A Computer Tool to Analyze the LMS Equalizer

Luciano L. Mendes[1], Francisco J. Fraga[2], Sandro A. Fasolo[3] and Estevan M. Lopes [4]

Abstract *— The purpose of this paper is to explore the concepts involved in adaptive equalization, using a didactical computer tool. The software developed uses the Least Mean Square algorithm to equalize a bit sequence transmitted in base band through a channel defined by the user. The program allows the user to define the size and the nature of the signal used as training sequence, the number of the taps of the equalizer filter and also the channel impulse response. All results are presented in graphics, which allow the user to analyze each step of the equalization process. This program was developed using the Matlab® platform with a Graphic User Interface (GUI).*

Index Terms — Adaptive Equalizer, Didactical Computer Tool, Training Sequence.

INTRODUCTION

This paper presents an educational approach for the study of adaptive equalizers using the LMS (Least Mean Square) algorithm [1]. This approach is realized using a simulation program developed with Matlab® platform, which allows the user to define the parameters of the system, and them analyze the effects of these parameters in the equalization process.

Equalization is fundamental in the reception of digital signals transmitted in channels that presents multiple paths and introduces Intersymbol Interference (ISI) [2][3]. The concepts involved in this process are complexes and must be clearly presented to the students. For this reason it is important to present approaches that allows the students to analyze the system and verify each step of this process. In this paper, the results obtained with the simulator will be presented and analyzed to demonstrate the didactical potential of the developed computer tool. Figure 1 presents the block diagram of a generic adaptive equalizer used in this paper.

SIMULATION STRUCTURE

The simulator was based in the block diagram presented by Figure 2 [1]. In this structure, at the beginning of the communication ($t = t_0$) the transmitter sends a training sequence (a_n) that is known by the receiver. The channel introduces ISI using the impulse response (b_n) provided by the user. The additive white gaussian noise (AWGN) can also be added to the signal and the received signal (r_n) is delivered to the adaptive equalizer. The LMS algorithm uses an error signal provided by the difference between the signal at the output of the equalizer and the training sequence generated at the receiver (reference sequence) to calculate the coefficients. The channel and the equalizer can be modeled as FIR [4] filters that can introduce delays in the signal. Thus, in this case, the training sequence must be properly delayed at the receptor to provide an accurate error signal to feedback the equalizer, allowing the equalizer to compare the reference signal (a_n) with the signal at the equalizer output. The obtained error signal (e_n) is then introduced in the LMS algorithm to update the coefficients that will be used to minimize the ISI introduced by the channel.

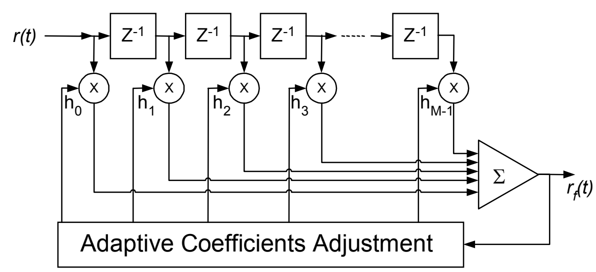

FIGURE 1
BLOCK DIAGRAM OF AN ADAPTIVE EQUALIZER.

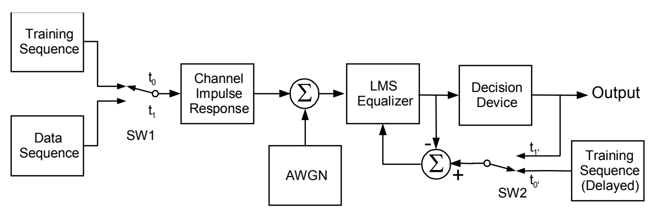

FIGURE 2
SIMULATION BLOCK DIAGRAM.

At the end of the training sequence ($t = t_1$), the equalizer has estimated the channel and the switches 1 and 2 change their positions. The transmission of the data sequence (c_n) begins and, at the receiver, the equalizer assumes that the decision device always estimates the received sequence correctly. The estimated sequence ($\hat{c}_n$) is then used to calculate the error signal that will be used as a feedback to the LMS algorithm. The bit error rate (BER) must be low enough to guarantee that the estimated sequence corresponds to the desired reference sequence. It means that the training

[1] Luciano Leonel Mendes, INATEL, Av. João de Camargo, 510, 37540-000, Santa Rita do Sapucaí, MG, Brazil, lucianol@inatel.br
[2] Francisco José Fraga, INATEL, Av. João de Camargo, 510, 37540-000, Santa Rita do Sapucaí, MG, Brazil, fraga@inatel.br
[3] Sandro Aadriano Fasolo, INATEL, Av. João de Camargo, 510, 37540-000, Santa Rita do Sapucaí, MG, Brazil, sandro.fasolo@inatel.br
[4] Estevan Marcelo Lopes, INATEL, Av. João de Camargo, 510, 37540-000, Santa Rita do Sapucaí, MG, Brazil, estevan@inatel.br

sequence must be long enough to allow the equalizer to estimate the channel correctly and, thus, to reduce the BER.

