channel states*, can be written as:

$$\begin{aligned}
\bar{x}(n) &= a(n)f_0 + a(n-1)f_1 + \cdots + a(n-N+1)f_{N-1} \\
\bar{x}(n-1) &= a(n-1)f_0 + a(n-2)f_1 + \cdots + a(n-N+2)f_{N-1} \\
\bar{x}(n-2) &= a(n-2)f_0 + a(n-3)f_1 + \cdots + a(n-N+3)f_{N-1} \\
\vdots \quad & \qquad \vdots \qquad\qquad\quad \vdots \qquad\qquad\qquad \vdots
\end{aligned} \tag{2}$$

And then, the prediction error corresponds to:

$$e(n) = x(n) - P(\mathbf{x}(n-1)) \tag{3}$$

where $\mathbf{x}(n-1) = [\ x(n-1)\ x(n-2)\ \cdots\ ]^T$, $x(n) = \bar{x}(n) + b(n)$ and $P$ is a function which provides a prediction of $x(n)$.

The simplest structure is obtained by choosing $P$ to be linear. Although it is the most frequently used function, the corresponding application of the linear predictor is limited to minimum or maximum phase channels [3, 5]. Some works (for instance [5] and references therein) have proposed a combined structure to treat nonminimum phase channels.

To remove temporal redundancies, the prediction error equation is rewritten in the form:

$$\begin{gathered} e(n) = a(n)f_0 + a(n-1)f_1 + \cdots + b(n) \\ -P\left(\bar{x}(n-1) + b(n-1) + \bar{x}(n-2) + b(n-2) + \cdots\right) \end{gathered} \tag{4}$$

Using a linear filter with discrete finite impulse response $\mathbf{p} = [\,p_1\ p_2\ p_3 \cdots\ p_k]$, as a predictor device, we have:

$$\begin{gathered} e(n) = \underbrace{a(n)f_0 + a(n-1)f_1 + \cdots + b(n)}_{x(n)} \\ -\underbrace{[x(n-1)p_1 + x(n-2)p_2 + \cdots + x(n-k)p_k]}_{\hat{x}(n)} \end{gathered} \tag{5}$$

where $p_i$ is the $i$-th prediction filter coefficient.

Expanding $\hat{x}(n)$ leads to:

$$\begin{gathered} \hat{x}(n) = \left(a(n-1)f_0 + a(n-2)f_1 + \cdots + b(n-1)\right)p_1 \\ + \left(a(n-2)f_0 + a(n-3)f_1 + \cdots + b(n-2)\right)p_2 + \cdots \\ + \left(a(n-k)f_0 + a(n-k+1)f_1 + \cdots + b(n-k)\right)p_k \end{gathered} \tag{6}$$

Combining Equations (5) and (6) leads to:

$$\begin{gathered} e(n) = a(n)f_0 + b(n) + a(n-1)\left[f_1 - f_0p_1\right] \\ -b(n-1)p_1 + a(n-2)\left[f_2 - f_1p_1 - f_0p_2\right] - b(n-2)p_2 \\ + \cdots - a(n-N+1)f_{N-1}\left[p_k\right] - b(n-N+1)p_k \end{gathered} \tag{7}$$

The goal here is to recover $a(n)f_0$. For this purpose, we must remove the undesired symbols by adapting the prediction filter in order to force them to zero. Unfortunately, not all coefficients can be canceled at once.

It becomes evident that, there is a *residue* in the prediction error expression and this residue cannot be cancelled by a finite linear filter. For equalization to be achieved, the samples of the prediction error sequence must to be uncorrelated and this residue must also be negligible with respect to $a(n)f_0$. A possible solution to this problem is to increase the predictor order which decreases the contribution of the residue. The scale factor multiplying $a(n)$ is recovered by the AGC that matches the power of sequence $\mathrm{e}(n)$ and transmitted sequence $\mathbf{a}(n)$.

However it is known that in the nonminimum phase channel case, it does not work and in any case the noise itself cannot be removed.

Since the linear mapping of a linear predictor may be not enough for equalization, we have tried to find a structure able to perform a nonlinear mapping in a satisfactory way. We chose the function implemented by an ANN that has the following form:

$$\psi = \sum_i \beta_i \cdot \operatorname{sign}(x - \theta_i) \tag{8}$$

where sign$(\cdot)$ is the signum function.

