The 7th International Telecommunications Symposium (ITS 2010)

Optimized Blind Algorithms for Widely Linear Beamforming

Adilson Chinatto[1,3]
Spectrum Line Ltda[1]
Paulínia, Brazil
chinatto@espectro-eng.com.br

Cynthia Junqueira[2,3]
Institute of Aeronautics and Space[2]
São José dos Campos, Brazil
cynthia.junqueira@iae.cta.br

João M. T. Romano[3]
School of Electrical and Computing Engineering, University of Campinas[3]
Campinas, Brazil
romano@dmo.fee.unicamp.br

***Abstract*—The use of array beamforming in modern wireless communication systems has been increasingly investigated, due to their potential in rejecting interference and improving the system capacity. The so-called blind algorithms have been considered in such application in order to avoid training procedures and improve the overall transmission throughput. Most works in the field consider circular signals and classical linear signal processing framework. However, the alternative scenario of Widely Linear Processing was shown to be more suitable under other circularity conditions, particularly in rectilinear modulations schemes. This paper deals with blind algorithms for array beamforming in a Widely Linear Processing framework. We consider the well-known CMA and NCMA algorithms and derive corresponding Widely Linear formulations with optimized performance. In addition, we show that their implementation requires less computational cost than the original CMA and NCMA. Simulation results confirm that both proposed algorithms perform better than their strictly linear counterparts.**

***Keywords*-widely linear processing; blind algorithms; CMA; NCMA; beamforming**

I. INTRODUCTION

Array beamforming techniques are considerably important in modern communication systems especially due to the recent increase in demand for wide-band mobile communication. In practical scenarios, serious communications problems are related to interference among systems and multipath fading effects. Antenna arrays using adaptive beamforming techniques can reject interfering signals and these capabilities can be exploited to improve the capacity of communication systems.

Conventionally, an adaptive array is employed with the aid of a training sequence known to both the transmitting and receiving ends. This training session, however, can be rather costly or even unrealistic in certain applications such as asynchronous wireless networks. To improve the overall throughput of a transmission system, a training period is avoided by performing blind recovery on the receiver side.

During the last decade, blind adaptive algorithm has received considerable interest for its application in mobile communication systems [1]. The constant modulus criterion [2], [3], ranks among the most widely employed methods for blind signal restoration and it can be used with constant modulus as well as with non-constant modulus such as the QAM signals. Godard [2] and Treicheler and Agee [3], based on the constant modulus criterion, introduced the well-known constant modulus algorithm (CMA), which is extremely simple to implement and converges to minima close to the Wierner beamforming. However, CMA is commonly noisy, presents slow convergence and can erroneously converge to a local minimum. To overcome those disadvantages, Hilal and Duhamel [4] developed the Normalized CMA (NCMA), which provides fast convergence using a calculated optimal step-size.

Concurrently, it has been shown that under some circularity situations the Widely Linear Processing (WLP), proposed by Brown and Crane in 1969 [5], presents enhanced performance in comparison to the usual Strictly Linear Processing (SLP) in some important problems related to array processing [6], [7], [8].

Some authors [9] address the constant modulus criterion applied to equalizers and widely linear processing showing the ability to perfect recovery of the symbols in the absence of noise introducing the widely linear version of CMA.

In the present paper, we propose suitable modifications into the NCMA in order to achieve an alternative version of this algorithm in the context of the widely linear processing (NCMA-WL). Also, we introduce optimized versions of CMA-WL and NCMA-WL, based upon restrictions on the desired signal. These modifications provide less computational cost without lose of performance.

This paper is composed of the following sections: Section II presents the mathematical basis of the widely linear processing. Section III includes a brief review of the CMA and NCMA algorithms. The proposed algorithms, including a computational cost study, are reported in section IV. Finally, section V presents simulation results and performance evaluation in the context of blind beamforming. Conclusions and future trends are stated in section VI.

