A NEW STRATEGY FOR THE BLIND MMSE EQUALIZATION

Khaled F. Abu-Salem, Chunbo Luo, and Yu Gong

School of Systems Engineering, University of Reading
RG6 6AY, Reading, UK
phone: + (44) 118 3788588 your phone, fax: + (44) 118 3788583 , email: y.gong, c.luo, k.f.abusalem@reading.ac.uk
web: www.rdg.ac.uk

ABSTRACT

This paper describes a new strategy for the blind equalization so that the blind Constant Module Algorithm (CMA) can be smoothly switched to the decision- directed (DD) equalization. First, we propose a combination approach by running the CMA and DD equalization simultaneously to obtain a smooth switch between them. We then describe an "anchoring process" to eliminate the effect from the CMA at the steady state to achieve low residual noise. The overall equalization can be regarded as the DD equalization being anchored by the combination approach. Numerical simulations are given to verify the proposed strategy.

Index terms - Blind equalization, CMA, adaptive filtering.

1. INTRODUCTION

The blind minimum-mean-square-error (MMSE) equalizer is of particular interest in many systems as it requires no training symbols. Perhaps the simplest blind equalizer is the so-called decision-directed (DD) equalizer, which is achieved by replacing the training symbols in a classic training-based MMSE equalizer with the hard decision of the equalizer outputs.

Although the DD equalizer can converge to the optimum tap setting in absence of noise [1], it is difficult to start up for channels causing severe inter-symbol interference (ISI), or when the "eye" diagram is closed. In [2], a "stop-and-go" strategy was proposed so that the decision-directed adaptation takes place only when the reliability of the hard decision can be determined. But the convergence of such a strategy may be slow especially when the ISI becomes so severe that few reliable equalization output is available.

The well known Constant Modulus algorithm (CMA), on the other hand, can achieve blind equalization not relying on hard decision [3]. Based on higher order statistics, however, the CMA suffers from slow convergence rate and high residual noise. It thus usually switches to the DD mode after the convergence. This brings up an issue of when such switch should take place. Traditional implementations usually make the switch at some pre-set time instant determined either by experience or to the worst case scenario, implying that often the switch is too late and sometimes is too early for severe channels.

In general, an ideal switching strategy between the CMA and DD equalization should achieve the following requirements:

- Smoothness: Switch occurs automatically and smoothly so that, when the equalizer starts to converge, the DD equalization gradually takes over the CMA at the earliest possible time, long before the traditional abrupt switch is able to happen, leading to fast convergence; on the other hand, the DD equalization can also automatically switch back to the CMA if it loses convergence due to, for example, a sudden change in the channel.
- Completeness: At the steady-state when the equalizer has converged, the effect from the CMA should be completely removed and only the DD equalization determines the overall equalization so that the residual noise is minimized.

It is clear that the traditional switch does not satisfy the smoothness requirement for the ideal switch. More flexible strategies are desirable. There exist some algorithms for making a smooth switch between the CMA and DD equalization. A typical of them is the so-called GPEA-G algorithm [4] which was derived following a similar approach to that in [5] for the Sato's algorithm. In the GPEA-G algorithm, the smooth switch is achieved by including the error signals from both the DD and the CMA equalization, denoted as e^{DD} ande^{CMA} respectively, into an overall pseudo error signal, e^{GPEA-G}, for the filter adaptation. Specially, $e^{GPEA-G}= k_1.e^{DD}+k_2.|e^{DD}|.e^{CMA}$, where k_1 and k_2 are two small constants. There are two problems related to this approach: First, the GPEA-G algorithm can not achieve the complete switch, because both k1 and k2 are randomly chosen without optimization, and the effect from the CMA always exists after the equalizer converges, resulting in high residua noise; Secondly and more seriously, the GPEA-G algorithm may suffer from instability, be-cause when $|e^{DD}|$ becomes large, it indirectly increases the step-size parameter of the

CMA which must however be significantly small to maintain stability [3].

In a recent paper [6], an alternative approach was proposed to improve the convergence of the blind equalizer by combining two independently running blind equalizers with different step-size parameters. Although the overall equalizer can well balance the fast convergence and the low residual noise between the two individual equalizers, it still has significantly higher residual noise than the DD approach. This so-called convex combination CMA (CC-CMA) approach was in fact inspired by the convex combination LMS algorithm proposed in [7].

