On the Convergence of Normalized Constant Modulus Algorithms for Blind Equalization

Constantinos B. Papadias e-mail: papadias@eurecom.fr
Dirk T. M. Slock e-mail: slock@eurecom.fr

Institut EURECOM, 2229 route des Crêtes, Sophia Antipolis, 06560 Valbonne, FRANCE

Abstract

One of the most known classes of algorithms for blind equalization is the so-called class of Constant Modulus Algorithms (CMA's) [1], [2]. These adaptive algorithms use an instantaneous gradient-search procedure similar to that of the LMS algorithm in order to minimize a stochastic criterion that penalizes the deviations of the received signal's modulus with respect to the known modulus of the emitted input signal. However, as has been recently reported [4], [5], [6], [7], these algorithms might ill-converge if they are not properly initialized, due to false minima of their corresponding cost function. This holds even in cases where the equalizer can match exactly the inverse of the transmitting channel [6]. Recently, a variant of CMA algorithms, the so-called Normalized CMA (NCMA) has been introduced in [8] and a more general class of normalized CMA algorithms containing NCMA as its first member has been introduced in [11]. These algorithms have a stable operation for any value of their stepsize in the (0,2) range, in contrast to unnormalized algorithms for which the range of stable stepsize values depends on the input signal's statistics and is very hard to determine. The choice of a big stepsize thus leads to a much faster convergence of normalized algorithms as compared to their unnormalized counterparts. In this paper we show that choosing a big stepsize may also help them circumvent the undesirable local minima of the algorithm's cost function thus avoiding the problem of ill-convergence.

1 Introduction

The term *blind deconvolution* is used to denote an identification procedure that estimates an unknown system using the output signal produced by this system when excited by an unknown input for which we have some information about its statistical properties. One of the most common applications of blind deconvolution is the field of data communications. In this case, one wants to estimate the impulse response of a linear transmitting filter in order to remove the inter-symbol-interference often present at the receiver end. This is usually done by implementing a tapped-delay-line (equalizer) at the receiver whose taps are updated according to an adaptive algorithm. When no training signal is used at the receiver, the procedure is called blind equalization. A simplified diagram of a blind equalization setup is shown in Figure 1.

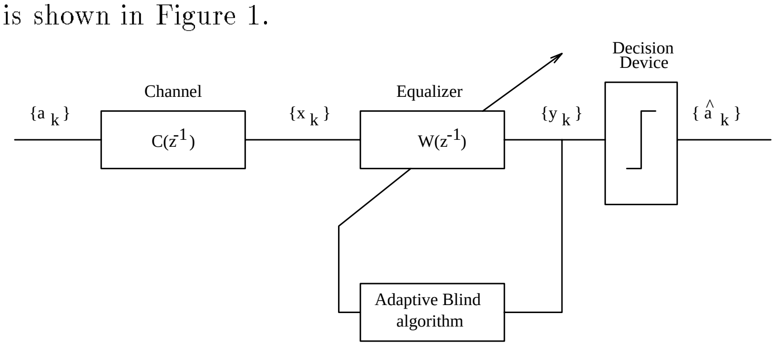

Figure 1: A typical blind equalization scheme

A very popular class of blind equalization algorithms based on the constant modulus property of the input signal are the so-called Godard blind equalizers [1], [2]. These algorithms minimize by a gradient-search procedure the following cost function:

$$J_p(W) = E\left\{\frac{1}{2p}(|y|^p - R_p)^2\right\} \qquad , \; p \in 1, 2, \dots \;, \tag{1}$$

where E denotes statistical expectation, y is the equalizer's output (which is complex in general) and R_p is a constant scalar called *dispersion constant* and defined by

$$R_p = \frac{E|a_k|^{2p}}{E|a_k|^p} \quad , \tag{2}$$

where $\{a_k\}$ is the emitted symbol sequence. The algorithm that minimizes the above cost function wih respect to W is given by:

