CONSTANT MODULUS ALGORITHM ASSISTED SOFT DECISION-FEEDBACK EQUALIZATION

S. Vlahoyiannatos, L. Hanzo

Dept. of Electr. and Comp. Sc., Univ. of Southampton, SO17 1BJ, UK.
Tel: +44-1703-593 125, Fax: +44-1703-594508
Email: {sv97r,lh}@ecs.soton.ac.uk
http://www-mobile.ecs.soton.ac.uk

ABSTRACT

A new blind equalizer is proposed, extending the well-known Constant Modulus Algorithm (CMA), which is based on a Finite Impulse Response (FIR) filter, to an Infinite Impulse Response (IIR) scheme. This equalizer is capable of equalizing channels having zeros on the unit circle with better accuracy than that of the classic CMA, while maintaining a similar complexity.

1. INTRODUCTION

Blind equalization has been intesively researched in recent years and a variety of methods have been proposed [1]. In this contribution we improve the well-known Constant Modulus Algorithm (CMA) based equalizer [2] by invoking soft feedback information. Similar work has been reported in [3], where a lattice-form CMA based Decision-Feedback Equalizer (DFE) using the Shtrom-Fan cost function was explored and in [4], where a predictive DFE-CMA was proposed. Furthermore, in [5] two different blind DFEs were explored using combinations of the CMA and the Shtrom-Fan cost functions for the feedforward and the feedback sections of the equalizer, respectively. Finally, in [6], a modified version of the CMA, namely the MCMA was extended to a DFE.

The paper is organized as follows. The proposed equalizer is described in Section 2. In Section 3 the convergence issues of this equalizer are discussed and in Section 4 the equalizer's performance is evaluated with the aid of computer simulations.

2. SYSTEM DESCRIPTION

The communications system under consideration is shown in Figure 1. In the proposed scheme the information bits are differentially encoded in order to be able to demodulate the

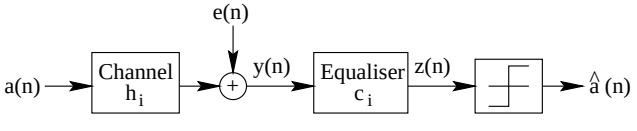

Figure 1: Equalized Communications system

information, when the blind equalizer converges to a sign-reversed solution. The differentially coded bits are then mapped to the Quadrature Amplitude Modulation (QAM) symbols $a(n)$. These symbols are convolved with the Channel Impulse Response (CIR) h_i and contaminated by the channel noise $e(n)$, yielding the received symbols $y(n)$ as:

$$y(n) = \sum_{i=-L_1}^{L_2} h_i \cdot a(n-i) + e(n). \quad (1)$$

The restoration of the original information bits is performed by the DFE-CMA equalizer. We will describe the operation of this equalizer below. The DFE-CMA equalizer performs channel equalization and delivers the equalized signal $z(n)$ as the output of a feedforward and a feedback filter. In mathematical terms the equalized signal is given by:

$$z(n) = \mathbf{c}^T \cdot \mathbf{y}(n) + \mathbf{w}^T \cdot \mathbf{z}(n-1), \quad (2)$$

where $\mathbf{c} = [c_0, c_1, \cdots, c_N]^T$ and $\mathbf{w} = [w_0, w_1, \cdots, x_M]^T$ are the feedforward and feedback equalizer coefficients respectively, $\mathbf{y}(n) = [y(n), y(n-1), \cdots, y(n-N)]^T$ is the received signal vector and $\mathbf{z}(n-1) = [z(n-1), z(n-2), \cdots, z(n-N-1)]^T$ is the equalized signal vector. If we consider the z-transform of this relationship, we arrive at:

$$Z(z) = \frac{C(z)}{1+W(z)} \cdot Y(z) = \frac{C(z)}{1+W(z)} \cdot H(z) \cdot A(z), \quad (3)$$

where $Z(z)$, $Y(z)$ and $A(z)$ are the z-transforms of the equalized, received and transmitted signals respectively, while $H(z)$, $C(z)$ and $W(z)$ are the CIR, the equalizer's feedforward and feedback vectors in the z-domain, respectively. If the channel was described by a linear model having a finite number of taps, then all that this system would have to do for equalizing $Y(z)$ would be to find $W(z)$, so that $1 + W(z) = H(z)$. By contrast, if the channel's transfer function also contains poles, then the feedforward section would also have to be active, so that $(1 + W(z))/C(z) = H(z)$. By considering the CMA cost function [2], we can formulate the cost function for this algorithm as:

$$J(n) = |e(n)|^2 = \left(|z(n)|^2 - R_p\right)^2, \quad (4)$$

where R_p is a constant used in the CMA. By utilising the classic gradient descent algorithm, we can now find the

