Recirculation of input data in frequency-domain adaptive filtering

Filtrage adaptatif dans le domaine de fréquence par récirculation de données d'entrée

By S.J. Chern *and* K.M. Wong, *Department of Electrical and Computer Engineering and Communications Research Laboratory, McMaster University, Hamilton, Ontario.*

The recirculation of the input data is proposed in the application of frequency-domain adaptive filtering when the supply of input data is limited. An analysis of the performance of such an algorithm is presented and it indicates that superior performance can be achieved. Examples of its applications in adaptive filtering confirm the results of analysis and demonstrate the effectiveness of the proposed algorithm.

En filtrage adaptatif dans le domaine de fréquence, lorsque le nombre des données est limité, on propose un algorithme basé sur la récirculation de données d'entrée. On présente une analyse de la performance d'un tel algorithme et démontre que l'on peu obtenir une performance améliorée. On donne des exemples d'application en filtrage adaptatif qui confirment les résultats d'analyse et qui démontrent l'efficacité de l'algorithme proposé.

Introduction

In many engineering applications, adaptive filters are essential. Such applications include array processing, noise and echo cancellation, statistical estimation, and channel equalization in data communication systems. Traditionally, an adaptive filter is implemented as a tapped delay line so that the output is the sum of the weighted input samples. The values of the tap weights are adjusted by a recursive algorithm – most commonly the least mean square (LMS) algorithm. An adaptive filter of the type described above is called a time-domain adaptive filter since the adjustment of its tap weights is based on the sample values in the time-domain. In time-domain adaptive filtering, the analysis involved in computing the statistics of weight misadjustment can be complicated when the desired signal and the input to the filter are highly correlated.

On the other hand, adaptive filtering can adopt a frequency-domain approach such that the computationally efficient fast Fourier transform (FFT) algorithm is used. The FFT algorithm transforms the input signal into the frequency domain and then performs the optimization of the tap weights for each frequency bin separately. Implementation of the LMS algorithm in the frequency-domain can give significant reductions in computation over the conventional time domain. Furthermore, under certain conditions, both the mean and the variance of the top weights of the frequency domain adaptive filter can be obtained relatively simply allowing statistical analysis to be performed[1-5], and can often be used to predict the performance of time-domain adaptive filtering.[2]

In this paper, we introduce the recirculation of the input data in adaptive filtering the study and analysis of which are carried out in the frequency domain for the reasons stated above.

The method of recirculating the data is similar to the approach discussed in References 10-12, where the on-line algorithm or the adaptive algorithm is applied with off-line or batch data to the identification problem. The advantages for using such an adaptive algorithm for batch data are:

- If one already has recursive software available, it may be more expensive to develop batch software than to use the available software.
- The cost function minimum can sometimes be found faster with a recursive algorithm. If the recursive estimates from one pass are close to the minimum, then the initial conditions for the next pass (being close to the minimum) will essentially bring the estimate to the minimum.
- The recursive algorithms can be used to detect non-stationarities in the data and the system properties.

In the LMS adaptation algorithms, the modes of the adaptive process converge at different rates such that the rate of each mode is determined by the associated eigenvalues of the input autocorrelation matrix. For a large disparity of eigenvalues, in order to keep the algorithms stable, the LMS adaptive process may converge slowly, and the algorithm suggested by Ogue et. al[14] will be useful in this case.

However, in this paper, we focus on the effects of the recirculating data in the adaptive algorithm via the frequency-domain implementation. Furthermore, in our application to broadband noise cancellation problems, we consider cases when noise power is much higher than signal power. Thus, the power spectrum does not vary much with frequency, which results in autocorrelation matrix eigenvalues of similar magnitudes. Also, the convergence rates in different modes are close. Therefore, the other algorithms which are suitable when large eigenvalue disparity exists are not discussed here but may be examined in references 13-16.

We first briefly review the frequency-domain adaptive filtering algorithm. The circular convolution frequency-domain adaptive filter, illustrated in Figure 1, is different from the conventional time-domain filter in that the input signal is processed in blocks. The input signal $\{x(l)\}$ is sectioned into M signal vectors each consisting of N data samples such that the mth input signal vector is given by

$$\underline{x}_m = [x(mN)x(mN+1)\dots x(mN+N-1)]^T \tag{1}$$

An N-point FFT of this signal vector is achieved yielding N frequency samples, $X_k(m)$, $k = 0, \dots, N-1$, where

$$X_k(m) = \sum_{n=0}^{N-1} x(mN + n)e^{-j\frac{2\pi}{N}nk} \tag{2}$$

Each of these frequency samples is then multiplied by a tap weight $W_k(m)$ to produce an output frequency sample $Y_k(m)$. An N-point IFFT is then performed on these frequency samples to obtain the output vector $\underline{y}_m$ so that

$$\underline{y}_m = [y(mN)y(mN + 1) \ldots y(mN + N - 1)]^T \tag{3}$$

where

$$y(mN + n) = \frac{1}{N}\sum_{k=0}^{N-1} Y_k(m)e^{j\frac{2\pi}{N}nk},\ n = 0, \ldots, N - 1 \tag{4}$$

with

$$Y_k(m) = W_k(m)X_k(m) \tag{5}$$

If the mth desired signal vector is $\underline{d}_m$ such that

$$\underline{d}_m = [d(mN)\ d(mN + 1) \ldots d(mN + N - 1)]^T \tag{6}$$

and its kth frequency sample is given by

$$D_k(m) = \sum_{n=0}^{N-1} d(mN + n)e^{-j\frac{2\pi}{N}nk} \tag{7}$$

The object of frequency-domain adaptive filtering is to adjust the tap weights $W_k(m)$ so that mean square error ϵ^2_0 is minimised where

$$\epsilon_0^2 = \overline{|E_k(m)|^2} = \overline{|D_k(m) - Y_k(m)|^2} \tag{8}$$

$\overline{[\cdot]}$ denotes the ensemble average. The tap weight, $W_k(m)$, for each frequency sample is adjusted according to the steepest descent algorithm[6,7]

$$W_k(m + 1) = W_k(m) + \mu E_k(m)X_k^*(m),\ 0 \leqq m \leqq M - 1 \tag{9}$$

where * denotes the complex conjugate, until a minimum ϵ^2 is reached.