In the nonlinear case, Equation (3) is rewritten by replacing function $P$ by a nonlinear function $\psi_{NN}$ where the subscript stands for a neural network.

$$e(n) = x(n) - \psi_{NN}(\mathbf{x}(n-1)) \tag{9}$$

According to Equation (8), $\psi_{NN}$ is a sum of weighted and shifted copies of sign$(\cdot)$, whose parameters $\beta_i$ and $\theta_i$ would be found by means of an a priori knowledge of the channel coefficients. However, since we do not have such an *a priori* knowledge, all parameters of $\psi_{NN}$ are stochastically adjusted by means of the new algorithm described in Section 3.

Expanding Equation (9), it follows that:

$$\begin{aligned} e(n) = a(n)f_0 + a(n-1)f_1 + \cdots + b(n) \\ -\psi_{NN}(x(n-1)\ ,\ x(n-2)\ ,\ \cdots) \end{aligned} \tag{10}$$

It is possible to find a function $\psi_{NN}$ such that we can exactly cancel the term: $a(n-1)f_1 + \cdots + a(n-N+1)f_{N-1}$. Moreover this function can only explicitly depend on $x(n-1)$ since it has all information about past symbols that we need. So, rewriting Equation (10) it follows that:

$$\begin{aligned} e(n) = a(n)f_0 + a(n-1)f_1 + \cdots + b(n) - \\ \underbrace{\psi_{NN}(a(n-1)f_0 + a(n-2)f_1 + \cdots + b(n-1))}_{a(n-1)f_1 + a(n-2)f_2 + \cdots + a(n-N+1)f_{N-1}} \end{aligned} \tag{11}$$

In this case we can obtain no residue. It is worth noting that the noise $b(n)$ is assumed to be an white Gaussian random variable and, consequently, it is not predictable, therefore the best the ANN can do is to cancel redundancies in the time sequence $\mathbf{e}(n)$.

Figure 3 shows, a two-dimensional illustration of a nonlinear mapping done by the ANN.

Clearly, the parameters $\theta_i$, in Equation (8), have a crucial role on the construction of $\psi_{NN}$. So, the problem of finding good parameters for the ANN is addressed in Section 3.

## NEW SELF-ORGANIZED LEARNING ALGORITHM

In classical techniques, such as backpropagation [4], the updating of parameters $\theta_i$ and of the synaptic weights is usually done through the same pro-

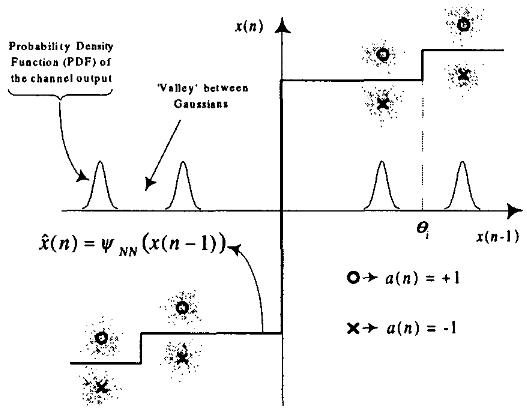



Figure 3: Nonlinear Mapping Function.

cedure. However, in this equalization problem such a procedure may not be able to quickly provide the fast transitions shown in Figure 3[1].

In order to solve this problem in a satisfactory way, we propose a new self-organized learning algorithm that is based on the minimization of a cost function in order to correctly find the $\theta_i$ of the neurons and to simplify the task of finding the interpolation surface.

It is easy to show that the probability density function (PDF) of the received signal is a mixture of Gaussians. Furthermore, the variance of each Gaussian is the noise variance and their means are channel-characteristic-related.

It can be seen from Figure 3 that the function referred to in Equation (8) can achieve correct interpolation if parameters $\theta_i$, associated to the step transitions, are well placed between the 'valleys'. Then, we minimize a cost function that permits us to find those parameters by lookin for a function that can fit in those 'valleys'. Since the valleys have the shape of a "V", perhaps the simplest function, similar to a "V" we can use is $N(x, \theta_i) = |x - \theta_i| + \kappa$ where $\kappa > 0$ (see Figure 4). We use this function in order to simplify the resulting algorithm.

The constant $\kappa$ is inserted to avoid instability problems when $|x - \theta_i|$ is very small and to guarantee a strictly positive function $N(x, \theta_i)$.

In order to measure function similarities, we have chosen the Kullback-Leibler divergence (KLD), which is indeed a distance measure in the Riemann space [4] given by:

$$D_{h(x)||g(x)} = \int_{-\infty}^{\infty} h(x) \ln\left(\frac{h(x)}{g(x)}\right) dx \qquad (12)$$

[1] In a previous work we have used the backpropagation algorithm for the equalization of channels with low intersymbol interference (ISI) [2].