$$W_{k+1} = W_k - \mu X_k y_k |y_k|^{p-2}(|y_k|^p - R_p) \ \ , \tag{3}$$

where W_k is a $N \times 1$ vector containing the equalizer's coefficients at time instant k, μ is the algorithm's stepsize, $X_k = [x_k \ x_{k-1} \ldots x_{k-N+1}]^H$ is a $N \times 1$ vector containing the N most recent samples of the received signal, ^{H} denotes complex conjugate transpose and the *a priori* output of the equalizer at time instant k is denoted by y_k and equals $X_k^H W_k$. The popular SATO and CMA 2-2 algorithms are special cases of (3) when $p = 1$ or $p = 2$, respectively. The stationary points of the algorithm (3) are given by a non-linear system of equations with respect to the equalizer taps. Due to this non-linearity, its cost function has not only a global minimum but also a number of local minima. Thus, some initializations of the algorithm might result in convergence to a non-desirable stationary point of the algorithm (ill-convergence). In this paper we study this same problem of ill-convergence for some recently proposed *normalized* constant modulus algorithms. The rest of the paper is organized as follows. Section 2 contains a brief review of the ill-convergence of Constant Modulus Algorithms. Normalized Constant Modulus Algorithms will be presented in section 3 and their ill-convergence will be studied in section 4. Section 5 contains some simulations that verify the analysis of the previous section. Finally, our conclusions are contained in section 6.

2 Ill-convergence of constant modulus algorithms

Consider the CMA 2-2 algorithm:

$$W_{k+1} = W_k - \mu X_k y_k (|y_k|^2 - R_2) \ . \tag{4}$$

This algorithm minimizes the cost function given by:

$$J_2(W) = E(\frac{1}{4}(|y|^2 - R_2)^2) \qquad , \ p = 1, 2, \ldots \ . \tag{5}$$

In order to find the equilibria points of this algorithm, one should set the derivative with respect to W of the cost function in (5) equal to zero. This gives: [see also [3]]

$$E((|y_k|^2 - R_2) y_k X_k) = 0 \ . \tag{6}$$

Equation (6) is actually a system of N nonlinear equations for the coefficients of the equalizer filter W. The highly non-linear character of this system of equations results in a plenitude of solutions, some of which are minima of the cost function $J_2(W)$. This is why false minima possibilities exist for the CMA algorithm. Figure 2(a) shows an example of the cost function (5) in a case where the equalizer has two taps. The cost function is plotted with respect to the axes of the two coefficients. As can be seen, apart from the two desired global minima (which correspond to two opposite optimal equalizer settings), there exists also another pair of symmetric minima that gives a non-optimal equalizer setting. It is this existence of local minima that may lead a constant modulus algorithm to ill-convergence.

3 Normalized constant modulus algorithms

In [8] a Normalized Constant Modulus Algorithm (NCMA) has been derived by nulling the CMA's *a posteriori* error at each iteration:

$$\epsilon_k = |X_k^H W_{k+1}|^2 - R_2 = 0, \quad for\ all\ k\ =\ 1, 2, \ldots\ . \tag{7}$$

The thus derived NCMA has the following form:

$$W_{k+1} = W_k - \frac{1}{\|X_k\|^2} X_k y_k (1 - \frac{1}{|y_k|}) \quad , \tag{8}$$

where $\|.\|$ denotes the 2-norm of a vector in the Euclidean space. As shown in [11], the NCMA may also be derived by minimizing exactly at each iteration the following deterministic criterion (see also [10]):

$$\min_{W_{k+1}} \left\{ \frac{1}{\|X_k\|^2} |X_k^H W_{k+1} - sign(X_k^H W_k)|^2 + (\frac{1}{\bar{\mu}} - 1)\|W_{k+1} - W_k\|^2 \right\} \quad , \tag{9}$$

