A Soft Decision-Directed LMS Algorithm for Blind Equalization

Steven J. Nowlan and Geoffrey E. Hinton

***Abstract*—We propose a new adaptation algorithm for equalizers operating on very distorted channels. The algorithm is based on the idea of adjusting the equalizer tap gains to maximize the likelihood that the equalizer outputs would be generated by a mixture of two gaussians with known means. The familiar decision-directed least mean square (LMS) algorithm is shown to be an approximation to maximizing the likelihood that the equalizer outputs come from such an i.i.d. source. The algorithm is developed in the context of a binary PAM channel and simulations demonstrate that the new algorithm converges in channels for which the decision-directed LMS algorithm does not converge.**

I. Introduction

THE system which we investigate in this paper is illustrated in Fig. 1. A sequence of data (a_t)[1] is sent through a (linear) channel with unknown impulse response S. The data may take on one of a small number of discrete values (in the sequel we assume that a_t may take on one of two values $\pm a$). The output of the unknown channel (x_t) is then passed through a linear transversal filter W whose impulse response is approximately S^{-1}. The objective of W is to cancel out most of the distorting effects of the channel S so that the sequence (y_t) output by W is a near copy of the original input sequence (a_t). Finally, a zero-order (memoryless) nonlinear decision process (ZNL) examines each output y_t and replaces it with the closest value from the set of discrete input values, producing the sequence $(\hat{a}_t)$. We wish to consider the general case in which S may be noncausal, so W must introduce some delay into the system. As a result, we desire in general that $y_t \approx a_{t-N}$ or $\hat{a}_t = a_{t-N}$ where N represents the (known) delay of W.

Systems of the type just described occur frequently in digital communications where data must be converted into analog form for transmission and then converted back into digital form at the receiver (see [11], [14] for a general introduction to this problem). W is generally a filter with adjustable tap weights and the problem we explore is how to adjust these tap weights so W becomes a good approximation of S^{-1}. If the sequence (a_t) is known, the classical approach is to use an LMS or stochastic gradient descent procedure [12] to minimize $E[(y_t - a_{t-N})^2]$.[2] In practical situations the sequence (a_t) is not known. Instead, a two-step procedure is used to adjust the equalizer: 1) an initial settling phase in which the transmitter sends a *known* initialization sequence and the receiver performs LMS adjustment of the equalizer and 2) a phase in which the receiver uses the output of the ZNL $(\hat{a}_t)$ as an estimate for (a_t) in performing LMS adjustment of the equalizer.

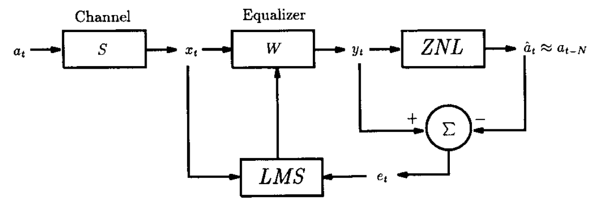

Fig. 1. Block diagram of system of interest.

The second step of the procedure is usually referred to as operating the equalizer in *decision-directed* mode since updates to the taps of the equalizer are controlled by the decisions made by the ZNL.

In the classical decision-directed LMS (DDLMS) algorithm [14], [13], the ZNL is a simple threshold device. For the binary channel we are considering, the output of this ZNL can be represented as $a\ \mathrm{sgn}(y_t)$ and the DDLMS algorithm can be regarded as minimizing $E[(y_t - a\ \mathrm{sgn}(y_t))^2]$. In this paper we derive a different form of ZNL for operating the equalizer in decision-directed mode. The modified ZNL is derived by proposing that (y_t) be modeled as the output of an independent, identically distributed (i.i.d.) random process with a gaussian mixture density. The ZNL then corresponds to the maximum likelihood estimate of $\hat{a}_t$ given y_t and the assumed model. The next section of the paper derives the basic algorithm, and Section III presents some simulation results on a simulated binary PAM channel which demonstrate that the modified equalization algorithm can converge reliably with initial channel error rates considerably greater than those allowed by the classical DDLMS procedure.