II. WYDELY LINEAR PROCESSING

In [10], Pincibono and Chevalier proposed the joint use of the received signal and its complex conjugate for the optimal filter derivation. From there, the filter output is:

$$y(k) = \mathbf{w_1}^H \mathbf{x}(k) + \mathbf{w_2}^H \mathbf{x}^*(k) \tag{1}$$

where $\mathbf{w_1}$ and $\mathbf{w_2}$ are complex filters, the parameters of which are obtained minimizing the mean square error (MSE):

$$\varepsilon(k) = E\left[\left|s_D(k) - y(k)\right|^2\right], \tag{2}$$

where $s_D(k)$ is the desired signal.

From the orthogonality principle, in order to reach the minimum of cost function expressed by (2), it is necessary and sufficient that the Wiener filter coefficients are such that the error ε is orthogonal to the samples of the filter input vector, that is, orthogonal to all vector elements of $\mathbf{x}$ and $\mathbf{x}^*$. Then, $E[y^*\mathbf{x}] = E[s_D^*\mathbf{x}]$ and $E[y^*\mathbf{x}^*] = E[s_D^*\mathbf{x}^*]$.

After some algebraic manipulations, we can write:

$$\mathbf{R_{xx}}\,\mathbf{w_1} + \mathbf{C_{xx}}\,\mathbf{w_2} = \mathbf{r} \quad (3)$$

$$\mathbf{C_{xx}}\,\mathbf{w_1} + \mathbf{R_{xx}}^*\,\mathbf{w_2} = \mathbf{z} \quad (4)$$

where $\mathbf{C_{xx}} = E[\mathbf{xx}^T]$, $\mathbf{r} = E[s_D^*\mathbf{x}]$ and $\mathbf{z} = E[s_D\mathbf{x}]$. The optimal solution for the widely linear weights is:

$$\mathbf{w_1} = [\mathbf{R_{xx}} - \mathbf{C_{xx}}(\mathbf{R_{xx}}^*)^{-1}\mathbf{C_{xx}}^*]^{-1}[\mathbf{r} - \mathbf{C_{xx}}(\mathbf{R_{xx}}^{-1})^*\mathbf{z}^*]$$
$$\mathbf{w_2} = [\mathbf{R_{xx}}^* - \mathbf{C_{xx}}^*\mathbf{R_{xx}}^{-1}\mathbf{C_{xx}}^*]^{-1}[\mathbf{z}^* - \mathbf{C_{xx}}^*\mathbf{R_{xx}}^{-1}\mathbf{r}]. \quad (5)$$

III. CONSTANT MODULUS ALGORITHM AND ITS NORMALIZED VERSION

The Godard proposal [2], showed the heuristic weight optimization $\mathbf{w}$ as a way to maintain the constant modulus property of the transmitted data. From this technique, Treicheler and Agee [3] created the Constant Modulus Algorithm (CMA) for the modulations schemes as FM, PSK, GMSK whose the update equation is:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu([|y(k)|^2 - 1]y^*(k)\mathbf{x}(k) \quad (6)$$

where μ is the step-size.

The Godard criterion penalizes the output samples $y(k)$ that do not have the constant modulus characteristics. To prevent the update weights from diverging and to make the algorithm more stable the Normalized-CMA (NCMA) was introduced by Hilal and Duhamel [4] in order to maximize de convergence speed of the Godard algorithm leading to the NCMA update equation:

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \frac{1}{\|\mathbf{x}(k)\|_2^2}\mathbf{x}(k)y^*(k)(1 - \frac{1}{|y(k)|}) \cdot \quad (7)$$

In this paper, for the sake of clearness, NCMA algorithm will be treated as Strictly Linear Normalized Constant Modulus Algorithm (NCMA-SL).

IV. PROPOSED ALGORITHMS

In this section, the main idea is to present modifications of the original CMA [3] and NCMA [4]. Firstly, we will deal with the generalized versions of CMA and NCMA in their widely linear sense. Following, the optimized versions of the algorithms for rectilinear signals are introduced, leading to Opt-CMA-R and Opt-NCMA-R algorithms. Finally, a brief discussion about the computational cost of these algorithms is performed.

