Non-linear RLS-based Algorithm for Pattern Classification

Emilio Soria-Olivas [a,1], Gustavo Camps-Valls [a], José D. Martín-Guerrero [a], Javier Calpe-Maravilla [a], Joan Vila-Francés [a], and Antonio J. Serrano-López [a]

[a] *Dept. Enginyeria Electrònica. Universitat de València. València, Spain.*

Abstract

A new non-linear Recursive Least Squares (RLS) algorithm is presented in the context of pattern classification problems. The algorithm incorporates the non-linearity of the filter's output in the updating rules of the classical RLS algorithm. The proposed method yields lower stationary error levels when compared to the standard LMS and RLS algorithms in a classical application of pattern classification, such as the channel equalization problem.

Key words: Pattern classification, LMS, RLS, non-linear, filter, channel equalization.

1 Introduction

Adaptive filters are a fundamental tool in digital signal processing [1]. The most usual filter's structure is based on the Finite Impulse Response (FIR) filter. Choosing the adaptive algorithm, which describes how filter parameters are updated, becomes of crucial importance for their performance. These algorithms are usually based on the minimization of a cost function, so that the minimum corresponds to the optimum performance of the system (zero error).

Many adaptive algorithms are available in the literature. They can be grouped into two main categories or families: Least Mean Squares (LMS) algorithms

[1] Correspondence address: Prof. Emilio Soria-Olivas. Escola Tècnica Superior D'Enginyeria. Dept. Enginyeria Electrònica. Universitat de València. C/ Dr. Moliner, 50. Burjassot (València). Spain. Tel.: +34 96 3160459; Fax: +34 96 3160466. E-mail address: emilio.soria@uv.es.

and Recursive Least Squares (RLS) algorithms. The implementation of the RLS algorithm is optimized by exploiting the inversion matrix lemma and provides faster convergence and smaller error rates than the LMS algorithm [1]. However, the RLS algorithm needs a linear relationship between input and output [1]. In fact, when RLS algorithm are used for classification purposes, and a non-linear threshold function is applied to the filter's output, this function is not taken into account to obtain the adaptation equations. Thus, the system is iteratively adapted according to the error signal obtained *before* the application of the non-linearity, as shown in Fig. 1, where the solid lines show the common updating rules.

In this communication, a new approach to implement the RLS algorithm is proposed, in which filter weights are updated following dashed lines in Fig. 1. It is worth noting that RLS-based algorithms carry out a linear approximation to the non-linear function of the filter's output. This approximation has been traditionally performed by using Taylor's expansion [2–7]. The proposed procedure has the additional advantage that approximations are no longer necessary. Summarizing, this approach tackles the classification problem in a more natural way by exploiting the information content of the output signal after passing through the non-linear function.

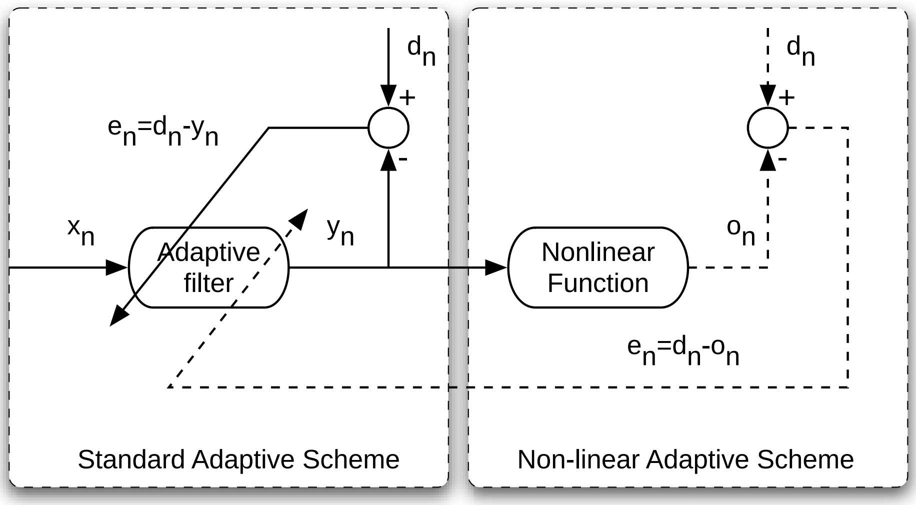

Fig. 1. Standard (solid) and proposed (dashed) weight updating rules of an adaptive scheme.