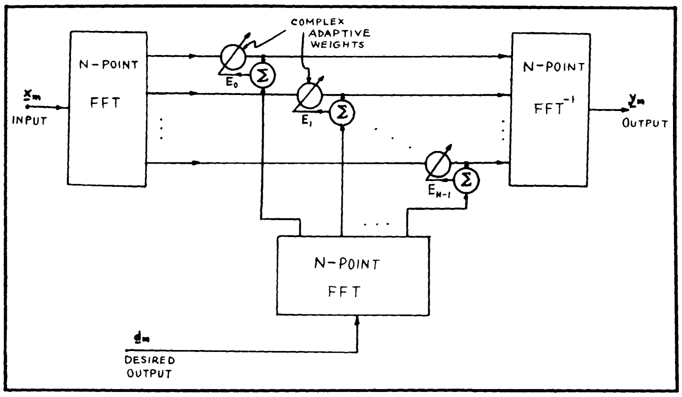

Figure 1: *An adaptive filter in the frequency domain.*

The circular convolution frequency-domain adaptive filter described above can be employed in many cases in which the time-domain adaptive filter is applicable. In this paper, we consider the special case of applying it to signal processing cases where the number of input signal samples available is often limited. Under such circumstances, the adaptive filter may not have sufficient supply of signal samples to converge to the state of minimum mean square error and, therefore, may not operate in its optimum state. In order to improve on this situation, an algorithm is proposed in next section that offers a limited supply of signal samples recirculated to the adaptive filter so that further adjustment of tap weights can be carried out. It is found that the use of this algorithm significantly improves the performance of the adaptive filter. An analysis of the effects of the signal recirculation is presented and confirmed by computer simulation. The performances of the adaptive filters with and without signal recirculation are then compared and the results presented.

Recirculation of input data in frequency-domain adaptive filtering

To facilitate our analysis, we assume that the frequency-domain adaptive filtering is employed in an environment such that the desired signal vector $\underline{d}_m$ and the input signal vector $\underline{x}_m$ contain a signal vector $\underline{s}_m$ buried in statistically independent white noise. Thus, the kth frequency component of $\underline{d}_m$ and $\underline{x}_m$ are given by

$$D_k(m) = \alpha_k S_k(m) + N_{1k}(m)$$
$$X_k(m) = S_k(m) + N_{2k}(m) \tag{10}$$

where α_k is a complex coefficient. This situation arises very often in signal processing especially in problems concerning noise cancellation and time-delay estimation.

Furthermore, we assume that the frequency components of both sequences $\underline{d}_m$ and $\underline{x}_m$ are zero-mean, white complex circular Gaussian processes.[8] This means that the signal and noise in the kth frequency components are related by the following equations

$$\overline{[S_k(m)S_k^*(n)]} = \sigma^2_{Sk}\delta_{mn}, \quad k = 0, \ldots, N - 1$$
$$\overline{[S_k(m)S_k(n)]} = 0 \quad \forall m, n \tag{11a}$$
$$\overline{[N_{ik}(m)[N_{ik}^*(n)]} = \sigma^2_N\delta_{mn}, \quad (i = 1, 2)$$
$$\overline{[N_{1k}(m)N_{2k}^*(n)]} = 0 \tag{11b}$$

where σ^2_{Sk} is the signal power in the kth bin of the FFT of the signal; σ^2_N is the noise power in every frequency component (white noise); and δ_{mn} is the Kronecker delta. The assumption of signal and noise properties in Equation (11) is valid in broad-band noise cancellation and in time-delay estimation when the number of time-delay samples is small compared to the order of the FFT employed.[3]

The configuration of a frequency-domain adaptive filter having the ability of recirculation of input data is shown in Figure 2. This configuration is similar to a simple circular convolution frequency-domain adaptive filter. The additional features are the delay elements which carry out the function of storing the signals

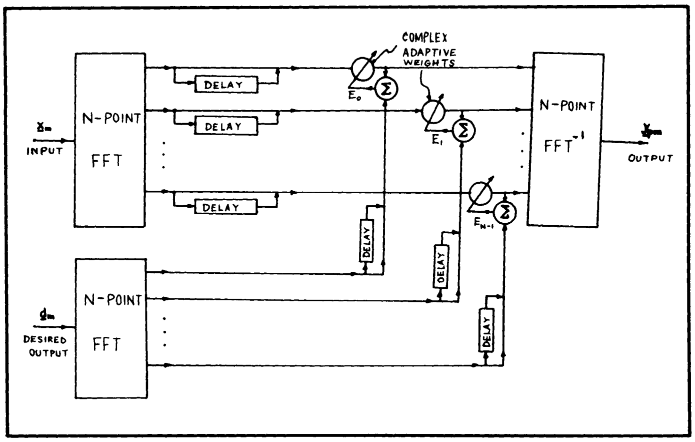

Figure 2: *A frequency-domain adaptive filter with input recirculation facility*

$X_k(m)$ and $D_k(m)$. These delays must be at least of length MN samples so as to allow all the input signal samples to be processed first before the next recirculation begins. With this extra facility, the notations for the output signal vector, the error vector, and the tap weights of the adaptive filter have to be slightly modified. We let:

