AN IMPROVED PNLMS ALGORITHM

Jacob Benesty and Steven L. Gay

Bell Laboratories, Lucent Technologies
700 Mountain Avenue
Murray Hill, NJ 07974, USA
E-mail: {jbenesty, slg}@bell-labs.com

ABSTRACT

Recently, the proportionate normalized least mean square (PNLMS) algorithm was developed for use in network echo cancelers. In comparison to the normalized least mean square (NLMS) algorithm, PNLMS has very fast initial convergence and tracking when the echo path is sparse. Unfortunately, when the impulse response is dispersive, the PNLMS converges much slower than NLMS. This implies that the rule proposed in PNLMS is far from optimal. In many simulations, it seems that we fully benefit from PNLMS only when the impulse response is close to a delta function. In this paper, we propose a new rule that is more reliable than the one used in PNLMS. Many simulations show that the new algorithm (improved PNLMS) performs better than NLMS and PNLMS, whatever the nature of the impulse response is.

1. INTRODUCTION

Recently, the proportionate normalized least mean square (PNLMS) algorithm was developed for use in network echo cancelers [1]. In comparison to the normalized least mean square (NLMS) algorithm, PNLMS has very fast initial convergence and tracking when the echo path is sparse. The idea behind PNLMS is to update each coefficient of the filter independently of the others by adjusting the adaptation step size in proportion to the estimated filter coefficient. Unfortunately, when the impulse response is dispersive, the PNLMS converges much slower than NLMS. This implies that the rule proposed in PNLMS is far from optimal. In this case, one may ask: What is sparse? What is dispersive? For what kind of impulse responses will the algorithm break down and why? In many simulations, it seems that we fully benefit from PNLMS only when the impulse response is close to a delta function.

More recently, the so-called PNLMS++ was proposed in [2]. PNLMS++ partially solves the above mentioned problem by alternating the update process each sample period between NLMS and PNLMS algorithms. However, this solution is far from general.

Obviously, an optimal rule should better exploit the shape of the estimated echo path and, the algorithm should always have better performance (convergence speed) than NLMS with non-dispersive impulse responses and similar performance with highly dispersive impulse responses. In other words, the echo path does not need to be sparse in order to be able to accelerate the convergence rate of the algorithm.

In this paper, we propose a new rule that is more reliable than the one used in PNLMS. Many simulations show that the obtained algorithm (improved PNLMS) performs better than NLMS and PNLMS, whatever the nature of the impulse response is.

2. THE NLMS AND PNLMS ALGORITHMS

In this section, we briefly explain the NLMS and PNLMS algorithms. In derivations and descriptions, the following notation is used:

$$
\begin{aligned}
x(n) &= \text{Far-end signal},\\
y(n) &= \text{Echo and background noise},\\
\mathbf{x}(n) &= [x(n) \cdots x(n-L+1)]^T, \text{ Excitation vector},\\
\mathbf{h} &= [h_0 \cdots h_{L-1}]^T, \text{ True echo path},\\
\hat{\mathbf{h}}(n) &= [\hat{h}_0(n) \cdots \hat{h}_{L-1}(n)]^T, \text{ Estimated echo path}.
\end{aligned}
$$

Here L is the length of the adaptive filter, and n is the time index.

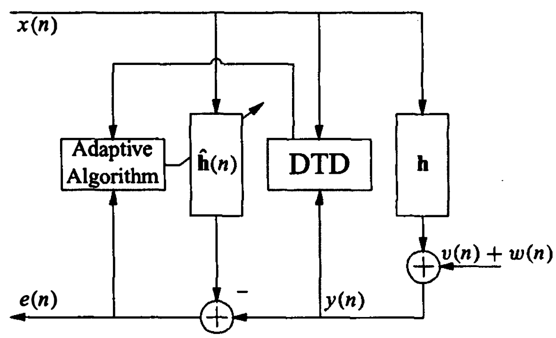

Figure 1. Block diagram of the echo canceler and double-talk detector.

