An efficient algorithm for Decision Feedback Blind Equalization of QAM Signals

João Mendes Filho, Maria D. Miranda & Magno T. M. Silva

***Abstract*— We propose a symbol-based decision algorithm for blind equalization of quadrature amplitude modulation signals. It jointly update the feedforward and feedback filters of a decision feedback equalizer and performs similarly to a supervised adaptive algorithm, such as the normalized least mean-squares algorithm. Besides presenting strategies to speed up its convergence, we provide sufficient conditions for its stability. Its good behavior is illustrated through simulation results.**

***Index Terms*— Adaptive filters, blind equalization, constant modulus algorithm, decision-directed algorithm, decision feedback equalizer, quadrature amplitude modulation.**

I. Introduction

Modern communication systems must deliver high amounts of data within short time intervals, pursuing low symbol error rates (SER), and also considering different environment conditions. To accomplish such objective, many modulation schemes were proposed in the literature, as is the case of high-order quadrature amplitude modulation (QAM), which uses the available bandwidth in an efficient manner [1], [2], [3]. The performance of a communication system that employs high-order QAM signals strongly depends on the signal-to-ratio (SNR), as we can observe in Fig. 1. This figure shows SER curves as a function of SNR for an additive white Gaussian noise (AWGN) channel and assuming different QAM orders. To achieve acceptable symbol error rates, the higher the QAM order is, the higher the signal-to-noise ratio must be.

In practical situations, the channel is noisy and dispersive, which demands an efficient equalizer to mitigate the effect of intersymbol interference, mainly when high-order QAM signals are used in single carrier systems. Optimum equalizers are usually developed from the perspective of known channel characteristics and have high computational cost, as is the case of the maximum likelihood sequence estimation (MLSE) method [4]. On the other hand, solutions based on the linear transversal equalizer (LTE) are relatively simple, but perform poorly in difficult environments such as channels with long and sparse impulse response, non-minimum phase, spectral nulls or nonlinearities [5], [6], [7], [8].

In this context, decision feedback equalizers (DFEs) present a favorable tradeoff between computational cost and efficient behavior, independently of the channel type. Besides employing QAM, the use of bandwidth can be improved if DFEs are blindly adapted. When the feedforward and feedback filters of a DFE are jointly adapted by blind algorithms, as is the case of algorithms based on the constant-modulus cost function [9], certain actions must be taken. This is due to the fact that constant-modulus-based algorithms may converge to so-called degenerative solutions, which occur when the signal at the equalizer output is independent of its input. This problem was addressed in [6], where modifications in the constant modulus criterion were proposed to avoid such undesired solutions, leading to a stochastic algorithm named DFE-CMA-FB (constant modulus algorithm for adaptation of DFE with constraint in the feedback filter). Although DFE-CMA-FB avoids degenerative solutions, it still presents some inherent drawbacks of constant-modulus-based algorithms as the impossibility of solving phase ambiguities introduced by the channel and a relatively large misadjustment when used to recover nonconstant modulus signals, as is the case of high-order QAM signals (see, e.g., [10] and its references). The phase rotation can be avoided by using, for example, the phase tracking algorithm as in [6] or the philosophy of the multimodulus algorithm (MMA), which minimizes the dispersion of the real and imaginary parts of the equalizer output separately [11], [12]. The MMA approach does not reduce significantly the misadjustment of DFE-CMA-FB since its updating error is zero only when the equalizer output is zero or when its magnitude is equal to the square root of the dispersion constant.

To reduce the misadjustment of DFE-CMA-FB, [13] and

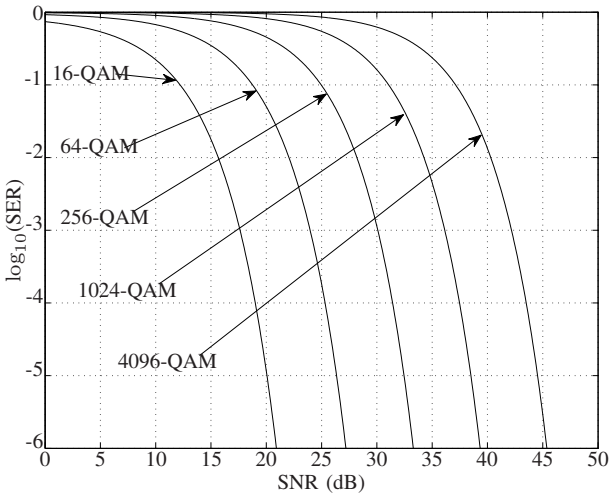

Fig. 1. Logarithm of SER as function of SNR (dB) for 16, 64, 256, 1024 and 4096-QAM, and assuming an AWGN channel.

Manuscrito recebido em 30 de setembro de 2011; revisado em 10 de outubro de 2011.

João Mendes Filho (jmendes@lps.usp.br), Maria D. Miranda (maria@lcs.poli.usp.br) e Magno T. M. Silva (magno@lps.usp.br) are with Escola Politécnica, University of São Paulo. Av. Prof. Luciano Gualberto, 158, trav. 3, Butantã, São Paulo - SP - Brazil - 05508-010.

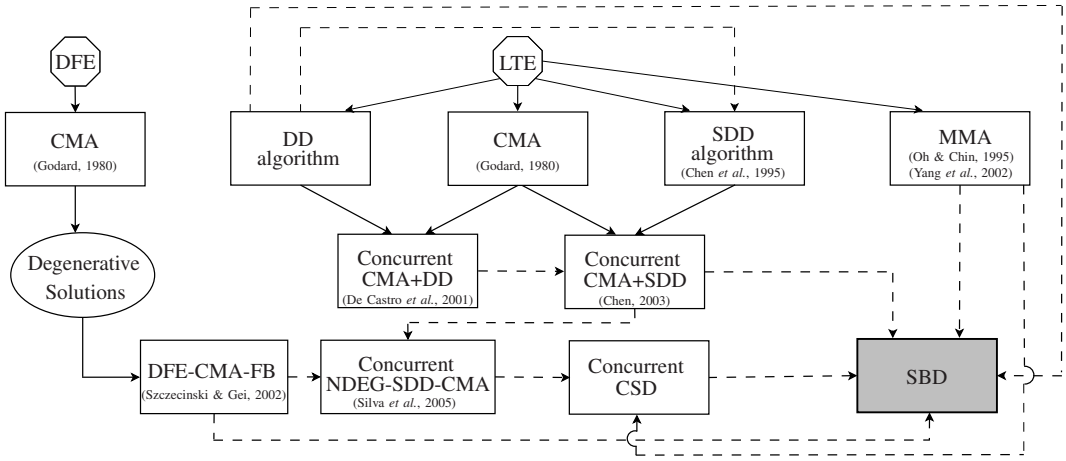

Fig. 2. Relations among blind algorithms for equalization of QAM signals.