$W_k(p, m)$ be the kth complex tap weight for the mth N-point signal block;

$Y_k(p, m)$ be the kth output sample for the mth N-point signal block;

$E_k(p, m)$ be the kth error sample for the mth N-point signal block;

where p denotes the number of times the input data have been recirculated. Thus, for a simple frequency-domain adaptive filter with no recirculation of input data, $p = 0$. The procedure of steepest descent is again used for updating the tap weights,

$$W_k(p, m+1) = W_k(p, m) + \mu E_k(p, m) X_k^*(m) \quad m = 0, 1, \ldots, M-1 \tag{12}$$

Note that

$$W_k(p+1, 0) = W_k(p, M) \tag{13}$$

Also, since $X_k(m)$ and $D_k(m)$ are the recirculated signals, they are independent of p.

Since

$$E_k(p, m) = D_k(m) - Y_k(p, m) = D_k(m) - W_k(p, m) X_k(m) \tag{14}$$

Equation (13) can be rewritten as

$$W_k(p, m+1) = W_k(p, m)[1 - \mu|X_k(m)|^2] + \mu D_k(m) X_k^*(m) \tag{15}$$

which is a first order difference equation. The solution of which can be written as

$$W_k(p, m+1) = W_k(p, 0) \prod_{i=0}^{m} [1 - \mu|X_k(i)|^2] + \mu \sum_{j=0}^{m} D_k(j) X_k^*(j) \prod_{l=j+1}^{m} [1 - \mu|X_k(l)|^2] = W_k(p, 0) A_k(m) + B_k(m) \tag{16}$$

where

$$A_k(m) = \prod_{i=0}^{m} [1 - \mu|X_k(i)|^2] \tag{17a}$$

and

$$B_k(m) = \mu \sum_{j=0}^{m} D_k(j) X_k^*(j) \prod_{l=j+1}^{m} [1 - \mu|X_k(l)|^2] \tag{17b}$$

Now, if we start the tap weights at zero value, i.e. $W_k(0,0) = 0$, then at the beginning of the first recirculation, we have from Equations (13) and (16),

$$W_k(1, 0) = W_k(0, M) = B_k(M-1)$$

and at the end of the first recirculation, we have

$$W_k(1, M) = B_k(M-1) A_k(M-1) + B_k(M-1) = B_k(M-1)[1 + A_k(M-1)]$$

Again, using Equations (13) and (16) and continuing the recursive relationship, we have, at the end of the pth recirculation of input data,

$$W_k(p, M) = \sum_{n=0}^{p} B_k(M-1) A_k^n(M-1) \tag{18}$$

Since $D_k(m)$ and $X_k(m)$ are random variables, $A_k(M-1)$ and $B_k(M-1)$ are also random variables. However, they are correlated. In order to compute for the mean and mean-square values of the tap weights, it is necessary to separate $A_k(M-1)$ and $B_k(M-1)$ into their correlated and uncorrelated parts. From Equations (17a) and (17b),

$$B_k(M-1) A_k^n(M-1) = \left\{ \mu \sum_{j=0}^{M-1} D_k(j) X_k^*(j) \prod_{l=j+1}^{M-1} [1 - \mu|X_k(l)|^2] \right\} \left\{ \prod_{i=0}^{M-1} [1 - \mu|X_k(i)|^2]^n \right\}$$

$$= \sum_{j=0}^{M-1} \mu D_k(j) X_k^*(j) [1 - \mu|X_k(j)|^2]^n \prod_{i=0}^{j-1} [1 - \mu|X_k(i)|^2]^n \prod_{l=j+1}^{M-1} [1 - \mu|X_k(l)|^2]^{n+1} \tag{19a}$$

$$= \sum_{j=0}^{M-1} \left\{ \Theta_k(j, n) \prod_{i=0}^{j-1} \Phi_k(i, n) \prod_{l=j+1}^{M-1} \Psi_k(l, n) \right\} \tag{19b}$$

where

$$\Theta_k(j, n) = \mu D_k(j) X_k^*(j) [1 - \mu|X_k(j)|^2]^n \tag{20a}$$

$$\Phi_k(i, n) = [1 - \mu|X_k(i)|^2]^n \tag{20b}$$

and

$$\Psi_k(l, n) = [1 - \mu|X_k(l)|^2]^{n+1} \tag{20c}$$

Now, $\Theta_k(j, n)$, $\Phi_k(i, n)$ and $\Psi_k(l, n)$ are uncorrelated since they contain input signals from different blocks. Therefore,

$$\overline{[W_k(p, M)]} = \sum_{n=0}^{p} \overline{[B_k(M-1) A_k^n(M-1)]}$$

$$= \sum_{n=0}^{p} \left\{ \sum_{j=0}^{M-1} \overline{[\Theta_k(j, n)]} \prod_{i=0}^{j-1} \overline{[\Phi_k(i, n)]} \prod_{l=j+1}^{M-1} \overline{[\Psi_k(l, n)]} \right\} \tag{21}$$