A. Generalized Version of CMA-WL and NCMA-W

In order to propose the widely linear formulation of the CMA and NCMA algorithm, (1) can be re written in a synthetic form as

$$y(k) = \tilde{\mathbf{w}}^H\tilde{\mathbf{x}} \quad (8)$$

where $\tilde{\mathbf{x}} = [\mathbf{x}, \mathbf{x}^*]^T$ is the new input vector and $\tilde{\mathbf{w}} = [\mathbf{w}_1, \mathbf{w}_2]^T$ is the widely linear weights vector.

Considering the new input vector in (6) and (7), with some algebraic manipulations, it is possible to reach the Widely Linear Constant Modulus Algorithm (CMA-WL)

$$\tilde{\mathbf{w}}(k+1) = \tilde{\mathbf{w}}(k) - \mu([|y(k)|^2 - 1]y^*(k)\tilde{\mathbf{x}}(k) \quad (9)$$

and the Widely Linear Normalized Constant Modulus Algorithm (NCMA-WL) adaptation equations

$$\tilde{\mathbf{w}}(k+1) = \tilde{\mathbf{w}}(k) + \frac{\mu}{\|\tilde{\mathbf{x}}(k)\|^2}\tilde{\mathbf{x}}(k)\left[1 - \frac{1}{|y(k)|}\right]y^*(k) \cdot \quad (10)$$

B. Optimized CMA-WL and NCMA-WL for Rectilinear Desired Signal

In the widely linear processing, when the desired signal is real, the terms $\mathbf{r}$ and $\mathbf{z}$ in (5) have the same value, implying that $\mathbf{w_1} = \mathbf{w_2} = \mathbf{w}_{WL}$, and consequently

$$y(k) = 2\Re(\mathbf{w}_{WL}^H\mathbf{x}) \quad (11)$$

where $\Re$ is the real operator.

Equation (11) can be written as

$$y(k) = 2(\mathbf{w}_R^T\mathbf{x}_R + \mathbf{w}_I^T\mathbf{x}_I) \quad (12)$$

where $\mathbf{w}_R = \Re(\mathbf{w}_{WL})$, $\mathbf{w}_I = \Im(\mathbf{w}_{WL})$, $\mathbf{x}_R = \Re(\mathbf{x})$ $\mathbf{x}_I = \Im(\mathbf{x})$

and $\Im$ is the imaginary operator.

Using these assumptions it is possible to derive the optimized version of the widely linear constant modulus algorithm (Opt-CMA-R) and of the widely linear normalized constant modulus algorithm (Opt-NCMA-R) for rectilinear signals.

1) Opt-CMA-R

Using the CMA cost function definition [3]

$$J_{CMA} = (|y|^2 - 1)^2 \quad (13)$$

it can be verified that the gradient of J_{CMA} with respect to the weight vector $\mathbf{w}_{WL}$ is given by

$$\nabla J_{CMA} = 2(|y|^2 - 1)(2\mathbf{xx}^H\mathbf{w} + \mathbf{xx}^T\mathbf{w}^*) = 2(|y|^2 - 1)(2\mathbf{x}y^* + \mathbf{x}y). \quad (14)$$

Considering $y^* = y$, y real,

$$\nabla J_{CMA} = 6(|y|^2 - 1)y\mathbf{x} \,. \quad (15)$$

So, the updating equation becomes

$$\mathbf{w}(k+1) = \mathbf{w}(k) - 6\mu([|y(k)|^2 - 1]y(k)\mathbf{x}(k) \quad (16)$$

As μ is generally an arbitrated value, highly dependent of the application, we can consider the Opt-CMA-R update equation as

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \mu_R([|y(k)|^2 - 1]y(k)\mathbf{x}(k) \quad (17)$$

where μ_R is the step-size.