[14] proposed to operate it concurrently with the soft decision-directed algorithm (SDD) [15] for equalization of QAM signals. The resulting concurrent algorithm, denoted NDEG-SDD-CMA (Non-Degenerative SDD-CMA), presents an improvement in equalization performance over DFE-CMA-FB at the cost of a moderate increase in computational complexity. To the best of our knowledge, the first concurrent algorithm for blind equalization of QAM signals was proposed in [16] and later improved in [17], both for the adaptation of linear transversal equalizers. In [14], the algorithm of [17] was extended to the blind adaptation of DFEs, taking into account the criterion of [6] in order to avoid degenerative solutions.

In this paper, inspired by the NDEG-SDD-CMA characteristics and using the MMA approach, we introduce the symbol-based decision (SBD) algorithm, which can be interpreted as an extension of the decision-directed algorithm (DD) for blind equalization of QAM signals. In a similar manner, it avoids degenerative solutions if the criterion of [6] is adopted. For sake of comparison, we also consider the MMA approach for NDEG-SDD-CMA, which is referred hereinafter to as the concurrent soft-decision (CSD) algorithm. A summary of the relations among the algorithms is shown in Fig. 2, where the dashed lines indicate that an algorithm was used as an inspiration source.

The paper is organized as follows. In Section II, we present the problem formulation. Then, we explore different error functions and propose the SBD algorithm in Section III. In Section IV, we discuss the convergence and stability of the proposed algorithm. Finally, in sections V and VI, we present the simulations and conclusions, respectively.

II. Problem formulation

We assume a fractionally decision feedback equalizer ($T/2$-DFE) as shown in Fig. 3, due to its inherent advantages (see, e.g., [18], [19], [6] and their references). The independent and identically distributed (i.i.d.) and non-Gaussian signal $a(n)$ is transmitted through an unknown communication channel, modeled by the impulse response vectors

$$\mathbf{h}_{\mathrm{e}} = [\, h_0 \; h_2 \; \cdots \; h_{2N-2} \,]^T$$

and

$$\mathbf{h}_{\mathrm{o}} = [\, h_1 \; h_3 \; \cdots \; h_{2N-1} \,]^T,$$

and additive white Gaussian noise (AWGN), denoted as $\eta_{\mathrm{e}}(n)$ and $\eta_{\mathrm{o}}(n)$. The superscript T stands for transposition and $h_0, h_1, \cdots, h_{2N-1}$ are samples of a continuous-time channel model, sampled with twice the symbol rate. The signals $x_{\mathrm{e}}(n)$ and $x_{\mathrm{o}}(n)$ are distorted versions of the transmitted signal, due to the effects of the intersymbol interference and of the additive white Gaussian noise. These signals are filtered by finite impulse response (FIR) filters ($\mathbf{w}_{\mathrm{fe}}$ and $\mathbf{w}_{\mathrm{fo}}$), each one with $M_{\mathrm{f}}/2$ coefficients, forming the oversampled feedforward filter, whose output is denoted as $y_{\mathrm{f}}(n)$. The past decisions are fed back and filtered by a baud-rate FIR feedback filter $\mathbf{w}_{\mathrm{b}}$ with M_{b} coefficients, resulting in the output signal $y_{\mathrm{b}}(n)$. The sum of the filters' outputs, i.e., $y(n) = y_{\mathrm{f}}(n) + y_{\mathrm{b}}(n)$, enters to the decision device.

Defining the input regressor vectors as

$$\mathbf{x}(n) = [\, \mathbf{x}_{\mathrm{e}}^T(n) \; \mathbf{x}_{\mathrm{o}}^T(n) \,]^T \tag{1}$$

$$\hat{\mathbf{a}}_{\Delta}(n) = [\hat{a}(n-\Delta-1) \; \cdots \; \hat{a}(n-\Delta-M_b)]^T, \tag{2}$$

where

$$\mathbf{x}_{\mathrm{e}}(n) = [x_{\mathrm{e}}(n) \;\; x_{\mathrm{e}}(n-1) \; \cdots \; x_{\mathrm{e}}(n-M_f/2+1)]^T, \tag{3}$$

and

$$\mathbf{x}_{\mathrm{o}}(n) = [x_{\mathrm{o}}(n) \;\; x_{\mathrm{o}}(n-1) \; \cdots \; x_{\mathrm{o}}(n-M_f/2+1)]^T, \tag{4}$$

the outputs of the feedforward and feedback filters can be computed respectively as

$$y_{\mathrm{f}}(n) = \mathbf{x}^T(n)\mathbf{w}_{\mathrm{f}}(n-1) \tag{5}$$

and

$$y_{\mathrm{b}}(n) = \hat{\mathbf{a}}_{\Delta}^T(n)\mathbf{w}_{\mathrm{b}}(n-1), \tag{6}$$

being $\mathbf{w}_{\mathrm{f}}(n)$ the coefficient vector of the feedforward filter, formed by the concatenation of the coefficient vectors $\mathbf{w}_{\mathrm{fe}}(n)$ and $\mathbf{w}_{\mathrm{fo}}(n)$.

Using known statistics of the transmitted signal, the blind DFE must mitigate the channel effects and recover the signal $a(n)$ for some delay Δ.

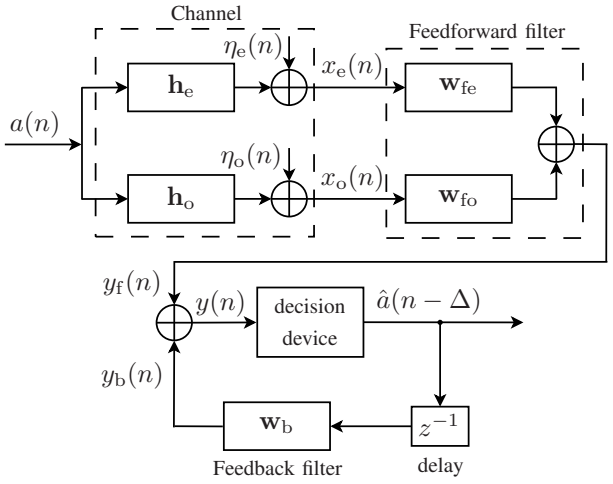

Fig. 3. Simplified communication system with a T/2-fractionally-spaced DFE.