Step 1:	Calculate the equalized signal according to (2)
Step 2:	Calculate the update of the derivatives according to (27) and (28)
Step 3:	Shift matrices $\mathbf{C}$ and $\mathbf{W}$ one column deleting the least recent column and replace with the column from the recursion of the previous step
Step 4:	Update the equalizer coefficients according to (5), (6) and (25), (25)

Table 1: The steps of the algorithm.

equalizer coefficient update procedure by differentiating the cost function of Equation (4) with respect to (wrt) both the feedforward and feedback coefficients, yielding the update algorithm of:

$$\mathbf{c}^{(n)} = \mathbf{c}^{(n-1)} - \lambda \cdot \frac{\partial J(n-1)}{\partial \mathbf{c}} \quad (5)$$

$$\mathbf{w}^{(n)} = \mathbf{w}^{(n-1)} - \lambda \cdot \frac{\partial J(n-1)}{\partial \mathbf{w}}, \quad (6)$$

where λ is the step-size parameter, controling the learning rate of the equalizer. As seen in Appendix A, the calculation of the above derivatives gives:

$$\begin{aligned} \mathbf{c}^{(n)} &= \mathbf{c}^{(n-1)} - \lambda \cdot e(n) \cdot \left(z^*(n) \cdot \bar{\mathbf{C}}(n-1) \cdot \mathbf{w}\right. \quad (7) \\ &+ \left. z(n)\left(\mathbf{y}^*(n) + \mathbf{C}(n-1) \cdot \mathbf{w}^*\right)\right) \\ \mathbf{w}^{(n)} &= \mathbf{w}^{(n-1)} - \lambda \cdot e(n) \cdot \left(z^*(n) \cdot \bar{\mathbf{W}}(n-1) \cdot \mathbf{w}\right. \quad (8) \\ &+ \left. z(n)\left(\mathbf{z}^*(n-1) + \mathbf{W}(n-1) \cdot \mathbf{w}^*\right)\right), \end{aligned}$$

where $\mathbf{C}(n)$, $\bar{\mathbf{C}}(n)$, $\mathbf{W}(n)$, $\bar{\mathbf{W}}(n)$ are the derivatives of the equalized vector with respect to the equalizer tap-vector as defined in Appendix A, where a recursion for their update is given. Here we give the proposed equalizer coefficient update algorithm in the form of a set of steps, which are summarized in Table 1.

3. CONVERGENCE ISSUES

In this section we discuss the convergence properties of the DFE-CMA algorithm. The stationary points of the algorithm are identified and the similarities with the CMA are highlighted. We commence the analysis by considering the joint transfer function of the channel plus equalizer system denoted by $\mathbf{t} = [t_0, t_1, \cdots, t_K]^T$. Naturally, K may tend to infinity, but we will consider it to be finite for the moment. With this notation, the stationary points of the algorithm can be found by setting the derivatives of the cost function with respect to the equalizer coefficients to zero, as seen in Equations (18) and (19):

$$\begin{aligned} \frac{\partial J(n)}{\partial \mathbf{c}} &= 2 \cdot e(n) \cdot [z^*(n) \cdot \frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{c}} \cdot \mathbf{w} \quad (9) \\ &+ z(n) \cdot (\mathbf{y}^*(n) + \frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{c}} \cdot \mathbf{w}^*)] = \mathbf{0} \\ \frac{\partial J(n)}{\partial \mathbf{w}} &= 2 \cdot e(n) \cdot [z^*(n) \cdot \frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{w}} \cdot \mathbf{w} \quad (10) \\ &+ z(n) \cdot (\mathbf{z}^*(n-1) + \frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{w}} \cdot \mathbf{w}^*)] = \mathbf{0}. \end{aligned}$$

We will assume that we can ignore the contributions of the terms $\frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{w}}$, $\frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{w}}$ and $\frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{c}}$, $\frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{c}}$ because they are relatively small compared to the received and the equalized signals, respectively. Under this assumption, Equations (10) and (11) become:

$$e(n) \cdot z(n) \cdot \mathbf{y}^*(n) = \mathbf{0} \quad (11)$$

$$e(n) \cdot z(n) \cdot \mathbf{z}^*(n-1) = \mathbf{0}. \quad (12)$$

The first of these equations is the same as the relevant equation for the CMA, studied in [7, 8, 9] in the context of a modified version. The stationary points defined by Equation (11) are as follows:

- Local minima are:
  - the points of the form: $\mathbf{t} = [0, \cdots, 0, e^{j\cdot\phi}, 0, \cdots, 0]^T$ and
  - the channel–dependent local minima exhibited by equalizers having a finite length.
- Saddle points are:
  - the point $\mathbf{t} = \mathbf{0}$ and
  - the points of the form $\mathbf{t} = [0, \cdots, 0, e^{j\cdot\phi_1}, \cdots e^{j\cdot\phi_M}, 0, \cdots, 0]^T$.

Note furthermore that for a point to be a local minimum of the DFE-CMA algorithm in terms of the cost function of Equation (4) it has to be a local minimum of both Equations (11) and (12). We will now consider Equation (12), in order to investigate the associated stationary points. After some averaging calculations over all possible transmitted symbols $\mathbf{a}(n)$, assuming that they are independent and identically distributed (i.i.d), this Equation (12) reduces to the following system:

$$\begin{pmatrix} t_0^* & t_1^* & t_2^* & \cdots & t_{K-1}^* \\ 0 & t_0^* & t_1^* & \cdots & t_{K-2}^* \\ 0 & 0 & t_0^* & \cdots & t_{K-3}^* \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & t_0^* \end{pmatrix} \begin{pmatrix} f(t_1) \\ f(t_2) \\ f(t_3) \\ \vdots \\ f(t_K) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad (13)$$

where $f(t_i)$ is defined as $f(t_i) = (m_4(|t_i|^2-1)+2m_2^2\sum_{l\neq i}|t_l|^2)\cdot t_i$, where we have defined $m_i = E[|a(n)|^i]$, with $a(n)$ representing the complex QAM input symbols. By examining Equation (13) we readily see that it accepts the trivial solution $\mathbf{t} = \mathbf{0}$, as well as solutions, which obey $t_i = 0$ for $i < \delta$ and $t_i \neq 0$, $f(t_i) = 0$ for $i \geq \delta$. This solution gives the following system of equations for $i \geq \delta$:

$$m_4(|t_i|^2 - 1) + 2m_2^2 \sum_{l\neq i} |t_l|^2 = 0, \quad i = \delta, \cdots, K. \quad (14)$$

Equation (14) can be rewritten as:

$$\begin{pmatrix} m_4 & 2m_2^2 & \cdots & 2m_2^2 \\ 2m_2^2 & m_4 & \cdots & 2m_2^2 \\ \vdots & \vdots & \ddots & \vdots \\ 2m_2^2 & 2m_2^2 & \cdots & m_4 \end{pmatrix} \begin{pmatrix} |t_\delta|^2 \\ |t_{\delta+1}|^2 \\ \vdots \\ |t_K|^2 \end{pmatrix} = \begin{pmatrix} m_4 \\ m_4 \\ \vdots \\ m_4 \end{pmatrix}. \quad (15)$$

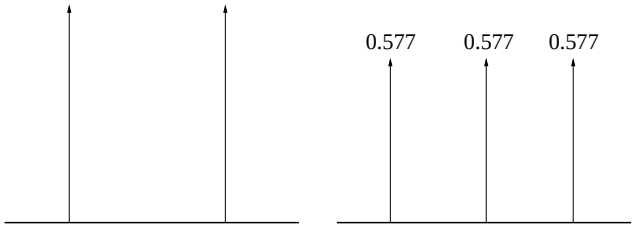

Figure 2: Two and three–path channel models used in the performance comparison.

