ADAPTIVE REGULARIZATION IN FREQUENCY-DOMAIN NLMS FILTERS[1]

Ayman Faza†, *Steven Grant*[2]†, *Jacob Benesty*‡

† Missouri University of Science and Technology, Rolla, MO, USA
‡INRS-EMT, University of Quebec, Montreal, QC, Canada

ABSTRACT

Regularization is an important part of adaptive filter design. Traditionally, the regularization parameter has been empirically selected, as there has not been a lot of work in the literature for determining an optimal method for finding its best value. In this work, we propose an adaptive method for finding the regularization parameter in the normalized least-mean-square (NLMS) algorithm. Furthermore, we apply this regularization approach in a frequency-domain version of the NLMS algorithm, in which a separate regularization parameter is computed for each frequency bin. Simulation results show that computing the regularization parameter for each frequency bin separately provided better performance for the filter, for the common case of colored noise excitation.

Index Terms— Regularization, adaptive filter, NLMS, frequency-domain NLMS.

1. INTRODUCTION

Regularization is an important parameter in the NLMS adaptive filter. It helps keep the filter stable and, when selected properly, it also helps the filter converge to the optimal solution faster. However, while this parameter is very important in the design of adaptive filters in general, and the NLMS filter in particular, very little work has been done into estimating its optimal value. In this paper, we focus on developing an optimal value for the regularization parameter that adaptively changes as the excitation signal and noise environments change in time. In developing an adaptive regularization algorithm, we guarantee good results at any given time during the filtering process.

Previous work by Benesty et al. [3] shows that the knowledge of the SNR, in addition to the knowledge of the excitation signal, can provide enough information to determine the optimum regularization parameter. This paper improves on that effort by adaptively determining the optimum regularization value rather than computing it beforehand.

In order to consider the effects of colored noise, it was necessary to divide the signal into a number of frequency bins, for which the regularization parameter can be computed separately. Therefore, for our work, we used a frequency-domain version of the NLMS algorithm, in which we set the number of frequency bins to be equal to the number of filter taps for the adaptive NLMS filter used. We simulated several scenarios, with different noise types and various SNRs, and in this paper we present our analysis and the results of these simulations.

The rest of the paper is organized as follows. Section 2 describes the frequency-domain version of the NLMS algorithm. In Section 3, we describe our method for adaptive regularization. In Section 4, we provide a method to estimate the background noise. Section 5 presents our simulation results and Section 6 concludes the paper.

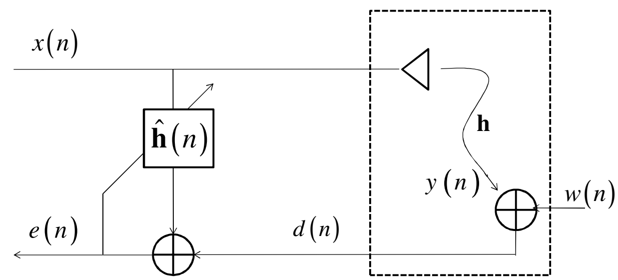

Figure 1 – Principle of Echo Cancellation.

2. FREQUENCY DOMAIN NLMS

Figure 1 depicts an adaptive filter in a typical echo canceller arrangement. The time-domain version of the NLMS algorithm can be summarized as the error calculation:

$$e(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}(n-1) \tag{1}$$

and the coefficient update calculation:

[1] This work was partly funded under the Wilkens Missouri Endowment.

[2] Formerly Steven L. Gay.

$$\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}(n-1) + \mu \frac{\mathbf{x}(n)e(n)}{\delta + \mathbf{x}^T(n)\mathbf{x}(n)}, \quad (2)$$

where $d(n)$ is the near-end or desired signal, $\mathbf{x}(n)$ is the far-end or excitation signal vector, $\hat{\mathbf{h}}(n)$ is the echo path impulse response estimate at time n, $e(n)$ is the error signal, μ is the step size, and δ is the regularization parameter. The desired signal, $d(n)$ is

$$d(n) = y(n) + w(n), \quad (3)$$

where $y(n)$ is the echo and $w(n)$ is the background noise; $y(n) = \mathbf{h}^T\mathbf{x}(n)$ where $\mathbf{h}$ is the true echo path impulse response vector of length N.

The algorithm for a frequency-domain LMS algorithm is described in [2]. Here we present a slightly modified version which includes bin-by-bin normalization as well as regularization.