The rest of the paper is outlined as follows. Section II shows the theoretical development of the RLS-based algorithm. Section III analyzes its performance in standard channel equalization problems, in terms of accuracy, robustness to channel noise, convergence speed and stationary error levels. We end up this paper with some conclusions in Section IV.

2 Algorithm development

For the sake of simplicity, we will use vector notation. Hence, $\mathbf{x}_n = [x_n, \ldots, x_{n-L+1}]^\top$ is the input vector, $\mathbf{w}_n = [w_1, \ldots, w_L]^\top$ represents the weight vector for an FIR filter at time instant n, the superscript $^\top$ indicates the transposition operation, and L is the tap length of the adaptive filter. In the following, we will assume a binary output signal, initially coded as $\{0, 1\}$. Extension to multi-classification schemes could be easily derived by including more adaptive filters.

The proposed Non-Linear RLS (NL-RLS) algorithm is obtained by following a similar approach to that presented in [8,9]. Given an i.i.d. labeled training set $\{\mathbf{x}_n, d_n\}$, the goal is to obtain a model which maximizes the probability $\mathrm{P}(\mathbf{x}_n, d_n)$. This probability can be expressed as the product of the conditional probability $\mathrm{P}(d_n|\mathbf{x}_n)$ times $\mathrm{P}(\mathbf{x}_n)$. Since $\mathrm{P}(\mathbf{x}_n)$ is unknown, it is considered to follow a uniform distribution. In a binary classification problem, in which the desired outputs are coded as $d_n \in \{0, 1\}$, $\mathrm{P}(d_n|\mathbf{x}_n)$ follows a Bernouilli distribution [9]. Therefore, given an i.i.d. training data set, we can define the conditional probability of obtaining the desired outputs from an input vector $\mathbf{x}_n$, as follows:

$$\mathrm{P}(d_n|\mathbf{x}_n) = \prod_{k=1}^{n} o_k^{d_k} (1 - o_k)^{(1-d_k)} \tag{1}$$

where o_k is the signal obtained after applying a non-linear function at the output of the adaptive filter, $y_k = \mathbf{w}_k^\top \mathbf{x}_n$. Manipulation and posterior maximization of (1) yields the entropy cost function [9]. We can now convert this system in order to make desired and output signals to take values between $\{-1, +1\}$, as follows:

$$\mathrm{P}(d_n|\mathbf{x}_n) = \prod_{k=1}^{n} \left(\frac{1 + o_k}{2}\right)^{\left(\frac{1+d_k}{2}\right)} \left(\frac{1 - o_k}{2}\right)^{\left(\frac{1-d_k}{2}\right)}, \tag{2}$$

which can be easily re-written as

$$\mathrm{P}(d_n|\mathbf{x}_n) = \exp\left\{\sum_{k=1}^{n} \frac{d_k}{2} \ln\left(\frac{1 + o_k}{1 - o_k}\right) + \frac{1}{2} \ln\left(\frac{1 - o_k^2}{4}\right)\right\} \tag{3}$$

Now, if we define

$$y_k = \ln\left(\frac{1 + o_k}{1 - o_k}\right) \tag{4}$$

then

$$o_k = \frac{e^{y_k} - 1}{e^{y_k} + 1}. \tag{5}$$

This is an important result of the method since the sigmoid-shaped expression for o_k is obtained naturally from the re-coding step. Now, by substituting (5) into (3), one can easily obtain:

$$\mathrm{P}(d_n|\mathbf{x}_n) = \exp\left\{\sum_{k=1}^{n} \frac{d_k}{2} y_k + \frac{1}{2}(y_k - 2\ln(1 + e^{y_k}))\right\} \tag{6}$$

Note that it is more straightforward to maximize the logarithm of this function than the original function. An iterative method for this maximization is the Newton-Raphson method [10], by which

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \left(\frac{\partial^2 \ln \mathrm{P}(d_n|\mathbf{x}_n)}{\partial \mathbf{w}_n \partial \mathbf{w}_n^\top}\right)^{-1}\Bigg]_n \cdot \left(\frac{\partial \ln \mathrm{P}(d_n|\mathbf{x}_n)}{\partial \mathbf{w}_n}\right)\Bigg]_n \tag{7}$$

