An adaptive radius blind equalization algorithm based on pdf fitting

Malek Messai, Souhaila Fki, Thierry Chonavel, Abdeldjalil Aissa El Bey

AN ADAPTIVE RADIUS BLIND EQUALIZATION ALGORITHM BASED ON PDF FITTING

Malek Messai, Souhaila Fki, Thierry Chonavel and Abdeldjalil Aïssa-El-Bey

Institut Télécom; Télécom Bretagne; UMR CNRS 3192 Lab-STICC
Technopôle Brest Iroise CS 83818 29238 Brest, France
Université européenne de Bretagne
Email: Firstname.Lastname@telecom-bretagne.eu

ABSTRACT

In this paper, we investigate blind equalization techniques based on information theoretic criteria. They involve estimating the probability density function (pdf) of transmitted data. Our work is based on previous studies where the Parzen window method has been used to estimate the pdf at the equalizer output. The equalizer is obtained by minimizing the distance between this equalized pdf and some target distribution. With a view to reduce algorithm complexity, we propose a reduced constellation implementation of the adaptive equalizer. We show complexity and performance gain against similar approaches in the literature.

Index Terms— Blind equalization, pdf, Parzen windowing

1. INTRODUCTION

Blind equalization has been an intensive research area for several decades. It aims at developping effective and low complexity algorithms that avoid bandwidth waste resulting from training data. Several approaches have been proposed to achieve this goal. They include the Sato algorithm [1] that was the first blind technique and the Godard algorithms [2]. Among these methods, the Constant Modulus Algorithm (CMA) is probably the most popular blind equalizer technique [3]. However, it requires a long data sequence to converge and shows relatively high residual error. To overcome these limitations, several approaches have been proposed like the Modified Constant Modulus Algorithm (MCMA), that performs blind equalization and carrier phase recovery simultaneously [4], the Multi-Modulus Algorithm (MMA) that measures the errors of the real part and imaginary part of the equalizer output separately [5] and the Normalized-CMA (NCMA), that accelerates convergence by estimating the optimal step size of the algorithm at each iteration [6]. In the last decade, new techniques for blind equalization, based on information theoretic criteria and pdf estimation of transmitted data, have been proposed. These criteria are optimized adaptively, in general by means of stochastic gradient techniques. Among pdf estimation techniques, kernel methods are very popular and in particular the one that involves a Gaussian kernel, also known as Parzen window [7]. In this paper, we are interested in such a Stochastic blind equalization approach that uses the Quadratic Distance (SQD) between the pdf at the equalizer output and the known constellation pdf as a cost function [8]. SQD outperforms CMA in terms of convergence speed and residual Intersymbol Interference (ISI). However, SQD has a computational burden which increases with the order of the QAM modulation signal. Recently, an interesting algorithm, named Low Complexity SQD (LCSQD), has been introduced in [9]. It decreases the SQD complexity and outperforms it. However, at low signal-to-noise ratio (SNR), it may diverge in some cases. In this paper, we suggest to modify the LCSQD to avoid this drawback. We also propose a new method, also based on SQD, but that is more efficient than LCSQD in terms of complexity, convergence speed and residual ISI.

The rest of the paper is organized as follows. In section 2, we give an overview of the system model and we introduce the SQD and LCSQD methods. In section 3, we first supply an improved version of the LCSQD algorithm. Then, we describe the new algorithm that we propose and that we named AR-LCSQD, with AR standing for Adaptive Radius. In section 4, we illustrate the good behavior of the AR-LCSQD through simulation examples. Conclusions of our work are given in section 5.

