RECURSIVE COST FUNCTION ADAPTATION FOR ECHO CANCELLATION

Corneliiu Rusu
Signal Processing Laboratory,
Tampere University of Technology,
P.O. BOX 553, SF-33101
Tampere, FINLAND
corneliiu@cs.tut.fi

Colin F.N. Cowan
Dept. of Electrical and Electronic Engineering,
The Queen’s University of Belfast,
Ashby Building, Stranmillis Road,
Belfast, BT7 3BY, U.K.
c.f.n.cowan@ee.qub.ac.uk

ABSTRACT
The goal of this paper is to introduce the RCFA (Recursive Cost Function Adaptation) algorithm. The derivation of the new algorithm does not use an estimator of the instantaneous error as the previous CFA (Cost Function Adaptation) algorithms did. In the RCFA case, the new error power is computed from the previous error power using an usual LMS recursive equation. The proposed method improves the sensitivity of the error power with respect to the noisy error, while the other benefits of the CFA algorithms in terms of the convergence speed and residual error remain. The properties of the new algorithm will be compared, using computer simulations, to standard LMS and LMF. The effect of the parameters involved in the design of the error power adaptive subsystem is also discussed.

1. INTRODUCTION

1.1. Quadratic and Non-quadratic Algorithms
The adaptive LMS (Least Mean Square) [1] algorithm has received a great deal of attention during the last decades, and it has been used in many applications due to its simplicity and relatively well-behaved performance. However, the convergence speed to optimal filter coefficients is relatively slow. This can be a drawback in the case of the digital echo cancellation, where one of the goals is to reduce the adaptation time, during which transmission of useful data is not possible. More recently, high order error power algorithms have been proposed. Walach and Widrow studied the use of the fourth power of the error, and the LMF (Least Mean Fourth) algorithm resulted [2]. Unfortunately, this algorithm has stability problems. Shah and Cowan investigated NQSGr (non-quadratic stochastic gradient algorithms) with arbitrary constant error power $r$ ($2 < r < 3$), and their results indicated that these improve stability [3].

1.2. Previous Cost Functions Algorithms
The CFA (Cost Function Adaptation) adaptive algorithm was first introduced in [4]. In this approach the error power is a function of the instantaneous error $r = r(e_k)$, and the new cost function $J_r = E[e_k^r]$. The derivation of this CFA stochastic gradient algorithm follows the cancellation of the posterior error output, encountered also in the affine projection and normalized LMS (NLMS) algorithms [5]. The resulted CFA algorithm is in fact a piecewise non-quadratic algorithm, and the error power is updated using the relationship:

$$r_{k+1} = r(e_k) = \frac{RE_{dB}}{|e_k|dB}$$

(1)

where $RE_{dB}$ is an arbitrary constant and $|e_k|dB$ is the error modulus, measured in dBs. The weights are computed using the simple recursive relation as in the case of non-quadratic error power $r_k$:

$$h_{k+1} = h_k + \mu BE_{k} \cdot x_k \cdot |e_k|^{r-1} \cdot \text{sgn}(e_k)$$

(2)

However, the error power must be updated in terms of a well-behaved estimator of the instantaneous error, otherwise instability can occur [6]. At the beginning two types of error mappings have been tried [4]: the running average of the modulus of the instantaneous error and the log running average of the squared instantaneous error, resulting the following CFA algorithms: the decreasing staircase power-error algorithm and the decreasing smooth power-error algorithm [4]. Then the normalised tap-error vector norm was used to reduce the sensitivity to the noisy error [6]. This error estimator is quite smooth, but sometimes it is difficult to calculate in practical problems.

A more general case was pointed out in [7], where the error power updating rule

$$r_{k+1} = r(e_k) = \frac{A}{|e_k|B^{r-1}}$$

(3)

was derived by enforcing the same direction of the instantaneous gradient as in the case of non-quadratic algorithms:

$$h_{k+1} = h_k + \mu B r_k \cdot x_k \cdot |e_k|^{r-1} \cdot \text{sgn}(e_k)$$

(4)

Here $A$ and $B$ are arbitrary constants and $B$ should be positive. If $B = 1$ we retrieve LMS ($r(e(k)) = 2$), LMF ($r(e(k)) = 4$) and NQSGr ($r(e(k)) = r =$const.). For $B = 2$ we have CFA.