II. Development of the Algorithm

We begin by introducing some mathematical notation, and then outline the basic method for adjusting the tap weights of the equalizer via a stochastic gradient descent procedure.

Paper approved by the Editor for Channel Equalization of the IEEE Communications Society. Manuscript received April 24, 1990; revised May 23, 1991 and August 12, 1991. This work was supported by a grant from the Ontario Information Technology Research Center.

S. J. Nowlan is with the Computational Neuroscience Laboratory, The Salk Institute, San Diego, CA 92186.

G. E. Hinton is with the Department of Computer Science, University of Toronto, Toronto, Ont. M5S 1A4, Canada.

IEEE Log Number 9207349.

[1] The notation (a_t) is used to denote a *sequence* of real values, while a_t will refer to one element of that sequence.

[2] Technically, this expectation is evaluated with respect to *ensemble* averages of the signal. However, usual practice is to assume stationarity of systems and processes and to substitute time averages for ensemble averages. In the sequel, except as noted, one may assume that all expectations are evaluated as time rather than ensemble averages.

We then develop our algorithm by introducing a particular functional to be optimized and relating this functional to the functional used in the classicial DDLMS procedure.

In the system depicted in figure 1, we assume that W represents an adjustable filter with $2N + 1$ taps and a delay of N, so

$$y_t = \sum_{k=-N}^{+N} w_k^t x_{t-k} \tag{1}$$

where the superscript on the term w_k^t indicates that the tap weights change over time. Define the vector $\boldsymbol{w}_t = (w_{-N}^t, \cdots, w_N^t)$ and the vector $\boldsymbol{x}_t = \{x_{t+N}, \cdots, x_{t-N}\}$. Then we may express (1) as

$$y_t = \boldsymbol{w}_t^{\boldsymbol{T}} \boldsymbol{x}_t \tag{2}$$

where the superscript $\boldsymbol{T}$ denotes the vector transpose.

As mentioned in the introduction, the objective of adjusting the tap weights in the equalizer is to make $y_t \approx a_{t-N}$. The approach we shall use is to treat (y_t) as a stochastic process and to minimize a functional

$$\min_{\boldsymbol{w} \in R^{2N+1}} \mathcal{J}(\boldsymbol{w}) = E[\Psi(y_t)]. \tag{3}$$

Following the classical approach (assuming stationary ergodic processes), we replace ensemble averages in (3) with time averages and we get the following stochastic gradient update procedure:

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \epsilon e_t \boldsymbol{x}_t \tag{4}$$

where

$$e_t = \frac{\partial \Psi(y_t)}{\partial y_t}$$

and ϵ is a (fixed) stepsize. General convergence results for this class of stochastic approximation algorithms may be found in [3]. To complete the algorithmic specification, we simply need to define $\Psi(y_t)$.

Since we do not have direct access to (a_t) when updating our tap weights, $\Psi(y_t)$ must contain some assumptions about a_{t-N}. In the classical DDLMS algorithm for a binary PAM channel with input values $\pm a$, $\Psi(y_t) = (y_t - a \operatorname{sgn}(y_t))^2$ and $e_t = y_t - a \operatorname{sgn}(y_t)$. Informally, the assumption in the classical algorithm is that if $y_t \geq 0$ then $a_{t-N} = a$, otherwise $a_{t-N} = -a$.

We propose instead that we define

$$\Psi(y_t) = -\log f_Y(y_t) \tag{5}$$

where

$$f_Y(y_t) = \left(\frac{\rho_1}{\sqrt{2\pi}\sigma} e^{-(y_t+a)^2/2\sigma^2} + \frac{\rho_2}{\sqrt{2\pi}\sigma} e^{-(y_t-a)^2/2\sigma^2} \right). \tag{6}$$

$f_Y(y_t)$ is the density function of a gaussian mixture with two components centered at $\pm a$.[3] ρ_1 and ρ_2 are the proportions of the two gaussians in the mixture, and σ is their standard deviation. In the sequel, we will assume that $\rho_1 = \rho_2 = 0.5$ which corresponds approximately to the assumption that the values $\pm a$ are equally likely in the data sequence (a_t). We assume for now σ is also fixed.