Easy manipulations of (6) lead to

$$\frac{\partial \ln \mathrm{P}(d_n|\mathbf{x}_n)}{\partial \mathbf{w}_n}\Bigg]_n = \sum_{k=1}^{n} \left(\frac{d_k - o_k}{2}\right) \mathbf{x}_k \tag{8}$$

and

$$\left(\frac{\partial^2 \ln \mathrm{P}(d_n|\mathbf{x}_n)}{\partial \mathbf{w}_n \partial \mathbf{w}_n^\top}\right)\Bigg]_n = -\frac{1}{2} \sum_{k=1}^{n} (1 - o_k^2) \mathbf{x}_k \mathbf{x}_k^\top \tag{9}$$

Now, if we define the signal $\mathbf{u}_k = \sqrt{1 - o_k^2}\mathbf{x}_k$ and $\mathbf{R}_n = \sum_{k=1}^{n}(1 - o_k^2)\mathbf{x}_k\mathbf{x}_k^\top$, then $\mathbf{R}_n$ can be recursively expressed as follows:

$$\mathbf{R}_{n+1} = \mathbf{R}_n + \mathbf{u}_{n+1}\mathbf{u}_{n+1}^\top, \tag{10}$$

which can be easily integrated into an RLS-based minimization procedure. In fact, it is possible to apply the matrix inversion lemma of the recursive algorithms to $\mathbf{R}_n^{-1}$ [1]. Therefore, the NL-RLS reduces to the standard RLS algorithm in which one substitutes $\mathbf{x}_k$ with $\mathbf{x}_k\sqrt{1 - \mathbf{o}_k}$, where $\mathbf{o}_k = f(\mathbf{y}_k)$ and f is the non-linear function defined in (5). Two main advantages are obtained with our proposal: (1) the new algorithm includes the non-linearity in a more natural way, and (2) no approximations are necessarily made.

(1) Initialize $\mathbf{w}_o$ and $\mathbf{P}_o = \delta^{-1}\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and δ is a small value.

(2) For $n = 1, 2, \ldots, k$

  (a) Compute output signal $y_n = \mathbf{w}_{n-1}^{\top}\mathbf{x}_n$, $o_n = (e^{y_n} - 1)/(e^{y_n} + 1)$

  (b) Compute error signal: $e_n = d_n - o_n$

  (c) Compute signal $\boldsymbol{\pi}_n = \mathbf{x}_n^{\top}\mathbf{P}_{n-1}\sqrt{1 - o_n^2}$

  (d) Compute tunable gain signal $\mathbf{k}_n = \dfrac{\mathbf{P}_{n-1}\mathbf{x}_n}{\lambda + \boldsymbol{\pi}_n\mathbf{x}_n}$, $\lambda < 1$.

  (e) Weight update: $\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{k}_n e_n$

  (f) Compute signal $\mathbf{P}'_{n-1} = \mathbf{k}_n\boldsymbol{\pi}_n$

  (g) Obtain matrix $\mathbf{P}_n = \dfrac{1}{\lambda}(\mathbf{P}_{n-1} - \mathbf{k}_n\boldsymbol{\pi}_n)$

(3) End

Fig. 2. Pseudocode procedure for the proposed recursive algorithm.

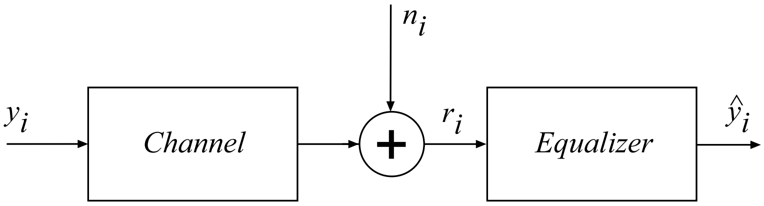

Fig. 3. *Schematic of a data transmission system.*

The proposed procedure has an obvious drawback: when the adaptive system is near the optimal system, the gradient term (8) does not equal zero since it considers all contributions from the very beginning, and thus it keeps updating for all iterations. The simplest solution to this problem is just to consider the last term in the summatory of the gradient expression. The NL-RLS algorithm is given in pseudo-code form in Fig. 2.