All sequences involved in this simulation are shown in graphics that allow the user to analyze each step of the equalization process.

SIMULATOR

Following, a group of simulations are presented to demonstrate the use of the simulator.

Simulation with Square Training Sequence

In this simulation, a square training sequence with 256 bits has been used to estimate the channel frequency response. The number of taps of the equalizer has been set to eight and the channel has been modeled for Medium ISI (which means that the channel impulse response is [1 0.2 -0.2 0 0]), with signal to noise ratio (SNR) of 60dB, which almost eliminates the interference of the noise. The data sequence length has been set to 512 bits. Figure 3 shows the transmitted training sequence and this sequence at the channel output. Note that the training sequence is a cyclic square waveform where each period is formed by five bits 1 (+1) and five bits 0 (-1). The amplitude distortion in the signal at the channel output is caused by the ISI.

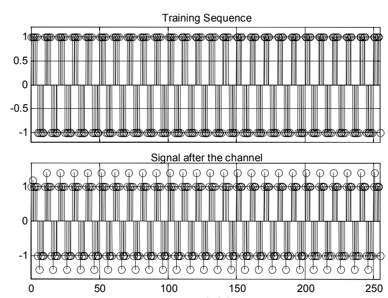

FIGURE 3
SQUARE TRAINING SEQUENCE AND THE OUTPUT OF THE CHANNEL.

The equalizer updates its taps at each received bit from the training sequence. Once defined the delay introduced by the channel, the received training sequence is subtracted from the reference sequence to obtain the error signal, which will be used to feedback the LMS algorithm and update the taps. The update process can be viewed in a dynamic graphic that shows the equalizer impulse response. This graphic is displayed whenever the taps are update. Figure 4 shows the equalizer impulse response at the end of the training sequence, where it is possible to note that the channel has not been correctly estimated, because the equalizer impulse response does not annul the channel impulse response. The distortions introduced by the channel are canceled when (1) is satisfied.

$$c(n) * h(n) = \sum_{k=0}^{N+K-1} h(k) \cdot b(k-n) = \delta(n) \tag{1}$$

Where $\delta(n)$ is the discrete-time impulse function and $h(n)$ is the equalizer impulse response.

Equation (1) shows that the convolution between the channel and equalizer impulse responses must result in a Dirac impulse. Figure 5 shows that the condition presented in (4) has not been reached.

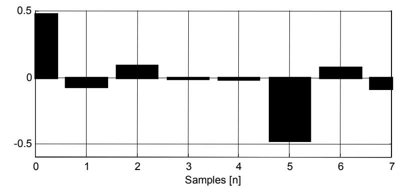

FIGURE 4
EQUALIZER IMPULSE RESPONSE.

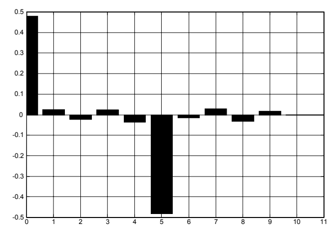

FIGURE 5
CONVOLUTION: EQUALIZER AND CHANNEL IMPULSIVE RESPONSE.

It is also important to analyze the equalization process in the frequency domain. The Fourier Transform of (1) allows one to define the condition to obtain a perfect equalization of the channel, in the frequency domain. Equation (2) shows this condition.

$$\begin{gathered} H(k) \cdot C(k) = 1 \\ |H(k) \cdot C(k)| = 1 \\ \angle H(k) + \angle C(k) = 0 \end{gathered} \tag{2}$$

Where $H(k)$ is the equalizer frequency response and $C(k)$ is the channel frequency response.

Figure 6 shows the frequency and phase response of the equalizer and channel, where it is possible to observe that the equalizer frequency response only satisfies (2) for some specifics values of frequency (-π, -0.6π, -0.2π, 0.2π, 0.6π e π). This occurs because the square training sequence has discrete spectral components, as shown in Figure 7.