The solution of Equation (15) is given by:

$$|t_i|^2 = P^2 = \frac{m_4}{m_4 + 2m_2^2(K-\delta)}, \quad i = \delta, \cdots, K, \qquad (16)$$

which corresponds exactly to the CMA's stationary points, as it was shown in [7]. We have found that the stationary points can assume one of the following forms:

- $[0, \cdots, 0, e^{j\cdot\phi_\delta}, \cdots, e^{j\cdot\phi_{K-\delta}}, 0, \cdots]^T$, where $K - \delta + 1$ defines the number of non-zero vector entries,
- $[0, \cdots, 0]^T$,

which are similar to the CMA's stationary points. The sole difference that in this case the CIR does not play a multiplicative role. Therefore the requirement of an infinite number of equalizer taps is not necessary in this case. At this stage we have to ascertain, which of the stationary points are minima, saddle points and maxima. Explicitly, we only want the points of the form $[0, \cdots, 0, e^{j\cdot\phi}, 0, \cdots, 0]^T$ to be local minima and all the other points to be unstable equilibria. It is easy to see that this is true, since the cost function of this algorithm is the same as the cost function of the CMA. Foschini [7] has studied the stationary points of this cost function, which are the same for the proposed DFE-CMA as well. Foschini also proved that only the tap vector having a single non-zero element - i.e. the vectors of the form $[0, \cdots, 0, e^{j\cdot\phi_0}, 0, \cdots, 0]^T$- constitute a minimum and all the others - i.e. the vectors of the form $[0, \cdots, 0]^T$ as well as $[0, \cdots, 0, e^{j\cdot\phi_\delta}, \cdots, e^{j\cdot\phi_{K-\delta}}]^T$, $K \neq \delta$ - are saddle points. This also proves that the DFE-CMA cost function, which is the same, also assumes only the above non-zero tap vector as a minimum.
Having discussed the convergence of the DFE-CMA, we will now characterize its performance.

4. PERFORMANCE RESULTS

In this section the proposed algorithm is benchmarked against the classic CMA [2], demonstrating the improved performance of the DFE-CMA, when the channel is difficult to equalize, since the CIR contains two or three equal paths. In Figure 2 the channel models used are shown. In Figure 3 the equalizer's Mean Squared Error (MSE) learning curves are plotted for Quadrature Phase Shift Keying (QPSK) using the two-path channel model of Figure 2 for a Signal-to-Noise-Ratio (SNR) of $30dB$. Observe the substantially improved performance of the DFE-CMA, explained by the fact that a reduced number of taps is needed, thus inflicting less convolutional noise. The step–size parameter λ is $5\text{x}10^{-4}$. In Figure 4 the same performance curves are given

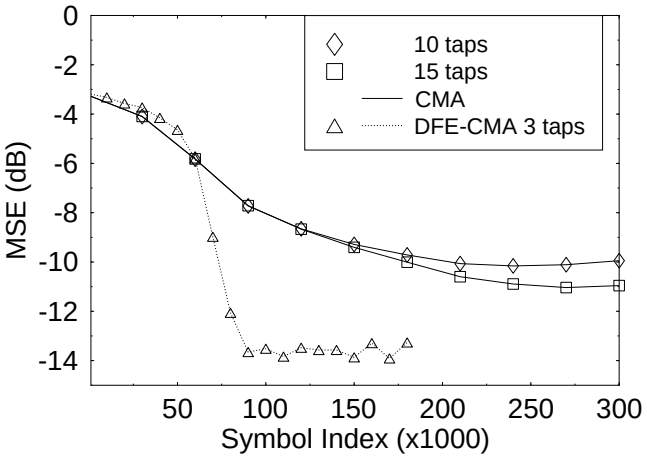

Figure 3: The MSE learning curves for QPSK, using the two-path model of Figure 2.

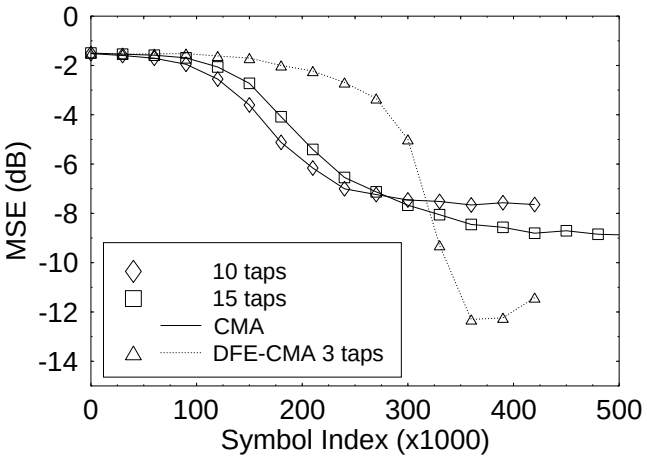

Figure 4: The MSE learning curves for 16-QAM, using the three-path model of Figure 2.