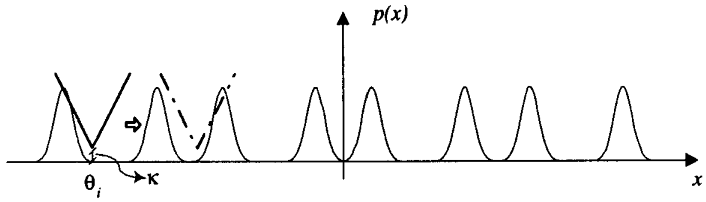



Figure 4: Looking for the 'valleys'.

where $h(x)$ and $g(x)$ are two strictly positive functions. Since the PDF of $x(n-1)$ and $N(x,\theta_i)$ are strictly positives, we apply KLD in order to measure similarities between them.

Eliminating the term which does not depend on $\theta_i$ we obtain one cost function for each neuron given by:

$$\begin{aligned} J(\theta_i) &= \int_{-\infty}^{\infty} p(x).\ln\left(\frac{1}{N(x,\theta_i)}\right)dx \\ J(\theta_i) &= -\mathbb{E}\{\ln(N(x,\theta_i))\} \end{aligned} \tag{13}$$

In order to find the minima of this cost function we have to set $\frac{\partial J(\theta_i)}{\partial \theta_i} = 0$, where

$$\frac{\partial J(\theta_i)}{\partial \theta_i} = -\mathbb{E}\left\{\frac{\operatorname{sign}(x-\theta_i)}{|x-\theta_i|+\kappa}\right\} \tag{14}$$

The minimum value of $\theta_i$ can thus be obtained by using a simple stochastic version of the gradient algorithm:

$$\theta_i(n+1) = \theta_i(n) - \lambda\frac{\operatorname{sign}(x(n-1)-\theta_i(n))}{|x(n-1)-\theta_i(n)|+\kappa} \tag{15}$$

which is a local adaptation rule of the Anti-Hebbian kind [4].

Finally, this adaptation rule is applied on the first layer while the second one is updated by a stochastic LMS algorithm.

## SIMULATION RESULTS

In order to investigate the applicability of the nonlinear predictor in blind equalization we compare it with the linear predictor and with a classical algorithm, the Constant Modulus Algorithm (CMA) [3] using two kinds of channels, firstly the minimum phase and then the nonminimum phase.

The order of the nonlinear predictor (number of neurons) and of the linear predictor (number of coefficients) were chosen as a result of several trials with different possibilities aiming to obtain the best results.

Using BPSK modulation and a Signal-to-Noise (SNR) defined as SNR= $10\log_{10}\left(\frac{\sigma_a^2\sum_{i=0}^{N-1} f_i^2+\sigma_b^2}{\sigma_b^2}\right)$ where $\sigma_a^2$ and $\sigma_b^2$ are the symbol and noise variance respectively. Both the Decison Squared Error (DSE) $(\varepsilon(n) = y - \mathrm{Dec}(y))^2$ and the Symbol Error Ratio (SER) were averaged by means of 100 Monte-Carlo trials.

### Minimum Phase Channel

The minimum phase channel used in the computational simulations has the following impulse response:

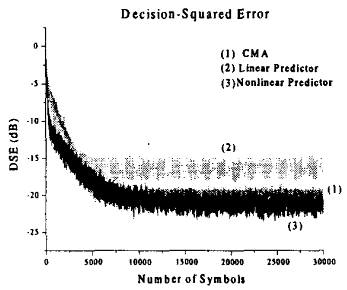
$$f(z) = 1 + 0.8z^{-1} + 0.4z^{-2}$$

In the linear predictor we used a filter with 25 coefficients, a step factor equal to $10^{-3}$ and the initialization is done by setting the vector of filter coefficients at zero.

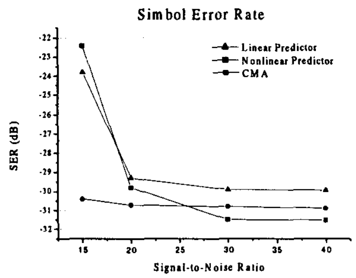
For the CMA, the linear filter has 8 coefficients and the step factor equals $10^{-3}$. The initialization is done by setting the vector of filter coefficients at zero except that at the middle, set at 1.

The nonlinear predictor has one input, 15 neurons in the hidden layer and one output. For this structure we used the following parameters: supervised learning rate equals $10^{-3}$, $\lambda$ equals $5.10^{-4}$. The number of symbols for finding the $\theta_i$ was set to 50 and $\kappa = 10^{-7}$. The algorithm for the AGC [5] has a step size equal to $10^{-3}$ and the weights in the output layer were initialized at zero. The $\theta_i$ were randomly initialized from an uniformly distributed interval: [-1.5,1.5].