2. SIGNAL AND EQUALIZER MODEL

2.1. Signal model

The baseband model of a transmission system with an adaptive blind channel equalizer is shown in Fig.1, where $s(n)_{n\in\mathbb{Z}}$ is the transmitted symbol at time n, that is assumed to be drawn from a QAM constellation, $\boldsymbol{h} = [h_0, h_1, ..., h_{L_h-1}]^T$ is the channel finite impulse response filter (FIR) with order L_h, while $(.)^T$ denotes the transpose operater, $b(n)_{n\in\mathbb{Z}}$ is an additive white Gaussian noise, $x(n)_{n\in\mathbb{Z}}$ is the equalizer input, $\boldsymbol{w} = [w_0, w_1, ..., w_{L_w-1}]^T$ is the equalizer impulse response,

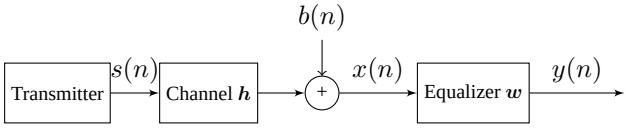


Fig. 1. Baseband model of a transmission system with an adaptive blind channel equalizer.

with length L_w and $y(n)$ is the equalized signal at time n. $x(n)$ and $y(n)$ can be modeled as

$$x(n) = \sum_{i=0}^{L_h-1} h_i s(n-i) + b(n) \quad (1)$$

$$y(n) = \sum_{i=0}^{Lw-1} w_i x(n-i) = \boldsymbol{w}^T \boldsymbol{x}(n) \quad (2)$$

where $\boldsymbol{x}(n) = [x(n), x(n-1), ..., x(n-Lw+1)]^T$.

2.2. SQD and LCSQD algorithms

2.2.1. SQD

The SQD algorithm [8] aims at minimizing the quaratic distance between the pdf of the equalizer output and the pdf of the noisy constellation. Its cost function is given by

$$J(\boldsymbol{w}) = \int_{-\infty}^{\infty} (p_{Y^P}(z) - p_{S^P}(z))^2 dz \quad (3)$$

where $Y^P = \{|y(n)|^P\}$, $S^P = \{|s_n|^P\}$, $|.|$ denotes the absolute value and $p_Z(z)$ denotes the pdf of Z at z.
By using the Parzen window method with a window involving the L previous symbols, the estimates of the current pdfs are

$$\hat{p}_{Y^P}(z) = \frac{1}{L} \sum_{i=0}^{L-1} K_\sigma(z - |y(n-i)|^P) \quad (4)$$

$$\hat{p}_{S^P}(z) = \frac{1}{N_s} \sum_{i=0}^{N_s-1} K_\sigma(z - |s_i|^P) \quad (5)$$

where K_σ is a Gaussian kernel with standard deviation σ

$$K_\sigma(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (6)$$

and N_s is the number of complex symbols in the constellation. According to [8], for $P = 2$ and a Parzen window with length $L = 1$, the expression of the cost function is given by

$$J(\boldsymbol{w}) = \frac{1}{N_s^2} \sum_{i=1}^{N_s} \sum_{j=1}^{N_s} K_\sigma(|s_j|^2 - |s_i|^2) - \frac{2}{N_s} \sum_{i=1}^{N_s} K_\sigma(|y(n)|^2 - |s_i|^2). \quad (7)$$

Then, the equalizer coefficient weights are adapted by

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) - \mu \nabla_{\boldsymbol{w}} J(\boldsymbol{w}) = \boldsymbol{w}(n) - \mu^{'} \epsilon_p \boldsymbol{x}(n)^* \quad (8)$$

where $\mu^{'} \propto \mu$ is a fixed step-size and ϵ_p is given by

$$\epsilon_p = \sum_{i=0}^{N_s-1} y(n)(|y(n)|^2 - |s_i|^2) K_\sigma(|y(n)|^2 - |s_i|^2). \quad (9)$$

2.2.2. LCSQD

In the LCSQD algorithm [9], instead of using the whole constellation as in SQD algorithm, only neighboring values of the symbol constellation pdfs are considered. These symbols are selected as shown in Fig.2. More specificaly, neighborhoods selection is performed as follows:

- If $\max(|y_r(n)|, |y_i(n)|) < A_{max} + R$, where $R > 0$ is an experimental constant, A_{max} denotes the maximum constellation amplitude, $y_r(n) = \Re\{y(n)\}$ and $y_i(n) = \Im\{y(n)\}$, then the selected points are the set:

$$S = \{s_k : \|y(n) - s_k\| < R\} \quad (10)$$

- If $\min(|y_r(n)|, |y_i(n)|) \geq A_{max} + R$, then four symbols of the constellation are selected:

$$\mathcal{S} = \left\{ \begin{array}{c} s_{k,r} = \text{sign}(y_r(n))(A_{max} - ld) \\ s_{k,i} = \text{sign}(y_i(n))(A_{max} - ld) \end{array} \right\} \begin{array}{c} l = 0, 1 \\ k = 1, .., 4 \end{array} \quad (11)$$

where, d is the minimum constellation distance.

