Thank you for downloading this document from the RMIT Research Repository.

The RMIT Research Repository is an open access database showcasing the research outputs of RMIT University researchers.


Citation:

See this record in the RMIT Research Repository at:

Version: Accepted Manuscript

Copyright Statement: © 2011 IEEE

Link to Published Version:
N/A
Single-bit Adaptive Channel Equalization for Narrowband Signals

Tayab Din Memon
School of Electrical and Computer Engineering
Royal Melbourne Institute of Technology
Melbourne, Victoria
tayab.memon@rmit.edu.au

Paul Beckett
School of Electrical and Computer Engineering,
Royal Melbourne Institute of Technology
Melbourne, Victoria, Australia
pbeckett@rmit.edu.au

Amin Z Sadik,
School of Electrical and Computer Engineering
Royal Melbourne Institute of Technology
Melbourne, Victoria

Peter O’Shea
School of Electrical and Computer Engineering
Queensland University of Technology,
Brisbane, Queensland

Abstract—In this paper, a new design of a single-bit adaptive channel equalization is proposed using sigma delta modulation and a single-bit block LMS algorithm. With correlated narrowband input signals, this model is capable to converge and provide equivalent equalization filter with improvement in the SNR and very low Symbol Error Rate (SER). The input, filter coefficients and output values are all in single-bit and ternary format that results in a reduction in hardware complexity compared to traditional multi-bit channel equalization. Additionally, the technique avoids the need for successive conversion from multi-bit to single bit and back at the receiver and transmitter stages.

Keywords: Single-bit Channel Equalization, Sigma Delta Modulation, Single-bit Block LMS, Ternary Filter

I. INTRODUCTION

Channel equalization has widespread applications in contemporary communication systems, both wired and wireless [1, 2]. It is routinely used to mitigate the effects of inter symbol interference (ISI) caused by limited channel bandwidth and noise disturbances [3]. Various efficient algorithms have been developed to resolve these issues in last few decades while keeping the channel equalizer coefficients and input in multi-bit domain. Using a multi-bit format for coefficients and input implies higher chip area and lower performance from a hardware perspective [1, 2]. It also increases the complexity of the multiplication stages that ultimately impacts the system performance.

Recently, a single-bit adaptive algorithm has been proposed to suppress noise in narrowband signals using sigma delta modulated filters [4, 5]. In this filter, the primary inputs, internal signals, adaptive filter coefficients, error term and final output all are in a single-bit format. In [6-8], it was shown that ternary FIR filters implemented in FPGAs using pipelined and non-pipelined organizations could exhibit superior performance compared to their multi-bit counterparts. Thus, single-bit DSP systems have the tendency to reduce the multiplier complexity that gives better performance with comparable chip area.

In this paper we extend our single-bit adaptive filter work of [4, 5] towards novel single-bit channel equalization using narrowband signal (Figure 1). All signals are kept in single-bit format, including the final output after using second order sigma delta modulator. As a sigma delta modulator operates at a higher over sampling ratio (OSR) compared to the Nyquist rate, it offers better noise shaping. In this work, it is assumed that the input signal is oversampled that was modulated using 2-PAM. Hence, sign function was used to take the samples \( \pm 1 \) in MATLAB. This oversampled signal is shaped by the channel impairments and then passed through sigma delta modulation to bring it in single-bit format as channel equalization may be performed in single-bit domain. In this way overall system remains in single-bit domain and there is no need to convert the multi-bit signals back into single-bit and vice versa at transmitter and receiver ends again.

The multi-bit channel equalization model currently used in the contemporary communication is modified with a novel single-bit model as shown in figure 1[1]. In this model, the received signal, equalization coefficients, and final output all are in single-bit domain. The delayed version of the input is used as desired input signal.

The remainder of this paper proceeds as follows. In section 2, we discuss the system design of the channel equalizer using Block LMS and the single-bit BLMS algorithm. In section 3, simulation results of the single-bit channel equalizer are presented. Finally, we summarize and conclude the paper and point to future work.