It can be shown (*See Appendix 1*) that

$$\overline{[\Theta_k(j, n)]} \simeq \mu\alpha_k\sigma_{Sk}^2\{1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)\} \tag{22a}$$

$$\overline{[\Phi_k(i, n)]} \simeq 1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2) \tag{22b}$$

and

$$\overline{[\Psi_k(l, n)]} \simeq 1 - (n + 1)\mu(\sigma_{Sk}^2 + \sigma_N^2) \tag{22c}$$

where $\sigma^2{}_{Sk}$ and $\sigma^2{}_N$ are the values of the signal and noise power in the kth frequency component. From Equations (22), we see that the value of $\overline{[W_k(p, M)]}$ is a function of $\mu(\sigma^2{}_{Sk} + \sigma^2{}_N)$. To simplify the notations, we define the normalized step size $\tilde{\mu}_k$ as

$$\tilde{\mu}_k = \mu(\sigma_{Sk}^2 + \sigma_N^2) \tag{23}$$

It is generally chosen such that $\tilde{\mu}_k \ll 1$. Using this condition and substituting Equations (22) into Equation (21), the mean of the kth tap weight can be expressed as (*see Appendix 1*)

$$\overline{W_k(p, M)} \simeq \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)}[1 - \{1 - (p + 1)\tilde{\mu}_k\}^{M+2}] \tag{24}$$

Equation (24) gives the mean value of the kth tap weight. It is clear that if there is a large number of data blocks (i.e., if $M \to \infty$), then the final mean value of the kth tap weight is given by

$$W_{k\infty} = \lim_{M\to\infty} \overline{W_k(p, M)} = \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} \tag{25}$$

which is reminiscent of the optimum Wiener solution. Thus, if we have a large number of data blocks, there is no necessity to recirculate the input data. However, if the number of data blocks is limited, it can be deduced from Equation (24) that by recirculating the input (i.e., $p > 0$), the mean value of the tap weight will be closer to the final optimum value.

To find the mean square of the tap weight, we use Equation (18). After simplification, we obtain (*see Appendix 2*)

$$\overline{|W_k(p, M)|^2} = \overline{C}_{1k}(p, M - 1) + \overline{C}_{2K}(p, M - 1) + \overline{C}_{3k}(p, M - 1) \tag{26}$$

where

$$\overline{C}_{1k}(p, M - 1) \simeq \frac{1}{2}\tilde{\mu}_k\left\{|W_{k\infty}|^2 + \left(\frac{|\alpha_k|^2 - 1}{\alpha_k}\right)W_{k\infty} + 1\right\}$$

$$\left[1 - \{1 - 2(p + 1)\tilde{\mu}_k\}^{M+2} + 2\sum_{n=1}^{p}\{1 - n\tilde{\mu}_k\}^{M+2} - \{1 - (p + n + 1)\tilde{\mu}_k\}^{M+2}\right] \tag{27}$$

$$\overline{C}_{2k}(p, M - 1) = \overline{C}_{3k}(p, M - 1)$$

$$\simeq \frac{1}{2}|W_{k\infty}|^2[1 - 2\{1 - (p + 1)\tilde{\mu}_k\}^{M+4} + \{1 - 2(p + 1)\tilde{\mu}_k\}^{M+4}] \tag{28}$$

Again, for a large number of data blocks

$$\lim_{M\to\infty} \overline{|W_k(p, M)|^2} = \frac{1}{2}\tilde{\mu}_k\left\{|W_{k\infty}|^2 + \left(\frac{|\alpha_k|^2 - 1}{\alpha_k}\right)W_{k\infty} + 1\right\} + |W_{k\infty}|^2 \tag{29}$$

The last term in Equation (29) is the square of the mean of the tap weight for a large number of data blocks. The first term is the variance which is proportional to the normalized step size μ_k. The smaller is $\tilde{\mu}_k$, the smaller is the variance.

The mean-square error for the frequency-domain adaptive filter having the input data recirculated p times can be simply evaluated as follows:

$$\epsilon_p^2 = \overline{|D_k(m) - Y_k(p, m)|^2}$$
$$= \overline{|\alpha_k S_k(m) - W_k(p, m)S_k(m)|^2} + \overline{|N_{1k}(m)|^2} + \overline{|W_k(p, M)N_{2k}(m)|^2} \tag{30}$$

Using Equation (11), we can write

$$\epsilon_p^2 = \sigma_{Sk}^2\,\overline{[|\alpha_k - W_k(p, M)|^2]} + \sigma_N^2\,\overline{[1 + |W_k(p, M)|^2]} \tag{31}$$

Substituting the values of $\overline{W_k(p, M)}$ and $\overline{|W_k(p, M)|^2}$ from Equations (24) and (26) respectively into Equation (31), the value of the mean-square error after recirculating the input data p times can be evaluated. Note that if $p = 0$, the values of $\overline{W_k(0, M)}$ and $\overline{|W_k(0, M)|^2}$ can be more accurately obtained by using Equations (A.10) and (A.24) respectively in Appendices 1 and 2. Again, if there is a large number of data blocks (i.e., for $M \to \infty$), then the mean-square error becomes

$$\epsilon_\infty^2 = \lim_{M\to\infty} \epsilon_p^2 = \epsilon_{\min}^2 + \epsilon^2 \tag{32}$$

where

$$\epsilon_{\min}^2 = \sigma_N^2\left[\frac{|\alpha_k|^2\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} + 1\right] \tag{33}$$

$$\epsilon^2 = \frac{\tilde{\mu}_k}{2}\left[\frac{|\alpha_k|^2\sigma_{Sk}^4}{(\sigma_{Sk}^2 + \sigma_N^2)^2} + (|\alpha_k|^2 - 1)\frac{\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} + 1\right] \tag{34}$$

