Pseudo-Maximum Likelihood Data Estimation of Digital Sequences in the Presence of Intersymbol Interference

Hamid R. Sadjadpour and Charles L. Weber
Communication Sciences Institute, Department of Electrical Engineering-Systems
University of Southern California, Los Angeles, CA 90089-2565

Abstract

A Pseudo-Maximum Likelihood Data Estimation (PML) algorithm for discrete channels with finite memory in additive white Gaussian noise is developed. Unlike the traditional methods which utilize the Viterbi Algorithm (VA) for data sequence estimation, the PML algorithm offers an alternative solution to the problem which is much easier to implement. There is a tradeoff in the PML algorithm between computational complexity and performance. The performance of PML is compared with the optimum receiver (VA with known channel).

1 Introduction

The term blind equalization or deconvolution in communication and signal processing applications refers to recovering the input data sequence which is applied to an unknown linear time-invariant channel by knowing its output only. The Viterbi Algorithm [7] is the common method in blind equalization techniques under the maximum likelihood (ML) criterion. [1-2] In order to utilize VA, correct channel coefficient estimation is needed. Although the performance of these methods is equivalent to the performance of the optimum receiver, they are computationally complex, which make them impractical for many applications. These techniques are estimating a *sequence* of data. In this paper, we suggest a new algorithm to estimate the data using the ML criterion. This method does not apply VA for estimation of the data sequence and it is a symbol-by-symbol estimation. The paper is organized as follows: The next section will describe the problem and assumptions. In section 3, the mathematical derivation of the PML algorithm and its physical interpretation is presented. The computer simulation results are given in section 4. A summary and some suggestions for future work conclude the paper.

2 Problem statement and assumptions

Throughout this paper, it is assummed that the channel is modeled as a discrete time-invariant FIR filter with length L. The discrete channel can be thought of as the convolution of the transmitter filter, the channel and the receiver filter. The received signal ($\{r_k\}$) is the convolution of the data sequence ($\{a_k\}$) and the channel coefficients ($\{h_k\}$) added to a sequence of independant identically distributed (i.i.d.) Gaussian random noise samples ($\{n_k\}$). The data sequence is also an i.i.d. sequence. According to this model, r_k can be written as

$$r_k = \sum_{i=1}^{L} h_i a_{k-i+1} + n_k. \tag{1}$$

There are no other restrictions on the nature of the channel, i.e. with or without zeros in its frequency response, linear or non-linear phase. The transmitter and the receiver are assumed synchronous.

3 PML algorithm

3.1 Derivation of the PML algorithm

Equation (1) in vector form is

$$\underline{r}_N = A_N \underline{h} + \underline{n}_N \tag{2}$$

where $\underline{r}_N^t = [r_{k+1} \ \dots \ r_{k+N}]$, $\underline{n}_N^t = [n_{k+1} \ \dots \ n_{k+N}]$, $\underline{h}^t = [h_1 \ \dots \ h_L]$ and

$$A_N = \begin{bmatrix} a_{k+1} & a_k & \dots & a_{k-L+2} \\ a_{k+2} & a_{k+1} & \dots & a_{k-L+3} \\ \vdots & \vdots & & \vdots \\ a_{k+N} & a_{k+N-1} & \dots & a_{k+N-L+1} \end{bmatrix}$$

From the above equation, the conditional probability density function (pdf) of the received signal given A_N

and $\underline{h}$ is

$$p(\underline{r}_N/A_N,\underline{h}) = \frac{1}{(\sigma\sqrt{2\pi})^N}\, exp[\frac{|\underline{r}_N - A_N\underline{h}|^2}{-2\sigma^2}]. \quad (3)$$

Maximization of the conditional pdf is equivalent to minimization of the following likelihood function.

$$C(A_N,\underline{h}) = |\underline{r}_N - A_N\underline{h}|^2 \quad (4)$$

For a given A_N, the maximum likelihood estimate of $\underline{h}$ can be obtained by differentiating the likelihood function with respect to $\underline{h}$ as

$$\frac{\partial C(A_N,\underline{h})}{\partial \underline{h}} = 2A_N^t\underline{r}_N - 2A_N^tA_N\underline{h} \quad (5)$$

where A_N^t is the transpose of matrix A_N. If the above equation is set to zero, the maximum likelihood estimate of $\underline{h}$ is obtained, namely

$$\hat{\underline{h}}_{ML} = (A_N^tA_N)^{-1}A_N^t\underline{r}_N \quad (6)$$

Substituting $\hat{\underline{h}}_{ML}$ into (4) will result the following likelihood function.