III. Error functions and the proposed algorithm

To overcome the drawbacks of the constant-modulus-based DFE, we focus on the following class of normalized algorithms

$$\begin{bmatrix} \mathbf{w}_f(n) \\ \mathbf{w}_b(n) \end{bmatrix} = \begin{bmatrix} \mathbf{w}_f(n-1) \\ [1-\tilde{\mu}(n)\lambda(n)]\,\mathbf{w}_b(n-1) \end{bmatrix} + \tilde{\mu}(n)e(n)\mathbf{u}^*(n) + \tilde{\mu}(n)\lambda(n)\begin{bmatrix} \mathbf{g}_f(n) \\ \mathbf{0}_{M_b} \end{bmatrix}, \quad (7)$$

where $e(n) = e_R(n) + je_I(n)$ is the estimation error with the real and imaginary parts computed separately as in MMA [12], the column vector $\mathbf{u}(n)$ is formed by the concatenation of the input vectors of the filters, i.e.,

$$\mathbf{u}(n) = [\,\mathbf{x}^T(n)\;\; \hat{\mathbf{a}}_\Delta^T(n)\,]^T, \quad (8)$$

$(\cdot)^*$ stands for the complex-conjugate, $\mathbf{0}_{M_b}$ is a null vector with M_b elements,

$$\tilde{\mu}(n) = \frac{\mu}{\delta + \|\mathbf{u}(n)\|^2}, \quad (9)$$

being $0 < \mu < 2$, δ a regularization factor, and $\|\cdot\|$ the Euclidean norm. The Lagrange multiplier $\lambda(n)$ and the vector $\mathbf{g}_f(n)$ appear in (7) to avoid degenerative solutions.

The degenerative solutions occur when the signal at the equalizer output is independent of its input, presenting a constant or oscillatory behavior. To avoid these undesirable solutions, [6] imposed a constraint in the constant modulus criterion, leading to the algorithm DFE-CMA-FB. This algorithm computes the variable

$$c(n) = \|\mathbf{w}_b(n-1)\|^2 - E_{y_f}(n),$$

in which $E_{y_f}(n)$ represents an estimate of the power of $y_f(n)$. According to the proof presented in [6], to avoid degenerative solutions, $c(n)$ must be always less than or equal to zero. Therefore, if $c(n) \leq 0$, the Lagrange multiplier $\lambda(n)$ is made equal to zero and DFE-CMA-FB works like the conventional CMA. On the other hand, if $c(n) > 0$, it adjusts the updating of $\mathbf{w}_f$ and $\mathbf{w}_b$ with $\lambda(n) = \ell_o$ (positive constant). The vector $\mathbf{g}_f(n)$ represents an estimate of the cross correlation between the vector $\mathbf{x}(n)$ and the output of the feedfoward filter $y_f(n)$. Here, we extend the constraint of [6] to the class of algorithms (7).

In the sequel, we analyze different error functions $e(n)$, in order to verify some desirable characteristics for the equalization of QAM signals. All the error functions considered here can be used in (7) to obtain different versions of algorithms to blindly adapt a DFE.

In the multimodulus algorithm, the estimation error $e(n)$ is defined in terms of its real and imaginary parts separately, i.e,

$$e_{MMA}(n) = [r - y_R^2(n)]y_R(n) + j[r - y_I^2(n)]y_I(n), \quad (10)$$

where $y_R(n)$ and $y_I(n)$ are the real and imaginary parts of $y(n)$, respectively, and r is the dispersion factor [12]. In the case of square QAM constellations, r is the dispersion constant, being the same for both real and imaginary parts, i.e.,

$$r = \frac{\mathrm{E}\{a_R^4(n)\}}{\mathrm{E}\{a_R^2(n)\}} = \frac{\mathrm{E}\{a_I^4(n)\}}{\mathrm{E}\{a_I^2(n)\}}, \quad (11)$$

where $\mathrm{E}\{\cdot\}$ is the expectation operator, and a_R (resp., a_I) represents the real (resp., imaginary) part of all possible transmitted symbols.

Fig. 4 shows the real part of the MMA error, denoted by $e_{MMA,R}(n)$, as a function of $y_R(n)$, assuming a 64-QAM signal (the figure for the imaginary counterpart is identical). The real part of the MMA error is equal to zero when $y_R^2(n)$ is null or when $y_R^2(n)$ is equal to the dispersion constant r. Furthermore, $|e_{MMA,R}(n)|$ assumes a value from the set $\{36,\ 60,\ 84\}$, when $y_R(n)$ is equal to one of the symbols coordinates $\{\pm 1,\ \pm 3,\ \pm 5,\ \pm 7\}$. Therefore, similarly to CMA, MMA exhibits a large steady-state mean-square error (MSE) for nonconstant modulus signals.

In order to reduce the steady-state MSE of MMA, different approaches were proposed in the literature. This is the case of Sliced-MMA proposed in [20], where the dispersion constant is weighed based on the constellation size and on the magnitude of the transmitted symbols. Although its error function is reduced when $y(n)$ is equal to the constellation symbols, it is not enough to reduce substantially the steady-state MSE of MMA.

Another approach was proposed in [17], where CMA operates concurrently with the last stage of the soft decision-directed (SDD) algorithm. This algorithm was extended in [13] and [14] to blindly adapt a DFE, considering DFE-CMA-FB rather than CMA. Through simulations, it was shown in [17], [13], [14] that, at the cost of a moderate increase in computational complexity, the concurrent algorithms CMA+SDD and NDEG-SDD-CMA can present an improvement in equalization performance over CMA and DFE-CMA-FB, respectively. To illustrate the error function of the CSD algorithm (the MMA version of NDEG-SDD-CMA), Fig. 5 shows the real part of the error $e_{CSD,R}(n)$ as a function of the real part of the equalizer output $y_R(n)$ for 64-QAM. Unlike MMA, the error of CSD is close to zero when the equalizer output

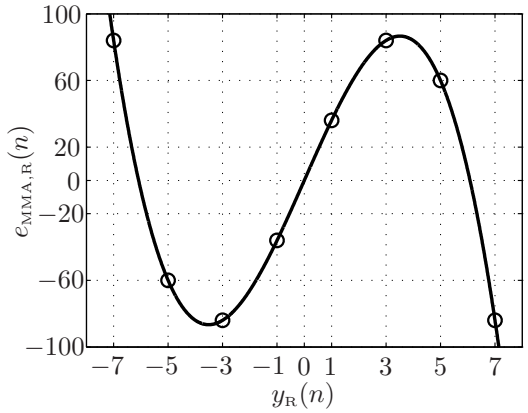

Fig. 4. Real part of the error of MMA as a function of $y_{\mathrm{R}}(n)$ for 64-QAM.

