ON THE NONLINEAR PREDICTION FOR BLIND EQUALIZATION: IS THAT A GOOD CHOICE?

CHARLES C. CAVALCANTE(1), JOÃO C. M. MOTA(2), JUGURTA R. MONTALVÃO F.(3), B. DORIZZI(4)

(1) Universidade Federal do Ceará, Fortaleza-CE, Brazil, charlesc@dee.ufc.br
(2) Universidade Federal do Ceará, Fortaleza-CE, Brazil, mota@dee.ufc.br
(3) Universidade Tiradentes, Aracaju-SE, Brazil, Jugurta.Montalvao@int-evry.fr
(4) Institut National des Télécomm., Evry, France, Bernadette.Dorizzi@int-evry.fr

ABSTRACT

In this paper we investigate the conditions of applicability of a nonlinear predictor based on neural networks to perform blind equalization using only one input instead of several ones as usual in linear strategies. Linear and nonlinear prediction concepts are revisited in order to clarify some believes. A new self-organized learning algorithm is proposed in order to improve the performance of the nonlinear equalizer. Computational simulations illustrate and compare the performance of the proposed approach with classical structures for equalization.

1. INTRODUCTION

Blind equalization in digital communication systems stands for the estimation of a transmitted symbol with no training sequence in acquisition period, using statistics of the transmitted data. Several techniques have appeared in the last two decades to perform blind equalization, namely, Constant Modulus Algorithm (CMA), Benveniste-Goursat and Shalvi-Weinstein [1, 2, 3] among others.

Most part of those strategies use linear filters with finite impulse response (FIR) and insert nonlinearity in the applied cost function. Although their efficiency, it is well known that the use of a FIR equalizer may provoke an excessive increasing of noise power as well its unability to equalize spectral nulls channels [4].

Aiming to cope with that restriction, several works have proposed the use of nonlinear filter structures, specially neural networks, to blind equalization [5, 6, 7, 8]. The choice of neural networks lay on their characteristics of embedded nonlinearity and also in the saturation characteristics that could provide some improvement to the problem of increasing noise power.

Considering the transmitted symbols to be uncorrelated, equalization can be done by means of prediction [9]. In second-order statistics context, the use of linear prediction is limited to the case of minimum and maximum phase channels [1, 9].

Some works [10, 11, 12] using the concept of prediction have proposed combined structures in order to perform equalization of nonminimum phase channels (NMPC). Those approaches use the well known fact that a NMPC can be separated in its minimum and maximum parts and use linear filters and decision feedback to recover the symbols.

In a classification approach, the equalization problem based on linear prediction is limited due to the linear mapping provided by the filter. Nonetheless, it is quite easy to show that, in most cases, the ideal mapping is nonlinear (see example in Section 2). In this work we propose to investigate the use of a neural network-based predictor to blind equalization. In a previous work [13] we have used the backpropagation learning algorithm to equalize channels with low intersymbol interference.

In order to avoid local minima on the used cost function a new self-organized algorithm is proposed to update the network parameters. This new strategy permits to divide in to two steps the learning process. The first one self-organized and then a classical supervised algorithm based on prediction error.

In Section 2 we explore the linear and nonlinear concepts. Section 3 is devoted to the new self-organized algorithm. Section 4 presents computational simulations comparing the new strategy with classical ones and, in the last section conclusions are presented.

2. PREDICTION CONCEPTS

In digital communication systems, the implicit goal of applying prediction is to remove redundancies in the received signal, which can be used in blind equalization. The representation of a prediction-based equalizer is shown in Fig. 1, where $\mathbf{x}(n)$ is the noisy channel output sequence, $\mathbf{b}(n)$ is the noise, $\hat{x}(n)$ is the predicted signal, $e(n)$ is the prediction error, P is a prediction filter and g is an Automatic Gain Control (AGC).

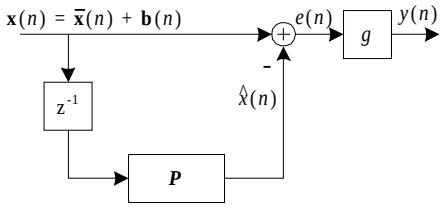

Figure 1: Prediction-Based Equalizer.

The channel is modeled as a linear filter with finite impulse response (FIR) and its transfer function given 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 can also 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: $\mathbf{a}(n) = [\ a(n)\quad a(n-1)\quad \cdots\quad a(n-N+1)\ ]^T$ (transmitted sequence), $\mathbf{x}(n) = [\ x(n)\quad x(n-1)\quad \cdots\quad x(n-N-M+1)\ ]^T$ and $\mathbf{b}(n) = [\ b(n)\quad b(n-1)\quad \cdots\quad b(n-N-M+1)\ ]^T$ where M is the order of the equalizer.

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

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

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. In this case, considering power criterion on the prediction error, equalization based on linear prediction is performed only to minimum or maximum phase channels [1, 9]. On the other hand, some works (for instance [9] and references therein) have proposed a nonlinear filter structure to treat nonminimum phase channels.

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

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

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:

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

where p_i is the i-th prediction filter coefficient.