$$\begin{aligned} C(A_N,\hat{\underline{h}}_{ML}) &= |\underline{r}_N - A_N(A_N^tA_N)^{-1}A_N^t\underline{r}_N|^2 \\ &= \underline{r}_N^t\underline{r}_N - \underline{r}_N^tA_N(A_N^tA_N)^{-1}A_N^t\underline{r}_N \quad (7)\end{aligned}$$

Minimization of the above equation with respect to A_N is equivalent to maximization of the following likelihood function.

$$C'(A_N) = \underline{r}_N^tA_N(A_N^tA_N)^{-1}A_N^t\underline{r}_N \quad (8)$$

$C'(A_N)$ is a function of the received signal (r_k) and the transmitted data sequence (a_k).

To maximize this function, the following assumptions has been made. The data sequence is known (or estimated) prior to time $k+N+1$ and our objective is to estimate the data at time $k+N+1$. At time k, the known (or estimated) data will be shown as $\hat{a}_k$ and the data to be estimated as a_k. So the matrix A at time $k+N+1$ can be written as

$$A_{N+1} = \begin{bmatrix} A_N \\ ------------ \\ \underline{a}_{N+1}^t \end{bmatrix}$$

where $\underline{a}_{N+1} = [a_{k+N+1}\ \hat{a}_{k+N} \dots \hat{a}_{k+N-L+2}]^t$ and A_N is given above. $\underline{r}_{N+1}$ is

$$\underline{r}_{N+1} = \begin{bmatrix} \underline{r}_N \\ ---- \\ r_{k+N+1} \end{bmatrix}$$

where $\underline{r}_N$ is the same as before. $C'(A_{N+1})$ is then given by

$$C'(A_{N+1}) = \underline{r}_{N+1}^tA_{N+1}(A_{N+1}^tA_{N+1})^{-1}A_{N+1}^t\underline{r}_{N+1} \quad (9)$$

Our objective is to maximize $C'(A_{N+1})$. $C'(A_{N+1})$ can be expressed as a function of A_N, $\underline{a}_{N+1}$, $\underline{r}_N$ and r_{k+N+1}. To calculate $C'(A_{N+1})$, the inverse of $(A_{N+1}^tA_{N+1})$ is needed, where

$$A_{N+1}^tA_{N+1} = A_N^tA_N + \underline{a}_{N+1}\underline{a}_{N+1}^t \quad (10)$$

The inverse of the above equation can be found using the inverse theorem [5]. The following parameters are defined:

$$k_1 = \underline{a}_{N+1}^t(A_N^tA_N)^{-1}A_N^t\underline{r}_N \quad (11)$$

$$k_2 = \underline{a}_{N+1}^t(A_N^tA_N)^{-1}\underline{a}_{N+1} \quad (12)$$

where k_1 and k_2 are scalars. $C'(A_{N+1})$ can then be written as

$$C'(A_{N+1}) = C'(A_N) + \Delta(k_1,k_2) \quad (13)$$

where

$$\Delta(k_1,k_2) = 2r_{k+N+1}k_1 + r_{k+N+1}^2k_2 - \frac{(k_1 + r_{k+N+1}k_2)^2}{1+k_2}. \quad (14)$$

This recursion in (13) should be maximized at each time period with respect to a_{k+N+1}. $C'(A_N)$ is constant and independant of a_{k+N+1}, so it is sufficient to maximize $\Delta(k_1,k_2)$. The only unknown variable in $\Delta(k_1,k_2)$ is a_{k+N+1} which is the first element of vector $\underline{a}_{N+1}$. By differentiating $\Delta(k_1,k_2)$ with respect to a_{k+N+1} [1] and setting the result equal to zero, the condition to maximize $\Delta(k_1,k_2)$ can be determined.

$$\begin{aligned} \frac{\partial \Delta(k_1,k_2)}{\partial a_{k+N+1}} &= \frac{\partial \Delta(k_1,k_2)}{\partial k_1} \times \frac{\partial k_1}{\partial a_{k+N+1}} \\ &+ \frac{\partial \Delta(k_1,k_2)}{\partial k_2} \times \frac{\partial k_2}{\partial a_{k+N+1}} \quad (15)\end{aligned}$$

$$\frac{\partial \Delta(k_1,k_2)}{\partial k_1} = 2 \times \frac{r_{k+N+1} - k_1}{1+k_2} \quad (16)$$

$$\frac{\partial \Delta(k_1,k_2)}{\partial k_2} = \frac{(r_{k+N+1} - k_1)^2}{(1+k_2)^2} \quad (17)$$

The solution to (15) is

$$k_1 = r_{k+N+1}. \quad (18)$$

[1] a_{k+N+1} has a discrete value and in order to differentiate with respect to a_{k+N+1}, we allow a_{k+N+1} to take on a continuum of values.