II. SYSTEM DESIGN

The transmitted narrow band input signal in channel equalization can be described as [3, 9]:

\[
x(n) = \sum_{p=0}^{m} d_p g_T(n - pT)
\]  

(0.1)

where \( g_T(n) \) is the basic pulse shape that is selected to control the spectral characteristics of the transmitted signal, \( d_p \) is the sequence of the transmitted information symbols from a signal constellation consisting of \( M \) points, and \( T \) the signal interval: \( (1/T) \) is symbol rate[3]. In our case, we are not taking into ac-
count the pulse shaping filter for the sake of simplicity so the transmitted input sequence is:

\[ x(n) = d_n \quad (0.2) \]

The transmitted signal is shaped by the inter-symbol interference (ISI) due to channel impairments and additive noise that can be mathematically represented as:

\[ r(n) = \alpha \left( \sum_{i=0}^{N-1} h_d \cdot x(n-i) + v(n) \right) \quad (0.3) \]

where \( h_d \) represents the channel impulse response and \( v(n) \) is additive noise, considered to be white Gaussian noise with zero mean and variance \( \sigma^2 \). This received signal is passed through a second order sigma delta modulator to convert the input into a binary \{+1,-1\} as shown in Fig. 1. However, to maintain the dynamic range (DR) of the second order sigma delta modulator i.e., \{+1,-1\}, a gain parameter of \( \alpha \) has been introduced to ensure that the convolution sum stays within the prescribed dynamic range.

A. Single-bit Channel Equalization

The primary objective of the channel equalization filter is to reduce the ISI affects and increase the output SNR. Various algorithms have been reported to date, including zero forcing equalizer (ZFE), LMS, Block LMS, NLMS, and RLS etc. Most of the proposed algorithms except ZFE use Minimum Mean Square Error (MMSE) methods to adjust the equalization coefficients. We have used Block LMS due to its ability to adjust more samples as compared to the LMS while maintaining equivalent performance with small computational complexity [10].

The single-bit Block LMS algorithm (SBLMS) can be derived using a standard Block LMS (BLMS) algorithm. The general block diagram of the SBLMS is shown in Fig. 3. The input to the SBLMS is the sigma delta modulated Gaussian noise corrupted signal, \( r(n) \). The \( N \times 1 \) single-bit input signal vector at the time index \( n \) can be expressed as:

\[ r(n) = [r(n), r(n-1), \ldots, r(n-N+1)]^T \quad (0.4) \]

where \([,]^T\) indicates transposition, \( N \) represents the interpolated single-bit filter order, \( q \) is the multi-bit channel filter order.

The \( N \times 1 \) single-bit coefficient vector of the equalization filter \( h_{eq} \) at time index \( n \) can be denoted by:

\[ h(n) = [h_1(n), h_2(n), \ldots, h_{N-1}(n)]^T \quad (0.5) \]

and the single-bit estimation output vector at time \( n \) as:

\[ y_{eq}(n) = [y_{eq}(n), y_{eq}(n-1), \ldots, y_{eq}(n-N+1)]^T \quad (0.6) \]

As we are considering here the block LMS so to work with blocks, let \( j \) refers to the block index that is related to the original sampling index \( n \) as follows:

\[ n = j\Delta + i, \quad j = 1,2,3,4, 5, \ldots; \quad i = 0,1,2, \ldots, N-1 \quad (0.7) \]

where \( \Delta \) denotes the block length and \( i \) is the block index, \( j \) the number of blocks index so that \( j = n/\Delta \). The LMS algo-
ithm is a special case of the BLMS where the block length is 1. Generally, block length is considered with reference to the order of the filter i.e., $\Delta \geq N$, $\Delta < N$, or $\Delta = N$. In general, the second and third cases are preferred to the first. In this paper we have considered the second case i.e., $\Delta < N$. Additionally, due to the higher order of the single-bit equalization filter we consider $\Delta$ and the filter order ($N$) in the power-of-2.