For a filter of size N, we define $\mathbf{\Omega}(k)$ as the Fourier transform of the estimated filter, padded with N zeros,

$$\mathbf{\Omega}(k) = \mathbf{F}\begin{bmatrix} \hat{\mathbf{h}}^T(k) \\ \mathbf{0}_N \end{bmatrix}, \quad (4)$$

where $\mathbf{0}_N$ is a column vector of N zeros and $\mathbf{F}$ is the $2N \times 2N$ DFT matrix. We define

$$\mathbf{X}(k) = \mathbf{F}\begin{bmatrix} \mathbf{x}(k-1) \\ \mathbf{x}(k) \end{bmatrix}, \quad (5)$$

where

$$\mathbf{x}(k) = \begin{bmatrix} x(kN) & \cdots & x(kN+N-1) \end{bmatrix}^T. \quad (6)$$

Then, we get

$$\mathbf{y}(k) = \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \end{bmatrix} \mathbf{F}^{-1}\left[\mathbf{\Omega} \circ \mathbf{X}(k)\right], \quad (7)$$

where the symbol $\circ$ denotes the Hadamard product, $\mathbf{0}_{N\times N}$ is an all zero $N \times N$ matrix, and $\mathbf{I}_{N\times N}$ is the $N \times N$ identity matrix. The error can then be found as

$$\mathbf{E}(k) = \mathbf{F}\begin{bmatrix} \mathbf{0}_N \\ \mathbf{d}(k) - \mathbf{y}(k) \end{bmatrix}. \quad (8)$$

We define

$$\mathbf{\Delta}(k) = \begin{bmatrix} \mathbf{I}_{N\times N} & \mathbf{0}_{N\times N} \end{bmatrix} \mathbf{F}^{-1}\left(\mathbf{E}(k) \circ \mathbf{X}^*(k) \circ \mathbf{P}(k)\right), \quad (9)$$

where the symbol $*$ denotes conjugation,

$$\mathbf{P}(k) = diag\left\{\left(\mathbf{S}(k) + \delta\mathbf{I}\right)^{-1}\right\}, \quad (10)$$

where $diag\{\bullet\}$ means to convert the diagonal of a matrix into a vector. Furthermore,

$$\mathbf{S}(k) = \lambda\mathbf{S}(k-1) + (1-\lambda)\mathbf{X}(k)\mathbf{X}^H(k), \quad (11)$$

where the superscript H denotes the Hermitian operator and λ is a forgetting factor.

Finally, the update equation is

$$\mathbf{\Omega}(k+1) = \mathbf{\Omega}(k) + \mu\mathbf{F}\begin{bmatrix} \mathbf{\Delta}(k) \\ \mathbf{0}_N \end{bmatrix}. \quad (12)$$

By using a frequency-domain version of the NLMS algorithm, it is possible to measure the SNR separately in each frequency bin, which will allow calculating a different regularization parameter for each frequency bin as well. In the following section, we outline our algorithm for adaptively estimating the regularization parameter.

3. ADAPTIVE REGULARIZATION FOR THE NLMS ALGORITHM

To adaptively optimize the regularization parameter δ in the algorithm above, we use the criterion described in [3], where in the time domain, the variance of the a posteriori error is set equal to the variance of the background noise,

$$\sigma_\varepsilon^2 = \sigma_w^2, \quad (13)$$

where the a posteriori error is defined as

$$\varepsilon(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}(n). \quad (14)$$

Using (2) it is easily shown that

$$\varepsilon(n) = \left[1 - \mu\frac{\mathbf{x}^T(n)\mathbf{x}(n)}{\delta + \mathbf{x}^T(n)\mathbf{x}(n)}\right]e(n). \quad (15)$$

Taking the variance of $\varepsilon(n)$, we get

$$\sigma_\varepsilon^2 = E\left[\left(1 - \mu\frac{\mathbf{x}^T(n)\mathbf{x}(n)}{\delta + \mathbf{x}^T(n)\mathbf{x}(n)}\right)^2 e^2(n)\right]. \quad (16)$$

Assuming that we can approximate $E[\mathbf{x}^T(n)\mathbf{x}(n)] \approx N\sigma_x^2$, then using (16) in (13) and rearranging terms we derive

$$\delta = N\sigma_x^2\left[\frac{\mu\sigma_e}{\sigma_e - \sigma_w} - 1\right]. \quad (17)$$

This equation can be further simplified, by assuming the step size $\mu = 1$, in which case, we get

$$\delta = N\sigma_x^2\frac{\sigma_w}{\sigma_e - \sigma_w}. \quad (18)$$

In our simulations, we use this equation to find the value of δ for each frequency bin. As such, the values of σ_x^2, σ_w, and σ_e are also separately calculated for each frequency bin. We also note that in the frequency bins, there is only one adaptive filter tap per bin, so $N = 1$.