- If $\max(|y_r(n)|, |y_i(n)|) \geq A_{max} + R$:
  - if $|y_r(n)| < A_{max} + R$, then the selected constellation points are :

$$\mathcal{S} = \left\{ \begin{array}{c} s_{k,r} : \|\text{sign}(y_r(n)) s_{k,r} - [y_r(n)]\| < R \\ s_{k,i} : \text{sign}(y_i(n))(A_{max} + ld) \end{array} \right\}_{l=-1,0} \quad (12)$$

  - if $|y_i(n)| < A_{max} + R$, then the selected constellation points are :

$$\mathcal{S} = \left\{ \begin{array}{c} s_{k,r} : \text{sign}(y_r(n))(A_{max} + ld) \\ s_{k,i} : |\text{sign}(y_i(n)) s_{k,i} - [y_i(n)]| < R \end{array} \right\}_{l=-1,0} \quad (13)$$

  where $[.]$ is the integer operator.

3. IMPROVEMENT OF THE LCSQD AND AR-LCSQD ALGORITHM

3.1. Improvement of the LCSQD algorithm

We have checked on simulations that when using large constellations such as a 64-QAM or more at low SNR, typicaly

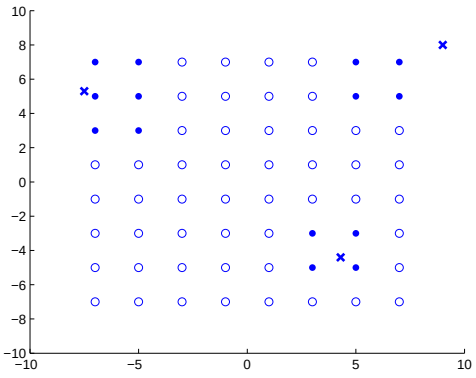


Fig. 2. Local decision regions for a 64 QAM constellation: the crosses represent values of $y(n)$ while the filled circles represent the corresponding constellation neighbors.

less than about 20 dB, we can notice a convergence problem in the LCSQD algorithm. This results from a scenario in the first test ($\max(|y_r(n)|, |y_i(n)|) < A_{max} + R$) which is omitted. We come across this scenario when we get an equalized symbol that matches the first condition and is very close to boundaries of the square limited by the lines at $\pm(A_{max}+R)$. In this case, there is no symbol in the considered constellation that satisfies the inequality $\|y(n) - s_k\| < R$. Indeed, when $y(n)$ lies in the shaded area in Fig.3, the distance between the equalized symbol to the nearest symbol of the constellation is then greater than R. The cases omitted by the LCSQD can prevent the implementation of the algorithm from converging. To solve this problem, a larger value of the radius can be used.

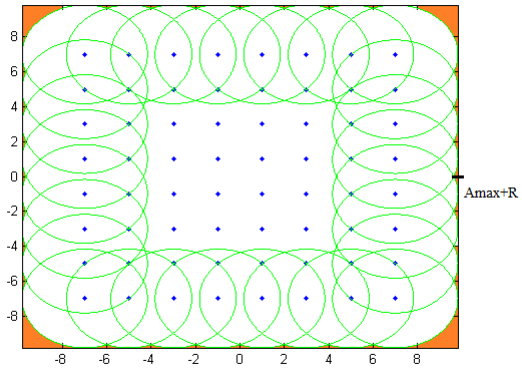


Fig. 3. Omitted areas in the LCSQD algorithm.

Then, we propose to add a subtest in the first test of the LCSQD algorithm:
If $\max(|y_r(n)|, |y_i(n)|) < A_{max} + R$, then the selected constellation points are :

$$S = \{s_k : \|y(n) - s_k\| < R\}. \quad (14)$$

If S is empty, then the selected constellation points are :

$$S = \{s_k : \|y(n) - s_k\| < \sqrt{2}R\}. \quad (15)$$

We can also use the value $\sqrt{2}R$ for both cases. But, this may decrease the performance of the algorithm in terms of residual ISI since when the equalized symbol lies within the constellation, the number of selected symbols for the equalizer updating is larger than necessary.