To check that this value maximizes $\Delta(k_1, k_2)$, the second derivative of $\Delta(k_1, k_2)$ with respect to a_{k+N+1} is needed. It can be shown that this derivative is negative if $1 + k_2$ is positive [3].

The performance of the PML algorithm can be improved significantly by estimating l data instead of one data at a time. Let's define A_{N+l} as

$$A_{N+l} = \begin{bmatrix} A_N \\ \hline a^t_{N+l} \end{bmatrix}$$

where A_N is the same as before and

$$a^t_{N+l} = \begin{bmatrix} a_{k+N+1} & \hat{a}_{k+N} & \cdots & & \hat{a}_{k+N-L+2} \\ \vdots & & & & \vdots \\ a_{k+N+l} & \cdots & a_{k+N+1} & \hat{a}_{k+N} \cdots & \hat{a}_{k+N+l-L+1} \end{bmatrix}.$$

$\underline{k}_1$ and k_2 are defined as before except $\underline{k}_1$ is a vector of length l and k_2 is a $l \times l$ square matrix, so that

$$\underline{k}_1 = a^t_{N+l}(A^t_N A_N)^{-1} A^t_N \underline{r}_N \quad (19)$$

$$k_2 = a^t_{N+l}(A^t_N A_N)^{-1} a_{N+l}. \quad (20)$$

$\underline{r}_{N+l}$ can be written as

$$\underline{r}_{N+l} = \begin{bmatrix} \underline{r}_N \\ \hline \underline{r}_l \end{bmatrix}$$

where $\underline{r}_N$ is the same as before and $\underline{r}_l = [r_{k+N+1} \; \cdots \; r_{k+N+l}]^t$. $C'(A_{N+l})$ is given by

$$C'(A_{N+l}) = C'(A_N) + \Delta(\underline{k}_1, k_2) \quad (21)$$

where

$$\begin{aligned}\Delta(\underline{k}_1, k_2) &= -\underline{k}^t_1(k_2 + I_l)^{-1}\underline{k}_1 + 2\underline{r}^t_l\underline{k}_1 + \underline{r}^t_l k_2 \underline{r}_l - \\ &\quad \underline{r}^t_l k_2 (k_2 + I_l)^{-1} k_2 I_l - 2\underline{r}^t_l k_2 (k_2 + I_l)^{-1}\underline{k}_1 \end{aligned} \quad (22)$$

Ideally, we would like to minimize $C'(A_{N+l})$ over all the unknown data values in a_{N+l}. This is a tedious task, so alternatively, we choose to minimize $C'(A_{N+l})$ with respect to the vector $\underline{k}_1$, realizing this optimization corresponds to a unique value for a_{N+l}, via (19). Proceeding, the first derivative of $\Delta(\underline{k}_1, k_2)$ with respect to $\underline{k}_1$ is

$$\frac{\partial \Delta(\underline{k}_1, k_2)}{\partial \underline{k}_1} = -2(k_2 + I_l)^{-1}\underline{k}_1 + 2\underline{r}_l - 2(k_2 + I_l)^{-1} k_2 \underline{r}_l. \quad (23)$$

If (23) is set equal to zero, the result is

$$\underline{k}_1 = \underline{r}_l. \quad (24)$$

To prove that the above result (eq. 24) is the global maximum[1] of $\Delta(\underline{k}_1, k_2)$ assuming that $\underline{k}_1$ can take on an arbitrary continuum of values through a_{N+l}, it can be proved that $C(A_N) = C(A_{N+l})$ [3].