According to Fig. 1, the role of the adaptive filter is to estimate the echo path so that it can subtract a replica of the returned echo $y(n)$. Traditionally, the NLMS algorithm has been the work-horse in echo canceler implementation. It therefore serves as a reference algorithm. The error signal and the coefficient update equation of the NLMS algorithm are given by [3]

$$
\begin{aligned}
e(n) &= y(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n), && (1)\\
\hat{\mathbf{h}}(n) &= \hat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{x}(n) + \delta_{\text{NLMS}}}, && (2)
\end{aligned}
$$

where μ $(0 < \mu < 2)$ is the adaptation step and δ_{NLMS} is the regularization factor.

The PNLMS algorithm was proposed in [1]. In this algorithm, an adaptive individual step-size is assigned to each filter coefficient. The step-sizes are calculated from the last estimate of the

0-7803-7402-9/02/$17.00 ©2002 IEEE

filter coefficients in such a way that a larger coefficient receives a larger increment, thus increasing the convergence rate of that coefficient. This has the effect that active coefficients are adjusted faster than non-active coefficients (i.e. small or zero coefficients). Hence, PNLMS converges much faster than NLMS for sparse impulse responses (i.e., responses for which only a small percentage of coefficients is significant). Most impulse responses in the telephone network have this characteristic.

The PNLMS algorithm is described by the following equations:

$$e(n) = y(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n), \quad (3)$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{G}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{G}(n-1)\mathbf{x}(n) + \delta_{\text{PNLMS}}} \quad (4)$$

$$\mathbf{G}(n-1) = \text{diag}\{g_0(n-1), \ldots, g_{L-1}(n-1)\}, \quad (5)$$

where $\mathbf{G}(n-1)$ is a diagonal matrix that adjusts the step-sizes of the individual taps of the filter, μ is the overall step-size parameter (the same as in NLMS to achieve a given misadjustment), and δ_{PNLMS} is the regularization parameter. The diagonal elements of $\mathbf{G}(n)$ are calculated as follows [1]:

$$\gamma_l(n) = \max\{\rho \max[\delta_p, |\hat{h}_0(n)|, \ldots, |\hat{h}_{L-1}(n)|], |\hat{h}_l(n)|\}, \quad (6)$$

$$g_l(n) = \frac{\gamma_l(n)}{\sum_{i=0}^{L-1} \gamma_i(n)}, \; 0 \le l \le L-1. \quad (7)$$

Parameters δ_p and ρ are positive numbers with typical values $\delta_p = 0.01$, $\rho = 5/L$. The first term in (6), ρ, prevents $\hat{h}_l(n)$ from stalling when it is much smaller than the largest coefficient and δ_p regularizes the updating when *all* coefficients are zero at initialization.

A variant of this algorithm is the PNLMS++ [2]. In this algorithm, for odd-numbered time steps the matrix $\mathbf{G}(n)$ is calculated as above, while for even-numbered steps it is chosen to be the identity matrix

$$\mathbf{G}(n) = \mathbf{I}, \quad (8)$$

which results in an NLMS iteration. Alternating between NLMS and PNLMS iterations has the advantage of making the convergence rate not much worse than NLMS; as a result, the PNLMS++ algorithm is less sensitive to the assumption of a sparse impulse response than PNLMS. However, PNLMS++ is not a completely satisfactory solution because it does not fully exploit the structure of the estimated impulse response. Indeed, switching between the two algorithms will work well only in the two extreme cases when the impulse response is sparse or highly dispersive. But if the impulse response is something between sparse and dispersive, PNLMS++ will likely converge about as fast as NLMS since the rule used in PNLMS does not work for this case.

3. AN IMPROVED PNLMS (IPNLMS) ALGORITHM

In this section, we introduce an improved PNLMS algorithm. Our objective is to derive a rule that better exploits the "proportionate" idea than the original PNLMS. The fact that PNLMS is slower than NLMS with dispersive impulse responses means that (6) has to be modified. Intuitively, the brutal choice (maximum) in (6) between $|\hat{h}_l|$ and one other positive number can have a disastrous effect on the convergence of the algorithm when the estimate of the coefficients is not accurate. In the following, we propose to change this part and make the choice smoother.