coincides with the transmitted signal, which is responsible for the reduction of its misadjustment. Unless a scale factor, it is possible to recognize in the figure an error pattern which repeats in regions containing the real part of two symbols of the constellation. We can also observe that the error function presents three zero-crossings in each region. However, only two zero-crossings are necessary since each region contains two symbol coordinates. It is important to notice that the good behavior of the CSD algorithm depends on the shape of its error function, which in turn, depends on the ratio of step-sizes $\mu_{\mathrm{SDD}}/\mu_{\mathrm{MMA}}$. If, for example, $\mu_{\mathrm{SDD}}/\mu_{\mathrm{MMA}} = 100$, the error of CSD will be farther from zero when the equalizer output coincides with the transmitted signal, which deteriorates its performance. Therefore, it is not always easy to ensure a good performance of the CSD algorithm.

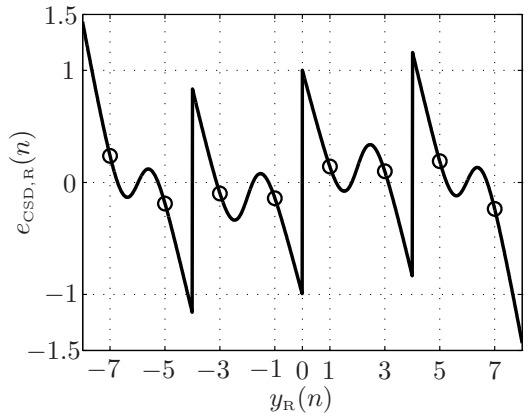

Fig. 5. Real part of the error of CSD as a function of $y_{\mathrm{R}}(n)$ for 64-QAM; $\mu_{\mathrm{SDD}}/\mu_{\mathrm{MMA}} = 500$.

Inspired in the CSD error function and using an MMA-like implementation, we propose the symbol-based decision (SBD) algorithm, which can be interpreted as an extension of the decision-directed algorithm for blind equalization of QAM signals. The error of the proposed algorithm is shown in Fig. 6, where the real part of the error $e_{\mathrm{SBD,R}}(n)$ is plotted as a function of the real part of the equalizer output $y_{\mathrm{R}}(n)$ for 64-QAM. Unlike CSD, the SBD error is null only when the equalizer output is equal to one of the constellation symbol coordinates, which ensures its better behavior when compared with MMA or with the CSD algorithm. It its important to notice that there is an envelope in the SBD error, which is essential for the recovery of the transmitted symbols as observed in [21]. Without this envelope, the error function coincides with that of the decision-direct algorithm, whose good behavior is ensured only when the equalizer is close to the optimal solution. The expression for the SBD error is given by

$$e_{\mathrm{SBD}}(n) = |\hat{a}_{\mathrm{R}}(n)|[\hat{a}_{\mathrm{R}}(n) - y_{\mathrm{R}}(n)] + j|\hat{a}_{\mathrm{I}}(n)|[\hat{a}_{\mathrm{I}}(n) - y_{\mathrm{I}}(n)], \quad (12)$$

where $\hat{a}_{\mathrm{R}}(n)$ (resp., $\hat{a}_{\mathrm{I}}(n)$) is the nearest real (resp., imaginary) part of the constellation symbol from $y_{\mathrm{R}}(n)$ (resp., $y_{\mathrm{I}}(n)$), and $|\hat{a}_{\mathrm{R}}(n)|$ and $|\hat{a}_{\mathrm{I}}(n)|$ are responsible by the envelope of the error function.

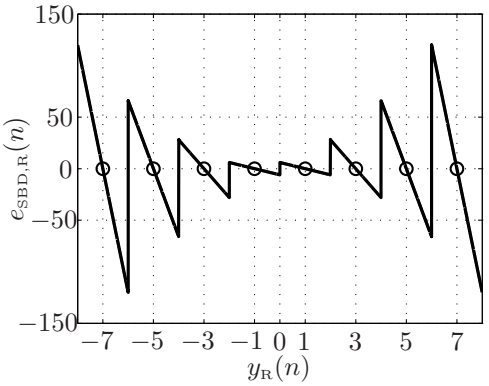

Fig. 6. Real part of the error of SBD as a function of $y_{\mathrm{R}}(n)$ for 64-QAM.

A summary of the SBD algorithm is shown in Table I, where $\mathrm{step}[x] = \{1 \text{ if } x \geq 0;\ 0 \text{ if } x < 0\}$ is the step function, α is a forgetting factor, and $\mathrm{dec}[x]$ is the nearest symbol coordinate from x. It is usual to assume $\ell_{\mathrm{o}} = 2$ and $\alpha = 0.95$.

The computational cost per iteration of SBD is shown in Table II, considering the number of real multiplications, real additions, real divisions, comparisons ($\mathcal{C}$), and exponential computations (exp). The scalar S represents the number of constellation symbols. The complexity of SBD is compared to those of DFE-MMA-FB (the MMA version of DFE-CMA-FB) and CSD. As we can observe in the table, the CSD algorithm requires more operations per iteration than SBD, which in turn, requires almost the same number of operations per iteration of DFE-MMA-FB.

IV. On the convergence and stability

In this section, we propose a method to improve the convergence rate of the SBD algorithm, exploring the neighborhood of the estimated symbol. We also analyze its stability, assuming $\lambda(n) = 0$.

A. Improving the convergence with the neighborhood

At the initial iterations, the coefficient vectors $\mathbf{w}_{\mathrm{f}}$ and $\mathbf{w}_{\mathrm{b}}$, updated with the SBD algorithm, can be very distant from the optimal solution, and the signal $\hat{a}(n-\Delta)$ can represent a

TABLE I
SUMMARY OF THE SBD ALGORITHM.

Initialize the algorithm by setting:

$\mathbf{w}_f(0) = [0 \ \cdots \ 0 \ 1 \ 0 \ \cdots \ 0]$, $\ell_o = 2$

$\mathbf{w}_b(0) = 0$, $\mathbf{g}_f(0) = \mathbf{0}$, $E_{yf}(0) = 0$, $0 \ll \alpha < 1$

$0 < \mu < 2/\sqrt{S}$, δ : small positive constante.