where the *sign* function of a complex number z is defined as $sign(z) = sign(re^{j\phi}) = e^{j\phi}$. The algorithm that minimizes the above criterion is the following:

$$W_{k+1} = W_k - \frac{\bar{\mu}}{\|X_k\|^2} X_k y_k (1 - \frac{1}{|y_k|}) \quad , \tag{10}$$

where $\bar{\mu}$ is a stepsize controlling the algorithm's convergence speed. The algorithm is stable for all values of $\bar{\mu} \in (0, 2)$ and has a maximum convergence speed when $\bar{\mu} = 1$. The relationship between CMA and NCMA is in full analogy with the one between the LMS and NLMS algorithms [9]. This means that NCMA provides a faster convergence speed as compared to the CMA and has a stable operation for all values of $\bar{\mu} \in (0, 2)$, whereas CMA needs a careful choice (normally determined by trial and error) of its stepsize parameter in order not to diverge.
In [11], a more general class of normalized algorithms called Normalized Sliding Window CMA's (NSWCMA's) that provide a faster convergence speed has been introduced. The algorithms of this class minimize exactly at each iteration a criterion similar to the one in (9), namely,

$$\min_{W_{k+1}} \left\{ \|(\mathbf{X}_k^H W_{k+1} - sign(\mathbf{X}_k^H W_k))\|^2_{(\mathbf{X}_k^H \mathbf{X}_k)^{-1}} + (\frac{1}{\bar{\mu}} - 1)\|W_{k+1} - W_k\|^2 \right\} \quad , \tag{11}$$

where $\|v\|_S^2 = v^H S v$, $\mathbf{X}_k^H = \begin{bmatrix} X_k^H \\ X_{k-1}^H \\ . \\ . \\ X_{k-M}^H \end{bmatrix} = \begin{bmatrix} x_k & x_{k-1} & \ldots & x_{k-N+1} \\ x_{k-1} & x_{k-2} & \ldots & x_{k-N} \\ . & . & \ldots & . \\ . & . & \ldots & . \\ x_{k-M} & x_{k-M-1} & \ldots & x_{k-M-N+1} \end{bmatrix}$

is a $(M+1) \times N$ matrix of input data and the *sign* of a vector is defined as a vector whose elements are the *signs* of the respective elements of the vector. The thus obtained algorithm has the following form:

$$W_{k+1} = W_k - \bar{\mu} \mathbf{X}_k (\mathbf{X}_k^H \mathbf{X}_k)^{-1} (\mathbf{X}_k^H W_k - sign(\mathbf{X}_k^H W_k)) \ . \tag{12}$$

It is obvious that when $M = 0$ the above algorithm reduces to the NCMA. We will study in the sequel the (ill) convergence of these normalized constant modulus algorithms.

4 Ill-convergence of normalized-constant modulus algorithms

We will focus on the constant modulus algorithms algorithms described by (12). These algorithms can also be seen as stochastic gradient algorithms for the following cost function:

$$E\left\{ \|\mathbf{X}_k^H W - sign(\mathbf{X}_k^H W)\|^2_{(\mathbf{X}_k^H \mathbf{X}_k)^{-1}} \right\} \ . \tag{13}$$

As CMA's cost function (5), the cost function in (13) also has local undesirable minima. An example of such a cost function in a case of two equalizer parameters and an AR(1) noiseless transmitting channel for NSWCMA (M=1) is shown in Figure 2(b). In the following analysis we consider the