To compute the values of σ_x^2, σ_w, and σ_e we use leaky integrators. For example, for a certain frequency bin f, the k^{th} iteration for the excitation variance is

$$\begin{aligned} &\sigma_x^2(f,k) \\ &\quad = \lambda\sigma_x^2(f,k-1) + (1-\lambda)\mathrm{X}(f,k)\mathrm{X}^*(f,k), \end{aligned} \quad (19)$$

where $X(f,k)$ is the f^{th} element in $\mathbf{X}(k)$ corresponding to frequency bin f. Similarly, σ_w, and σ_e were found using

$$\sigma_w^2(f,k) = \lambda\sigma_w^2(f,k-1)+(1-\lambda)W(f,k)W^*(f,k) \tag{20}$$

and

$$\sigma_e^2(f,k) = \lambda\sigma_e^2(f,k-1)+(1-\lambda)E(f,k)E^*(f,k), \tag{21}$$

where $W(f,k)$ and $E(f,k)$ represent the noise and error elements at frequency f and block k, respectively. Our adaptive regularization parameter is thus,

$$\delta(f,k)=\sigma_x^2(f,k)\frac{\sigma_w(f,k)}{\sigma_e(f,k)-\sigma_w(f,k)}. \tag{22}$$

The problem with this result is that we assume that the frequency noise variance $\sigma_w^2(f,k)$ is known. This, however, is often not the case, so we estimate the variance of the background noise using the following analysis.

4. ESTIMATION OF THE BACKGROUND NOISE VARIANCE

In the previous section, we pointed out that in most cases it is not possible to know the background noise beforehand, and it is therefore necessary to estimate it. As we will see in the results section later in this paper, the known noise will lead to much better results, but the estimated noise variance still yields good results.

The estimate for the noise variance is obtained in the following manner [2]. The cross-correlation vector between the excitation signal and the error is defined as

$$\mathbf{r_{ex}}(n)=E[\mathbf{x}(n)e(n)]. \tag{23}$$

The error can be expressed as

$$\begin{aligned} e(n) &= \mathbf{h}^T\mathbf{x}(n)+w(n)-\hat{\mathbf{h}}^T(n)\mathbf{x}(n) \\ &= \Delta\mathbf{h}^T(n)\mathbf{x}(n)+w(n), \end{aligned} \tag{24}$$

where

$$\Delta\mathbf{h}(n)=\mathbf{h}-\hat{\mathbf{h}}(n). \tag{25}$$

Therefore, assuming that $x(n)$ is independent of $w(n)$ it is easily shown that

$$\mathbf{r_{ex}}(n)=\mathbf{R_{xx}}\Delta\mathbf{h}(n), \tag{26}$$

where $\mathbf{R_{xx}}=E\left[\mathbf{x}(n)\mathbf{x}^T(n)\right]$. The error variance is also easily shown to be

$$\sigma_e^2(n)=E[e^2(n)]=\Delta\mathbf{h}^T\mathbf{R_{xx}}\Delta\mathbf{h}+\sigma_w^2. \tag{27}$$

After rearranging the terms, the noise signal estimator can be defined as

$$\sigma_w^2=\sigma_e^2(n)-\Delta\mathbf{h}^T\mathbf{R_{xx}}\Delta\mathbf{h}. \tag{28}$$

If we assume (or approximate) the excitation signal is white then we can use (26) to calculate the background noise as

$$\sigma_w^2\approx\sigma_e^2(n)-\frac{\mathbf{r_{ex}}^T(n)\mathbf{r_{ex}}(n)}{\sigma_x^2(n)}. \tag{29}$$

In the frequency domain, $\mathbf{r_{ex}}(n)$ reduces to the scalar $r_{ex}(f,k)$ since there, the adaptive filter length is $N=1$. Calculating $r_{ex}(f,k)$ is once again done using the leaky integrator,

$$r_{ex}(f,k) = \lambda r_{ex}(f,k-1)+(1-\lambda)X(f,k)E^*(f,k). \tag{30}$$

The time varying noise estimate in the bins then becomes,

$$\sigma_w^2(f,k)\approx\sigma_e^2(f,k)-\frac{\left|r_{ex}(f,k)\right|^2}{\sigma_x^2(f,k)}. \tag{31}$$

This is used in (22) for the frequency-domain time-varying optimal regularization estimate.

5. SIMULATIONS AND RESULTS

For our simulations, we compared three scenarios. In all cases, the regularization parameter $\delta(f,k)$ was calculated adaptively. In each of the simulations for one plot, one $\delta(n)$ was calculated in the time domain and used for all frequency bands, in the other plot $\delta(f,k)$ is updated each frame and in each frequency bin where the number of frequency bins is equal to the number of filter.