3.2. AR-LCSQD algorithm

Although it behaves pretty well, the main drawback of the LCSQD algorithm is that the radius R is fixed throughout the algorithm. Thus, at the beginning, when the equalizer is far from convergence, the number of constellation symbols selected to update the equalizer [9] is not enough to compute the cost function. This can slow down the convergence speed of the algorithm. Whereas, when the equalizer is very close to convergence and the error at the equalizer output becomes small, the number of constellation symbols in the updating region is unnecessarily large for the cost function calculation. This can prevent further reduction of the ISI.
To overcome this shortcoming, we propose a new approach that we name Adaptive Radius-LCSQD (AR-LCSQD). It consists in adaptively controlling the radius of the updating region. This allows for a large radius in the blind stage, that involves almost the whole constellation to reinforce convergence speed, while only keeping the nearest symbol in the constellation at convergence. Then, when convergence is achieved, we obtain a low residual ISI because our algorithm will be very similar to a decision-directed equalizer.
It can be mentioned here that with the AR-LCSQD algorithm we select, whatever the position of the equalized symbol, the points of the constellation that satisfy the following system:

$$S = \{s_k : \|y(n) - s_k\| \leq R(n)\}. \quad (16)$$

Evolution criterion for the radius of the judged region
To adapt the radius that determines the symbols of the constellation that should be used for the equalizer updating, we have employed a simple criterion based on the following dispersion function:

$$f_d = |\ \mathbb{E}\{|y(n)|^2\} - R_2\ | \quad (17)$$

where, $R_2 = \frac{\mathbb{E}\{|s_n|^4\}}{\mathbb{E}\{|s_n|^2\}}$. It was shown in [10] that f_d is equivalent, up to a constant factor, to the CMA cost function given by

$$J_{\text{CMA}} = \mathbb{E}\{(|y(n)|^2 - R_2)^2\}. \quad (18)$$

f_d characterizes the amount of intersymbol interference at the equalizer output. Thus, it takes a great value during first iterations and decreases until reaching a minimum at convergence.

This makes it a meaningful tool to adapt the radius of the AR-LCSQD algorithm. This choice is validated by simulation results in the next section. Since f_d is time varying, it is adapted iteratively using a forgetting factor λ :

$$f_d^s(n) = \lambda f_d^s(n-1) + (1-\lambda) f_d \tag{19}$$

where f_d^s refers to the smoothed value of f_d. At time n, we define a linear relationship between the instantaneous value of the radius $R(n)$ and $f_d^s(n)$ as follows :

$$\begin{aligned} R(n) &= \alpha\, f_d^s(n) + \beta \\ &= \lambda R(n-1) + (1-\lambda)(\alpha\, f_d(n) + \beta) \end{aligned} \tag{20}$$

where α and β are chosen empirically. We have tested the selected values over multiple channels, like the channels of Proakis and channels with an exponential decay profile and we checked the independence of these parameters with respect to the channel used. Then, $R(n)$ is used to determine the judged region according to Eq.(16).

4. SIMULATION RESULTS

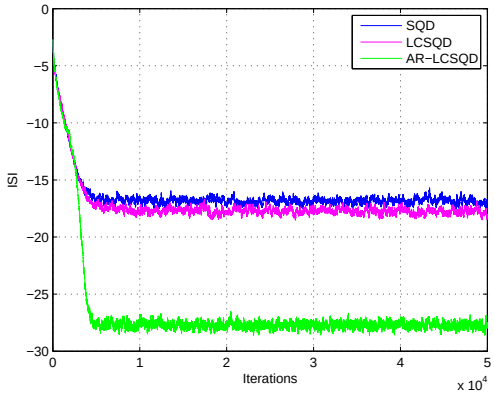
In simulations, we use one typical digital radio channel that was used in [9]:

$$\begin{aligned} H_1(z) &= (0.041 + i0.0109) + (0.0495 + i0.0123)z^{-1} \\ &+ (0.0672 + i0.017)z^{-2} + (0.0919 + i0.0235)z^{-3} \\ &+ (0.792 + i0.1281)z^{-4} + (0.396 + i0.0871)z^{-5} \\ &+ (0.2715 + i0.0498)z^{-6} + (0.2291 + i0.0414)z^{-7} \\ &+ (0.1287 + i0.0154)z^{-8} + (0.1032 + i0.0119)z^{-9} \end{aligned} \tag{21}$$