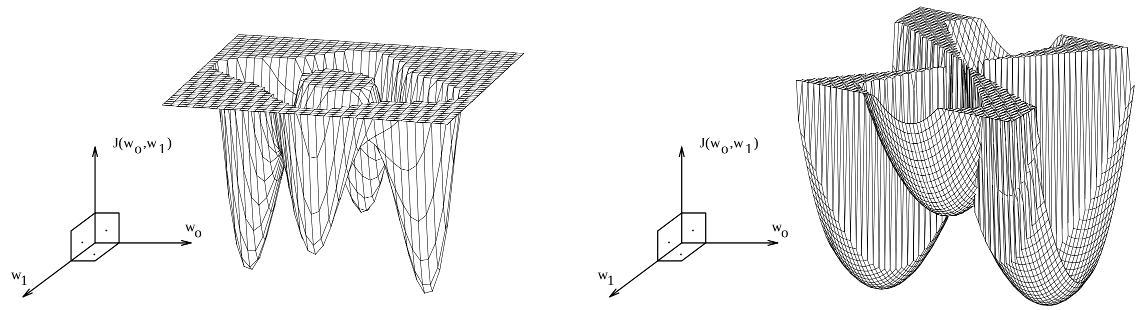

Figure 2: (a) (b)
CMA and NSWCMA (M=1) cost functions in a case of two equalizer parameters

communications channel to be modeled as an AR($N-1$) channel and the equalizer is FIR with N coefficients. Then the equalizer will be able to exactly match the channel's inverse and problems arising from the under-parameterization of the equalizer will be avoided. Also for simplicity we consider the case of a 2-PAM emitted constellation, i.e. the emitted symbol sequence is a white noise that may take on the real values 1 or -1 with equal probability:

$$Pr(a_k = 1) = Pr(a_k = -1) = \frac{1}{2} \qquad k = 1, 2, \dots \ , \tag{14}$$

where Pr denotes the probability of an event. However the results of the analysis below are extendable to other kinds of constellations such as QAM constellations [11]. The fact that the channel is AR($N-1$) is described by the following equation:

$$X_k^H W^o = a_k \ \ , k = 1, 2, \dots \ , \tag{15}$$

where W^o is a column vector that contains the AR channel's coefficients and a_k is the symbol transmitted at time instant k. According to (15) W^o is the equalizer's optimal setting. The same is true for the opposite-to-the-optimal equalizer setting W^o which is also an acceptable point of convergence since differential coding techniques can eliminate the π-phase ambiguity present in the received signal. We will now show that these two optimal stationary points are the only ones where the algorithm (in the ideal case of an AR noiseless channel) perfectly stops. Such a stationary point should satisfy:

$$\mathbf{X}_k^H W = sign(\mathbf{X}_k^H W) \quad k = 1, 2, \dots \ . \tag{16}$$

If one writes eq. (15) at $M+1$ successive time instants, one obtains:

$$\mathbf{X}_k^H W^o = [a_k \ a_{k-1} \ \cdots a_{k-M}]^T \quad k = 1, 2, \dots \ . \tag{17}$$

It is obvious from the above equation that the optimum equalizer settings $\pm W^o$ satisfy eq. (16) since $sign(a_{k-i}) = a_{k-i}$. Therefore the algorithm will exactly stop if it attains one of its two optimal settings. The question now is if it can exactly stop at another stationary point. Let us denote by H the overall input-output linear filter consisting of the cascade of the transmission channel and the equalizer and let $\{h_i\}$ be its impulse response. Then the output of the equalizer at time instant k is given by:

$$y_k = \sum_{i=-\infty}^{+\infty} h_i a_{k-i} \ . \tag{18}$$

Consider now that the equalizer's setting corresponds to a stationary point that causes the algorithm to stop exactly. Then the equalizer's output will equal $y_k = \pm 1$ and therefore, $E y_k^2 = 1 \Rightarrow$