for 16-QAM and for the three-path channel model of Figure 2. Again, we observe better performance for the DFE-CMA, combined in this case with slower convergence. It is expected that the convergence of the DFE-CMA should be faster than that of the CMA, as a consequence of its reduced number of taps. On the other hand, since a DFE is typically less stable than its open–loop counterpart, it may require more time to move towards the point of equilibrium. Following the above arguments, what we actually see in Figure 4 and also in Figure 3 is that after an initial period of slow or no convergence the DFE "switches" to fast convergence mode and then converges in a "waterfall" fashion. On the other hand, the conventional CMA would require more taps to give a similar performance, and hence the convolutional noise would increase the MSE. In Table 2 we give a complexity estimate for the update of the DFE-CMA and also for the classic CMA, based on the number of real additions and multiplications required at each symbol interval. In this table M is the number of feedback taps, while N is the number of feedforward taps. It is clear from Table 2 that in general the DFE-CMA is more complex, because its complexity depends on M^2, even when we use only

Algorithm	Additions	Multiplications
DFE-CMA	$16M(M+N)+31M+15N-16$	$8M(M+N)+29M+13N$
DFE-CMA ($N=1$)	$16M^2+47M+15$	$8M^2+37M+21$
CMA	$4N+2$	$4N+4$

Table 2: Complexity estimation of the DFE-CMA and the CMA.

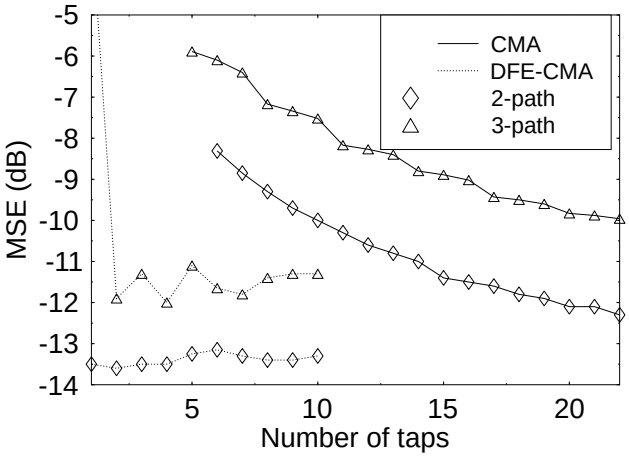

Figure 5: The minimum MSE as a function of the equalizer's number of taps for 16-QAM using the two- and three-path channel models of Figure 2 at an SNR of $30dB$.

one feedforward tap. However, in the situations of interest the number of feedback taps required is determined by the channels' delay spread expressed in terms of the number of symbol intervals, which is rather low in the scenario considered. However, for these channels a high-order CMA based equalizer would be required. As an example, in Figure 5 we plot the minimum MSE as a function of the number of equalizer taps at an SNR of $30dB$. We observe that the DFE-CMA MSE function has a minimum, depending on the channel model and also on the SNR. In this graph we observe the expected minimum introduced as a result of two factors:

- The additive channel noise, implying that the MSE cannot be less than a certain threshold and
- The convolutional noise, which exists because of:
  - the finite equalizer length and
  - due to the fact that this is a blind equalizer and it does not converge with infinite precision, especially in the presence of additive channel noise.

Similar minima exist in the CMA MSE function at a higher number of taps. It is clear that in the two-path channel scenario the DFE-CMA requires only one feedback tap, while the CMA requires at least 20 taps. In Table 2 we also observe that the corresponding complexities are 78 additions, 64 multiplications for the DFE-CMA and 82 additions, 84 multiplications for the CMA. Therefore, the CMA is more complex in this scenario and its performance is inferior to that of the proposed algorithm.

5. CONCLUSIONS

In summary, a novel blind equalizer was proposed and studied, extending the conventional CMA to the DFE-CMA. The benefit of this equalizer in comparison to the conventional CMA is its better performance, when used to equalize channels exhibiting zeros on the unit circle. Over other types of channels it exhibits similar performance to the CMA but at a higher complexity. Finally, unlike the CMA, this equalizer cannot converge to any CIR-duration dependent undesirable local minima.