In this paper, we describe a novel switch strategy between the CMA and DD equalization so that both requirements for the ideal switch can be satisfied. Also inspired by the convex combination LMS algorithm [7], we propose to combine the CMA and DD equalizer which are running independently. Unlike the convex combination LMS algorithm and its blind equalization counterpart in [6], however, here the optimum combination cannot be achieved by minimizing the output error because there are no training symbols available for the DD equalizer. Rather, a so-called anchored process will be introduced to minimize the effect from the CMA in the steady state. The proposed equalizer has significantly better performance than existing approaches including the GPEA-G algorithm and the CC-CMA algorithm pro-posed in [6]. The rest of this paper is organized as follows: Section 2 describes a combination approach to include both the CMA and DD equalization into one framework so that the smooth switch can be obtained; Section 3 describes an anchoring approach to suppress the effect of the CMA equalization from the overall process after the convergence, so that the completeness of the switch can be achieved; Section 4 gives numerical simulations to verify the proposed algorithm; Finally, Section 5 summarizes the paper.

2. A COMBINATION APPROACH

2.1 System Model

Without losing generality, we assume the decision delay for the equalization is zero. The details of the optimum decision delay search can be found in [8] and [9]. The received signal vector is given by:

$$y(n) = Hs(n) + n(n) \tag{1}$$

where H is the channel matrix, $s(n)$is the transmission signal vector, and $n(n)$ is the white channel noise vector.

In the CMA [3], the equalizer coefficient vector is adapted according to:

$$w_g(n+1) = w_g(n) + \mu_g e_g(n) y^*(n) \tag{2}$$

Where $e_g(n) = z_g(n).(R_2 - |z_g(n)|^2)$ which is called pseudo error signal, $z_g(n) = w_g^H(n)y(n)$ which is the equalization output, $R_2 = E|s(n)|^4/E|s(n)|^2$ which is determined by the high order statistics of the transmission data, and μ_g is the step-size parameter.

Since the CMA suffers from slow convergence and high residual noise, after (2) converges, it should switch to the DD mode as:

$$w_d(n+1) = w_d(n) + \mu_d e_d(n) y^*(n) \tag{3}$$

where $e_d(n) = \hat{s_d}(n) - z_d(n)$ and $\hat{s_d}(n)$ is the hard decision of the equalizer output $z_d(n)$. For clarity of exposition, we choose the subscripts of "g" and "d" to represent the CMA and DD modes respectively. Note that usually $\mu_g \ll \mu_d$ so the CMA converges much slower than its DD counterpart.

2.2 Convex Combination

In [7], Arenas-Garcia (et al) described a "convex combination" LMS algorithm in order to balance the conflict requirements of low MSE and fast convergence. This was achieved by proportionally summing the outputs from two independently running LMS filters with different step-size parameters. Similarly, we have both the CMA and DD equalizers run simultaneously, and combine them with an adjustable combination factor $\lambda(n)$ so that the overall equalizer becomes:

$$w_c(n) = \lambda(n) w_d(n) + (1 - \lambda(n)) w_g(n) \tag{4}$$

Then the smooth switch can be obtained by appropriately adapting $\lambda(n)$.

Unlike the convex combination LMS algorithm [7], unfortunately, the $\lambda(n)$in (4) cannot be adapted based on minimizing the overall mean square error (MSE). This is because there are no training symbols available in the blind equalization. Instead, the hard decision of the symbol estimates has to be used to adapt both the DD tap weight, i.e. $w_d(n)$, and $\lambda(n)$. This makes the $w_d(n)$, and $\lambda(n)$ adaptation be highly interactive to each other, diverging both of them. In fact, we found that even with the training symbols being used to adapt $\lambda(n)$ (but not $w_d(n)$) in a way similar to that in [7], the overall process is still unstable. Hence we let $\lambda(n)$ be adjusted in a "fixed" manner as:

$$\lambda(n) = exp\left(\frac{e_{p,d}^2(n)}{e_{p,g}^2(n)}\right) \tag{5}$$

where $e_{p,d}(n) = \hat{s_c}(n) - w_d^H(n+1)y(n)$, $e_{p,g}(n) = \hat{s_g}(n) - w_g^H(n+1)y(n)$, $\hat{s_c}(n)$ and $\hat{s_g}(n)$ are hard decision of the overall and the CMA outputs respectively. Since $\lambda(n)$ doesn't depend on its previous value $\lambda(n-1)$, the stability problem can be largely avoided. Moreover, we deliberately apply the a posterior error signals of $e_{p,d}(n)$ and