[3] Upper case letters denote random variables (RV) and corresponding lower case letters denote values of the RV.

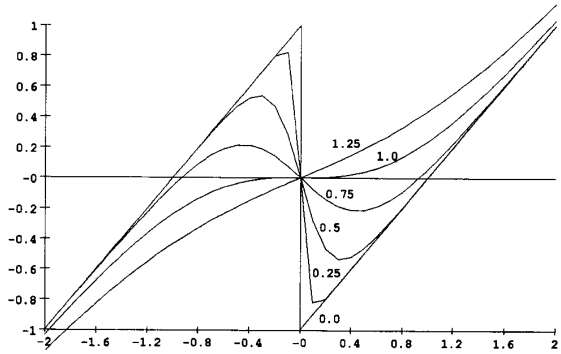

Fig. 2. Soft decision error function for different values of σ. ($\sigma = 0$ corresponds to a hard decision.)

Minimizing $\Psi(y_t)$ as defined in (5) corresponds to maximizing the (log) likelihood that the sequence (y_t) is generated by an i.i.d. source with density $f_Y()$. One can justify this model for (y_t) by assuming that (a_t) is generated by an i.i.d. source with density $f_A(a_t) = 0.5\delta_{a_t+a} + 0.5\delta_{a_t-a}$ and that the combined effect of S followed by W on (a_t) is simply to add 0 mean gaussian noise (and a delay of N) to (a_t). The added noise is composed of channel noise and an intersymbol interference term. Since the intersymbol interference term contains the contributions from a large number of random symbols, the characterization of the overall noise constribution as gaussian is not unreasonable.

To evaluate e_t based on (5) we must deal with the sum of exponentials in (6). One simple approximation is to ignore the smaller of the two exponential terms. Under this assumption we find

$$e_t \approx \begin{cases} \frac{1}{\sigma^2}(y_t + a) & y_t < 0 \\ \frac{1}{\sigma^2}(y_t - a) & \text{otherwise.} \end{cases} \tag{7}$$

We can simply include the $1/\sigma^2$ term into our step size ϵ and we find that this approximate expression for e_t is identical to the expression for e_t in the classical DDLMS algorithm.

If we do not make any approximations, the expression for e_t based on (5) is

$$e_t = \frac{1}{\sigma^2} \left(y_t - a \frac{e^{2ay_t/\sigma^2} - 1}{e^{2ay_t/\sigma^2} + 1} \right). \tag{8}$$

Comparing (8) to the classical DDLMS error, we see that we have replaced the sign function with a more complicated non-linearity. We refer to this particular non-linearity as a "soft" decision as compared to the "hard" decision represented by the sign function in the DDLMS algorithm. The choice of these terms is apparent from Fig. 2 where we have plotted (8) as a function of y for several values of σ. As $\sigma \to 0$ (8) becomes identical to the error in the DDLMS algorithm. This error has a sharp discontinuity at $y = 0$, and as σ increases this discontinuity is smoothed away.

The weakness of the hard decision used in the classical DDLMS algorithm is apparent when we realize that when y_t is very near 0 a decision based on the sign of y_t is most likely to be incorrect, due to the effects of random noise. Yet it is in these cases that the tap weights are changed the most. The "soft" decision nonlinearity has a much smaller magnitude when y_t is near 0, so it makes only small weight changes in these highly ambiguous cases.