$\epsilon^2{}_{\min}$ is a constant for given values of α_k, signal and noise power. $\epsilon^2{}_{\min}$ represents the lower bound of the mean-square error when the frequency-domain adaptive filter is applied to the input signal and noise characterized by Equation (11). On the other hand, ϵ^2 is a non-negative quantity dependent on $\tilde{\mu}_k$. In theory, if we have an infinite amount of input data, we can reduce the mean-square error arbitrarily close to the lower bound by choosing a very small μ_k. For limited supply of data, however, the dependence of $\epsilon^2 p$ on $\overline{W_k(p, M)}$ and $\overline{|W_k(p, M)^2|}$ means that by recirculating the input signal, the mean-square error will be reduced further. As a comparison of the recirculating algorithms with various values of p, we plot the normalized step-size $\tilde{\mu}_k$ against the relative mean-square error $\epsilon^2{}_{Rp}$ which is defined as

$$\epsilon_{Rp}^2 = \frac{\epsilon_p^2 - \epsilon_{\min}^2}{\epsilon_{\min}^2}, \quad p = 0, 1, \ldots \tag{35}$$

where ϵ^2_{min} is the minimum mean-square error given by Equation (33), and ϵ^2_p, the mean-square error of the filter having the input data recirculated p times, is given by Equation (31). The results are shown in Figure 3, where M, the number of blocks, is taken to be 160, and the signal-to-noise ratio is selected at -5 dB. The range of values of $\tilde{\mu}_k$ is chosen so that $\tilde{\mu}_k \ll 1$, an assumption necessary to arrive at Equation (31).

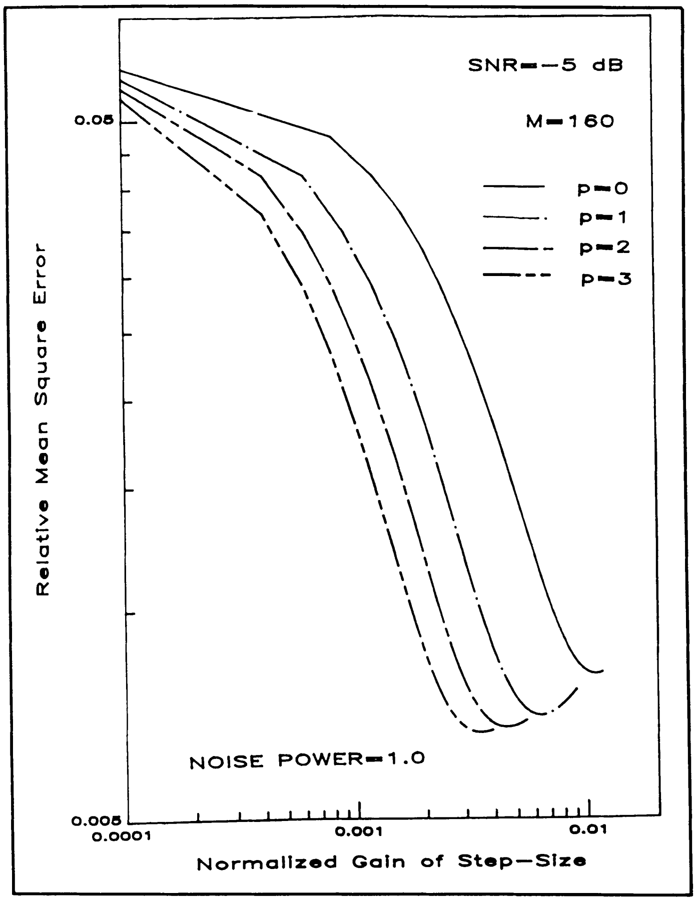

Figure 3: *Comparison of performances of adaptive filters with various number of input recirculations.*

The graphs for various values of p exhibit common characteristics:

- For the same value of $\tilde{\mu}_k$, $\epsilon^2_{Rp} < \epsilon^2_{Rq}$ for $p > q$. This result is obvious from Equations (24), (26) and (31) where it can be observed that ϵ^2_p decreases with p.
- Each of the curves decreases as $\tilde{\mu}_k$ increases until a minimum value of ϵ^2_{Rp} is reached. Then the relative mean-square error increases again beyond this minimum. This result is reasonable because a very small normalized step size $\tilde{\mu}_k$ will take a long time (i.e., a large amount of data) to converge to the minimum error. For a limited supply of input data ($M = 160$ in this case), a larger $\tilde{\mu}_k$ will yield smaller error. However, a normalized step-size $\tilde{\mu}_k$ too large will yield larger error again since the adjustment will then be too coarse. The optimum normalized step-sizes $\tilde{\mu}_{opt}$ for various values of p in this particular case are shown in Table 1. The corresponding values of ϵ^2_{Rp} are also tabulated.

Table 1
Optimum normalized step-size for different values of p

p	$\tilde{\mu}_{opt}$	ϵ^2_{Rp}
0	0.0108	8.044×10^{-3}
1	0.0063	7.023×10^{-3}
2	0.0044	6.756×10^{-3}
3	0.0035	6.647×10^{-3}

- It is observed that the optimum normalized step-size $\tilde{\mu}_{opt}$ decreases as p increases. Furthermore, the optimum normalized step-size obeys approximately the relationship that

$$(p + 1)\tilde{\mu}_{opt} = 1.24 \times 10^{-2} \tag{36}$$

This again is intuitively reasonable because recirculating the input data several times is analogous to having larger supply of input data for which a smaller $\tilde{\mu}_k$ yields smaller error.