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

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

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

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

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 negligeable 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 $e(n)$ and $a(n)$.

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 [9].

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:

$$\psi\left(x, \beta_i, \theta_i\right) = \sum_i \beta_i \cdot \text{sign}\left(x - \theta_i\right) \tag{8}$$

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

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

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

According to Equation (8), ψ_{NN} is a sum of weighted and shifted copies of sign$(\cdot)$, whose parameters β_i and θ_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 ψ_{NN} are stochastically adjusted by means of the new algorithm described in Section 3.

Expanding Equation (9), it follows that:

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

It is possible to find a function ψ_{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 to remove redundancies. So, rewriting Equation (10) it follows that:

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

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 $\mathbf{e}(n)$.

Fig. 2 shows, a two-dimensional illustration of a nonlinear mapping done by the ANN.

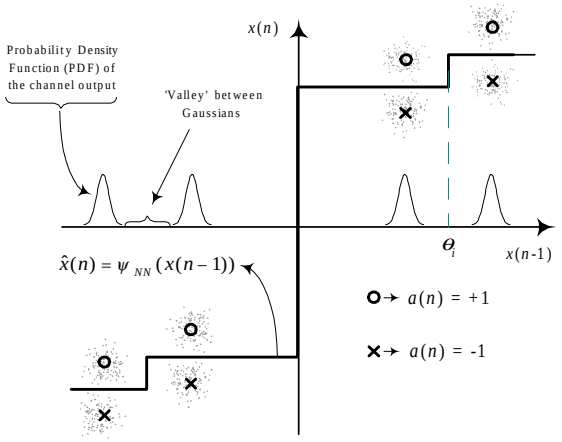

Figure 2: Nonlinear Mapping Function.

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

3. NEW SELF-ORGANIZED LEARNING ALGORITHM

In classical techniques, such as backpropagation [14], the updating of parameters θ_i and of the synaptic weights is usually done through the same procedure. However, in this equalization problem such a procedure may not be able to quickly realize the fast transitions shown in Fig. 2.

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 θ_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 Fig. 2 that the function referred to in Equation (8) can achieve correct interpolation if parameters θ_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 looking 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 Fig. 3). We use this function in order to simplify the resulting algorithm.

The constant κ 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 [14] given by:

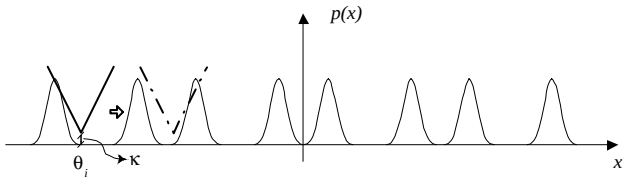

Figure 3: Looking for the 'valleys'.

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

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 θ_i we obtain a cost function 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

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

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

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

which is a local adaptation rule of the Anti-Hebbian kind [14].

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

4. SIMULATION RESULTS

In order to investigate performance of the proposed structure computational simulations of minimum and nonminimum phase channel have been used. In both cases results have been compared with linear predictor and CMA. Performance results were also compared with the limits of Wiener solution [1].

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 σ_a^2 and σ_b^2 are the symbol and noise variance respectively. Decison Squared Error (DSE) $(\varepsilon(n) = y - \text{Dec}(y))^2$ is averaged by means of 100 Monte-Carlo trials.

4.1 Minimum Phase Channel

The selected channel has the following transfer function:

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

For the linear predictor we have used a transversal filter with 25 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 CMA has a filter with 30 coefficients and the step size equals 10^{-3}. The initialization was done in the same way that of linear predictor.

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}, λ equals 5.10^{-4}. The number of symbols for finding the θ_i was set to 50 and $\kappa = 10^{-7}$. The algorithm for the AGC [9] has a step size equal to 10^{-3} and the weights in the output layer were initialized at zero. The θ_i were randomly initialized from an uniformly distributed interval: [-1.5,1.5].

Fig. 4 shows the evaluation of DSE for those strategies in a SNR = 40 dB environment.

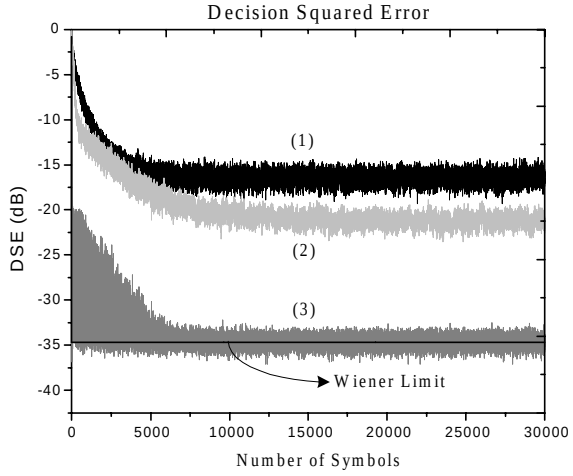

Figure 4: Decision Squared Error (SNR = 40 dB) - Minimum Phase Channel. (1)Linear Predictor (2)Nonlinear Predictor (3)CMA.