$E\sum_{i,j} h_i h_j a_{k-i} a_{k-j} = 1$. But by definition $Ea_{k-i}a_{k-j} = \delta_{ij}$ where δ_{ij} denotes Kronecker's delta funcion. Therefore:

$$\sum_{i=-\infty}^{+\infty} h_i^2 = 1 \ . \tag{19}$$

Also, as $y_k = \pm 1$, $|y_k| = 1 \Rightarrow |\sum_{i=-\infty}^{+\infty} h_i a_{k-i}| = 1$. As this should be true *for all* possible sequences $\{a_i\}$, it should also be valid for the particular choice $a_{k-i} = sign(h_i)$, which gives:

$$\sum_{i=-\infty}^{+\infty} |h_i| = 1 \ . \tag{20}$$

Combining now (18) and (19) one gets:

$$h_i = \pm\delta_{li} \ for\ some\ integer\ l. \tag{21}$$

This last equation means that a stationary point that causes the algorithm to stop exactly will always match exactly the inverse of the transmitting channel's impulse response, except for an arbitrary time shift. Therefore even in the case of an exactly invertible noiseless channel, the algorithm never stops exactly at a local stationary point, but continues turning around it. It will only stop at a global (optimal) minimum of the cost function. This property is also true for an unnormalized algorithm, since the same steps in the above proof apply also for unnormalized CMA's.

What we just proved implies that maybe the choice of a big stepsize for a normalized algorithm could help it "escape" from a local minimum since it could amplify the amplitude of the "movement" around the false minimum and finally succeed to override it, thus avoiding ill-convergence. The simulations in the next section will show the validity of this argument.

5 Simulations

In order to provide evidence to the above claims, the following simulation was carried out: a 2-PAM input sequence was transmitted through an AR(1) noiseless channel with two coefficients $c_0 = 1$ and $c_1 = 0.25$. The channel's output is passed through an FIR equalizer (with two coefficients w_0 and w_1) that is recursively updated at each iteration by a blind-equalization algorithm. In a first experiment the CMA 2-2 algorithm is employed with a stepsize $\mu = 0.05$ which has been found by trial and error to guarantee stability. Fig. 3a shows the algorithm's trajectories after 300 iterations for fourty different initializations on a circle of radius 2. In a second experiment, the NSWCMA with M=1 was implemented, first for a stepsize $\bar{\mu} = 0.05$ and then for $\bar{\mu} = 1$ and the corresponding trajectories after

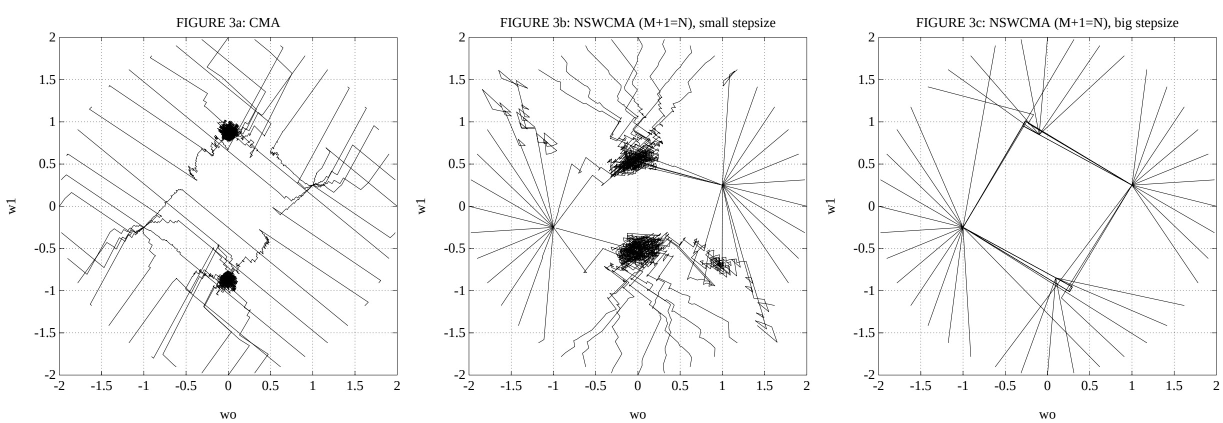

Figure 3: A comparison of CMA and NSWCMA (M=1) for an AR(1) channel

30 iterations are shown in figures 3b and 3c, respectively. As can be seen, CMA is trapped by the local minima on the w_1 axis for some initializations, and so is NSWCMA (M=1) for $\bar{\mu} = 0.05$. However,