Simulation and comparison of performances

To evaluate the performance of the new algorithm of recirculating input data, and to compare it with the normal circular convolution frequency-domain adaptive filtering, a computer program was written to simulate the two methods. The simulation program can essentially be represented by the block diagram shown in Figure 4.

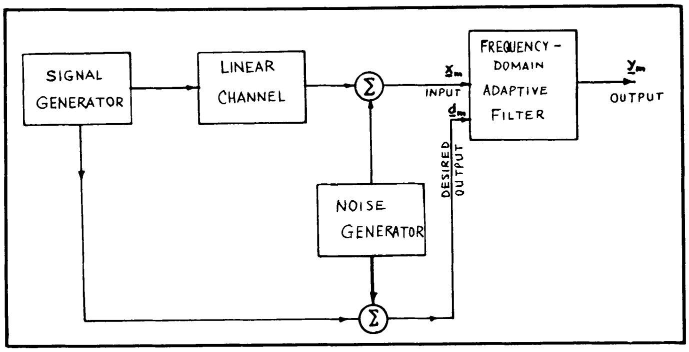

Figure 4: *Block diagram for computer simulation.*

The signal generator produces a discrete circular Gaussian process, the Fourier transform of which obeys Equation (11a). This signal is separated into two branches. The lower branch carries the signal $\{s\}$ which encounters additive circular Gaussian noise $\{n_1\}$ generated by a noise-generating subroutine. This mixture of signal and noise represents the desired signal sequence $\{d\}$. The upper branch passes the generated signal through a linear channel and then a circular Gaussian noise sequence $\{n_2\}$ is added. This represents the received signal sequence $\{x\}$.

The frequency-domain adaptive filters with and without input sample recirculation are simulated according to Figures 1 and 2 respectively. In our simulation, both the adaptive filters utilise an FFT of 32 points (i.e. $N = 32$) and both use the complex LMS algorithm developed by Widrow et al.[7] The comparison of performances is carried out between the adaptive filter without input sample recirculation and the adaptive filter which recirculates the input samples once only. With m blocks of input data, the filter without recirculation facility would have m iterations of adjusting the tap-weights. The mean-square error, $\epsilon^2_0(m)$ which is a function of m, is evaluated. The relative mean-square error, $\epsilon^2_{R0}(m)$ defined in Equation (35) for $p = 0$ is then calculated.

Now, with the same m blocks of input data, the performance of the adaptive filter with the facility of recirculating the input data once is tested. This is equivalent to having a normal frequency-domain adaptive filter with no recirculation facility but having the same m blocks of input data twice. Again the mean-square error $\epsilon^2_1(m)$ and the relative mean-square error $\epsilon^2_{R1}(m)$ are calculated. The values of $\epsilon^2_{R0}(m)$ and $\epsilon^2_{R1}(m)$ are plotted for various values of m. These theoretical mean-square errors for the two filters are also plotted for comparison. The following examples will illustrate the comparison of the performances of the two filters.

Example 1

In this example we choose the linear channel shown in Figure 4 to be a pure delay, i.e., the desired and received signals are respectively given by

$$d(n) = s(n) + n_1(n)$$

$$x(n) = s(n - \nu_0) + n_2(n)$$

where ν_0 is a constant. The variance of $s(n)$ is chosen to be unity and the signal-to-noise ratio at the input of the filter is 9.55 dB so that the variances of $n_1(n)$ and $n_2(n)$ are each equal to 0.1109. The total data length used for simulation is 4800 samples and is divided into 150 blocks each having 32 samples. These blocks are processed by the frequency-domain adaptive filter. For every m blocks of data the relative mean-square errors $\epsilon^2_{R0}(m)$ and $\epsilon^2_{R1}(m)$ are evaluated and are plotted in Figure 5. Also shown in Figure 5 are the theoretical values of these relative mean-square errors. It can be seen that the mean-square errors from simulation agree closely with their theoretical counterparts.

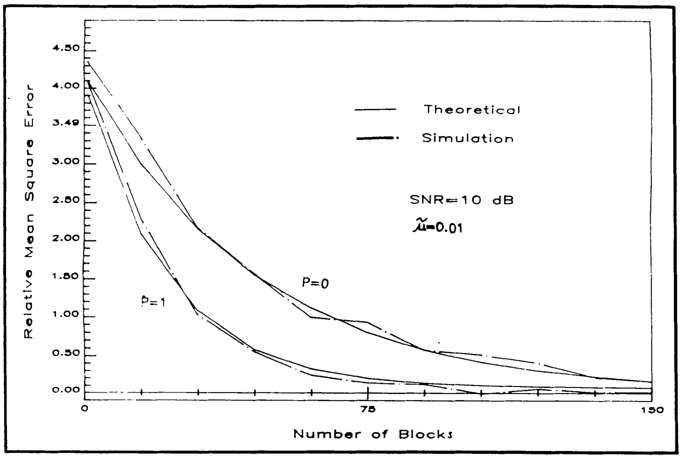

Figure 5: *Comparison of performances for the adaptive filters in example 1.*

Other examples were tested, and it was observed that the relative mean-square errors from simulations are in very close agreement with their theoretical values. Thus, in order to depict the comparison of the performances of the two filters so that curves of very close values are avoided, only the theoretical values of the relative mean-square errors are plotted. Note that, the relative mean-square error defined in Equation (35) is the ratio of the difference of the mean-square error and the minimum MSE. This means that in both the recirculation and without recirculation adaptive filtering algorithms we compared the results with the reminiscent frequency-domain optimum Wiener solution. Since the Dentino-type frequency-domain adaptive algorithm we considered is not the direct realization of the conventional time domain LMS algorithm, the reminiscent frequency-domain optimum solution is not necessarily equal to the time-domain Wiener solution. This implies that the minimum mean-square error in the time domain is, in general, not equal to the Dentino-type frequency-domain minimum mean-square error. To match the assumptions we made in Equations (11) for the frequency-domain algorithm, the signals and noises in time domain were assumed to be white Gaussian random processes which results in similar values of the minimum MSE for both time domain and frequency-domain algorithms.