$e_{p,g}(n)$ in (5) to further "detach" the interaction between the tap weight and $\lambda(n)$ adaption for better stability. It is clear from (5) that $0 < \lambda(n) < 1$. To be specific, at the beginning when DD equalizer doesn't converge, we have $e_{p,g}^2(n) < e_{p,d}^2(n)$, so that $\lambda(n)$ is small and the overall equalizer is more determined by the CMA. As the CMA starts to converge, the output from the overall equalizer becomes more reliable, or $e_{p,d}^2(n)$ becomes smaller. This forces $\lambda(n)$ to increase, gradually increasing the contribution from the DD equalization to the overall process. And smooth switch is thus reached.

3. AN ANCHORING APPROACH

In order to achieve the completeness of the ideal switch, the value of $\lambda(n)$ needs to go up to one after the equalizer converges. Unfortunately, this cannot be achieved by (5) which does not depend on an optimum criterion. So the contribution from the CMA cannot be completely removed out of the overall equalization at the steady-state and a high residual noise will be observed. Further process is thus desirable.

3.1 Anchoring Process

In this section, we introduce an "anchoring process" such that only at specific intervals, namely the anchoring intervals, does the overall equalizer operate in a combination manner as in (4). At all other intervals, the overall equalizer simply runs in the traditional DD mode. This modifies the adaption rule for the overall equalizer to:

$$\begin{cases} w_c(n) = \lambda(n) w_d(n) + (1-\lambda(n)) w_g(n), if\ n\ mod\ N_{anc}(n) = 0 \\ w_c(n) = w_c(n-1) + \mu_c e_c(n-1) y^*(n-1), otherwise \end{cases} \quad (6)$$

where $e_c(n-1) = \hat{s_c}(n-1) - w_c^H(n-1)y(n-1)$, and $N_{anc}(n)$ is the time space between two adjacent anchoring intervals at time n. It is obvious when $N_{anc}(n) = 1$, (6) reduces to the normal combination approach of (4).

Such process can be viewed as a standard DD equalization being anchored by the combination approach. It is required that, at the initial stage, the anchoring space $N_{anc}(n)$is small, so that the overall process is intensively anchored by the combination approach and the convergence can be guaranteed. After the equalizer converges, $N_{anc}(n)$ becomes large so that the anchoring happens little and the contribution from the CMA can be ignored. In order to satisfy this requirement, $N_{anc}(n)$ needs to be increased when the equalizer is converging or has converged, and decreased when the equalizer diverges due to, for example, a sudden change in channel.
This translates into an adaptation rule as:

$$\begin{cases} N_{anc}(n+1) = N_{anc}(n) + \delta, if\ E[e_c^2(n)] =< E[e_c^2(n-\Delta)] \\ N_{anc}(n+1) = N_{anc}(n) + \delta, if\ E[e_c^2(n)] > E[e_c^2(n-\Delta)] \end{cases} \quad (7)$$

where both δ and Δ are positive integers which will be explained in more detail later.

3.2 Fractional Anchoring Space

Adapting $N_{anc}(n)$ directly based on (7) is very sensitive to the noise, if it converges at all. This is not only because $E[e_c^2(n)]$ is difficult to be accurately tracked in real time, but also because of the constraint that $N_{anc}(n)$ must be an integer.