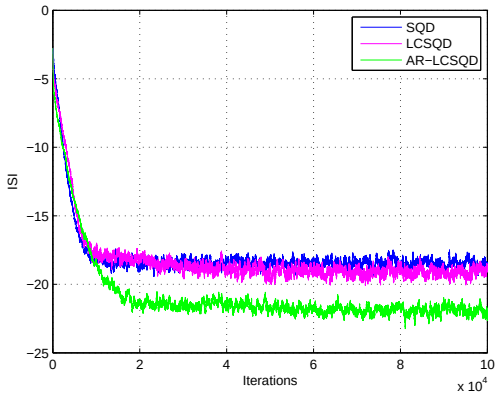
Besides, we employ an equalizer of length $L_w = 31$ initialized with a tap-centered strategy. As a measure of performance we use the ISI, defined by

$$ISI(n) = \frac{\sum_n |\boldsymbol{h} * \boldsymbol{w}(n)|^2 - max|\boldsymbol{h} * \boldsymbol{w}(n)|^2}{max|\boldsymbol{h} * \boldsymbol{w}(n)|^2} \tag{22}$$

where $\boldsymbol{h} * \boldsymbol{w}$ is the combined channel-equalizer impulse response. The kernel size, σ, of the function $K_\sigma(x)$ was updated according to [8]. To smooth the criterion function f_d for the radius adaptation, we use a forgetting factor such that $(1-\lambda)$ is equal to 5×10^{-3} and 5×10^{-2} for 16-QAM and 64-QAM modulations. In 16-QAM case, the following parameters have been employed: μ of 3×10^{-4}, 1.7×10^{-4} and 3×10^{-4} for SQD, LCSQD and AR-LCSQD respectively, α of 0.5 and β of -1.5 for AR-LCSQD and R of $2\sqrt{2}$ for LCSQD. In 64-QAM case, the following parameters have been employed: μ of 8.5×10^{-6}, 2×10^{-5} and 8×10^{-6} for SQD, LCSQD and AR-LCSQD respectively, α of 0.1 and β of -1 for AR-LCSQD and R of $2\sqrt{2}$ for LCSQD. The performance of the AR-LCSQD algorithm was compared with the SQD and LCSQD algorithms for SNR $=$ 30dB when using 16-QAM and 64-QAM modulations. The learning curves of the algorithms are presented in Fig.4. We can notice that the AR-LCSQD algorithm which we propose outperforms the SQD and LCSQD algorithms in terms of convergence speed and residual ISI. We also compare the performance of the different algorithms, in Fig.5, when using a discrete-time frequency selective channel H_2 with an exponential decay profile: $h_2(l) \sim \mathcal{N}(0, Ge^{-\rho l})$ where $l = 0, ..., L_h - 1$ and G is chosen such that $\sum_{l=0}^{L_h-1} \mathbb{E}[|h_2(l)|^2] = 1$. For simulations, we chose $\rho = 0.7$. Fig.5 also shows that AR-LCSQD outperforms the SQD and LCSQD algorithms.