Example 2

In this example, the signal model and channel model are chosen to be identical to those in Example 1. The purpose here is to show the effect of step-size on the relative mean-square error. We illustrate two cases, the first has a high signal-to-noise ratio of 10 dB, and the second has a low signal-to-noise ratio of -5 dB. In each case we use normalized step-size $\tilde{\mu}_k$ of 0.01, 0.0025 and 0.001; and in each case we compare the adaptive filter without recirculation of input data to that with input data recirculated once. The results are shown in Figures 6 and 7 for the high and low signal-to-noise ratio respectively. The scale of relative mean-square error in Figure 6 is of two orders higher than that in Figure 7. This is because the noise power is the same in both cases whereas the signal-to-noise ratios are vastly different. However, since the desired signal $D_k(n)$ consists of noise of equal power, the minimum mean-square errors $\epsilon^2_{\min}$ in both cases would be of similar order as shown in Equation (33). Indeed, the minimum mean-square errors are 1.91 and 1.24 respectively for the high and the low signal-to-noise ratio cases. In all cases, as observed from Figures 6 and 7, recirculating input data results in significant performance improvement.

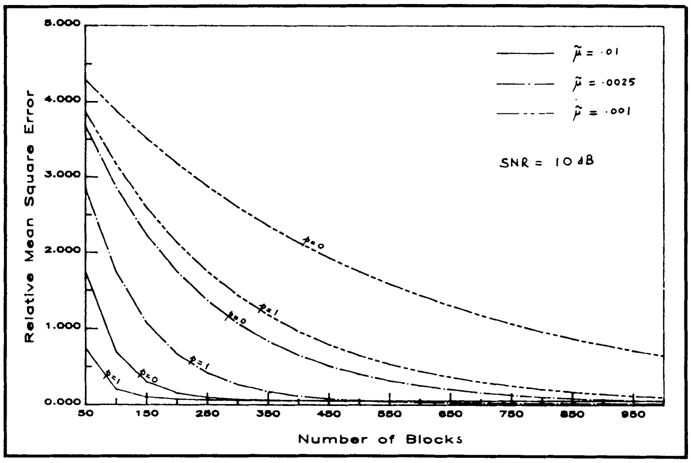

Figure 6: *Comparison of performances for the adaptive filters in example 2.*

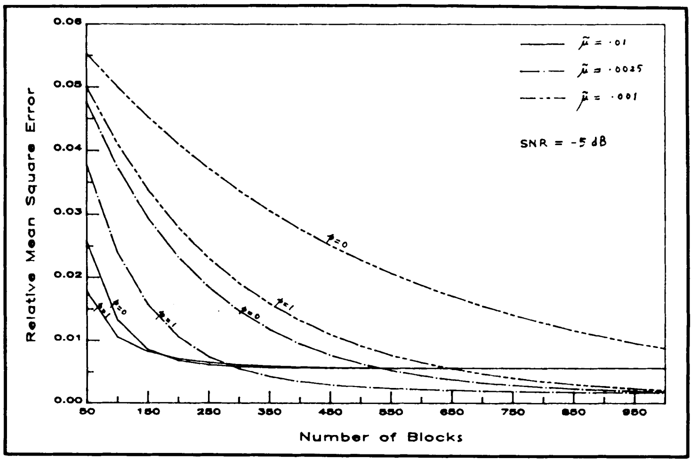

Figure 7: *Comparison of performances for the adaptive filters in example 2.*

Conclusion

In this paper we have introduced an algorithm of recirculating input data to a frequency-domain adaptive filter. In employing this new algorithm to a situation often encountered in noise cancellation and time-delay estimation, we have developed an analysis which expresses the mean-square error of the filter as a function of p, the number of data recirculation, and M, the number of data blocks available, for given values of signal power, noise power, and step-size of tap adjustment. This analysis is confirmed by computer simulation. In all test cases, there is significant improvement of the adaptive filter performance in terms of mean-square error by recirculating the input data.

In the cases where the number of data blocks is relatively small, recirculating the input data once almost invariably reduces the relative mean-square error to half of that yielded by the filter without data recirculation. Further recirculation of the data (i.e., $p \geqq 2$), however, results in less dramatic improvements. Figure 8 shows the relative mean-square errors for different number, p, of data recirculation. Here, we have chosen that

$$(p + 1)\tilde{\mu} = 1.24 \times 10^{-2}$$

The reason for this has been stated in Observation 3 for Figure 3 where the optimum normalized step-size obeys Equation (36). Once this relationship is satisfied, we are ensured that an optimum normalized step-size is chosen. Furthermore, in the development of Equations (A-9), (A-20), (A-21), and (A-22), we have assumed that the normalized step-size is very much less than unity. Equation (36) again ensures this condition to be satisfied.

The asymptotic value reached by having large value of p in Figure 8 is not necessarily the minimum mean-square error. The minimum mean-square error is achieved only if M is sufficiently large. If the number of data blocks, M, is not large, then the amount of information that can be extracted is limited. To exhaust the amount of available information, we recirculate the samples several times so that the asymptotic value of mean-square error in Figure 8 is reached. In practical applications of adaptive filtering, the supply of input data is often limited. It is in such cases that the algorithm of recirculation of input data becomes attractive.