Inspired by the "fractional filter length" algorithm proposed in [10] to adapt the filter length which must also be an integer, we introduce a concept of pseudo-fractional anchoring space, $n_{anc}(n)$, which can take fractional values. The anchoring space adaptation is then re-constructed based on $n_{anc}(n)$:

$$n_{anc}(n+1) = (n_{anc}(n+1) + \beta) - \gamma.(e_c^2(n) - e_c^2(n-\Delta)) \quad (8)$$

And the "true" anchoring space $N_{anc}(n)$ is obtained as:

$$N_{anc}(n) = \begin{cases} \lfloor n_{anc}(n+1) \rfloor, |N_{anc}(n) - n_{anc}(n+1)| \geq \delta \\ N_{anc}(n), otherwise \end{cases} \quad (9)$$

where δ and Δ are defined in the same way as in (7), γ is the step-size parameter, β is a small positive constant used to prevent $n_{anc}(n)$ from dropping after the equalizer converges, and $\lfloor . \rfloor$ rounds the embraced value to the nearest integer.

Specifically, a larger δ makes $N_{anc}(n)$ adaptation be less sensitive to the noise, but achieves less accuracy at the steady state. On the other hand, Δ is used to reduce the effect of the adaption noise on tracking the error signals. Therefore, the larger the Δ is, the smoother the adaption we can obtain, but the longer the delay we will observe. Moreover, in order to ensure stability, $n_{anc}(n)$ should be limited such that $N_{min} \leq n_{anc}(n) \leq N_{max}$. Normally we let N_{min}= 1 and N_{max} be a large value.

3.3 Discussions

Based on instantaneous rather than mean square errors, (8) adapts the anchoring space in an LMS manner with little complexity imposed. Taking expectation on both sides of (8) yields:

$$E[n_{anc}(n+1)] = (E[(n_{anc}(n)] + \beta) - \gamma.(E[e_c^2(n)] - E[e_c^2(n-\Delta)]) \quad (10)$$

Without losing generality, we assume $N_{anc}(0)$= $n_{anc}(0) = 1$. Then initially the overall equalizer is equivalent to a normal combination equalizer and able to converge, we have $E[e_c^2(n)] \leq E[e_c^2(n-\Delta)$, and then further from (10) we have

$E[n_{anc}(n+1)] > E[n_{anc}(n)]$. Thus at the steady-state, $N_{anc}(n)$ always reaches its maximum value of N_{max} in the mean. Further from (4), we can obtain the contribution from the CMA to the overall equalization at the steady state as:

$$P_{CMA} = \left(1-\lambda(\infty)\right).\frac{1}{N_{max}} \quad (11)$$

where $\lambda(\infty)$ is calculated according to (5) which is usually a value close to 1. As an illustration, we assume N_{max}= 50 and $\lambda(\infty)$ = 0.9 after the equalizer converges. It is clear from (11) that, at the steady state, the contribution from the CMA is only P_{CMA} = 0.2% which can be ignored in practice. The complete switch is thus reached.

On the other hand, when the equalizer diverges due to, for example, a sudden change in the channel, we have $E[e_c^2(n)] \gg E[e_c^2(n-\Delta)$. This makes $E[n_{anc}(n+1)] < E[n_{anc}(n)]$ so that the overall equalization becomes more and more determined by the combination approach and the convergence can be resumed.

It is interesting to note that if the combination approach is replaced by the traditional CMA, the anchoring process will not work. This is because that, unlike the combination approach, the CMA is a self-adapted process with no interaction with the overall equalization, constantly lifting the residual noise to a high level.

3.4 The Algorithm

With the above statements, a full description of the proposed algorithm is summarized as follows:

For every equalizer input y(n), n = 1, 2, 3,....

Update the DD and CMA equalizers simultaneously:

$$z_g(n) = w_g^H(n)y(n), z_d(n) = w_d^H(n)y(n),$$
$$e_g(n) = z_g(n).(R_2 - |z_g(n)|)^2, e_d(n) = \hat{s_c}(n) - z_d(n),$$
$$w_g(n+1) = w_g(n) + \mu_g e_g(n)y^*(n),$$
$$w_d(n+1) = w_d(n) + \mu_d e_d(n)y^*(n),$$

Update $N_{anc}(n)$:

$$n_{anc}(n+1) = (n_{anc}(n+1)+\beta) - \gamma.(e_c^2(n) - e_c^2(n-\Delta))$$

Limit $n_{anc}(n+1)$ in the range of $[N_{max}, N_{min}]$.