3.2 Intuitive discussion

The above condition ($\underline{k}_1 = \underline{r}_l$) has a significant physical interpretation. Let's define the vector $\underline{\hat{h}}$ as

$$\begin{aligned} \underline{\hat{h}} &= (A^t_N A_N)^{-1} A^t_N \underline{r}_N \quad (25) \\ &= [\hat{h}_1 \hat{h}_2 \ldots \hat{h}_L]^t \quad (26) \end{aligned}$$

By comparing (25) with (6), it can be concluded that the $\underline{\hat{h}}$ is the maximum likelihood estimate of the channel coefficients based on the estimated data sequence from time $k - L + 2$ to time $k + N$. When each row of the a_{N+l} is multiplied by the $\underline{\hat{h}}$, this is equivalent to the convolution of the transmitted data sequnce with the estimated channel coefficients. Obviously, we expect this value to be equal to the received signal at that time period. The received signals are the elements of the $\underline{r}_l$ vector. If $l = 1$, then PML is equivalent to a Decision Feedback Equalizer (DFE) [6]. DFE is an example of symbol-by-symbol detection, where in each time interval, the ISI is subtracted from the received signal and the output is fed back to a slicer which makes the decision about the current symbol. This is exactly the case for the PML algorithm when $l = 1$. The following recursive algorithm is proposed to estimate the data sequence.

3.3 Pseudo Maximum Likelihood Data Estimation (PML) Algorithm

1. At each time period, create the matrix A_N. Then calculate the $\underline{\hat{h}}$ vector.
2. There are 2^L possible values for a^t_{N+l} matrices or equivalently, $\underline{k}_1$. (This is for the binary case.)
3. Multiply each candidate a^t_{N+l} by $\underline{\hat{h}}$ and find the minimum mean square error (MSE) between the vector $\underline{k}_1$ and $\underline{r}_l$. The MSE between each $\underline{k}_1$ and $\underline{r}_l$ is defined as

   $$MSE = |\underline{r}_l - \underline{k}_1|^2 = (\underline{r}_l - \underline{k}_1)^t(\underline{r}_l - \underline{k}_1)$$

4. If the best data sequence which has the minimum MSE between $\underline{k}_1$ and $\underline{r}_l$ is defined as

[1] We can not infer from this development that the ML demodulation will result, since the transmitted data sequence must be constrained to discrete values.

$(a'_{k+N+1} \cdots a'_{k+N+l})$, then a'_{k+N+1} is equal to $\hat{a}_{k+N+1}$. The remaining estimated symbols in the sequence $(a'_{k+N+2} \cdots a'_{k+N+l})$ will not be used.

5. Return to step 1 and continue the algorithm to estimate the next data.

3.4 Remarks

1. To start this algorithm, a training sequence is needed to create the A_N matrix. In general, $N+L-1$ data is needed to start the algorithm. (N is the number of rows in A_N and L is the length of the channel)

2. At each time period, the estimated channel coefficients are updated by calculating the new $\underline{\hat{h}}$. This suggests that the PML algorithm can be applied for time-variant channels.

3. When the length of the channel (L) increases, l should be increased also. However, we have not yet found any specific relationship between l and the length of the channel (L).

4 Simulation results

This section compares the performance of the PML algorithm to the performance of the optimum receiver (VA with known channel) for the binary signal set. The channel a in (fig. 1) has a deep null in its frequency response but no phase distortion [4]. α is defined as

$$\alpha = \frac{\#\, of\ rows\ in\ A_N}{length\ of\ the\ channel} = \frac{N}{L}$$

Figure 2 compares the performance of the optimum receiver to the performance of the PML algorithm for different values of α and constant $l (l = 9)$. If α increases, it is more probable that $(A_N^t A_N)^{-1}$ exists and also more estimated data are being used for estimating the next l data elements, so the performance will improve. Figure 3 compares the performance of the optimum receiver to the performance of the PML algorithm for different values of l and constant $\alpha (\alpha = 15)$. From Figures 2 and 3 it can be concluded that increasing α and/or l will improve performance. Figure 4 compares the performance of the PML algorithm for Channel a and for two cases: 1- The length of the channel is known, namely L =3, 2- The length of the channel is unknown and it is assumed to be 5. It is obvious from this figure that the PML algorithm can

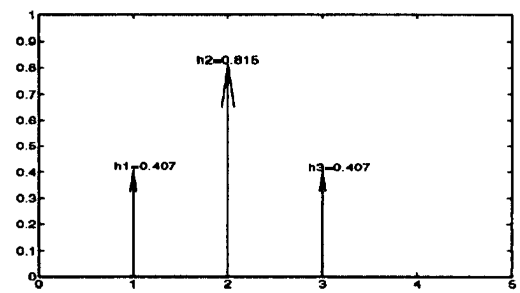

Figure 1: Impulse response of Channel a.

perform well even if the length of the channel is assumed longer than its correct value. From all these results, it can be concluded that the performance of the PML algorithm can come close to the performance of the optimum receiver. To apply the VA, it is necessary to first estimate the channel coefficients via another method. The new algorithm (PML), estimates the channel coefficients simultaneously and implicitly with the data. It seems that the PML algorithm can be an alternative for the VA, and it has some advantages (computational complexity) over the VA. This algorithm has been tested for a variety of channels and the results are satisfactory [3].