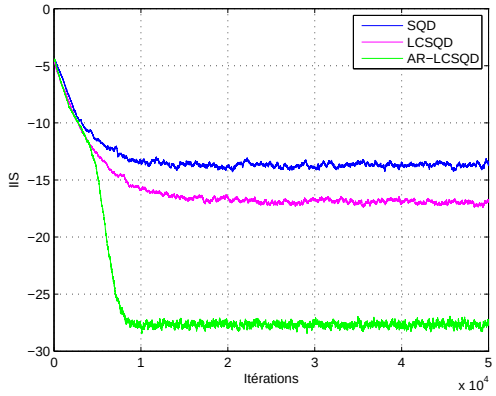
(a) 16-QAM



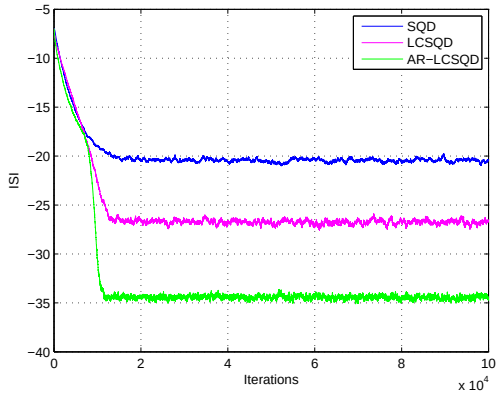
(b) 64-QAM

Fig. 4. Convergence curves of SQD, LCSQD and AR-LCSQD algorithms using H_1 for SNR$= 30dB$.

The complexity of the algorithms is factor of the number of symbols that are selected to adapt the equalizer. With the AR-LCSQD algorithm, this number decreases from one iteration to another until keeping at convergence the nearest symbol, in the constellation, to the equalized one. To measure the complexity of the algorithms, we average the total number of symbols used in the convergence stage over the iterations. Table 1 shows the complexity gain provided by the AR-LCSQD algorithm when using the channel H_1.



(a) 16-QAM



(b) 64-QAM

Fig. 5. Convergence curves for SQD, LCSQD and AR-LCSQD algorithms using H_2 for SNR$= 30dB$.

Table 1. Computation burdens for SQD, LCSQD and AR-LCSQD algorithms when using H_1. $N_s = 16$ for a 16-QAM and $N_s = 64$ for a 64-QAM.

	$\times$	Exponent	Complex $\times$
SQD	$2(N_s + 1)$	N_s	$2N_s + L_w$
LCSQD	$2(K + 1)$	$K \approx 6$	$2K + L_w$
AR-LCSQD	$2(M + 1)$	$M \approx 2$	$2M + L_w$

5. CONCLUSION

In this paper, we have modified the LCSQD algorithm and proposed a new approach for blind equalization based on pdf fitting Method. It consists in adaptive constellation implementation from adaptive serch region. This adaptive control allows AR-LCSQD to outperform the SQD and LCSQD algorithms in terms of complexity, convergence speed and residual ISI.

6. REFERENCES

[1] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems," *IEEE Transactions on Communications*, vol. 23, no. 6, pp. 679–682, June 1975.

[2] D. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867–1875, November 1980.

[3] J. Treichler and B. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 31, no. 2, pp. 459–472, April 1983.

[4] Kil Nam Oh and Y.O. Chin, "Modified constant modulus algorithm: blind equalization and carrier phase recovery algorithm," *IEEE ICC Seattle, 'Gateway to Globalization'*, vol. 1, pp. 498–502, June 1995.

[5] Kil Nam Oh and Y.O. Chin, "New blind equalization techniques based on constant modulus algorithm," *GLOBECOM*, vol. 2, pp. 865–869 vol.2, November 1995.

[6] D.L. Jones, "A normalized constant-modulus algorithm," *Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 694 –697, November 1995.

[7] C. Archambeau, M. Valle, A. Assenza, and M. Verleysen, "Assessment of probability density estimation methods: Parzen window and finite gaussian mixtures," in *International Symposium on Circuits, Systems and Proceedings*, May 2006.

[8] M. Lazaro, I. Santamaria, D. Erdogmus, K.E. Hild, C. Pantaleon, and J.C. Principe, "Stochastic blind equalization based on pdf fitting using parzen estimator," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 696–704, February 2005.

[9] C. Zhang, B. Lin, R. Liu, and B. Wang, "Low complexity blind equalization based on parzen window method," *5th International Conference on wireless Communications, Networking and Mobile Computing*, pp. 1–4, September 2009.

[10] C. Laot and N. Le Josse, "A closed-form solution for the finite length constant modulus receiver," *Information Theory. ISIT. International Symposium on Proceedings*, pp. 865–869, September 2005.