A. DERIVATIVES

In this appendix we find the expressions for the derivatives of Equation (4) wrt $\mathbf{c}$ and $\mathbf{w}$:

$$\begin{aligned}\frac{\partial J(n)}{\partial \mathbf{c}} &= 2\cdot e(n)\cdot[z^*(n)\cdot\frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{c}}\cdot\mathbf{w} \\ &+ z(n)\cdot(\mathbf{y}^*(n)+\frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{c}}\cdot\mathbf{w}^*)]\end{aligned} \quad (17)$$

and

$$\begin{aligned}\frac{\partial J(n)}{\partial \mathbf{w}} &= 2\cdot e(n)\cdot[z^*(n)\cdot\frac{\partial \mathbf{z}^T(n-1)}{\partial \mathbf{w}}\cdot\mathbf{w} \\ &+ z(n)\cdot(\mathbf{z}^*(n-1)+\frac{\partial \mathbf{z}^H(n-1)}{\partial \mathbf{w}}\cdot\mathbf{w}^*)].\end{aligned} \quad (18)$$

As we observe from Equations (18) and (19), the calculation of the gradients requires an estimate of the derivative of the equalized vectors with respect to the equalizer's feedforward and feedback tap-vectors. These derivatives are constituted by matrices and their elements will be close to zero, when the equalizer is close to convergence, however these low–valued elements are needed in order to bring the equalizer to this point. In order to estimate these derivatives we develop a recursive loop. Firstly, we define the following variable matrices and vectors:

$$\mathbf{c} = \mathbf{c_R} + j\mathbf{c_I} \quad (19)$$

$$\mathbf{w} = \mathbf{w_R} + j\mathbf{w_I} \quad (20)$$

$$\begin{pmatrix}\mathbf{C_{RR}}(n) & \mathbf{C_{RI}}(n)\\ \mathbf{C_{IR}}(n) & \mathbf{C_{II}}(n)\end{pmatrix} = \begin{pmatrix}\frac{\partial \mathbf{z}_R^T(n)}{\partial \mathbf{c}_R} & \frac{\partial \mathbf{z}_R^T(n)}{\partial \mathbf{c}_I}\\ \frac{\partial \mathbf{z}_I^T(n)}{\partial \mathbf{c}_R} & \frac{\partial \mathbf{z}_I^T(n)}{\partial \mathbf{c}_I}\end{pmatrix} \quad (21)$$

$$\begin{aligned}\mathbf{C}(n)=\frac{\partial \mathbf{z}^T(n)}{\partial \mathbf{c}} &= \mathbf{C_{RR}}(n)-\mathbf{C_{II}}(n)+j(\mathbf{C_{RI}}(n)+\mathbf{C_{IR}}(n))\\ \bar{\mathbf{C}}(n)=\frac{\partial \mathbf{z}^H(n)}{\partial \mathbf{c}} &= \mathbf{C_{RR}}(n)+\mathbf{C_{II}}(n)+j(\mathbf{C_{RI}}(n)-\mathbf{C_{IR}}(n))\end{aligned} \quad (22)$$

$$\begin{pmatrix} \mathbf{W_{RR}}(n) & \mathbf{W_{RI}}(n) \\ \mathbf{W_{IR}}(n) & \mathbf{W_{II}}(n) \end{pmatrix} = \begin{pmatrix} \frac{\partial \mathbf{z}_R^T(n)}{\partial \mathbf{w}_R} & \frac{\partial \mathbf{z}_R^T(n)}{\partial \mathbf{w}_I} \\ \frac{\partial \mathbf{z}_I^T(n)}{\partial \mathbf{w}_R} & \frac{\partial \mathbf{z}_I^T(n)}{\partial \mathbf{w}_I} \end{pmatrix} \quad (23)$$

$$\begin{aligned} \mathbf{W}(n) &= \frac{\partial \mathbf{z}^T(n)}{\partial \mathbf{w}} \\ &= \mathbf{W_{RR}}(n) - \mathbf{W_{II}}(n) + j(\mathbf{W_{RI}}(n) + \mathbf{W_{IR}}(n)). \\ \bar{\mathbf{W}}(n) &= \frac{\partial \mathbf{z}^H(n)}{\partial \mathbf{w}} \\ &= \mathbf{W_{RR}}(n) + \mathbf{W_{II}}(n) + j(\mathbf{W_{RI}}(n) - \mathbf{W_{IR}}(n)). \end{aligned} \quad (24)$$