The LCFA (Linear Cost Function Adaptation) algorithm is a special case of this family. The linear error power $r$ is adjusted in such manner that the error power is linearly decreasing during the time of adaptation. A new error mapping was implemented. This was done using the technique of the peak detector in classical amplitude modulation. We pass the logarithmic modulus of the instantaneous error through a first order recursive digital filter, the equivalent of the low-pass RC filter. The instantaneous error is usually very noisy, and its spectrum is quite flat. If the error is processed as mentioned above, the logarithmic output is linear decreasing, and also the error power is linear decreasing [7].
2. THE RECURSIVE COST FUNCTION ADAPTATION ALGORITHM

2.1. The adaptive filter framework

The simplified block diagram of the main echo-path identification system (EPIS) is shown in Figure 1. The vectors \( x_k \), \( \hat{h}_k \) and \( h_k \) are the transpose of the input observations vector, of the estimated filter coefficients vector, and respectively of the echo-path filter coefficients vector:

\[
x_k = [x_k, x_{k-1}, \ldots, x_{k-N+1}]^T,
\]

\[
\hat{h}_k = [\hat{h}_0, \hat{h}_1, \ldots, \hat{h}_{N-1}]^T,
\]

\[
h_k = [h_0, h_1, \ldots, h_{N-1}]^T.
\]

\( N \) is the number of filter coefficients. The echo path output signal \( y_k \) and the synthetic echo signal \( \hat{y}_k \) are given by

\[
y_k = h_k^T x_k, \quad \hat{y}_k = \hat{h}_k^T x_k.
\]

Inserting the attenuated far-end signal \( f_k \) in the error signal \( e_k \), we obtain

\[
e_k = y_k + f_k - \hat{y}_k = f_k - (\hat{h}_k - h_k)^T x_k,
\]

and with the tap-error vector \( \Delta h_k = \hat{h}_k - h_k \), it results

\[
e_k = f_k - \Delta h_k^T x_k.
\]

2.2. Derivation of the proposed algorithm

Consider the LMS algorithm

\[
\hat{h}_{k+1} = \hat{h}_k + 2\mu x_k e_k,
\]

where \( \mu \) is the step-size of the echo-path identification system. From the equation (8) we have

\[
\hat{h}_{k+1} = \hat{h}_k + 2\mu x_k (f_k - \Delta h_k^T x_k).
\]

If the channel is slowly varying, then we can subtract the echo-path filter coefficients vector from both sides of the equation (10). It follows

\[
\Delta h_{k+1} \approx \Delta h_k + 2\mu x_k (f_k - \Delta h_k^T x_k).
\]

It is the interest to determine the relationship between the instantaneous error \( e_k \) of the adaptive filter and the new error power \( r_{k+1} \) which is to be used to update the adaptive filter coefficients with the equation (2).

Suppose that we are not certain of the "unknown error power" \( r_{k+1} \). Reasoning as above, we can use the LMS algorithm and compute an "estimate of the new error power" \( \hat{r}_{k+1} \). For this reason we need the "near-end signal", and this will be the instantaneous error \( e_k \), because \( r_k \) is expected to be a function of \( e_k \). We need also the "attenuated far-end signal". It must have similar statistics properties as the "near-end signal". From the available signals we select the input observation sample \( x_k \), which is subject to an attenuation \( \phi_k \). The attenuation might be constant or not, whether we use some appropriate averages of the attenuated far-end signal \( f_k \) or simply \( f_k \). We thus have the useful equations:

\[
\Delta r_k = \hat{r}_k - r_k,
\]

\[
e_k = r_k + \phi_k x_k - \hat{r}_k = \phi_k x_k - \Delta r_k e_k,
\]

\[
\hat{r}_{k+1} = \hat{r}_k + 2 \cdot \rho \cdot e_k \cdot e_k,
\]

where \( e_k \) is the error signal and \( \rho \) is the step-size of the error power adaptation subsystem (EPAS). Now the equations (12) can give us
the estimated error power, and thus $\hat{r}_k$ can be used to update the weights with the equation

$$\hat{h}_{k+1} = \hat{h}_k + \mu \hat{r}_k \hat{w}_k |\epsilon_k|^2 \cdot \text{sgn}(\epsilon_k).$$  

The equations (12) for EPAS, and (13) for EPIS, together with Figure 1 and Figure 2 define the proposed recursive cost function algorithm.