The 1-norm of the adaptive filter is defined as:

$$\|\hat{\mathbf{h}}(n)\|_1 = \sum_{l=0}^{L-1} |\hat{h}_l(n)|.$$

An alternative to (6) is:

$$\kappa_l(n) = (1-\alpha)\frac{\|\hat{\mathbf{h}}(n)\|_1}{L} + (1+\alpha)|\hat{h}_l(n)|, \quad (9)$$
$$l = 0, 1, \ldots, L-1,$$

where, $-1 \le \alpha < 1$. Taking the 1-norm of vector $\boldsymbol{\kappa}$:

$$\|\boldsymbol{\kappa}(n)\|_1 = \sum_{l=0}^{L-1} |\kappa_l(n)| = \sum_{l=0}^{L-1} \kappa_l(n) = 2\|\hat{\mathbf{h}}(n)\|_1, \quad (10)$$

we deduce the IPNLMS algorithm:

$$e(n) = y(n) - \hat{\mathbf{h}}^T(n-1)\mathbf{x}(n), \quad (11)$$

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \frac{\mu \mathbf{K}(n-1)\mathbf{x}(n)e(n)}{\mathbf{x}^T(n)\mathbf{K}(n-1)\mathbf{x}(n) + \delta_{\text{IPNLMS}}} \quad (12)$$

$$\mathbf{K}(n-1) = \text{diag}\{k_0(n-1), \ldots, k_{L-1}(n-1)\}, \quad (13)$$

where

$$k_l(n) = \frac{\kappa_l(n)}{\|\boldsymbol{\kappa}(n)\|_1} = \frac{1-\alpha}{2L} + (1+\alpha)\frac{|\hat{h}_l(n)|}{2\|\hat{\mathbf{h}}(n)\|_1}, \quad (14)$$
$$l = 0, 1, \ldots, L-1.$$

In practice, in order to avoid a division by zero in (14), especially at the beginning of the adaptation where all the taps of the filter are initialized to zero, we propose to use a slightly modified form:

$$k_l(n) = \frac{1-\alpha}{2L} + (1+\alpha)\frac{|\hat{h}_l(n)|}{2\|\hat{\mathbf{h}}(n)\|_1 + \varepsilon}, \quad (15)$$

where ε is a small positive number. At initialization, since all the taps of the filter start with zero, vector $\mathbf{x}$ is multiplied by $(1-\alpha)/2L$. This suggests that the regularization parameter for the IPNLMS algorithm should be taken as:

$$\delta_{\text{IPNLMS}} = \frac{1-\alpha}{2L}\delta_{\text{NLMS}}. \quad (16)$$

For $\alpha = -1$, it can easily be checked that the IPNLMS and NLMS algorithms are identical. For α close to 1, the IPNLMS behaves like the PNLMS. In general, (9) is the sum of two terms. The first one is an average of the absolute value of the coefficients of the estimated filter and the second one is the absolute value of the coefficient itself. While the second term ("proportionate") is very important to improve the convergence rate when the impulse response is sparse, it can also be harmful because it is just an approximation of the coefficient and if this value is far from the true one, it can have just the opposite effect on the gradient. The first term of (9) is more accurate, since it is an average, and balances the errors introduced in the second term. In practice, good choices for α are 0 or -0.5. With those choices and in simulations, IPNLMS always behaves better than NLMS and PNLMS, whatever the impulse response is.

4. SIMULATIONS

In telephone networks that involve the connection of 4-wire and 2-wire links, an echo is generated at the hybrid. This echo has a disturbing influence on the conversation and must therefore be cancelled. Figure 1 shows the principle of a network echo canceler (EC). The far-end speech signal $x(n)$ goes through the echo path represented by a filter $\mathbf{h}$, then it is added to the near-end talker signal $v(n)$ and ambient noise $w(n)$. The composit signal is denoted $y(n)$. Most often the echo path is modeled by an adaptive FIR filter, $\hat{\mathbf{h}}(n)$, which subtracts a replica of the echo and thereby achieves cancellation. Here, we do not consider the double-talk situation (i.e. $v(n) = 0$).

In this section, we wish to compare, by way of simulation, the NLMS, PNLMS, and IPNLMS algorithms in the context of a network EC. As shown in Fig. 2, we use three different echo paths $\mathbf{h}$ of length $L = 512$. The same length is used for the adaptive filter $\hat{\mathbf{h}}(n)$. The sampling rate is 8 kHz and the signal-to-noise ratio is equal to 39 dB. The input signal $x(n)$ is either a speech signal or a white Gaussian noise. The parameter settings chosen for all the simulations are:

- $\mu = 0.2, \hat{\mathbf{h}}(-1) = \mathbf{0}$.
- $\alpha = 0$.
- $\delta_\text{p} = 0.01,\ \rho = 0.01$.
- $\delta_\text{NLMS} = \sigma_x^2,\ \delta_\text{PNLMS} = \delta_\text{NLMS}/L,\ \delta_\text{IPNLMS} = \delta_\text{NLMS}/2L$.