Although the nonlinear predictor outperforms the linear one, we can easily see that the CMA outperforms both of them.

4.2 Nonminimum Phase Channel

The considered channel has the following transfer funtion:

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

For the linear predictor we have used a transversal filter with 25 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 CMA has a filter with 30 coefficients and the step size equals 10^{-3}. The initialization was done in the same way that of linear predictor.

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 λ equals 10^{-4}. The number of symbols for finding the θ_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 θ_i from an uniform distribution [-2,2].

Fig. 5 shows the evaluation of DSE of those startegis in a SNR = 40 dB environment.

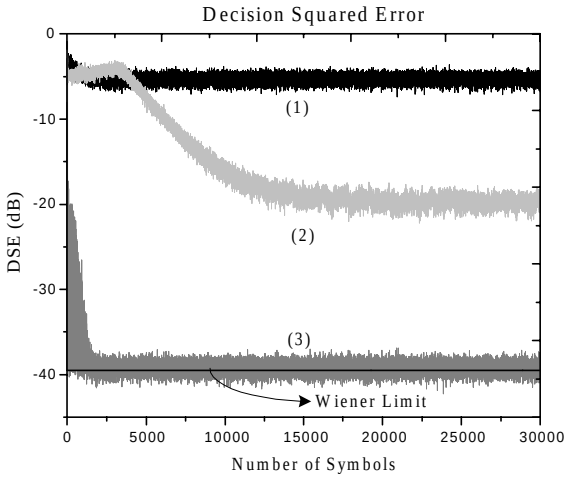

Figure 5: Decision Squared Error (SNR = 40 dB) - Nonminimum Phase Channel. (1)Linear Predictor (2)Nonlinear Predictor (3)CMA.

As expected, the linear predictor has failed in equalizing the NMPC. The nonlinear predictor has obtained better results but is outperformed by CMA.

5. 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 realizes 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 well as to avoid the local minimum found 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, where 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 κ. This improvement is actually under development.

On the other hand, its performance when compared with the CMA seems to be inferior in most cases. See for example Fig. 5, where the ANN predictive approach has higher complexity, slower convergence and worse performance.

Then, it 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.

ACKNOWLEDGEMENTS

The authors would like to thank FUNCAP and CNPq by their partial financial support.

6. REFERENCES

[1] Haykin, S., *Adaptive Filter Theory.* 3^{rd} edition. Prentice-Hall International, 1996.

[2] Benveniste, A. and M. Goursat, "Blind Equalizers", IEEE Trans. Communication, Vol. COM-32, No. 8, August 1984.

[3] Shalvi O. and E. Weinstein, "New criteria for blind deconvolution of non minimum phase systems", IEEE Trans. on Inform. Theory, Vol. 36, pp. 312-321. 1990.

[4] Montalvão J.R., J.C.M. Mota and B. Dorizzi, "Some Theoretical Limits Of Efficiency Of Linear And Nonlinear Equalizers", Journal of the Brazilian Telecommunication Society, Vol. 14, No. 2, pp. 85-92, December, 1999

[5] Mulgrew, B., "Applying Radial Basis Functions", IEEE Signal Proc. Magazine, Vol. 13, No. 2, pp. 50-65, March, 1996.

[6] Mulgrew, B., "Nonlinear Signal Processing for Adaptive Equalization and Multi-user Detection", In Proc. IX European Signal Processing Conference, EUSIPCO'98", Pp. 312.8.1-312.5.5, 1998.

[7] Oneg S., S.Choi and D. Hong, "A Decison Feedback Recurrent Neural Equalizer as an Infinite Impulse Response Filter", IEEE Trans. on Signal Processing, Vol. 45, No. 11, November, 1997.

[8] Mo S. and B. Shafai, "Blind Equalization Using Higher Order Cumulants and Neural Network", IEEE Trans. on Signal Processing, Vol. 42, No. 11, pp. 3209-3217, November, 1994.

[9] Macchi, O. "L'égalisation numérique en communications", Annales des Télécomm. 53, No 1-2, pp. 39-58,1998.

[10] Rocha C.A and O. Macchi, "A Novel Self-Learning Adaptive Recursive Equalizer with Unique Optimum for QAM", In Proceedings ICASSP-94, pp. III.481-III.484, April, 1994.

[11] Cavalcanti F.R.P. and J.C.M.Mota, "A Predictive Constant Modulus Algorithm for Blind Equalization in QAM Systems", In Proceedings ICC'97, 1997.

[12] Labat J., O. Macchi and Christophe Laot, "Adaptive Decision Feedback Equalization: Can You Skip the Training Period?", IEEE Trans. on Comm., Vol. 46, No. 7, pp. 921-930, July, 1998.

[13] Cavalcante C.C., J.C.M. Mota, B. Dorizzi and J.R. Montalvão, "Preliminary Results in Blind Equalization with Neural Network-Based Prediction", In Proc. XVII Brasilian Telecommunication Symposium, pp.508-513, Vila Velha, Brazil, 1999.

[14] Haykin, S., *Neural Networks: A Comprehensive Foundation.* 2^{nd} edition. Prentice Hall International, 1998.