3 Experimental results

Despite the fact that the proposed method is generally applicable to any classification problem, we illustrate its performance in channel equalization schemes. In digital communication systems, the transmitted signal is affected by different kinds of interferences, such as power degradation and fades, multi-path time dispersions, and background thermal noise [11]. In this context, design and application of efficient adaptive algorithms that can recover the transmitted signal from the noisy received one is very important.

The system considered in this work is depicted in Fig. 3. The input to the channel is assumed to be a sequence, y_i, of independent symbols extracted from a specific alphabet. The channel output is corrupted by random noise, n_i, which is considered to be an additive white Gaussian process. The task of the equalizer is to recover the input sequence, y_{i-D}, from the received sequence $\mathbf{r}_i = \{r_i, r_{i-1}, r_{i-2}, \ldots, r_{i-L+1}\}$. The integers L and D are the order (number of taps) of the equalizer and the signal delay, respectively. In the context of binary phase shift keying (BPSK) baseband communication systems, data is binary encoded and transmitted, and hence $y_i \in \{\pm 1\}$.

In order to illustrate the capabilities of the proposed algorithm, two different channels, previously proposed in [12], are considered in the simulations. The first one is represented in z-transform as $H_1(z) = 0.35 + z^{-1} - 0.35z^{-2}$, and we used $L = 15$ and $D = 8$. This channel produces a misadjustment in the eigenvalues of the adaptive system input autocorrelation matrix of 1.45. The second channel is represented in z-transform as $H_2(z) = 0.35 + z^{-1} + 0.35z^{-2}$, where a misadjustment of 28.7 is observed. We used 3000 input symbols for training and 5000 symbols for testing the performance of the standard LMS, RLS and the proposed NL-RLS algorithm. For a given Signal-To-Noise Ratio (SNR) varied between 5 and 20 dB (in 3 dB steps), we performed 100 simulations, which represents a reasonable confidence margin for the measured Bit Error Rate (BER).

Figures 4(a) and 4(b) show the obtained BER for $H_1(z)$ and $H_2(z)$, respectively. We observe that the proposed NL-RLS shows a clear improvement in BER as compared to LMS and RLS algorithms. In Figures 4(c) and 4(d) we illustrate the evolution of the MSE for $H_1(z)$ and $H_2(z)$ in the particular case of SNR=10dB. One can observe that a lower stationary error is obtained with the proposed algorithm and with similar convergence speed to the standard RLS and higher convergence speed than the LMS.

4 Conclusions

This paper presented a non-linear RLS-based adaptive algorithm in the context of classification problems. The method reformulates the standard RLS updating rules in a straightforward way, by incorporating the information provided by the non-linear function at system's plant output. A good trade-off between error rates and distortion has been obtained in our experiments with a channel equalization problem. These results motivate us to propose it as an efficient alternative to the standard LMS or RLS algorithms. Our future work is tied to extensive analysis of the convergence and stability of the proposed algorithm.