2) Opt-NCMA-R

According to [4] and considering (11), in order to determine an optimal value to the step-size in the CMA update equation (6) it is necessary to define two types of errors:

- The *a priori* error provided at time *k* by the previous filter

$$e_1(k) = |y(k)|^2 - 1 = |\mathbf{w}_{WL}^H(k)\mathbf{x}(k) + \mathbf{x}^H(k)\mathbf{w}_{WL}(k)|^2 - 1 \quad (18)$$

- The *a posteriori* error provided at time *k* by the new filter

$$e_2(k) = |\mathbf{w}_{WL}^H(k+1)\mathbf{x}(k) + \mathbf{x}^H(k)\mathbf{w}_{WL}(k+1)|^2 - 1 \quad (19)$$

The objective is to find the step-size μ_{opt} for which

$$e_2(k) = 0, \ \forall k \quad (20)$$

Rewriting (6) in terms of $e_2(k)$, the condition (20) implies a modified expression for the optimized step-size, expressed by

$$\mu_{opt}^2 E^2(k)(y(k) + y^*(k))^2$$
$$-2\mu_{opt}E(k)y(k)(y(k) + y^*(k)) + 1 = 0 \quad (21)$$

where

$$E(k) = \mathbf{x}^H(k)\mathbf{x}(k) = \|\mathbf{x}(k)\|_2^2 \ . \quad (22)$$

Solving (21) in μ_{opt} the optimal value of the step-size is

$$\mu_{opt} = \frac{|y(k)| - 1}{2E(k)y(k)(|y(k)|^2 - 1)} \quad (23)$$

By substituting (23) in (6) we get the update expression for Opt-NCMA-R

$$\mathbf{w}_{WL}(k+1) = \mathbf{w}_{WL}(k) - \frac{1}{2\|\mathbf{x}(k)\|_2^2}\mathbf{x}(k)y(k)(1 - \frac{1}{|y(k)|}) \quad (24)$$

where $y(k)$ is given by (11).

C. Computational Cost study

As far as the computational effort is concerned, different ways have been performed with the only purpose of evaluating the performance of the code. Several computational cost indicators are available in the literature, and the most usual are related to code execution time (software) or number of internal operations (hardware). Here, the choice was do this evaluation based on utilized hardware resources, measuring the computational cost through the number of operations: sum and multiplications of real numbers, inversions and trigonometric operations (i.e., sine, cosines, tangent, etc. and the inverses).

The computational cost is directly related with the number of the taps of the filter $\mathbf{w}_{WL}$. Considering that $\mathbf{w}_{WL}$ has *M* taps, Table I shows the computational cost of the presented algorithms.

According to the data presented in Table I, we can observe that the proposed Opt-CMA-R algorithm requires less computational resources than its counterpart CMA-SL. Precisely, the optimized algorithm needs only half of the multiplications and sums used in the strictly linear algorithm.

In the case of NCMA-SL and Opt-NCMA-R, a slight reduction in multiplications and sums could be noted. However, a large impact in the computational cost comes from the reduction in the number of inversions and trigonometric operations.

A well-known fact about the implementation of trigonometric functions in hardware is the need of use of recursive devices as CORDIC (COordinate Rotation DIgital Computer) [11], which implement this kind of function based on vectoring and rotation operations, or by mapping these functions into memories like Look-up Tables [12]. CORDIC devices, in addition to require large amounts of embedded logic, also demand fairly long run times, which can derail their use in systems operating at high symbol rates. Look-up Tables, on the other hand, imply the use of memories, which also involve a large consumption of logic.

V. Simulation Results

The objective of this set of simulations is to compare the performance of the proposed algorithms (Opt-CMA-R and Opt-NCMA-R) with their strictly linear counterparts (CMA-SL and NCMA-SL). The comparison between the algorithms Opt-CMA-R and CMA-SL was named Case Study 1, whereas the comparison between Opt-NCMA-R and NCMA- SL, Case Study 2.