Figure 3 compares the misalignment, $\|\mathbf{h}-\hat{\mathbf{h}}\|/\|\mathbf{h}\|$, of the three algorithms: (a) NLMS, (b) PNLMS, and (c) IPNLMS, with a white noise as the far-end signal and a sparse impulse response (Fig. 2a). We can see that PNLMS and IPNLMS converge much faster than NLMS with a small advantage for IPNLMS. Figure 4 shows the same thing but with a dispersive impulse response (Fig. 2b) this time. While NLMS and IPNLMS have the same behavior, PNLMS deteriorates. Figure 5 compares the algorithms in a tracking situation when after 3 seconds the sparse impulse response of Fig. 2a is shifted on the right by 12 samples. The other conditions are the same as in Fig. 3. Both PNLMS and IPNLMS track better than NLMS.

In Figs 6 and 7, we repeat the same simulations done for Figs. 3 and 4 but with speech as the far-end signal. We can draw the same conclusions either the input signal is white or speech.

Figure 8 compares the misalignment of the IPNLMS algorithm with different values of the parameter α when the far-end signal is a white noise and the impulse response is sparse (Fig. 2a). Note that for $\alpha = -1$, the IPNLMS and NLMS are the same. The IPNLMS algorithm performs very well with $\alpha = 0$ or $\alpha = -0.5$. Figure 9 compares the same algorithm with a dispersive impulse response (Fig. 2b). Even in this situation, the IPNLMS algorithm (for $\alpha = -0.5$) performs a little better than NLMS.

Finally, in the last simulation with white noise as input signal, we compare again the three algorithms (NLMS, PNLMS, and IPNLMS with $\alpha = 0$) with the quasi sparse impulse response of Fig. 2c. Figure 10 shows the result of this simulation. It is clear that the proposed algorithm outperforms the two others for these simulations.

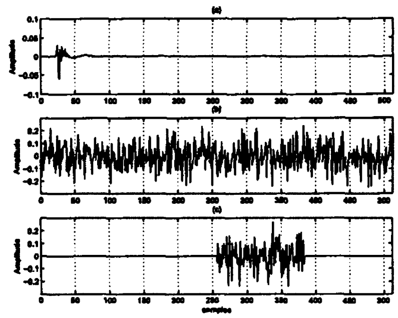

Figure 2. Three different impulse responses used in simulations: (a) sparse, (b) dispersive, and (c) quasi sparse.

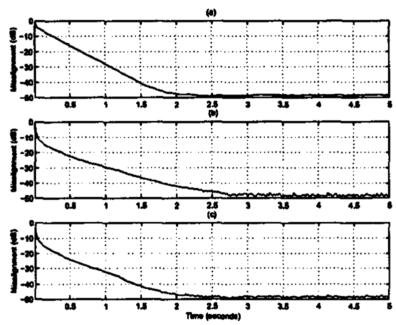

Figure 3. Misalignment of (a) NLMS, (b) PNLMS, and (c) IPNLMS, with a white noise as input signal and sparse impulse response of Fig. 2a.

5. CONCLUSIONS

While the PNLMS algorithm behaves very nicely and has a very fast initial convergence rate compared to NLMS when the impulse response is sparse, it has the annoying drawback of deteriorating quickly if the echo path is not sparse enough. It simply means that the rule used in PNLMS is not adequate and does not fully exploit the structure of the impulse response, though the algorithm works well in one particular case. In this paper, we derived a new and simple rule that overcomes this problem. In simulations, the proposed algorithm did no worse than NLMS. For very sparse impulse responses, the IPNLMS converges as well as PNLMS and for an impulse response that is between sparse and dispersive, it behaves much better than both NLMS and PNLMS. For future work, it will be interesting to evaluate the IPNLMS in the acoustic echo cancellation context, since acoustic impulse response are quasi sparse in general (not as sparse as in the hybrid but not very dispersive either).

6. REFERENCES

[1] D. L. Duttweiler, "Proportionate normalized least mean square adaptation in echo cancelers," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 508-518, Sept. 2000.