Obtain $N_{anc}(n+1)$ according to (9),

Update the overall equalizer:

If n MOD $N_{anc}(n+1) = 0$

$$e_{p,d}(n) = \hat{s_c}(n) - w_c^H(n)y(n),$$
$$e_{p,g}(n) = \hat{s_g}(n) - w_g^H(n)y(n),$$
$$\lambda(n) = \exp\left(-\frac{e_{p,d}^2(n)}{e_{p,g}^2(n)}\right),$$
$$w_c(n+1) = \lambda(n)w_d(n+1) + \left(1-\lambda(n)\right)w_g(n+1).$$

Else

$$w_c(n+1) = w_c(n) + \mu_c e_c(n)y^*(n),$$

Update the overall output:

$$z_c(n+1) = w_c^H(n+1)y(n+1),$$
$$\hat{s_c}(n+1) \textit{ is the hard decision of } z_c(n+1),$$
$$e_c(n+1) = \hat{s_c}(n+1) - w_c^H(n+1)y(n+1)$$

3.5 Numerical Simulations

In this section, the proposed approach is verified by numerical simulations. We consider two channels described in [2] and [11] with the impulse response vectors respectively given by:

Channel-I : [0.0410 + 0.0109j 0.0495 + 0.0123j 0.0672 + 0.0170j 0.0919 + 0.0235j 0.7920 + 0.1281j 0.3960 + 0.0871j 0.2715 + 0.0498j 0.2291 + 0.0414j 0.1287 + 0.0154j 0.1032 + 0.0119j]^T,

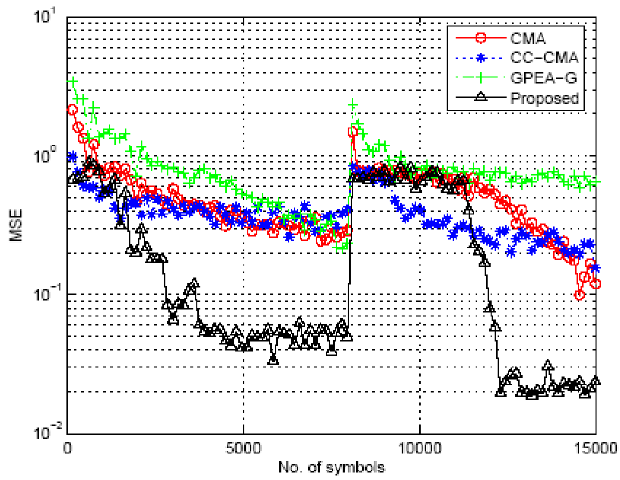

Fig. 3. The MSE learning curves for the channel with abrupt change.

Channel-II : [0-0.005j 0.009-0.024j 0.854-0.218j 0.049-0.016]^T (12)

In the below experiments, the channel noise for both channels are white Gaussian with signal- to-noise-ratio (SNR) at 20dB, 16QAM is used to modulate the transmission symbols, the filter length of the equalizer is 21, and the parameters for the anchoring space adaptation of (8) are set as follows: δ = 3, Δ= 50, γ= 5e-04, β = 1e-05, N_{min}= 0 and N_{max}= 50.

The proposed algorithm is compared with the classic CMA, the GPEA-G [4] and the convex combination CMA (CC-CMA) algorithms [6]. For fair comparison, the step-sizes for both the CMA and the GPEA-G algorithm are set as 1e-05, where the other parameters for the GPEA-G algorithm are chosen as same as those in [4]. The step-sizes for the CC-CMA algorithm are set as 5e-05 and 0.5e-05 for the two individual equalizers respectively.

All of the MSE learning curves below are obtained by averaging over 100 independent runs.

4. ABRUPT CHANNEL CHANGE EXPERIMENT

In this experiment, the channel is initially fixed at Channel-I and changed to Channel-II after 8,000 symbols. Fig. 3 compares the MSE performance for the CMA, CC-CMA, GPEA-G and proposed algorithms respectively, where the advantage of the proposed algorithm is clearly shown. It is interesting to point out that the GPEA-G algorithm converges to a very high MSE level after the abrupt channel change. This well matches our previous statement that a sudden rise in the MSE indirectly increases step-size of the CMA embedded in the GPEA-A algorithm, making it hard to converge.

