Partial Update Single Lag Autocorrelation Minimization for Channel Shortening

By Mahmud Grira & Jonathon A. Chambers
Center of Digital Signal Processing, Cardiff School of Engineering Queen’s Buildings, PO Box 925, Cardiff CF24 0YF, Wales, UK, {giraram, chambersj}@cf.ac.uk

Abstract

In this paper a novel partial update single lag autocorrelation minimization (PUS-LAM) algorithm is proposed. This algorithm has low computational complexity whilst retaining essentially identical performance to the single lag autocorrelation minimization (SLAM) algorithm proposed by Nawaz and Chambers. Simulation studies on eight representative digital subscriber line channels show the ability of PUSLAM to achieve channel shortening and hence a certain level of bitrate within a multicarrier system.

1. Introduction

In multicarrier modulation systems (MCM), such as asymmetrical digital subscriber line (ADSL) transceivers, each symbol consists of samples to be transmitted to the receiver plus a cyclic prefix (CP) of length v, (see for example Melsa, Younce & Rohrs (1996)). The CP is the last v samples of the original N samples to be transmitted. The CP is inserted between blocks to combat inter-symbol interference (ISI) and inter-channel interference (ICI). The length of the CP should at least be equal to the order of the channel impulse response. At the receiver the CP is removed, the remaining N samples are then processed by the receiver. To combat ISI, the impulse response of the channel must be of length v + 1 or shorter than a CP of length v. Since the efficiency of the transceiver is reduced by the introduction of the CP it is therefore desirable either to make v as small as possible or to choose a large N. Selecting large N will increase the computational complexity, system delay, and memory requirements of the transceiver. Also the length of the channel’s impulse response varies from channel to channel, so to achieve reasonable efficiency a large v and large N have to be chosen. To overcome these problems a short time-domain equalizer (TEQ), usually an FIR filter, is typically placed in the receiver. The purpose of this filter is to shorten the impulse response of the effective channel. The impulse response of the effective channel needs to be shorter than the length of the CP. The length of the shortened impulse response filter and CP are usually fixed a priori and not changed from channel to channel. A low complexity blind adaptive algorithm to design a TEQ, called sum-squared auto-correlation minimization (SAM) was proposed in (see for example Balakrishnan, Martin & Johnson (2003)) which achieves channel shortening by minimizing the sum-squared autocorrelation terms of the effective channel impulse response outside a window of a desired length. The drawback with SAM is that it has a significant computational complexity, whereas in the single lag autocorrelation minimization (SLAM) algorithm (see for example Nawaz & Chambers (2004)) identical channel shortening can be achieved for subscriber line channels by minimizing the square of only a single autocorrelation, with much reduced computational complexity.
2. System Model

The system model is shown in Figure (1). The input signal \( x(n) \) is the source sequence to be transmitted through a linear finite-impulse-response (FIR) channel \( h \) of length \((L_h + 1)\) taps. \( r(n) \) is the received signal, which will be filtered through an \((L_w + 1)\)-tap TEQ with an impulse response vector \( w \) to obtain the output sequence \( y(n) \). We denote \( c = h \cdot w \) as the shortened or effective channel assuming \( w \) is in steady-state. We also assume that \( 2L_c < N \) holds. The received sequence \( r(n) \) is

\[
r(n) = \sum_{k=0}^{L_h} h(k)x(n-k) + v(n)
\]

and the output of the TEQ \( y(n) \) is given by

\[
y(n) = \sum_{k=0}^{L_w} w(k)r(n-k) = w^T r_n
\]

where \( r_n = [r(n) \ r(n-1) \cdots r(n-L_w)]^T \) and \( w \) is the impulse response vector of the TEQ \( w = [w_0 \ w_1 \ w_2 \cdots w_{L_w}]^T \).

3. The notion of SAM and SLAM algorithms

The idea of SAM is based on the fact that for the effective channel \( c \) to have zero taps outside a window of size \((v+1)\), its autocorrelation values should be zero outside a window of size \((2v+1)\). In SAM the auto-correlation sequence of the combined channel-equalizer impulse response is given by

\[
R_{cc}(l) = \sum_{k=0}^{L_w} c(k)c(k - l)
\]
Partial Update Channel Shortening Algorithms

and for a shortened channel, it must satisfy
\[ R_{cc}(l) = 0, \forall |l| > v \]  
(3.2)

Then the cost function \( J_{v+1} \) in SAM is defined based upon minimizing the sum-squared auto-correlation terms, i.e.,
\[ J_{v+1} = \sum_{l=v+1}^{L_c} R_{cc}(l)^2 \]  
(3.3)