The $N \times \Delta$ single-bit input data for block $j$ is therefore defined by the set $[r(j\Delta + i)]_{i=0}^{\Delta-1}$, which can be expressed in matrix form as:

$$U(j) = [r(j\Delta), r(j\Delta + 1), \ldots, r(j\Delta + \Delta - 1)] \quad (0.8)$$

The tap weight vector $h(j)$ remains constant over this block of input data. The estimated output of this filter, $\{\hat{r}(j\Delta + i)\}$ produced by the equalization filter in response to the input signal vector $r(j\Delta + i)$ is given by:

$$\hat{r}(j\Delta + i) = h^T(j) r(j\Delta + i) \quad (0.9)$$

However, this expected output is the result of a convolution operation between single-bit input samples and single-bit coefficients so the output will be multi-bit. To keep the entire system within the single-bit domain this output is passed through the second order sigma delta modulation (Fig. 2).

As the dynamic range of the second order sigma delta modulator should be in the range of $[+1, -1]$ to achieve the best SNR, a scale factor is used to maintain this range. An important measure of the SDM is to keep flat signal frequency range over the desired band of the frequency. Thus, the SDM of the expected signal should not modify the specification of the estimated output. The single-bit version of the expected output can be described as:

$$y_o(j\Delta + i) = \text{sgn}(\hat{r}(j\Delta + i)) \quad (0.10)$$

where $\beta$ is a scaling factor and the sgn function is given by:

$$\text{sgn}(\delta) = \begin{cases} +1 & \delta > 0 \\ 0 & \delta = 0 \\ -1 & \delta < 0 \end{cases}$$

The second order sigma delta modulator used here has the following transfer function:

$$H(z) = S(z)z^{-1} + Q(z)(1 - 2z^{-1} + z^{-2}) \quad (0.11)$$

where $S(z)$ represents the signal transfer function and $E(z)$ the quantization noise transfer functions. The noise shaping effect of the $\Sigma \Delta M$ is evident from the presence of the filtering term, $(1 - 2z^{-1} + z^{-2})$ acting on the noise term, $E(z)$. This quantization effect of the sigma delta modulator can easily be approximated by using a linear approximation [11]. Therefore the expected output with quantization noise shaping can be expressed as:

$$y_o(j\Delta + i) = \beta \hat{r}(j\Delta + i) + q_{yo}(j\Delta + i) \quad (0.12)$$

where $q_{yo}$ represents the shaped quantization noise due to the modulation effect that is generated in the response to the convolution between noise impulse response coefficients and block of the quantization noise. Hence $\Delta \times 1$ quantization noise vector can be defined as:

$$q_{yo}(j) = [qyo(j\Delta), qyo(j\Delta + 1), \ldots, qyo(j\Delta + \Delta - 1)]^T \quad (0.13)$$

Thus, the single-bit output can be expressed as:

$$y_o(j\Delta + i) = \beta h^T(j) r(j\Delta + i) + q_{yo}(j\Delta + i) \quad (0.14)$$

or in matrix form as:

$$y_o(k) = \beta U^T(j) h(j) + q_{yo}(j) \quad (0.15)$$

where $U(j)$ is $N \times \Delta$ size matrix that can be generated using the Toeplitz built-in function in Matlab or by exploiting the matrix format.

Similar to the multi-bit block LMS, the coefficient update formula in single-bit domain takes into account the error term. The error is simply considered to be the desired signal subtracted from the expected output, defined in block terms as:

$$e(j\Delta + i) = r(j\Delta + i) - y_o(j\Delta + i) \quad (0.16)$$

In simple form the error is:

$$e(j) = r(j) - y_o(j) \quad (0.17)$$

Now the weights update formula for the single-bit domain can be described as:

$$h(j + 1) = \text{sgn}[h(k) + mu \ast e(k)] \quad (0.18)$$

where $mu$ is the controlling factor and $h(k)$ are the coefficients in the range of $[+1, 0, -1]$, that is the function of the ternary quantizer. The ternary format of the coefficients results a harsh quantization affect (i.e. it introduces quantization noise) that can be expressed by using linear approximation as shown previously. It is worth noting that we are not considering the averaging terms of input and error as usually done in multi-bit block LMS algorithms [10] as the single-bit nature of the system will not add any further improvement by including these terms. Therefore, the updating function can be approximated as:

$$h(j + 1) = h(j) + mu \ast [r(j) - y_o(j)] + q_w(j) \quad (0.19)$$

where the quantization noise $q_w(j)$ is a $N \times 1$ vector. Thus updating function becomes:

$$h(j + 1) = h(j) + mu \ast [r(j) - \beta U^T(j) h(j) + q_{yo}(j)] + q_w(j) \quad (0.20)$$

In these equations, all the single-bit adaptive process parameters and quantization error components are given.