Finally, we turn to the parameters of the gaussian mixture density $f_Y()$. There are three parameters for each component in the mixture, a mixing proportion ρ, a mean μ, and a variance σ^2. In the development so far we have assumed that the means have been fixed at $\pm a$ and the mixing proportions have been fixed at $\rho_1 = \rho_2 = 0.5$. These values come directly from our assumption that (a_t) is generated from a binary i.i.d. source with density

$$f_A(A) = 0.5\delta_{A+a} + 0.5\delta_{A-a} \tag{9}$$

where a is a constant and δ_x is the usual Kronecker delta function

$$\delta_x = \begin{cases} 1 & x = 0 \\ 0 & \text{otherwise.} \end{cases}$$

Knowledge of the source characteristics and transmission scheme will usually suggest reasonable choices for the values of ρ and μ for each component.

We have also assumed that the variance of the two components was equal and fixed at some unspecified value σ^2. To understand why both components would have the same variance, it is helpful to consider more carefully the nature of the "noise" which results when W is not an ideal inverse filter for S. In Fig. 1, x_t and y_t may be expressed as the following convolution sums:

$$x_t = \sum_{k=-\infty}^{\infty} s_k a_{t-k} \tag{10}$$

$$y_t = \sum_{k=-\infty}^{\infty} w_k x_{t-k} \tag{11}$$

where we have let $w_k = 0$ for $|k| > N$ and dropped the superscript on w_k for notational convenience. Let $W^* = (w^*_{-\infty}, \cdots, w^*_{\infty})$ denote the impulse response of an ideal inverse filter for S so

$$\sum_k w^*_k s_{l-k} = \delta_{l-N}. \tag{12}$$

If we replace W with W^* in Fig. 1 we would have

$$y_t = \sum_k w^*_k x_{t-k} = a_{t-N}. \tag{13}$$

We can regard W as an approximation to the ideal equalizer W^* and rewrite (11) as

$$\begin{aligned} y_t &= \sum_k w^*_k x_{t-k} + \sum_k (w_k - w^*_k) x_{t-k} \\ &= a_{t-N} + n^c_t \end{aligned} \tag{14}$$

where (n^c_t) is a convolutional noise sequence and we have used (13). The variance of each component in our mixture density is largely determined by the variance of the convolutional noise sequence (n^c_t). From (14) it is apparent that n^c_t contains contributions from a large number of transmitted symbols, so it is reasonable to assume that the properties of (n^c_t) are not dramatically different for different symbol values (i.e., samples representing the symbol a and samples representing the symbol $-a$ will have similar distributions for n^c). This suggests that the variance of each component in the mixture density $f_Y()$ is the same.[4]

Equation (14) also suggests that the characteristics of (n^c_t) will change as W changes, so the assumption that σ^2 should be fixed does not seem reasonable. Simulation studies have shown that better convergence may be obtained by adjusting σ in parallel with adjusting the equalizer tap weights [10], [7], [9]. The algorithm used to update σ is an incremental version of the standard maximum likelihood estimator for mixture components with tied variance [5], [8]. The actual update rule is

$$\sigma^2(t+1) = \kappa\sigma^2(t) + (1-\kappa)[\lambda(y_t + a)^2 + (1-\lambda)(y_t - a)^2] \tag{15}$$

where

$$\lambda = \frac{1}{1 + e^{2ay_t/\sigma^2}} \tag{16}$$

and κ is a decay rate slightly less than 1 for discounting past data. The decay rate for the exponential averaging of past data is based on the degree of stationarity of the data. A more detailed development of this expression may be found in [9].

Finally, we note that the model for the data given by (9) may be generalized to allow a larger class of discrete signal distributions while still allowing the distribution of the sequence (y_t) to be modeled as a mixture of gaussians. If we let A denote a random process for which (a_t) is a realization, the pdf of this i.i.d. source may be written in the general form

$$f_A(a) = \sum_i \rho_i \delta_{a-\mu_i}. \tag{17}$$

This source will be 0 mean and uncorrelated[5] as long as

$$\sum_i \rho_i \mu_i = 0. \tag{18}$$

An i.i.d. data source of the form (17) will lead to an i.i.d. model for (y_t) with a Gaussian mixture density in which the mixing proportions ρ_i and means μ_i are as specified in (17) and the variance of all components is the same.