With these definitions, Equations (18) and (19) become:

$$\begin{aligned} \frac{\partial J(n)}{\partial \mathbf{c}} &= 4 \cdot e(n) \cdot [0.5 \cdot z(n) \cdot \mathbf{y}^*(n) \\ &+ z_R(n) \cdot ((\mathbf{C_{RR}} \cdot \mathbf{w_R} - \mathbf{C_{IR}} \cdot \mathbf{w_I}) \\ &+ z_I(n) \cdot (\mathbf{C_{RR}} \cdot \mathbf{w_I} + \mathbf{C_{IR}} \cdot \mathbf{w_R}) \\ &+ j(z_R(n) \cdot ((\mathbf{C_{RI}} \cdot \mathbf{w_R} - \mathbf{C_{II}} \cdot \mathbf{w_I}) \\ &+ z_I(n) \cdot (\mathbf{C_{RI}} \cdot \mathbf{w_I} + \mathbf{C_{II}} \cdot \mathbf{w_R}))))] \\ &= 4 \cdot e(n) \cdot [0.5 \cdot z(n) \cdot \mathbf{y}^*(n) \\ &+ \begin{pmatrix} \mathbf{w_R} & -\mathbf{w_I} \\ j\mathbf{w_I} & j\mathbf{w_R} \end{pmatrix} \begin{pmatrix} \mathbf{C_{RR}} & -\mathbf{C_{RI}} \\ \mathbf{C_{IR}} & \mathbf{C_{II}} \end{pmatrix} \cdot \begin{pmatrix} z_R(n) \\ z_I(n) \end{pmatrix} \\ \frac{\partial J(n)}{\partial \mathbf{w}} &= 4 \cdot e(n) \cdot [0.5 \cdot z(n) \cdot \mathbf{z}^*(n-1) \\ &+ z_R(n) \cdot ((\mathbf{W_{RR}} \cdot \mathbf{w_R} - \mathbf{W_{IR}} \cdot \mathbf{w_I}) \\ &+ z_I(n) \cdot (\mathbf{W_{RR}} \cdot \mathbf{w_I} + \mathbf{W_{IR}} \cdot \mathbf{w_R}) \\ &+ j(z_R(n) \cdot ((\mathbf{W_{RI}} \cdot \mathbf{w_R} - \mathbf{W_{II}} \cdot \mathbf{w_I}) \\ &+ z_I(n) \cdot (\mathbf{W_{RI}} \cdot \mathbf{w_I} + \mathbf{W_{II}} \cdot \mathbf{w_R}))))] \\ &= 4 \cdot e(n) \cdot [0.5 \cdot z(n) \cdot \mathbf{z}^*(n-1) \\ &+ \begin{pmatrix} \mathbf{w_R} & -\mathbf{w_I} \\ j\mathbf{w_I} & j\mathbf{w_R} \end{pmatrix} \begin{pmatrix} \mathbf{W_{RR}} & -\mathbf{W_{RI}} \\ \mathbf{W_{IR}} & \mathbf{W_{II}} \end{pmatrix} \cdot \begin{pmatrix} z_R(n) \\ z_I(n) \end{pmatrix} \end{aligned}$$

From these definitions and from Equation (2) we can derive the recursion for the gradients as:

$$\begin{aligned} \mathbf{C_{RR}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{y_R}(n) \\ &+ (\mathbf{C_{RR}}(n-1) \cdot \mathbf{w_R} - \mathbf{C_{IR}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{C_{RI}}(n) \cdot \mathbf{f} &= -0.5 \cdot \mathbf{y_I}(n) \\ &+ (\mathbf{C_{RI}}(n-1) \cdot \mathbf{w_R} - \mathbf{C_{II}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{C_{IR}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{y_I}(n) \\ &+ (\mathbf{C_{IR}}(n-1) \cdot \mathbf{w_R} + \mathbf{C_{RR}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{C_{II}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{y_R}(n) \\ &+ (\mathbf{C_{II}}(n-1) \cdot \mathbf{w_R} + \mathbf{C_{RI}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{W_{RR}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{z_R}(n-1) \\ &+ (\mathbf{W_{RR}}(n-1) \cdot \mathbf{w_R} - \mathbf{W_{IR}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{W_{RI}}(n) \cdot \mathbf{f} &= -0.5 \cdot \mathbf{z_I}(n-1) \\ &+ (\mathbf{W_{RI}}(n-1) \cdot \mathbf{w_R} - \mathbf{W_{II}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{W_{IR}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{z_I}(n-1) \\ &+ (\mathbf{W_{IR}}(n-1) \cdot \mathbf{w_R} + \mathbf{W_{RR}}(n-1) \cdot \mathbf{w_I}) \\ \mathbf{W_{II}}(n) \cdot \mathbf{f} &= 0.5 \cdot \mathbf{z_R}(n-1) \\ &+ (\mathbf{W_{II}}(n-1) \cdot \mathbf{w_R} + \mathbf{W_{RI}}(n-1) \cdot \mathbf{w_I}) \end{aligned} \quad (25)$$