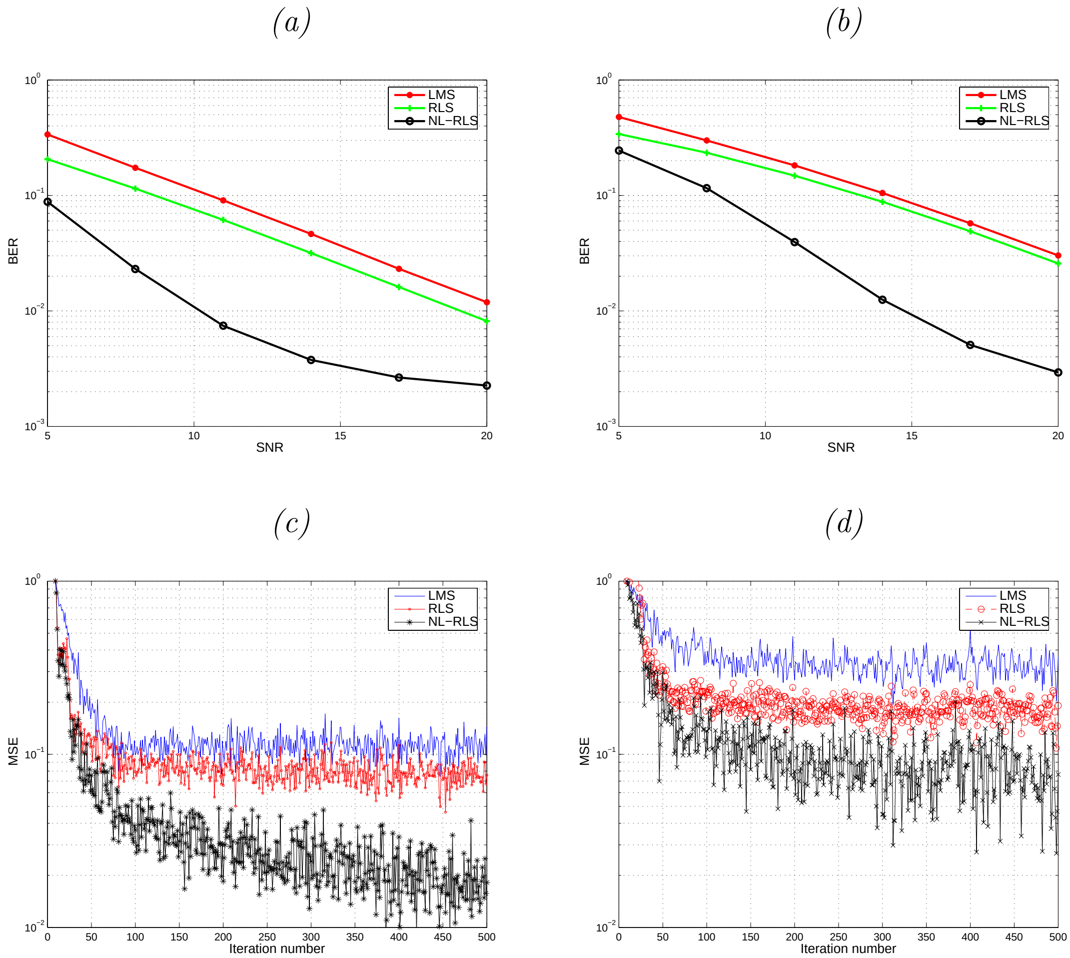

Fig. 4. **Top:** Bit Error Rate (BER) for different algorithms and channels: (a) $H_1(z)$ and (b) $H_2(z)$. **Bottom:** Evolution of the measured mean square error (MSE) as a function of the iterations for different algorithms and channels: (c) $H_1(z)$ and (d) $H_2(z)$.

References

[1] S. Haykin, Adaptive Filter Theory, Prentice Hall, Englewood Cliffs, New Jersey, 1999.

[2] M. R. Azimi-Sadjadi, S. Citrin, S. Sheedvash, Supervised learning process of multi-layer perceptron neural networks using fast least squares, in: Proc. of IEEE Inter. Conf. on Acoustics, Speech and Signal Processing (ICASSP' 90), 1990, pp. 1381–1394.

[3] S. C. Douglas, T. H. Y. Meng, Linearized least-squares training of multilayer feedforward neural networks, in: Proc. International Joint Conference on Neural Networks, 1991, pp. 307–312.

[4] R. Scalero, N. Tepedelenlioglu, A fast new algorithm for training feedforward

neural networks, IEEE Transactions on Signal Processing 40 (1) (1992) 202–210.

[5] G. J. Wang, C. C. Chen, A fast multilayer neural-network training algorithm based on thelayer-by-layer optimizing procedures, IEEE Transactions on Neural Networks 7 (3) (1996) 768–775.

[6] C. P. Chen, A rapid supervised learning neural networks for function interpolation and approximation, IEEE Transactions on Neural Networks 7 (5) (1996) 1220–1229.

[7] N. Karayiannis, N. Venetsanopoulos, Efficient learning algorithms for neural netorwks, IEEE Transactions on Systems, Man and Cybernetics 23 (5) (1993) 1372–1381.

[8] D. Rumelhart, R. Durbin, R. Golden, Y. Chauvin, Backpropagation: Theory, Architectures and Applications, Lawrence Erlbaum, Hillsdale, N.J., 1995, Ch. Backpropagation: The basic theory, pp. 1–34.

[9] C. Bishop, Neural Networks for Pattern Recognition, Clarendon Press, 1996.

[10] S. Kay, Fundamentals of Statistical Signal Processing, Prentice Hall, Englewood Cliffs, New Jersey, 1986.

[11] S. Qureshi, Adaptive Equalisation. Proc. IEEE, Vol. 73, 1985.

[12] Farhang, Adaptive Filters: Theory & Applications, Wiley & Sons, 1998.