III. Simulation Results

The discussion in the previous section comparing the "hard" and "soft" decision nonlinearities suggested that the difference between these two nonlinearities will be most apparent when a channel is poorly equalized and there are numerous incorrect decisions (i.e., cases where a_{t-N} and y_t have opposite signs). Simulations reported elsewhere [10], [7] have shown that the

[4] This argument is developed in greater detail in [9] and also in [6] in the context of nonlinear deconvolution.

[5] $E[A_iA_j] = a_i^2$ if $i = j$ and is 0, otherwise.

use of the soft decision nonlinearity can lead to more rapid initial convergence than the classical DDLMS algorithm in channels with moderate to severe noise and distortion. The simulations reported in this section focus on comparing the convergence limits of the DDLMS and soft decision-directed algorithms.

Simulations were carried out using a synthetic binary PAM channel. The output of an ideal nonredundant binary signal source with signal levels ±1 was sent through a raised cosine pulse generator and a non-causal finite impulse response (FIR) distortion filter. White zero mean noise was then added to the output of the FIR filter to produce the input to the equalizer. 50 different channels with different bit error rates were created by combining different distortion filters with different amounts of white noise. The distortion filters had a gain of 1 and phase distortion ranging from ±10° to ±60°. The signal-to-noise ratio varied from 30 dB to 0 dB.

The DDLMS and the soft-decision algorithm were used to equalize each of the 50 channels. In each simulation, the equalizer was initialized with a center tap weight of 1.0 with all other tap weights equal 0.0. An initial bit error rate for each channel was computed by comparing the sign of y_t to the sign of a_{t-N} for 500 channel outputs.[6] 1000 weight updates were then performed (using either the DDLMS or soft-decision algorithm) and the tap weights frozen. A final bit error rate was computed based on an additional 500 channel outputs. A figure of merit γ was then computed for each simulation:

$$\gamma = 1.0 - \frac{\text{Final bit error rate}}{\text{Initial bit error rate}}.$$

A value of 1.0 for γ indicates that the adjustment algorithm was able to reduce the bit error rate to 0 within 1000 updates,[7] $\gamma = 0.5$ indicates that the bit error rate was cut in half by the adjustment algorithm, and values of γ near 0 indicate that the adjustment algorithm has failed to improve the error rate at all. A range for ϵ between 0.1 and 10^{-8} was explored for each algorithm on each channel and results are reported for the value of ϵ which gave the best value for γ.[8]

The simulation results are summarized in Fig. 3 where γ has been plotted on the vertical axis and the initial bit error rate as a percentage on the horizontal axis. The shaded region bounded by the lines marked with + indicates the performance range of the DDLMS algorithm over the 50 channels. The shaded region bounded by the lines marked with $*$ indicates the performance range of the soft-decision algorithm. For channels with initial bit error rates up to about 5% the two algorithms are indistinguishable, however there are significant differences for larger initial bit error rates. The absolute convergence limit of each algorithm is indicated approximately by the bit error rate at which $\gamma = 0$. For the DDLMS algorithm this limit is between 16% and 18%[9] while the soft decision limit is between

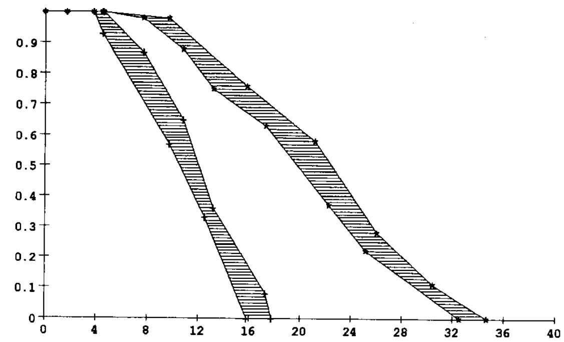

Fig. 3. Graph of γ versus initial bit error rate (%) for the DDLMS (+) and soft-decision ($*$) algorithms.