III. SINGLE-BIT EQUALIZATION ALGORITHM TERMS

A. Stability of the Algorithm

The estimated output of the input and the equalizer coefficients is in multi-bit format. To transform this output into the single-bit domain, a second order sigma delta modulator is introduced (Fig. 2). However, the dynamic range of the second order sigma delta modulator that results in the best SNR,
and assures the overall stability of the system is \( \{+1,-1\} \). To ensure stability, a gain parameter \( \beta \) is introduced as shown in Fig. 2. Considering the non-negative values of the expected output in the range \((1, N)\) then this factor may be defined as:

\[
\frac{2}{N} < \beta < 2
\]  
(0.21)

The precise value of \( \beta \) can be achieved by using any adaptive SDM, such as are reported in [12].

B. Rate of Convergence

The rate of convergence in single-bit adaptive systems depends upon the factor \( mu \) defined in (0.18). Unlike multi-bit block LMS or LMS algorithms, the rate of convergence in single-bit adaptive systems has limited flexibility due to its single-bit nature (i.e., quantization). It may be selected from the range of \( mu \leq 0.5 \) or \( mu > 0.5 \). Each value of \( mu \) has different affect upon the rate of convergence. In case of \( mu<0.5 \) the adapting process will stall in few iterations due to the dominance of the single-bit coefficient factor when \( h (-1, +1) \), upon the mu and error term multiplication as shown in (0.18).

In the other two cases, the adaptive process will continue and will depend upon the error term as well as the tap values. In case of \( mu=0.5 \), the coefficients and error term will be ternary in nature i.e., \(+1,0,-1\). In the case where \( mu>0.5 \), there would be no stalling, but multiplication factor of mu and error term would be in the range of \(+2,0,-2\) , which would be larger than the ternary taps \(+1,0,-1\). This setting would therefore result in harsh iterative convergence and more average errors with smaller SNR improvement. In this paper, we have used the mu value equal to 0.5.

C. Minimum Mean Square Error

Minimum mean squared error (MMSE) is the term used for the measurement of algorithm performance in LMS or BLMS cases. In single-bit systems, the error is not a continuous function but is bounded to the range \(+2,0,-2\) , which makes it harder to determine the gradient analytically. However, the mean squared error may be determined in the same way as current adaptive algorithms. In this way, an ensemble average learning curve of the sample can be defined as:

\[
P(j\Delta+i) = E[d^j(j\Delta+i) - y^j(j\Delta+i)]^2
\]  
(0.22)

where E denotes the expectation operator. The ensemble-average learning curve over the interval of \( 0 \leq j \leq N \) is defined as the average over the \( O \) realizations as:

\[
\hat{P}(j\Delta+i) = \frac{1}{O} \sum_{j=1}^{O} |e^j(j\Delta+i)|^2
\]  
(0.23)

where \( \hat{P}(j\Delta+i) \) is the sample-average approximation of the actual learning curve. The desired response here is the delayed version of the input signal \( d \).

IV. SIMULATION AND DISCUSSION

In this work, the SBLMS has been simulated in MATLAB. A narrowband 11-tap low pass filter channel model \( (He) \) was selected to create an equivalent channel equalizer filter \( (He) \) in single-bit domain. We originally intended to oversample the channel and convert it into single-bit domain and achieve the equalized filter \( (He) \). However, our simulation results proved that equalization filter convergence is very difficult and gives a very high symbol error rate and lower improvement in the SNR. However, oversampling the channel in the multi-bit domain creates great complexity with few advantages [13]. Thus single-bit systems that have inherent additional quantization noise are highly affected by the channel oversampling, so that we kept the channel filter in its original format.

In our tests, the sinusoidal input signal \( d(n) \) was arbitrarily chosen to be \( f_s = 2000Hz \) with an amplitude of \( A = 0.5 \). It was assumed that input signal \( d(n) \) is an oversampled single-bit signal throughout the simulation. The Nyquist rate of the channel filter order is selected as \( c=11 \), and the oversampling ratio is chosen as \( OSR=128 \) and the equivalent filter order was defined by using the relationship \( OSR \times c \).