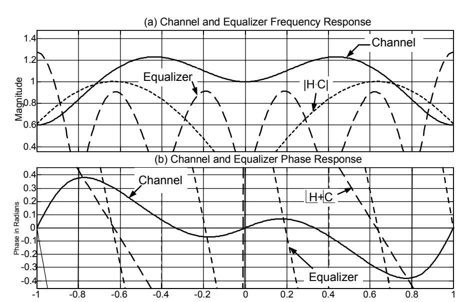

FIGURE 6
CHANNEL AND EQUALIZER FREQUENCY RESPONSE.

If the training sequence does not have spectral components in all frequencies, the equalizer cannot estimate the channel correctly. Because the equalizer corrected the channel in the frequencies where the training sequence has components, the interference in the sequence at the equalizer output has been eliminated.

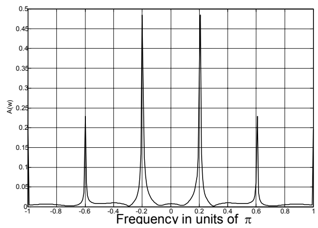

FIGURE 7

SPECTRUM OF THE SQUARE TRAINING SEQUENCE.

Figure 8 shows the error signal during the training sequence, where it is possible to note that the LMS algorithm is able to update the equalizer taps to minimize the error signal.

It can be seen that the equalizer approximates the received sequence to the desired sequence, removing the ISI introduced by the channel, as shown in Figure 9.

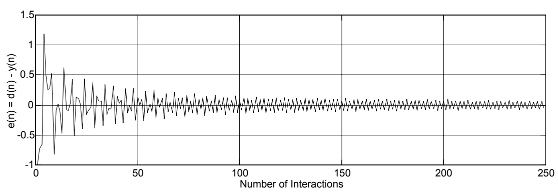

FIGURE 8

ERROR SIGNAL AND MEAN SQUARE ERROR.

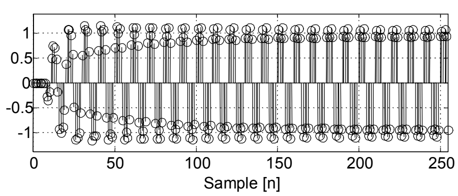

FIGURE 9

EQUALIZED TRAINING SEQUENCE.

At the end of the training sequence, the transmitter begins to send the data sequence and the equalizer begins to use the sequence at the output of the decision device as reference signal. To simulate a data sequence, the program generates a random bit sequence with uniform distribution [5]. The spectrum of a random sequence can be approximate to a constant value for all frequencies of interest, as showed in Figure 10.

To demonstrate the performance of the equalizer, the channel impulse response is randomly changed at the end of the training sequence. This procedure allows the user to analyze if the equalizer proposed is able to update its taps when the data sequence is transmitted. Figure 11 shows the transmitted data sequence and the received data sequence.

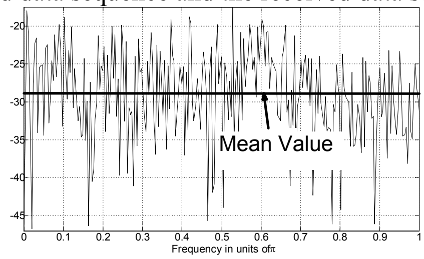

FIGURE 10

SPECTRUM OF THE DATA SEQUENCE.

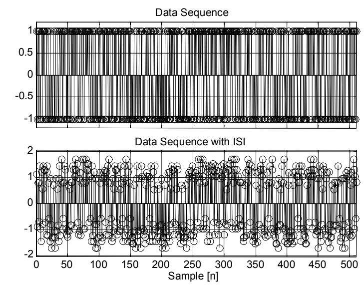

FIGURE 11

DATA SEQUENCE BEFORE AND AFTER THE CHANNEL.

Because the equalizer did not correctly estimate the channel during the training sequence, the BER at the beginning of the data sequence could be high. Thus, the reference sequence, that is the sequence estimated by the decision device, does not correspond to the desired sequence and the equalizer cannot remove the ISI from the received sequence. Figure 12 shows the equalizer and the channel frequency response. The equalizer frequency response does not represent the inverse of the channel frequency response, hence the data sequence delivered to the decision device is distorted by the ISI and the BER could be high. Figure 13 shows the equalized sequence.

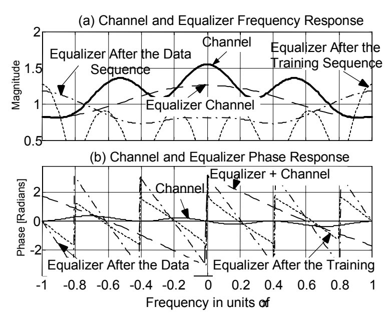

FIGURE 12

CHANNEL AND EQUALIZER FREQUENCY RESPONSE.