[2] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in *Proc. of Assilomar*, 1998.

[3] S. Haykin, *Adaptive Filter Theory*. Third Edition, Prentice Hall, Englewood Cliffs, N.J., 1996.

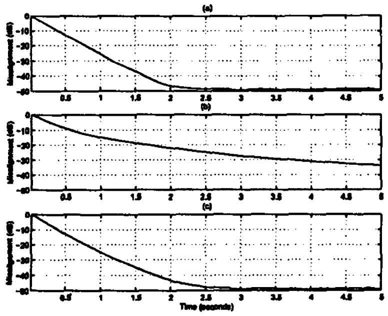

Figure 4. Misalignment of (a) NLMS, (b) PNLMS, and (c IPNLMS, with a white noise as input signal and dispersive impulse response of Fig. 2b.

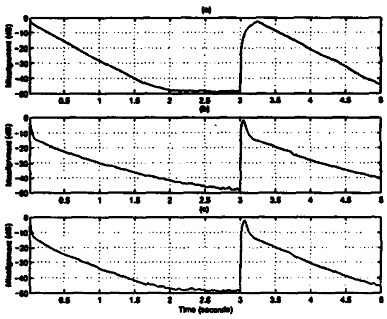

Figure 5. Misalignment during echo path change. The echo path changes at 3 seconds. Other conditions same as in Fig. 4.

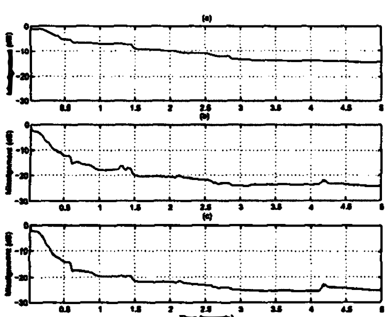

Figure 6. Misalignment of (a) NLMS, (b) PNLMS, and (c) IPNLMS, with a speech as input signal and sparse impulse response of Fig. 2a.

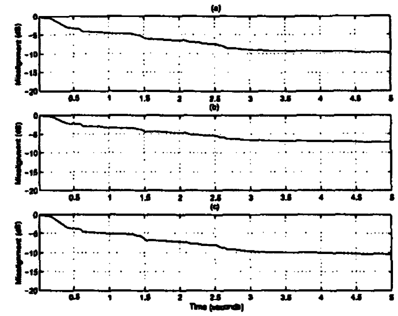

Figure 7. Misalignment of (a) NLMS, (b) PNLMS, and (c) IPNLMS, with a speech as input signal and dispersive impulse response of Fig. 2b.

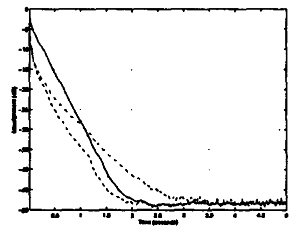

Figure 8. Misalignment of the IPNLMS algorithm with different values of the parameter α when the far-end signal is a white noise and the impulse response is sparse (Fig. 2a). (—) $\alpha = -1$ (equivalent to NLMS), (– –) $\alpha = -0.5$, (...) $\alpha = 0$, and (-. -.) $\alpha = 0.5$.

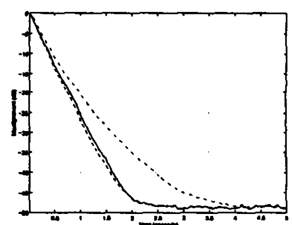

Figure 9. Misalignment of the IPNLMS algorithm with different values of the parameter α when the far-end signal is a white noise and the impulse response is dispersive (Fig. 2b). (—) $\alpha = -1$ (equivalent to NLMS), (– –) $\alpha = -0.5$, (...) $\alpha = 0$, and (-. -.) $\alpha = 0.5$.

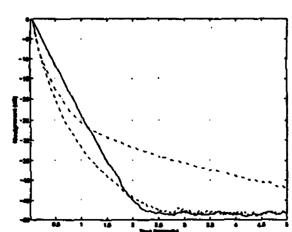

Figure 10. Misalignment of (—) NLMS, (-. -.) PNLMS, and (– –) IPNLMS, with a white noise as input signal and quasi sparse impulse response of Fig. 2c.