5 Summary

This paper introduced a new algorithm for data estimation in an ISI environment using the ML design criterion. Unlike conventional ML methods which apply VA, the new algorithm (PML) does not apply the VA. From computer simulation results, it is clear that the performance of the PML algorithm can perform well. In addition, for the PML algorithm, there is a tradeoff between the computational complexity and the performance and we can always find an appropriate performance level for each application. It is possible to reduce the computational complexity of this algorithm considerabely. This part is under investigation and a new algorithm is being developed.

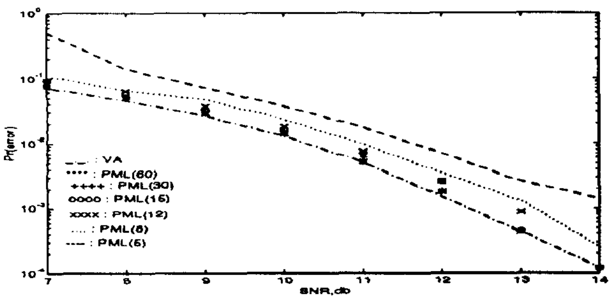

Figure 2: Performance of PML for channel a and $l = 9$ and $\alpha = 5$, 8, 12, 15, 30 and 60

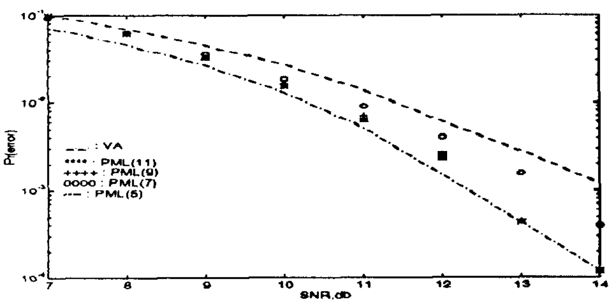

Figure 3: Performance of PML for channel a and $\alpha = 15$ and $l = 5$, 7, 9 and 11

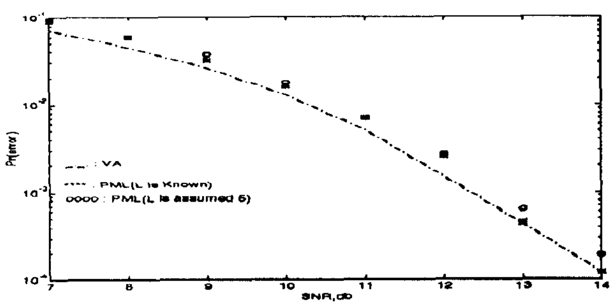

Figure 4: Performance of PML for channel a and $\alpha = 15$ and $l = 9$ and assuming that the length of the channel is 5.

References

[1] M. Ghosh and C. L. Weber, "Blind deconvolution using a Maximum Likelihood Channel Estimator," *Proceeding of the IPCCC*, 1991.

[2] M. Ghosh and C. L. Weber, "Maximum-Likelihood Blind Equalization," *The Journal of SPIE*, vol. 31, no.6, pp.1224-1229, June 1992.

[3] H. R. Sadjadpour, "Blind Equalization techniques Using Maximum Likelihood Criterion," *Ph.D. Thesis Proposal*, University of Southern California, CSI-09-01-94, September 1994.

[4] J. G. Proakis, *Digital Communications*, McGraw Hill, 1989.

[5] J.M. Mendel, *Lessons in Digital Estimation Theory*, Pretice-Hall, 1987.

[6] C. A. Belfiore and J. H. Park, Jr., "Decision feedback equalization ," *Proc. of the IEEE*, vol. 67, pp.1143-1156, Aug. 1979.

[7] G. D. Forney,Jr., "The Viterbi Algorithm," *Proceeding of the IEEE*, vol. 61, no.3, pp.268-278, March 1973.