For both cases studies it was considered an array of uniformly spaced linear antennas (ULA) composed of $M = 2$ omnidirecional elements spaced of half a wavelength. Three signals impinge the array: one desired signal (s_D) and two interfering signals (s_1 and s_2). This setup leads to the array to operate on a under parameterized manner, i.e. with more incident signals than antennas.

Table II summarizes the signals characteristics.

TABLE I: Computational cost comparison

Algorithm	Multiplications	Sum	Inversions	Trig.
CMA-SL	8*M*+6	8*M*	0	0
Opt-CMA-R	4*M*+3	4*M*	0	0
NCMA-SL	8*M*+8	7*M*+1	4	3
Opt-NCMA-R	5*M*+1	5*M*+1	1	0

TABLE II: SIGNAL CHARACTERISTICS FOR SIMULATIONS

Signal	DOA (°)	Modulation	SNR (dB)
s_D	– 90 to +90 $\Delta_{DOA} = 1$	BPSK	10
s_I	– 45	8-PAM	10
s_2	+ 45	8-PAM	10

In order to represent an I/Q imbalance over the signal s_2, we consider this signal formed by the following expression:

$$s_2 = \frac{A}{\left|\sqrt{(C-1)^2+1}\right|}\left(S_I - j(C_i-1)S_Q\right) \quad (25)$$

where A is the signal amplitude, C is the circularity coefficient, S_I and S_Q are the in-phase and quadrature component, respectively, with $0 \le S_I \le 1$ and $0 \le S_Q \le 1$. In this way, s_D and s_1 are rectilinear and s_2 changes from rectilinear to circular over the simulations. The parameters used to evaluate both case studies were the equivalent SNR and the gain. The equivalent SNR was calculated using the inverse of function Q, which is defined by

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{\left(-t^2/2\right)}dt \,, \quad (26)$$

and knowing that for an AWGN channel

$$SER = Q(\sqrt{2SNR})\,. \quad (27)$$

Finally, the gain is obtained by applying the following expression

$$G = 10\log\left(\frac{SNR_{OPT}}{SNR_{SL}}\right) \quad (28)$$

where SNR_{OPT} is the equivalent signal to noise ratio achieved at the output of the Opt-CMA-R or Opt-NCMA-R algorithms, and SNR_{SL} is applied to the strictly linear ones.

A. Case Study 1: Opt-CMA-R

For this Case Study, the step-size value was chosen taking into account the best results achieved for both CMA-SL and Opt-CMA-R. The considered value was $\mu_R = 0.0033$.

Fig. 1 shows the equivalent SNR for s_D at the output of Opt-CMA-R. The algorithm provides efficient interference mitigation for s_D DOA changing from –20^o to +20^o, where equivalent SNR reaches +10dB, which is the specified SNR at the input data at the simulations.

For the other DOA's the equivalent SNR fell below –10dB, which is a value underneath the noise level, i.e. showing an error situation. Those occurrences are independent on the s_2 circularity coefficient.

Fig. 2 represents the gain as defined in (28). In this case, despite the s_2 circularity, Opt-CMA-R exhibits at least a 3 dB gain over CMA-SL performance. Moreover for s_D DOA between –20^o to +20^o, Opt-CMA-R presents a gain of 40dB related to CMA-SL. It should be noted that the Opt-CMA-R reaches this gain performance using less computational resources than CMA-SL, as was punctuated in section IV.

B. Case Study 2: Opt-NCMA-R

The Case Study 2 compares the performances of NCMA-SL and Opt-NCMA-R algorithms. Fig. 3 and Fig. 4 show the equivalent SNR for s_D and the gain respectively.

As can be seen in Fig. 3, although applied to an under-parameterized array, Opt-NCMA-R provides efficient interference mitigation for s_D DOA changing from –20^o to +20^o. Within that range equivalent SNR reaches +10dB, in accordance to the specified SNR at Table II.