where $\mathbf{f} = [1, 0, 0, \cdots, 0]^T$ is a column vector used to extract the first column of the matrix, which it multiplies. The matrices $\mathbf{C}(n)$ and $\mathbf{W}(n)$ actually contain the N most recent columns of the derivative of the received and equalized signals respectively, with respect to the relevant equalizer tap-vector. Therefore, at every update, these columns are shifted so that the most obsolete one is replaced by the recently updated column from Equations (25) and (26). We can further streamline Equations (25) and (26) as:

$$\begin{aligned} \begin{pmatrix} \mathbf{C_{RR}} & \mathbf{C_{RI}} \\ \mathbf{C_{IR}} & \mathbf{C_{II}} \end{pmatrix}(n) \cdot \mathbf{f} &= 0.5 \cdot \begin{pmatrix} \mathbf{y_R} & -\mathbf{y_I} \\ \mathbf{y_I} & \mathbf{y_R} \end{pmatrix}(n) \\ &+ \begin{pmatrix} \mathbf{w_R} & -\mathbf{w_I} \\ \mathbf{w_I} & \mathbf{w_R} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{C_{RR}} & \mathbf{C_{RI}} \\ \mathbf{C_{IR}} & \mathbf{C_{II}} \end{pmatrix}(n-1) \end{aligned} \quad (27)$$

$$\begin{aligned} \begin{pmatrix} \mathbf{W_{RR}} & \mathbf{W_{RI}} \\ \mathbf{W_{IR}} & \mathbf{W_{II}} \end{pmatrix}(n) \cdot \mathbf{f} &= 0.5 \cdot \begin{pmatrix} \mathbf{z_R} & -\mathbf{z_I} \\ \mathbf{z_I} & \mathbf{z_R} \end{pmatrix}(n-1) \\ &+ \begin{pmatrix} \mathbf{w_R} & -\mathbf{w_I} \\ \mathbf{w_I} & \mathbf{w_R} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{W_{RR}} & \mathbf{W_{RI}} \\ \mathbf{W_{IR}} & \mathbf{W_{II}} \end{pmatrix}(n-1). \end{aligned} \quad (28)$$

B. REFERENCES

[1] L. Hanzo, W.T. Webb, T. Keller: Single- and Multi-carrier Quadrature Amplitude Modulation: Principles and Applications for Personal Communications, WATM and Broadcasting; IEEE Press-John Wiley, 2000

[2] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimensional data communication systems," *IEEE Transactions on Communications*, vol. COM-28, pp. 1867–1875, November 1980.

[3] A. Bouttier, "Truly recursive blind equalization algorithm," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, Part 6 (of 6)*, (Seattle, USA), pp. 3381–3384, May 12 – 15 1998.

[4] L. Tong and D. Liu, "Blind predictive decision-feedback equalization via the constant modulus algorithm," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, v 5, (1997)*, (Munich, Germany), pp. 3901–3904, April 21 – 24 1997.

[5] L. Tong, D. Liu, and H. Zeng, "On blind decision feedback equalization," in *Proceedings of the 1996 30th Asilomar Conference on Signals, Systems and Computers, Part 1 (of 2)*, (Los Alamitos, USA), pp. 305–309, Nov 3 – 6 1996.

[6] K. N. Oh and Y. O. Chin, "New blind equalization techniques based on constant modulus algorithm," in *Proceedings of the 1995 IEEE Global Telecommunications Conference, Part 2 (of 3)*, (Singapore), pp. 865–869, Nov 14 – 16 1995.

[7] G. J. Foschini, "Equalizing without altering or deleting data," *AT&T Technical Journal*, vol. 64, pp. 1885–1911, October 1985.

[8] Z. Ding, R. A. Kennedy, B. D. O. Anderson, and R. C. Johnson, "Ill-convergence of Godard blind equalizers in data communications systems," *IEEE Transactions on Communications*, vol. COM-39, pp. 1313–1327, September 1991.

[9] K. Wesolowsky, "Analysis and properties of the modified constant modulus algorithm for blind equalization," *European Transactions on Telecommunications and Related Technologies*, vol. 3, pp. 225–230, May–June 1992.