32% and 35%. From a practical standpoint, the region of interest is primarily for values of $\gamma > 0.5$. Even in this region, the soft decision algorithm maintains a significant advantage (10% to 12% for DDLMS with $\gamma = 0.5$ compared to 20% to 23% for the soft-decision algorithm).

IV. Conclusion

We have proposed a new algorithm for blind equalization of very distorted channels. The algorithm is based on the idea of adjusting the equalizer tap gains to maximize the likelihood that the equalizer outputs come from an i.i.d. source that is a mixture of two gaussians with known means. The approach is most similar to earlier work on blind deconvolution by Godfrey and Rocca [6] and blind equalization by Bellini [1], [2]. Our approach differs from these other approaches in that the algorithm is developed based on an assumed distribution for the equalizer output rather than assumptions about the distribution of signals at the channel input. This leads to an algorithm with an adaptive non-linearity for estimating the channel input given the equalizer output. This "soft" decision algorithm is compared to the classical decision-directed LMS algorithm on a binary PAM channel and simulations suggest that when operating in the blind mode the new algorithm can converge in channels with twice the initial bit error rate for which the DDLMS algorithm is convergent.

Acknowledgment

We thank Y. Le Cun and J. Bridle for helpful discussions.

References

[1] S. Bellini, "Bussgang techniques for blind equalization," in *IEEE GLOBECOM Conf. Rec.*, pp. 1634–1640, Dec. 1986.
[2] S. Bellini and F. Rocca, "Near optimal blind deconvolution," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* vol. 4, pp. 2236–2239, 1988.
[3] A. Benveniste, M. Goursat, and G. Ruget, "Robust identification of a nonminimum phase system: Blind adjustment of linear equalizer in data communications," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 385–399, 1980.
[4] J. A. C. Bingham, *The Theory and Practice of Modem Design.* New York: Wiley, 1988.
[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," in *Proc. Royal Statist. Soc.*, vol. B-39, pp. 1–38, 1977.

[6] In all cases, the initial bit error rate is >0 indicating that the "eye" of the equalizer output is initially partially closed.

[7] This does not mean the channel was perfectly equalized since it is possible to make no incorrect decisions and still have a high MSE.

[8] This optimal value of ϵ was not necessarily the same for both algorithms operating on the same channel.

[9] This figure is in agreement with other reported convergence limits for the DDLMS algorithm [4].

[6] R. Godfrey and F. Rocca, "Zero memory non-linear deconvolution," *Geophys. Prospecting,* vol. 29, pp. 189–228, 1981.

[7] G. E. Hinton and S. J. Nowlan, "The bootstrap Widrow–Hoff rule as a cluster-formation algorithm," *Neural Comput.,* vol. 2, pp. 355–362, 1990.

[8] G. J. McLachlan and K. E. Basford, *Mixture Models: Inference and Applications to Clustering.* New York: Marcel Dekker, 1988.

[9] S. J. Nowlan, "Soft competitive adaptation: Neural network learning algorithms based on fitting statistical mixtures," Ph.D. dissertation, School of Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, Apr. 1991.

[10] S. J. Nowlan and G. E. Hinton, Maximum likelihood decision directed adaptive equalization," University of Toronto, Tech. Rep. CRG-TR-89-8, 1989.

[11] S. U. H. Qureshi, "Adaptive equalization," *Proc. IEEE,* vol. 73, pp. 1349–1387, Sept. 1985.

[12] B. Widrow and M. E. Hoff, Jr., "Adaptive switching circuits," in *IRE WESCON Convention Rec.,* 1960, pp. 96–104.

[13] B. Widrow, J. M. McCool, M. G. Larimore, and C. R. Johnson, Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE,* vol. 64, pp. 1151–1162, Aug. 1976.

[14] B. Widrow and S. D. Stearns, *Adaptive Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1985.