For $n = 1, 2, 3 \ldots$, compute:

$\mathbf{u}(n) = \left[\mathbf{x}^T(n) \ \ \hat{\mathbf{a}}_\Delta^T(n)\right]^T$

$y_f(n) = \mathbf{x}^T(n)\mathbf{w}_f(n-1)$

$y_b(n) = \hat{\mathbf{a}}_\Delta^T(n)\mathbf{w}_b(n-1)$

$y(n) = y_f(n) + y_b(n)$

$y_R(n) = \mathrm{Re}[y(n)]; \quad y_I(n) = \mathrm{Im}[y(n)]$

$\hat{a}_R(n) = \mathrm{dec}[y_R(n)]; \quad \hat{a}_I(n) = \mathrm{dec}[y_I(n)]$

$e_R(n) = |\hat{a}_R(n)|[\hat{a}_R(n) - y_R(n)]$

$e_I(n) = |\hat{a}_I(n)|[\hat{a}_I(n) - y_I(n)]$

$e(n) = e_R(n) + je_I(n)$

$E_{y_f}(n) = \alpha E_{y_f}(n-1) + (1-\alpha)|y_f(n)|^2$

$\mathbf{g}_f(n) = \alpha\mathbf{g}_f(n-1) + (1-\alpha)\, y_f(n)\mathbf{x}^*(n)$

$c(n) = \|\mathbf{w}_b(n-1)\|^2 - E_{y_f}(n)$

$\lambda(n) = \ell_o \,\mathrm{step}[c(n)]$

$\tilde{\mu}(n) = \dfrac{\mu}{\delta + \|\mathbf{u}(n)\|^2}$

$\mathbf{w}_f(n) = \mathbf{w}_f(n-1) + \tilde{\mu}(n)[\lambda(n)\mathbf{g}_f(n) + e(n)\mathbf{x}^*(n)]$

$\mathbf{w}_b(n) = [1 - \tilde{\mu}(n)\lambda(n)]\mathbf{w}_b(n-1) + \tilde{\mu}(n)e(n)\hat{\mathbf{a}}_\Delta^*(n)$

end

TABLE II
COMPUTATION COST IN TERMS OF REAL OPERATIONS PER ITERATION.

Op.	DFE-MMA-FB	CSD	SBD
$\times$	$18M_f + 14M_b$ $+15$	$22M_f + 18M_b$ $+33$	$18M_f + 14M_b$ $+13$
$+$	$13M_f + 12M_b$ $+5$	$19M_f + 18M_b$ $+11$	$13M_f + 12M_b$ $+5$
$\div$	2	5	2
exp	$-$	4	$-$
$\mathcal{C}$	1	$\log_2(S)$	$\log_2(S)+1$

wrong decision, mainly in the presence of noise and for high-order constellations. This issue can be worse in a DFE due to the decision feedback, which also generates error propagation.

To improve the convergence of the SBD algorithm, we can use the philosophy proposed in [22]. Assuming a square S-QAM constellation, the real line can divided into $\sqrt{S}$ regions $\mathrm{A}_{k,\mathrm{R}}$ with symbol coordinates $a_{k,\mathrm{R}}$, being $k = -\sqrt{S}/2, \cdots, -1, 1, \cdots, \sqrt{S}/2$, as shown in Fig. 7 for the real part of 64-QAM. Assuming that the real part of the equalizer output falls in the region $A_{\ell,\mathrm{R}}$, the error should take into account not only the region $\mathrm{A}_{\ell,\mathrm{R}}$, but also the regions $\mathrm{A}_{\ell-1,\mathrm{R}}$ and $\mathrm{A}_{\ell+1,\mathrm{R}}$ in its neighborhood. Note that, $a_{\ell,\mathrm{R}} = \hat{a}_\mathrm{R}(n) = \mathrm{dec}[y_\mathrm{R}(n)]$ and $a_{\ell\pm1,\mathrm{R}} = \hat{a}_\mathrm{R}(n) \pm 2$. Furthermore, if $\mathrm{A}_{\ell,\mathrm{R}}$ is a region of the constellation edges, there will be only inner neighbors. In the example of Fig. 7, the main region is $\mathrm{A}_{-1,\mathrm{R}}$ and the neighboring regions are $\mathrm{A}_{1,\mathrm{R}}$ and $\mathrm{A}_{-2,\mathrm{R}}$. Thus, the real part of the SBD error can be computed as

$$e_{\mathrm{SBD,R}}(n) = \sum_{m=\ell-1}^{\ell+1} \gamma_{m,\mathrm{R}}|a_{m,\mathrm{R}}|\left[a_{m,\mathrm{R}} - y_\mathrm{R}(n)\right], \qquad (13)$$

where $\gamma_{m,\mathrm{R}} = 1$ for $m = \ell$ (main region) and $\gamma_{m,\mathrm{R}} = 2^{-2}$ for $m = \ell \pm 1$ (neighboring regions). The same procedure should be considered for the imaginary part $y_\mathrm{I}(n)$.

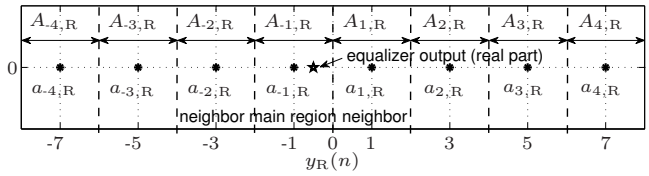

Fig. 7. Regions of the real part of 64-QAM for SBD.

Despite the improvement in convergence rate, the neighborhood of the estimated symbol may cause an increase in the steady-state MSE. Thus, the aid of the neighbors should be disregarded when the algorithm achieves the steady-state. For this purpose, instead of weighting the neighboring errors by $\gamma_{\ell\pm1,\mathrm{R}} = 2^{-2}$, we consider a time-variant function $\gamma_{\ell\pm1,\mathrm{R}}(n) = 2^{-p(n)}$, where

$$p(n) = 7.1467\frac{1 - \mathrm{e}^{8[\xi(n)-0.03]}}{1 + \mathrm{e}^{8[\xi(n)-0.03]}} + 9.1467, \qquad (14)$$

and $\xi(n) = \alpha\xi(n-1) + (1-\alpha)|e_d(n)|^2$ is an estimate of the mean-squared decision error, being $e_d(n) = \hat{a}(n-\Delta) - y(n)$ and $0 \ll \alpha < 1$ a forgetting factor. It should be notice that $2 \leq p(n) \leq 10$ and that the smaller the MSE, the larger is the value of $p(n)$, and consequently the smaller the weights $\gamma_{\ell\pm1,\mathrm{R}}(n)$. This function was experimentally chosen in [22]. Through simulations, we observe that $p(n)$ is important to make the MSE of the SBD algorithm smaller at the steady-state.