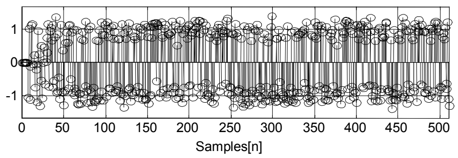

FIGURE 13
EQUALIZED SEQUENCE.

Simulation with Random Training Sequence

As showed in the previous session, it is not possible to use a square waveform to estimate the channel, because square waveforms have discrete spectral components. In this simulation, a random bipolar bit sequence with uniform distribution is used to estimate the channel. Any other parameter of the simulator has not been changed. Figure 14 shows the transmitted and received training sequence.

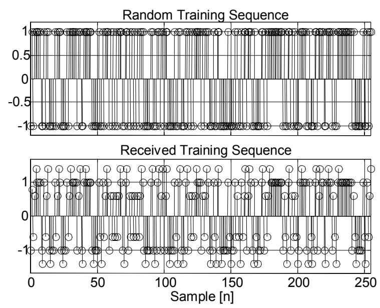

FIGURE 14
TRANSMITTED AND RECEIVED TRAINING SEQUENCE.

As showed in Figure 10, the spectrum of random sequences has components in every frequency. Thus, the equalizer can estimate the channel correctly and update its taps to remove the ISI introduced by the channel. Figure 15 shows the convolution between the channel and equalizer impulse responses, where it is possible to note that the equalizer was able to estimate and compensate the distortions introduced by the channel, once the condition presented in (1) has been satisfied. This conclusion can also be confirmed through Figure 16, where the channel and equalizer frequency responses are presented, as well the product between them.

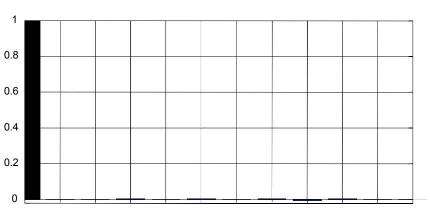

FIGURE 15
CONVOLUTION OF THE CHANNEL AND EQUALIZER IMPULSE RESPONSES

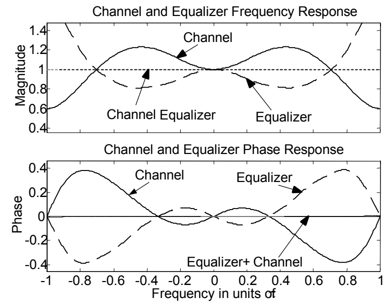

FIGURE 16
CHANNEL AND EQUALIZER FREQUENCY RESPONSE.

The equalized sequence is presented in Figure 17. Note that, after 90 interactions, the equalizer is able to remove the ISI from the signal, which means that after 90 bits of the training sequence, the reference sequence is almost equal to the equalized sequence. Figure 18 shows the equalizer and the channel frequency response after the transmission of the data sequence.

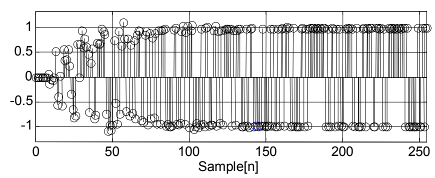

FIGURE 17
EQUALIZED TRAINING SEQUENCE.

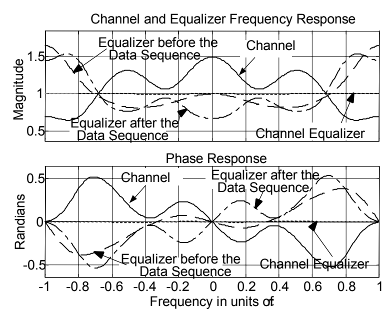

FIGURE 18
CHANNEL AND EQUALIZER FREQUENCY RESPONSE.

The equalizer frequency response is the inverse of the channel frequency response, therefor the statement presented by (2) was satisfied. Figure 19 shows the equalized sequence, where one can note that there are ISI in the beginning of the data sequence, because of the random variation in the channel impulse response, but the equalizer successfully update its taps to remove any interference introduced by the channel.

Although the random training sequence has characteristics that allow the equalizer to estimate correctly the channel, its use is not practical because the receiver can not generate the same random sequence generated by the transmitter. Thus it is not possible to obtain a reference sequence in the receiver to calculate the error signal.

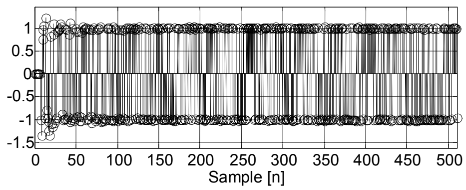

FIGURE 19
EQUALIZED DATA SEQUENCE.