in the case $\bar{\mu} = 1$, the latter algorithm escapes from these local minima and all its trajectories converge to the global minima $W = [w_0 \; w_1]^T = \pm[1 \; 0.25]^T$, thus avoiding ill-convergence. Another important remark about these figures is that for both algorithms the trajectories that arrive at the global minima perfectly stop there, whereas the trajectories that arrive at false minima continue on moving around them. This verifies our previous theoretical proof.

6 Conclusions

The issue of ill-convergence of normalized constant modulus algorithms has been addressed in this paper. The existence of local undesirable stationary points of these algorithms has been shown. A proof that even in a noiseless-channel case all constant modulus algorithms never actually stop at a false minimum of their cost function, but continue on turning around it, has been given. This suggested that augmenting the algorithm's stepsize would result in amplifying this motion around the undesirable stationary point and finally lead the algorithm to escape from it and converge to its optimum equalizer setting. A verification of this claim has been given via computer simulations that show that a big stepsize can help normalized algorithms to avoid ill-convergence. This gives an advantage to normalized algorithms with respect to their unnormalized counterparts, since the range of $\bar{\mu}$ for stability is known *a priori* and hence the choice of a big stepsize can be safely done without causing the algorithm to diverge.

REFERENCES

[1] D. N. Godard, "Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communications Systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 1867-1875, Nov. 1980.
[2] J. R. Treichler and B. G. Agee, "A new Approach to Multipath Correction of Constant Modulus Signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 459-472, Apr. 1983.
[3] O. Macchi and E. Eweda, "Convergence analysis of self-adaptive equalizers," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 162-176, Mar. 1984.
[4] C. R. Johnson, Jr., S. Dasgupta and W.A.Sethares, "Averaging Analysis of Local Stability of a Real Constant Modulus Algorithm Adaptive Filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 900-910, June 1988.
[5] Z. Ding, R. A. Kennedy, B. D. O. Anderson, C. R. Johnson, Jr., "Ill-Convergence of Godard Blind Equalizers in Data Communication Systems," *IEEE Trans. Commun.*, vol. 39, pp. 1313-1327, Sept. 1991.
[6] C. R. Johnson, Jr. and J. P. LeBlanc, "Toward Operational Guidelines for Memoryless-Error-Function-Style Blind Equalizers," *2nd COST 229 Workshop on Adaptive Algorithms in Communications*, Bordeaux, France, Oct. 1992.
[7] Z. Ding, C. R. Johnson, Jr. and R. A. Kennedy, "On the (Non) Existence of Undesirable Equilibria of Godard Blind Equalizers," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 40, pp. 2425-2432, Oct. 1992.
[8] K. Hilal and P. Duhamel, "A Convergence study of the Constant Modulus Algorithm Leading to a Normalized-CMA and a Block-Normalized-CMA," *Proc. EUSIPCO 92, VIth European Signal Processing Conference*, pages 135-138, Brussels, Belgium, Aug. 24-27, 1992.
[9] D. T. M. Slock, "On the Convergence Behaviour of the LMS and the NLMS Algorithms," To appear, *IEEE Trans. Acoust., Speech, Signal Processing*, submitted May 1990, revised May 1991.
[10] D. T. M. Slock, "The Block Underdetermined Covariance (BUC) Fast Transversal Filter (FTF) Algorithm for Adaptive Filtering," *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Oct. 26-28, 1992.
[11] C. B. Papadias and D. T. M. Slock, "A Normalized Sliding Window Constant Modulus Algorithm for Blind Equalization," 14^{th} *GRETSI Symposium on Signal and Image Processing*, Juan les Pins, France, Sept. 13-16, 1993.