The update equation for SAM can be written as
\[ w_{k+1} = w_k - 2\mu \sum_{l=v+1}^{L_c} \left\{ \left( \sum_{n=kN}^{(k+1)N-1} \frac{y(n)y(n-l)}{N} \right) \times \left( \sum_{n=kN}^{(k+1)N-1} \left\{ \frac{y(n)r_{n-l} + y(n-l)r_n}{N} \right\} \right) \right\} \]  
(3.4)

On the other hand, SLAM is based on the fact that a single autocorrelation at a lag greater than the guard interval is a measure of the presence of the channel outside the desired guard interval, therefore minimizing only this single autocorrelation also gives the channel shortening effect. This is particularly applicable to subscriber line channels which are essentially minimum phase. In SLAM the auto-correlation sequence of the combined channel-equalizer impulse response is given by
\[ R_{cc}(l) = \sum_{k=0}^{L_c} c(k)c(k-l) \]  
(3.5)

and for a shortened channel, it must satisfy
\[ R_{cc}(l) = 0, l = v + 1 \]  
(3.6)

Then the cost function \( J_{v+1} \) in SLAM is defined based upon minimizing the squared auto-correlation of the effective channel at lag \( l = v + 1 \), i.e.,
\[ J_{v+1} = R_{cc}(l)^2, l = v + 1 \]  
(3.7)

The update equation for SLAM can be written as
\[ w_{k+1} = w_k - 2\mu \{ E[y(n)y(n-l)] \} \{ E[y(n)r_{n-l} + y(n-l)r_n] \} \]  
(3.8)

4. Partial Update Channel Shortening Algorithms

As in any partial update algorithm, the aim of partial updating is to update a portion of the coefficients instead of the entire set of coefficients. Our proposal here is to apply the partial update method to the two channel shortening algorithms (Blind, Adaptive Channel Shortening By Sum-Squared Auto-Correlation Minimization SAM) algorithm (see for example Balakrishnan, Martin & Johnson (2003)), and Blind Adaptive Channel Shortening By Single Lag Autocorrelation Minimization (SLAM) algorithm (see for example Nawaz & Chambers (2004)), and achieve the same performance whilst reducing the computational complexity, the proposed algorithms are called the Partial Update SAM algorithm (PUSAM) and Partial Update SLAM algorithm (PUSLAM). In these algorithms the coefficients in the middle (in our simulation case eight will be the middle) are updated \( N_B - 1 \) times, that is achieved by introducing a vector which contains ones in the middle and zeros outside the middle, then at the \( N_B \)th time the outside ones are updated. The new vectors called “mask1,” and “mask2” are created as \( Mask_1 = [0000111111110000] \) \( Mask_2 = [1111000000001111] \). We define matrices \( M_k = diag(Mask_k) \), where \( k = 1, 2 \).
The partial-update SAM (PU-SAM) algorithm can therefore be written as:

$$w^{k+1} = w^k - 2\mu \times M_k \times$$

$$\sum_{l=0+1}^{L_c} \left\{ \sum_{n=kN}^{(k+1)N-1} \frac{y(n)y(n-l)}{N} \right\} \times \left\{ \sum_{n=kN}^{(k+1)N-1} \left( \frac{y(n)\Gamma(n-l)+y(n-l)\Gamma(n)}{N} \right) \right\}$$

(4.1)

In this work $N_B = 5$, so that if for the $N_B - 1$ times $M_k = M_1$ otherwise $M_k = M_2$. The Partial-Update SLAM (PU-SLAM) algorithm can be written as:

$$w^{k+1} = w^k - 2\mu \times M_k \times$$

$$\left\{ \sum_{n=kN_{avg}}^{(k+1)N_{avg}-1} \frac{y(n)y(n-l)}{N_{avg}} \right\} \times \left\{ \sum_{n=kN_{avg}}^{(k+1)N_{avg}-1} \left( \frac{y(n)\Gamma(n-l)+y(n-l)\Gamma(n)}{N_{avg}} \right) \right\}$$

(4.2)

and the same strategy for selection of $M_k$ is used as in PU-SAM. The proposed algorithms have essentially achieved the same performance as SAM and SLAM algorithms in terms of higher bit rates and shortening the channel as shown in the simulations results, the advantage of the proposed algorithms is that they essentially achieve the same performance whilst updating only half of the coefficients at each iteration which implies less computational complexity.

5. Simulations

The simulations show that the PUSAM and PUSLAM essentially achieve the same bit rate as SAM and SLAM. In the simulations, we use eight different channels as shown in Figures (3) and (4) to make sure that our proposed algorithms perform similarly with different channels.
6. Conclusion
New partial update blind channel shortening algorithms have been proposed. The proposed algorithms essentially achieve the same result in terms of reducing the effective channel length as SAM and SLAM with half the complexity, the disadvantage of (PUSAM) and (PUSLAM) is that they can converge slower than SAM and SLAM algorithms.

REFERENCES