Simulation with Pseudo Random Training Sequence.

The pseudo random sequences or pseudo noise (PN) sequences [2][3] present a solution for the training process because it behaves as a random sequence during each cycle of N samples. Once known the generator polynomial and the initial conditions, the receiver is able to generate the same sequence used by the transmitter. Thus, it is possible to use a PN sequence as a training sequence to estimate the channel frequency response. In this simulation, a PN sequence is used to train the equalizer with the same configuration presented in the previous sessions. Figure 20 shows the transmitted and received training sequence and Figure 21 shows its spectrum. It is possible to note that PN sequence does not have a uniform spectrum as a random sequence, but it presents a spectrum with energy well distributed in the bandwidth of interest. This allows the equalizer to estimate the channel and avoid high bit error rates during the transmission of the data sequence. Figure 22 shows the equalizer and channel frequency responses, as well, the product between them.

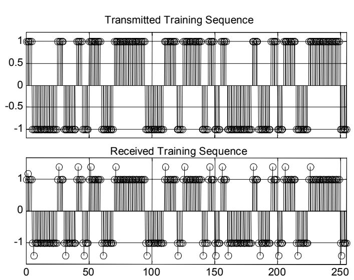

FIGURE 20
TRANSMITTED AND RECEIVED TRAINING SEQUENCE

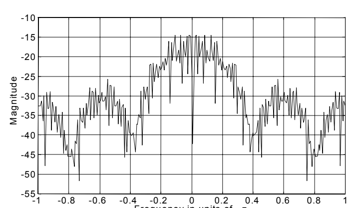

FIGURE 21
SPECTRUM OF THE TRANSMITTED TRAINING SEQUENCE.

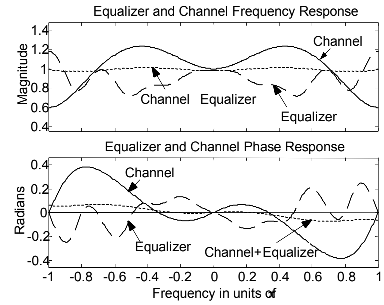

FIGURE 22
CHANNEL AND EQUALIZER FREQUENCY RESPONSE AFTER THE TRAINING SEQUENCE.

The resulting frequency response is not perfectly flat, but it is flat enough to avoid high BER during the transmission of the data sequence. Figure 23 shows that the equalizer has been able to correctly estimate the variation of the channel impulse response during the transmission of the data sequence.

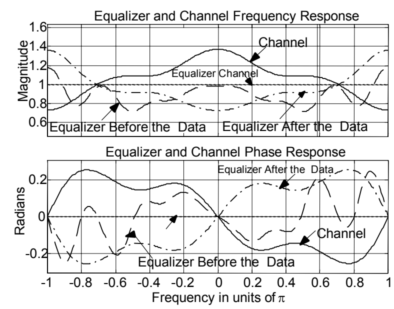

FIGURE 23
CHANNEL AND EQUALIZER FREQUENCY RESPONSE.

CONCLUSIONS

This paper has presented a didactic simulator to analyze a LMS adaptive equalizer, with some commented results obtained in three simulations. The first simulation showed that square waveform cannot be used to estimate the equalizer because cyclic waveforms have discrete spectral components, which does not allow the equalizer to correctly estimate the channel. The second simulation has presented the results obtained with a random training sequence. Although the obtained results demonstrate that a random sequence allows the equalizer to estimate the channel, this procedure cannot be used in practice because the receiver cannot generate the same random sequence used by the transmitter. Finally, the third simulation has presented a practical solution to train the equalizer properly. The PN sequences have spectra with energy distributed in the bandwidth of interest which allows the equalizer to estimate the channel. Although the energy distribution of a PN sequence spectrum is not uniform as in the random sequence spectrum, the PN sequence allows the equalizer to remove the ISI from the transmitted sequences, which can guarantee a low BER during the transmission of the data sequence.

REFERENCES

[1] Ingle, V and Proakis, J, *Digital Signal Processing Using MATLAB®*, Brooks, 2000.

[2] Haykin, S, *Communications Systems*, 2nd Edition , Wiley, 2000.

[3] Sklar, B, *Digital Communications Fundamentals and Applications*, Prentice Hall, 1988.

[4] Orfanidis, S, *Introduction to Signal Processing*, Prentice Hall, 1996.

[5] Yates, R and Goodman, D, *Probability and Stochastic Processes*, Wiley, 1999.