From the first two of the (12) equations we obtain

$$\Delta r_{k+1} = \Delta r_k + 2 \cdot \rho \epsilon_k \cdot \epsilon_k$$

$$= \Delta r_k + 2 \cdot \rho \epsilon_k \cdot (\varphi_k \epsilon_k - \Delta r_k \epsilon_k)$$

$$= 2 \rho \varphi_k \epsilon_k \epsilon_k + \Delta r_k (1 - 2 \rho \epsilon_k^2).$$  

(15)

Applying previous recursion $P$ times with respect to $k$, we conclude that

$$\Delta r_{P+1} = \alpha_P + \beta_P \alpha_{P-1}$$

$$+ \beta_P \beta_{P-1} \alpha_{P-2} + \cdots + \beta_P \beta_{P-1} \cdots \beta_1 \alpha_0$$

$$+ \beta_P \beta_{P-1} \cdots \beta_1 \beta_0 \Delta r_0,$$

where

$$\alpha_k = 2 \rho \varphi_k \epsilon_k, \quad \beta_k = 1 - 2 \rho \epsilon_k^2.$$  

(17)

The convergence of the error power adaptation subsystem is done by the convergence of the product

$$\prod_{k=0}^{\infty} (1 - 2 \rho \epsilon_k^2).$$  

(18)

This is equivalent [8] with the convergence of the series

$$\sum_{k=0}^{\infty} \epsilon_k^2.$$  

(19)

However the stability of the overall adaptive system is more difficult to predict and will be the goal of a future task.

3. SIMULATIONS

In order to test the proposed algorithm, the following framework was used:

- The channel considered has one zero at the origin and one pole at 0.8.
- The number of filter coefficients is $N = 40$.
- The input signal is a binary one ($x_k = \pm 1$).
- The attenuated far-end signal is modeled by an independent random bipolar sequence ($f_k = \pm f$).
- The level $f$ of the attenuated far-end signal was -20dB.
- The step-size of the main adaptive filter is $\mu = 5 \cdot 10^{-4}$.
- The performance measure is the normalised form of the tap-error vector norm:

$$p_k = \frac{\|\hat{h}_k - h_k\|}{\|h_k\|}.$$  

(20)

- The learning curves obtained are the average of 20 runs.
- The unknown error power $r_0$ is a constant function.
- The initial estimated error power is $\hat{r}_0 = 4$.
- The attenuation $\varphi_k$ is constant ($\varphi_k = f$).
Three types of results are presented in this paper.

Figure 3 shows a comparison between LMS, LMF and RCFA ($\rho = 0.001, r_k = 1$) algorithms, from the convergence speed and steady-state point of view. It is clear now that RCFA has a faster convergence than both algorithms, and the steady-state properties are the same as of the LMS algorithm.

Figures 4 (for learning curves) and Figure 5 (for estimated error power $r_k$) illustrate the performances of the RCFA algorithms if the EPAS step-size changes. For a small step-size ($\rho = 0.0005$), the estimated error power decreases slowly, and as a consequence the respective RCFA algorithm behaves closer to LMF or NQSGr, ($2.8 < r < 3.5$). For $\rho = 0.0015$, the corresponding RCFA has a faster convergence, but the steady-state is worst than for the LMS algorithm ($r_{\infty} = 1.4$).

The same type of comparison was done from the unknown error power $r_k$ point of view. The choice of the constant function $r_k$ affects both the convergence rate and the steady-state (Figure 6). Also the estimated error power $r_k$ is changed (Figure 7). Clearly, a trade-off should be done between the parameters involved in the design of this complex adaptive filter, i.e. $\rho, r_k$, and respectively $\mu$.

The plots from Figures 5 and 7 show also that the error power is not very sensitive to noisy error during adaptation and steady-state. The decrease of the estimated error power is smooth, and almost monotonic.

We notice also in our simulations that the RCFA algorithm has a better stability than the LMF and other CFA algorithms. An exact assessment of this effect might be the goal a future paper. However we suppose that the initial fast decrease of the estimated error power is one of the contributed factors.

4. CONCLUSIONS
In this paper, the new recursive cost function adaptation algorithm has been proposed. It has been shown that the RCFA algorithm gives better results compared to standard LMS and LMF for data echo cancellation. Also the behavior for different initialization parameters was discussed. However there is still a lot of work in the future, for instance the challenge of the stability for the overall RCFA algorithm.

5. REFERENCES