Depending on the number of the constellation symbols, it can be necessary to increase the number of neighbors. However, it is important to impose a distinction among the errors calculated in the neighborhood and that of the main region, i.e., the farther the neighbor, the smaller the weight $\gamma_{m,\mathrm{R}}$. Through simulations, we observed from 64 to 1024-QAM that two neighbors for the real part and two for the imaginary part are sufficient to improve significantly the convergence of SBD.

To include this improvement in the algorithm of Table I, the error $e_\mathrm{R}(n)$ should be replaced by (13) (analogously to the imaginary part). Additionally, the parameter $p(n)$ should also be computed at each iteration. Since this technique requires only computations of scalars and two exponentiations per iteration, the complexity cost of the resulting algorithm is slightly higher than that of Table II.

B. Stability issues

To facilitate the convergence analysis of the SBD algorithm, we assume $\lambda(n) = 0$, i.e., we do not include the mechanism

to avoid degenerative solutions. We first analyze the algorithm without neighbors and in the sequel, we consider two neighbors in the analysis.

In the case without neighbors, we particularize the error of the SBD algorithm by replacing the factors $|\hat{a}_{\rm R}(n)|$ and $|\hat{a}_{\rm I}(n)|$ by $\max\left\{|\hat{a}_{\rm R}(n)|, |\hat{a}_{\rm I}(n)|\right\}$, which leads to

$$e_{\rm SBD}(n) = \max\left\{|\hat{a}_{\rm R}(n)|, |\hat{a}_{\rm I}(n)|\right\}\left[\hat{a}(n-\Delta) - y(n)\right]. \quad (15)$$

Note that $\hat{a}(n-\Delta) \triangleq \hat{a}_{\rm R}(n) + j\hat{a}_{\rm I}(n)$. Using (15) in conjunction with convergence results for the normalized least mean-square (NLMS) algorithm (see, e.g., [23, p. 80]), we conclude that the SBD algorithm is stable if the step-size μ is chosen in the interval given by

$$0 < \mu < \frac{2}{\sqrt{S}} < \frac{2}{\max\left\{|\hat{a}_{\rm R}(n)|, |\hat{a}_{\rm I}(n)|\right\}}. \quad (16)$$

Assuming now that $y_{\rm R}(n)$ and $y_{\rm I}(n)$ fall respectively in $A_{\ell,{\rm R}}$ and $A_{k,{\rm I}}$ with neighbors $A_{\ell\pm1,{\rm R}}$ and $A_{k\pm1,{\rm I}}$, and using the symmetry properties $a_{\ell\pm1,{\rm R}} = \hat{a}_{\rm R}(n) \pm 2$ and $a_{k\pm1,{\rm I}} = \hat{a}_{\rm I}(n) \pm 2$, we can show that the stability of the SBD algorithm is ensured if μ is within the interval given by

$$0 < \mu < \frac{2}{\sqrt{S}(1+2\gamma_{\max})}, \quad (17)$$

where $\gamma_{\max} = \max\{\gamma_{\ell\pm1,{\rm R}}(n), \gamma_{k\pm1,{\rm I}}(n)\}$. For $\gamma_{\max} = 2^{-2}$, this interval reduces to $0 < \mu < 1.334/\sqrt{S}$.

V. Simulations

In this section, we compare the performance of the SBD algorithm with those of NLMS, CSD, and DFE-MMA-FB. We also consider the Wiener solution, assuming its best delay. For the performance evaluation of such algorithms it was considered both the MSE and symbol error rate (SER) curves. Constellation signals 64-QAM, 1024-QAM and 4096-QAM were used as test signals. For 4096-QAM, the SBD algorithm was implemented considering four neighbors for the real components and four for the imaginary ones. For 64-QAM and 1024-QAM constellations, only two neighbors for the real components and two for the imaginary ones were used. Normalized versions of all algorithms were implemented and their step-sizes were adjusted to ensure their better performance in terms of steady-state MSE. All algorithms adapt a $T/2$-DFE, with the lengths of the feedforward and feedback filters experimentally chosen. The feedforward filter was implemented with 118 coefficients considering center-spike initialization, and the feedback filter with 20 zero-initialized coefficients. The channel was obtained from "chan1.mat" of the database available at in http://spib.rice.edu/spib/cable.html. For these conditions a delay $\Delta = 61$ was chosen as the best delay.

Figures 8, 9, and 10 show the MSE along the iterations for 64-QAM, 1024-QAM, and 4096-QAM signals, respectively. For these three simulations, we can observe that SBD algorithm presents faster convergence an lower steady-state MSE when compared to DFE-MMA-FB or to the CSD algorithm. Additionally, only SBD achieves a steady-state MSE similar to that of NLMS. Despite CSD achieves low steady-state MSE for 64-QAM and 1024-QAM, the adjust of the CSD step-size becomes more difficult as the order of the constellation increases. This was particularly noted for the 4096-QAM signal since, in this case, the CSD algorithm converges to an MSE higher than that of SBD. It is also important to notice that the steady-state MSE for both SBD and NLMS is a little distant from steady-state MSE of the equivalent Wiener solution. This distance occurs due to the initialization of these algorithms, since they were initialized with the center-spike.

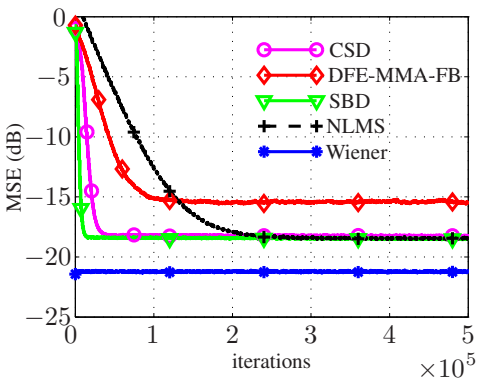

Fig. 8. MSE for DFE-MMA-FB ($\mu = 2\times10^{-4}$), CSD ($\mu_{\rm MMA} = 1\times10^{-5}$, $\mu_{\rm SDD} = 1\times10^{-2}$, $\rho = 0.6$), SBD ($\mu = 5\times10^{-3}$), and NLMS ($\mu = 2\times10^{-3}$); average of 50 runs, SNR = 35 dB; 64-QAM.