When s_D DOA lays around –45^o or +45^o, the equivalent SNR fell below –50dB, that is a full error situation. This fact occurs due to the constellation superposition and it is independent of the s_2 circularity coefficient.

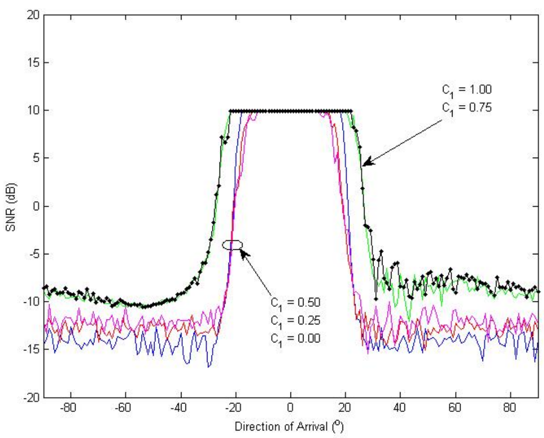

Figure 1. Equivalent SNR for Opt-CMA-R

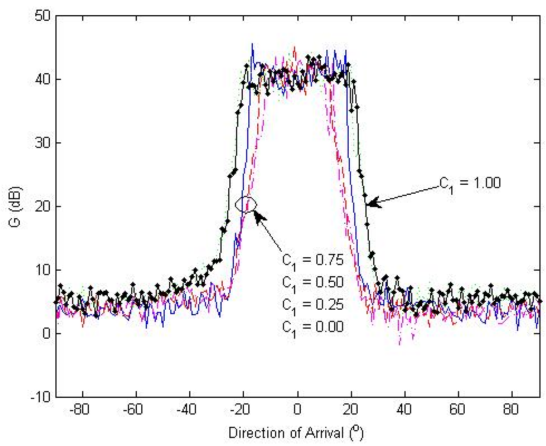

Figure 2. Opt-CMA-R gain over CMA-SL

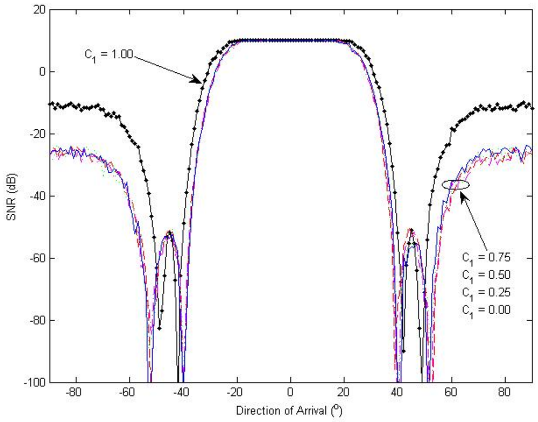

Figure 3. Equivalent SNR for Opt-NCMA-R

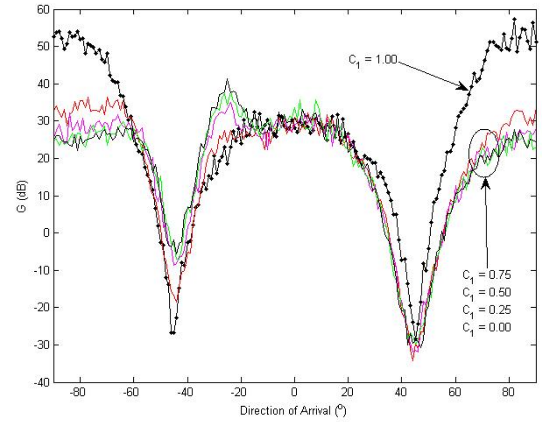

Figure 4. Opt-NCMA-R gain over NCMA-SL

The results of Opt-NCMA-R gain over NCMA-SL are presented in Fig. 4. In this case, for s_D DOA others than those where constellation superposition occurs, when s_2 circularity is unitary (C_1 = 1), Opt-NCMA-R clearly exhibits a better performance than NCMA-SL. As done before with in the Opt-CMA-R, it is important to highlight that the Opt-NCMA-R reach this gain performance with less computational operations than NCMA-SL, as was mentioned in section IV.