A. Symbol Error Rate (SER) at Varying input Training Samples

Initially, SER was calculated using varying input training samples that were recorded in decision directed mode. Hence, SDM oversampled input was filtered through the oversampled equalized channel filter model \( (He) \). However, in the single-bit domain it is not trivial to find the starting point due to delays introduced by the channel impairments. Thus, a for-loop was introduced to record the SER at 100 (starting from 0 to 99) values and extract their minimum value. The minimum SER recorded at the point was considered as the starting point, found to be near to 70-80s. The SER is shown at various input training samples in Fig. 4.

In a subsequent stage, the SER was recorded at varying SNR as shown in Fig. 5.

B. Signal-to-Noise Ratio (SNR)

The improvement in the SNR was a treated as a measure of the performance and was defined as the output divided by the input (in-band) SNR. An improvement in the output SNR was recorded at varying input SNR. Before doing this, an appropriate \( \beta \) factor was set to achieve the best SNR while keeping the in-band frequency same. Extensive simulations indicate an optimum around \( OSR=6.7 \). Simulations repeated under the same conditions sometimes show different performance due to the noise and ISI that continuously change the gain factor and therefore the dynamic range.

We have considered the best performance achieved through the simulation at \( OSR=128 \) and input sinusoid at \( f_o = 2000Hz \). It is evident that the best performance is
achievable at the full dynamic range of the expected output \(x\) that is \(\pm 1\). The results are given in Table 1.

![Figure 4. SER at varying input training samples](image)

![Figure 5. SER recorded at varying input SNR(dB)](image)

C. Minimum Mean Squared Error (MSE)

MMSE is calculated according to the definition given in (0.23). Averaged over 30 times, the mean squared error is shown in Fig. 5. It is evident from that the error trend is towards zero.

<table>
<thead>
<tr>
<th>No.</th>
<th>(\beta)</th>
<th>SNR</th>
<th>Psdb</th>
<th>SNR0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>OSR*6.7</td>
<td>2</td>
<td>-11.03</td>
<td>17.810</td>
</tr>
<tr>
<td>2</td>
<td>OSR*6.7</td>
<td>3</td>
<td>-12.03</td>
<td>17.44</td>
</tr>
<tr>
<td>3</td>
<td>OSR*6.7</td>
<td>4</td>
<td>13.03</td>
<td>17.35</td>
</tr>
<tr>
<td>4</td>
<td>OSR*6.7</td>
<td>5</td>
<td>-14.03</td>
<td>17.315</td>
</tr>
<tr>
<td>5</td>
<td>OSR*6.7</td>
<td>6</td>
<td>-15.03</td>
<td>17.99</td>
</tr>
<tr>
<td>6</td>
<td>OSR*6.7</td>
<td>7</td>
<td>-16.03</td>
<td>18.21</td>
</tr>
<tr>
<td>7</td>
<td>OSR*6.7</td>
<td>8</td>
<td>-17.03</td>
<td>17.637</td>
</tr>
<tr>
<td>8</td>
<td>OSR*6.7</td>
<td>9</td>
<td>-18.03</td>
<td>19.43</td>
</tr>
<tr>
<td>9</td>
<td>OSR*6.7</td>
<td>10</td>
<td>-19.03</td>
<td>19.48</td>
</tr>
<tr>
<td>10</td>
<td>OSR*6.7</td>
<td>15</td>
<td>-24.03</td>
<td>16.3</td>
</tr>
</tbody>
</table>

![Figure 6. Averaged MMSE recorded at 30-times](image)

V. CONCLUSION

In this paper, we have developed a novel single-bit adaptive channel equalization model for narrowband input signals. The overall system is kept within the single-bit domain including input, filter taps, final output and error terms. A narrow band low pass filter channel model was selected to demonstrate the model. The model exhibits improved SER over conventional techniques at varying training input pulses and at varying SNR. The average mean squared error was shown to trend towards zero. Improvement in the SNR was recorded at varying in-band input SNR. The model results in low hardware complexity, especially in FPGA devices.

VI. REFERENCES