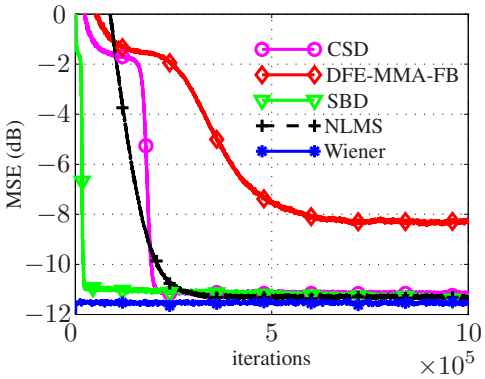

Fig. 9. MSE for DFE-MMA-FB ($\mu = 1\times10^{-6}$), CSD ($\mu_{\rm MMA} = 5\times10^{-7}$, $\mu_{\rm SDD} = 1\times10^{-2}$, $\rho = 0.6$), SBD ($\mu = 5\times10^{-3}$), and NLMS ($\mu = 5\times10^{-2}$); average of 50 runs, SNR = 40 dB; 1024-QAM.

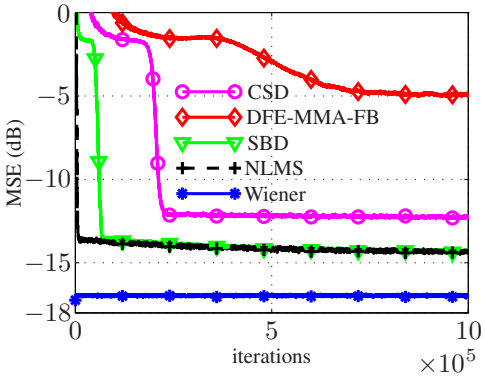

Fig. 10. MSE for DFE-MMA-FB ($\mu = 2\times10^{-7}$), CSD ($\mu_{\rm MMA} = 3\times10^{-7}$, $\mu_{\rm SDD} = 1\times10^{-2}$, $\rho = 0.6$), SBD ($\mu = 2\times10^{-3}$), and NLMS ($\mu = 5\times10^{-2}$); average of 50 runs, SNR = 50 dB; 4096-QAM.

Figures 11, 12, and 13 show SER curves as a function of the signal-to-noise ratio (SNR) for 64-QAM, 1024-QAM, and 4096-QAM signals, respectively. For each constellation, the SER curve of the AWGN channel was also shown. Lower SER values are found for SBD algorithm when compared to those of DFE-MMA-FB and CSD, specially for the 4096-QAM case. Again, the performance of SBD is very close to that of NLMS. As for the steady-state MSE, the behavior of both SBD and NLMS is worse than that of the Wiener solution, due to the initialization.

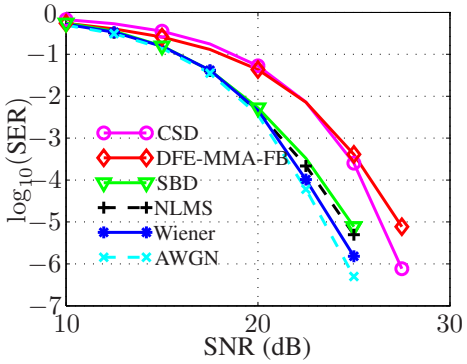

Fig. 11. Logarithm of SER as a function of SNR; DFE-MMA-FB ($\mu = 2 \times 10^{-4}$), CSD ($\mu_{\mathrm{MMA}} = 1 \times 10^{-5}$, $\mu_{\mathrm{SDD}} = 1 \times 10^{-2}$, $\rho = 0.6$), SBD ($\mu = 5 \times 10^{-3}$), and NLMS ($\mu = 2 \times 10^{-3}$); 64-QAM.

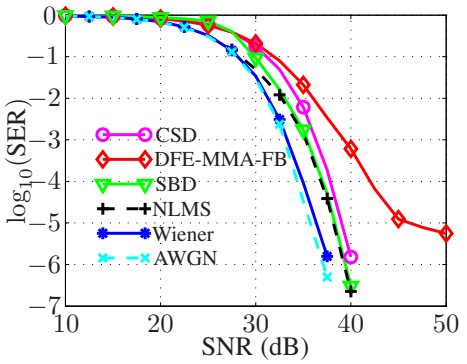

Fig. 12. Logarithm of SER as a function of SNR; DFE-MMA-FB ($\mu = 1 \times 10^{-6}$), CSD ($\mu_{\mathrm{MMA}} = 5 \times 10^{-7}$, $\mu_{\mathrm{SDD}} = 1 \times 10^{-2}$, $\rho = 0.6$), SBD ($\mu = 5 \times 10^{-3}$), and NLMS ($\mu = 5 \times 10^{-2}$); 1024-QAM.

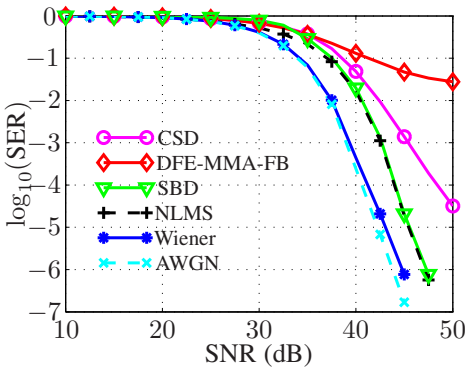

Fig. 13. Logarithm of SER as a function of SNR; DFE-MMA-FB ($\mu = 2 \times 10^{-7}$), CSD ($\mu_{\mathrm{MMA}} = 3 \times 10^{-7}$, $\mu_{\mathrm{SDD}} = 1 \times 10^{-2}$, $\rho = 0.6$), SBD ($\mu = 2 \times 10^{-3}$), and NLMS ($\mu = 5 \times 10^{-2}$); 4096-QAM.

VI. Conclusions

We introduced a symbol-based decision algorithm for blind equalization of quadrature amplitude modulation (QAM) signals using a decision feedback scheme. Independently of QAM order, it presents: (i) an error equal to zero when the equalizer output coincides with the transmitted signal; (ii) simultaneous recovery of the modulus and phase of the signal; (iii) a misadjustment close to that of the normalized least-mean squares (NLMS) algorithm; (iv) a fast convergence; and (v) the avoidance of degenerative solutions. Additionally, its convergence is ensured when the step-size is properly chosen and its computational cost is similar to that of DFE-MMA-FB.