VI. CONCLUSIONS

In this work we proposed modifications on adaptive blind algorithms CMA and NCMA in order to optimize their performances in cases where the involved signals are real. For this, we considered the application of the widely linear processing techniques and, based on the real characteristic of the signal, derived the optimal forms of those algorithms for rectilinear signals, leading to the new blind algorithms Opt-CMA-R and Opt-NCMA-R.

Analysis of the adaptation equations of the proposed algorithms allow to conclude that their implementation requires less computational cost, i.e., for real signals, using less hardware resources, these algorithm give better performance than that provided by the original CMA and NCMA.

In order to compare the performances of the proposed Opt-CMA-R and Opt-NCMA-R algorithms with the original CMA and NCMA, we presented a set of simulations where the algorithms were applied at the output of an underparameterized array whose objective was to provide interference mitigation. It was shown that the both proposed algorithms performed better than their strictly linear counterparts.

As future perspectives, the proposed algorithms Opt-CMA-R and Opt-NCMA-R should be implemented in a real test platform in order to explore their capability on equalization and beamforming. Such effective hardware implementation must reveal the real computational cost reduction of the Opt-CMA-R and Opt-NCMA-R algorithms when compared with the original CMA and NCMA.

REFERENCES

[1] S. Haykin (Eds.), Unsupervised Adaptive Filtering, John Willey & Sons, NY, 2000.

[2] D.N.Godard, "Self-recovering equalization and carrier tracking in a two-dimensional data communication system" *IEEE Trans. Communications*, vol.28, pp. 1867-1875, November. 1980.

[3] J. R. Treichler and B. G. Agee, " A new approach to multipath correction of constant modulus signals" *IEEE Trans. Acoustics, Speech,Signal Processing*, vol. 31, pp. 459-471, Apr. 1983.

[4] K. Hilal, P.Duhamel, "A convergence study of the constant modulus algorithm leading to a normalized-CMA and a block-normalized-CMA" Signal Processing VI: Theories and Applications, pp. 135-138, 1992.

[5] W. M. Brown and R. B. Crane, "Conjugate Linear Filtering", IEEE Trans. On communications, 51(1), pp. 37-42, Jan. 2003.

[6] P. Chevalier and A. Blin, "Widely linear MVDR Beamformers for the reception of an unknown Signal Corrupted by Noncircular Interferences," *IEEE Trans on Signal Processing,* vol. 55, n. 11, pp. 5323–53365, Nov. 2007.

[7] A. Chinatto, C. Junqueira, J.M.T. Romano " Interference mitigation using widely linear arrays", 17th European Signal Processing Conference, Eusipco 2009, Aug. 2009.

[8] P. Chevalier and F. Pipon, "New Insights Into Optimal Widely Linear Array Receivers for the Demodulation of BPSK, MSK, and GMSK Signals Corrupted by Noncircular Interferences — Application to SAIC," *IEEE Trans on Signal Processing,* vol. 54, n. 3, pp. 11–25, Mar. 2006

[9] A.S. Cacciapuoti, G. Gelli, L. Paura and F. Verde Widely-linear fractionally-spaced blind equalization of Frequency-selective channels, 14th European Signal Processing Coference, EUSIPCO 2006, Sep. 2006

[10] P. Chevalier and B. Picinbono, "Widely Linear Estimation with Complex Data," *IEEE Trans on Signal Processing,* vol. 43, n. 8, pp. 2030–2033, Aug. 1995.

[11] J. E. Volder, "The CORDIC trigonometric computing technique", *IEEE Trans. on Electron. Comput.*, vol. EC-8, pp. 330-334, 1959.

[12] A Xilinx Inc, "Using Block RAM in Spartan-3 Generation FPGAs", on http://www.xilinx.com/support/documetion/application_notes/xapp463.pdf, Mar. 2005.