To assess the performance of the algorithms and the effectiveness of the regularization method, we plot the normalized error coefficient as a function of time for each of the selected cases. The error coefficient is defined as

$$\xi(n)=10\log_{10}\left(\frac{\left(\mathbf{h}-\hat{\mathbf{h}}(n)\right)^T\left(\mathbf{h}-\hat{\mathbf{h}}(n)\right)}{\mathbf{h}^T\mathbf{h}}\right). \tag{32}$$

Figures 2 through 4 show the error coefficient defined in (32) for a number of input signal and background noise combinations. To demonstrate re-convergence and echo path change is effected in the middle of the simulations. Figure 2 shows the error coefficient when the input is a white noise signal while the background noise is also white. Results show that our algorithm for finding an adaptive regularization in the frequency domain does not provide improvement over that found using full band regularization. In fact, the full band calculation provides a slightly better error. This is to be expected since both the input and noise are white.

However, significant improvement is seen when a colored rather than a white input signal is used, as demonstrated by Figures 3 and 4. In Figure 3, the input signal is colored noise, and even though the background noise is white, we can clearly see that the frequency-domain adaptive regularization provides better performance than the full band regularization parameter. The results are similar when using colored noise in the background.

6. CONCLUSION

In this paper, we developed an adaptive algorithm for calculating the optimum regularization parameter (δ) for the NLMS algorithm in the frequency domain. To assess the ability of this algorithm to accurately determine the optimum value of δ in a colored noise environment, we used a frequency-domain version of the NLMS algorithm, and adaptively calculated a separate value for δ for each frequency bin. Simulations show that for the common case of colored excitation, the frequency domain adaptive regularization significantly outperforms adaptive regularization done in the time domain.

7. REFERENCES

[1] Ferrara, E. R., "Fast Implementation of LMS Adaptive Filters", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* Vol. ASSP-28, No. 4, August 1980

[2] M. A. Iqbal and S. L. Grant, "Novel Variable Step Size NLMS Algorithms for Echo Cancellation", *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP' 08)*, 2008

[3] J. Benesty, C. Paleologu, S. Ciochina, "On Regularization in Adaptive Filtering", *IEEE Transactions on Audio, Speech, and Language Processing,* Vol. 19, No. 6, August 2011

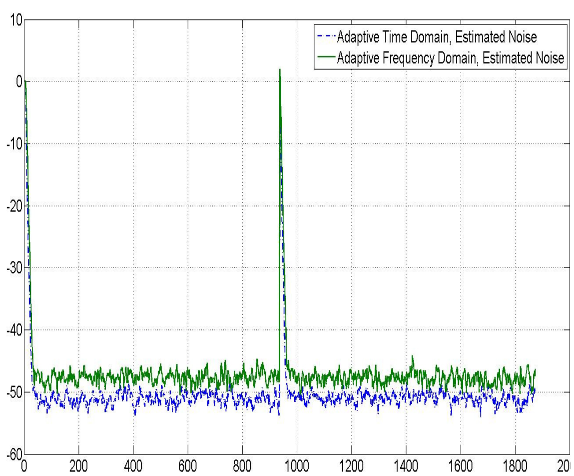

Figure 2. Coefficient error versus frame number. Both input and interfering signals are white.

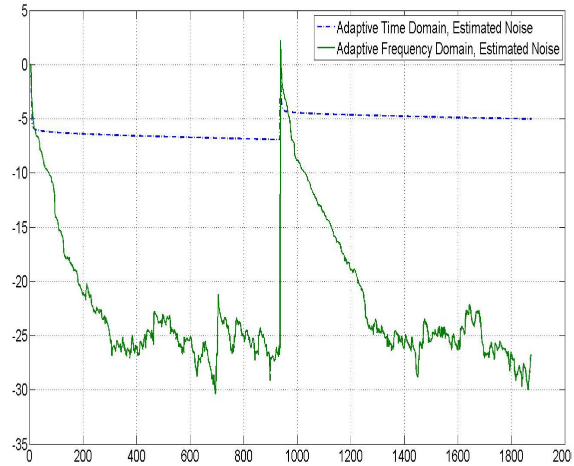

Figure 3. Coefficient error versus frame number. Input signal is colored and the interfering signal is white.

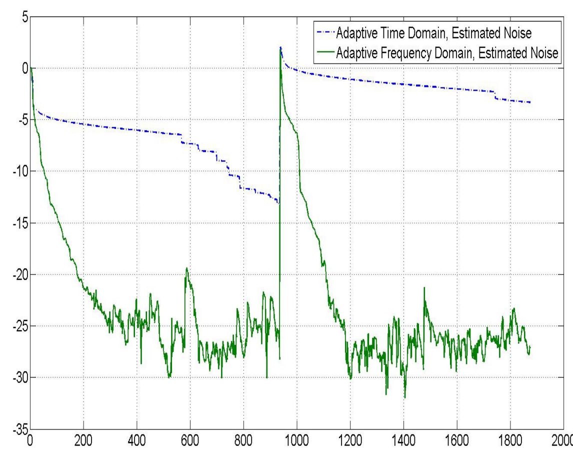

Figure 4. . Coefficient error versus frame number. Both input and interfering signals are colored.