References

[1] J. Proakis, *Digital Communications*, McGraw-Hill, NY, 4th edition, 2001.

[2] R. L. Howald, "QAM bulks up once again: Modulation to the power of ten," in *Proceedings of SCTE Cable-tec EXPO*, 2002. Avaiable at http://broadband.motorola.com/ips/pdf/QAM.pdf.

[3] W. Zhou and L. Zou, "Adaptive QAM transmission scheme to improve performance on an AWGN channel," U.S. Patent 2011/0090948A1, Apr. 21, 2011.

[4] G. D. Forney Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Transactions on Information Theory*, vol. IT-18, pp. 363–378, May 1972.

[5] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, Upper Saddle River, 4th edition, 2002.

[6] L. L. Szczecinski and A. Gei, "Blind decision feedback equalisers, how to avoid degenerated solutions," *Signal Processing*, vol. 82, pp. 1675–1693, Nov. 2002.

[7] S. Bouchired, D. Roviras, and F. Castani, "Equalisation of satellite mobile channels with neural network techniques," *Space Comm.*, vol. 15, pp. 209–220, 1998/1999.

[8] A. A. Rontogiannis and K. Berberidis, "Efficient decision feedback equalization for sparce wireless channel," *IEEE Transactions on Wireless Communications*, vol. 2, pp. 570–581, May 2003.

[9] D. N. Godard, "Self-recovering equalization and carrier tracking in two dimensional data communication system," *IEEE Trans. Commun.*, vol. 28, pp. 1867–1875, Nov. 1980.

[10] C. R. Johnson Jr. *et al.*, "Blind equalization using the constant modulus criterion: a review," *Proc. IEEE*, vol. 86, pp. 1927–1950, Oct. 1998.

[11] K. N. Oh and Y. O. Chin, "Modified constant modulus algorithm: Blind equalization and carrier phase recovery algorithm," in *Proc. of IEEE Int. Conf. Commun.*, 1995, vol. 1, pp. 498–502.

[12] J. Yang, J.-J. Werner, and G. A. Dumont, "The multimodulus blind equalization and its generalized algorithms," *IEEE J. Sel. Areas Commun.*, vol. 20, pp. 997–1015, Jun. 2002.

[13] M. T. M. Silva, M. Miranda, and R. Soares, "Concurrent blind decision feedback equalizer," in *Proc. of International Workshop on Telecommunications (IWT)*, Santa Rita do Sapucaí, MG, 2004, pp. 107–112.

[14] M. T. M. Silva, M. D. Miranda, and R. Soares, "Concurrent algorithm for blind adaptation of DFE," *Electronics Letters*, vol. 41, pp. 928–930, Aug. 2005.

[15] S. Chen, S. Mclaughlin, P. M. Grant, and B. Mulgrew, "Multi-stage clustering equaliser," *IEEE Trans. Commun.*, vol. 43, pp. 701–705, Feb./Mar./Abr. 1995.

[16] F. C. C. De Castro, M. C. F. De Castro, and D. S. Arantes, "Concurrent blind deconvolution for channel equalization," in *Proc. of ICC'2001*, 2001, vol. 2, pp. 366–371.

[17] S. Chen, "Low complexity concurrent constant modulus algorithm and soft directed scheme for blind equalization," *IEE Proceedings - Vision, Image, and Signal Processing*, vol. 150, pp. 312–320, Oct. 2003.

[18] J. R. Treichler, I. Fijalkow, and C. R. Johnson Jr., "Fractionally spaced equalizers," *IEEE Signal Process. Mag.*, vol. 13, pp. 65–81, May 1996.

[19] L. Qin *et al.*, "Fractionally spaced adaptive decision feedback equalizers with applications to atsc dtv receivers," *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 999–1003, Nov. 2004.

[20] S. Abrar and R. A. Axford Jr., "Sliced multi-modulus blind equalization algorithm," *ETRI Journal*, vol. 27, pp. 257–266, Jun. 2005.

[21] J. Mendes Filho, M. T. M. Silva, and M. D. Miranda, "A family of algorithms for blind equalization of QAM signals," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, Prague, Czech Republic, 2011, pp. 3388–3391.
[22] J. Mendes Filho, M. T. M. Silva, M. D. Miranda, and V. H. Nascimento, "A region-based algorithm for blind equalization of QAM signals," in *Proc. of the IEEE/SP 15th Workshop on Statistical Signal Processing*, Cardiff, UK, 2009, pp. 685–688.
[23] I. Mareels and J. W. Polderman, *Adaptive systems: an introduction*, Birkhäuser, Boston, 1996.

João Mendes Filho was born in São Paulo, Brazil. He received the B.S. degree in 1983 and the M.S. degree in 2007 from Mackenzie Presbyterian University, São Paulo, Brazil both in Electrical Engineering. Nowadays he is pursuing the doctoral degree at Escola Politécnica, University of São Paulo, São Paulo, Brazil, also in Electrical Engineering. As development engineer he has worked many years for Alcatel and NEC, both companies in São Paulo, Brasil.

Maria D. Miranda was born in Florianópolis, Brazil. She received the B.S. degree in 1983 and the M.S. degree in 1987 from Universidade Federal de Santa Catarina, Florianópolis, and the doctoral degree in 1996 from Escola Politécnica, University of São Paulo, São Paulo, Brazil, all in Electrical Engineering. She worked as a Postdoctoral Researcher at Escola Politécnica, University of São Paulo and at Institut National des Télécommunications, Evry, France, in 1997/1998 and 1998/1999, respectively. From February 2000 to July 2006 she was an Assistant Professor at Mackenzie Presbyterian University, São Paulo. Currently, she is an Assistant Professor in the Department of Telecommunications and Control Engineering, Escola Politécnica, University of São Paulo. Her teaching and research interests are in digital signal processing, adaptive signal processing, and specifically in applications and analysis of data-aided and blind algorithms.

Magno T. M. Silva was born in São Sebastião do Paraíso, Brazil, in 1975. He received the B.S. degree in 1999, the M.S. degree in 2001, and the Ph.D. degree in 2005, all in Electrical Engineering from Escola Politécnica, University of São Paulo, São Paulo, Brazil. From February 2005 to July 2006 he was an Assistant Professor at Mackenzie Presbyterian University, São Paulo. He is currently an Assistant Professor in the Department of Electronic Systems Engineering, Escola Politécnica, University of São Paulo. His research interests include linear and nonlinear adaptive filtering and machine learning techniques.