We can see in Figure 5(a) the performance achieved when SNR = 40 dB. Figure 5(b) shows the Symbol Error Rate (SER) for some SNRs computed for both predictors and for the CMA.




(a) Decision-Squared Error (SNR = 40 dB).




(b) Symbol Error Rate.

Figure 5: Minimum Phase Channel.

One can easily see that the performance of the nonlinear predictor is better than that of the linear one when SNR > 20 dB.

### Nonminimum Phase Channel

The channel considered has the following impulse response:

$$f(z) = 0.6 + 1z^{-1} - 0.7z^{-2}$$

The linear predictor has the same characteristics as in the previous case. The CMA uses a linear filter with 6 coefficients and a step factor equals $10^{-3}$ and vector of filter coefficients is initialized at zero except that at the middle, set at 1.

The nonlinear predictor has one input, 20 neurons in the hidden layer and one output. The parameters are: supervised learning rate equal to $10^{-3}$ and $\lambda$ equals $10^{-4}$. The number of symbols for finding the $\theta_i$ was 500. The step size of the AGC equals $5.10^{-2}$ and weights in the output layer were initialized at zero whereas the $\theta_i$ from an uniform distribution [-2,2].

Figure 6 shows the DSE of both predictors and CMA, for an SNR of 40 dB. As expected, the performances of the nonlinear predictor and CMA are better than the linear predictor. It can also be noted that the convergence rate of the CMA is much better than the nonlinear predictor one.

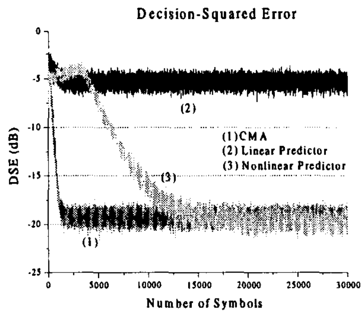



Figure 6: Decison-Squared Error (SNR = 40 dB) - Nonminimum Phase Channel.

## CONCLUSIONS

The strategy presented in this paper proposes a nonlinear prediction device based on Artificial Neural Networks with only one input. Thanks to this strategy, the use of prediction is extended to some cases of nonminimum phase channels. Furthermore, the nonlinear predictor outperforms the linear one even in the cases where it provides channel equalization.

The use of one single input in the nonlinear predictor to achieve equalization instead of several ones common in linear strategies, is presented as a plausible alternative.

The division of the learning task in to two steps: a self-organized for the hidden layer and a supervised for the output layer was proposed to accelerate the ANN abilities, as weel as to avoid the local minimum founs when a single MSE cost function is applied.

However, this strategy is limited to situations where the 'valleys' between Gaussians of the PDF of $x(n-1)$ are deep enough, this deepness depends on the noise power and channel characteristics. In cases where this condition does not hold, we must consider an adaptation of the previous algorithms, particularly acting on the parameter $\kappa$. This improvement is actually under development.

Besides, its performance when compared with the CMA seems to be inferior in most cases. See for example Figure 6, where the ANN predictive approach has higher complexity and slower convergence. This also seems to indicate that the use of neural network-based prediction is limited for some particular cases, indeed, nonlinear channels seem to be an interesting target for neural network-based approaches.

## ACKNOWLEDGMENTS

The authors would like to thank FUNCAP and CNPq from Brazil and ALFA Program/EU for the partial financial support. They would also like to thank INT-Evry France for the logistical support.

## REFERENCES

[1] A. Benveniste and M. Goursat, "Blind Equalizers," **IEEE Trans. on Communications**, vol. COM-32, no. 8, pp. 871–883, August 1984.

[2] C. Cavalcante, J. Mota, B. Dorizzi and J. R. Montalvão, "Preliminary Results in Blind Equalization with Neural Networks Based-Prediction," in **Proc. XVII Simpósio Brasileiro de Telecomunicações - SBT'99**, 1999, pp. 508–513, Vila Velha,ES - Brazil.

[3] S. Haykin, **Adaptive Filter Theory**, (Prentice-Hall Information and System Sciences Series), Prentice-Hall, $3^{rd}$ edn., 1996.

[4] S. Haykin, **Neural Networks: A Comprehensive Foundation**, Prentice Hall, $2^{nd}$ edn., 1998.

[5] O. Macchi, "L'égalisation numérique en communications," **Annales de Télécommunications**, vol. 53, no. 1-2, pp. 39–58, 1998.

[6] O. Shalvi and E. Weinstein, "New Criteria for Blind Deconvolution of Nonminimum Phase Systems (Channels)," **IEEE Trans. on Information Theory**, vol. 36, no. 2, pp. 312–321, March 1990.