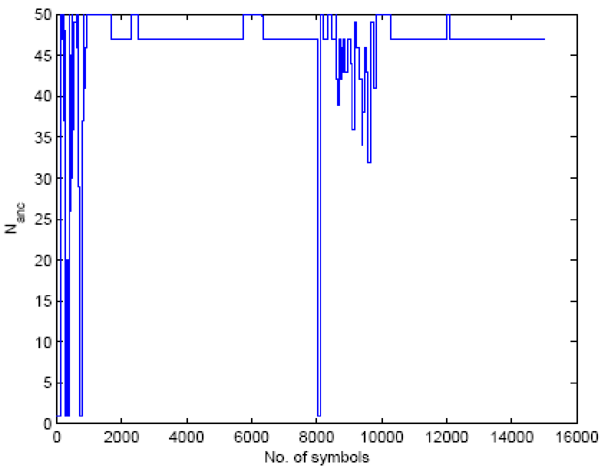

Fig. 4. The anchoring space learning curve for the channel with abrupt change.

Fig. 4 shows a typical learning curve of the anchoring space N_{anc} adaptation for a single run. As was expected, $N_{anc}(n)$ is small at the beginning or when the channel experiences abrupt change, and increases to a large value around N_{max} at the steady-state. A nearly ideal switch strategy is thus achieved.

5. CONCLUSIONS

This paper presented a new strategy for the blind equalization so that the blind CMA can be smoothly switched to the DD equalization. Compared with existing approaches, the proposed method of switch is smooth, complete and fast. Numerical simulation has been given to verify the analysis. The proposed approach describes a very useful method of implementing the blind equalizer in practice with only a little extra complexity being imposed.

6. REFERENCES

[1] O. Macchi and E. Eweda, "Convergence analysis of self-adaptive equalizers," IEEE Transactions on Information Theory, vol. IT-30, pp. 161-176, March 1984.

[2] G. Picchi and G. Prati, "Blind equalization and carrier recovery using a "stop-and-go" decision directed algorithm," IEEE Transactions on Communications, vol. COM-35, no. 9, pp. 877 - 887, Sept. 1987.

[3] D. N. Godard, "Self-recovering equalization and carrier tracking in two-dimentional data communication systems," IEEE Signal Communications, vol. COM-28, no. 11, pp. 1867 -1875, Nov. 1980.

[4] N. Deshpande, "Fast recovery equalization techniques for DTV," IEEE Transactions on broadcasting, vol. 43, no. 4, pp. 370 - 377, Dec. 1997.

[5] A. Benveniste and M. Goursat, "Blind equalizers," IEEE Transactions on communications,vol. COM-32, no. 8, pp. 871 - 883, Aug. 1984.

[6] M. T. M. Silva and V.H.Nascimento, "Convex combination of blind adaptive equalizers with different tracking capabilities," in Proceedings of ICASSP 2007, 2007, vol. III, pp. 457- 460.

[7] J. Arenas-Garcia, V. Gomez-Verdejo, and A. R. Figueiras-Vidal, "New algorithm for improved adaptive convex combination of LMS transversal ¯lters," IEEE Transaction on Signal Processing, vol. 54, no. 6, pp. 2239 - 2249, Dec. 2005.

[8] Y. Gong and C. F. N. Cowan, "Equalization with adaptive time lag," IEE Proceedings - Communications, vol. 152, no. 5, pp. 661 - 667, Oct. 2005.

[9] R. Lopez-Valcarce and F. Perez-Gonzalez, "Efficient reinitialization of the pre-whitened constant modulus algorithm," IEEE communications letters, vol. 5, no. 12, pp. 488 - 490, Dec. 1999.

[10] Y. Gong and C. F. N. Cowan, "An LMS style variable tap-length algorithm for structure adaptation," *IEEE Trans. on Signal Processing*, vol. 53, no. 7, pp. 2400 - 2407, July 2005.

[11] Cheolwoo You and Daesik Hong, "Nonlinear blind equalization schemes using complex-valued multilayer feedforward neural networks," *IEEE trans. on signal processing*, 1998.