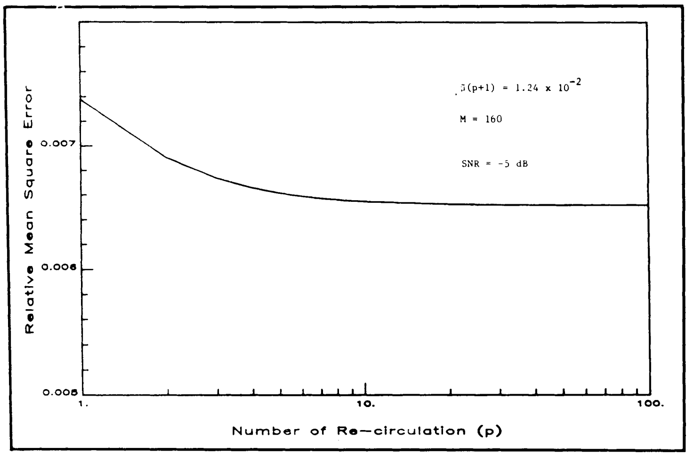

Figure 8: *Comparison of performance for the adaptive filter with different number of recirculation.*

Appendix 1

Here we present the evaluation of the values of $\Theta_k(j, n)$, $\Phi_i(j, n)$ and $\Psi_k(l, n)$. Using Equation (20a), we have,

$$\overline{\Theta_k(j, n)} = \overline{\{\mu D_k(j)X_k^*(j)[1 - \mu|X_k(j)|^2]^n\}}$$

$$= \overline{\left\{\mu D_k(j)X_k^*(j)\left[\sum_{l=0}^{n}\binom{n}{l}(-1)^l\mu^l|X_k(j)|_l^2\right]\right\}}$$

$$= \sum_{l=0}^{n}\binom{n}{l}(-1)^l\mu^{l+1}\overline{\{D_k(j)X_k^*(j)[X_k(j)X_k^*(j)]^l\}} \quad \text{(A.1)}$$

To find the average value of the terms inside the braces in Equation (A.1) we use the result of the moment theorem for a complex, zero-mean, circular Gaussian process[8] which states that:

If z_n are samples from a zero-mean, complex circular Gaussian process, then

$$\overline{[z_{m1}z_{m2}\ldots z_{ml}z_{n1}^*z_{n2}^*\cdots z_{nl}^*]}$$

$$= \sum_{\pi}\overline{(z_{m\pi(1)}z_{n1}^*)}\,\overline{(z_{m\pi(2)}z_{n2}^*)}\ldots\overline{(z_{m\pi(l)}z_{nl}^*)} \quad \text{(A.2)}$$

where π is a permutation of the set of integers $\{1, 2, \ldots, l\}$. If we let

$$z_{m1} = D_k(j),$$

$$z_{m2} = z_{m3} = \ldots = z_{m,l+1} = X_k(j)$$

$$z_{n1}^* = z_{n2}^* = \ldots = z_{n,l+1}^* = X_k^*(j)$$

then, using Equation (A.2) and noting that there are $(l + 1)!$ permutations among the $l + 1$ random variables, we have

$$\overline{\{D_k(j)X_k^*(j)[X_k(j)X_k^*(j)]^l\}}$$

$$= (l + 1)!\overline{\{D_k(j)X_k^*(j)\}}\,\overline{\{X_k(j)X_k^*(j)\}}^l \quad \text{(A.3)}$$

But from Equations (10) and (11),

$$\overline{\{D_k(j)X_k^*(j)\}} = \alpha_k\sigma_{Sk}^2$$

$$\overline{\{X_k(j)X_k^*(j)\}} = \sigma_{Sk}^2 + \sigma_N^2 \quad \text{(A.4)}$$

Therefore, substituting Equations (A.3) and (A.4) in Equation (A.1), we obtain

$$\overline{\Theta_k=(j, n)} = \sum_{l=0}^{n}\binom{n}{l}(-1)^l\mu^{l+1}(l + 1)!\alpha_k\sigma_{Sk}^2(\sigma_{Sk}^2 + \sigma_N^2)^l$$

$$\simeq \mu\alpha_k\sigma_{Sk}^2[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)] \quad \text{(A.5)}$$

where we have made the assumption that $\mu(\sigma_{Sk}{}^2 + \sigma_N{}^2) \ll 1$. Similarly, we can write,

$$\overline{\Phi_k(i, n)} = \overline{\{1 - \mu|X_k(1)|^2\}^n}$$

$$= \sum_{l=0}^{n}\binom{n}{l}(-1)^l\mu^l\overline{|X_k(i)|^{2l}}$$

$$= \sum_{l=0}^{n}\binom{n}{l}(-\mu)^l l!(\sigma_{Sk}^2 + \sigma_N^2)^l \simeq 1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2) \quad \text{(A.6)}$$

and

$$\Psi_k(l, n) = \overline{\{1 - \mu|X_k(l)|^2\}^{n+1}}$$

$$\simeq 1 - (n + 1)\mu(\sigma_{Sk}^2 + \sigma_N^2) \quad \text{(A.7)}$$

To evaluate the expected value of the tap-weight at the end of the recirculation of input data, we substituted Equations (A.5-A.7) into Equation (21), so that

$$\overline{W_k(p, M)} = \sum_{n=0}^{p}\sum_{j=0}^{M-1}\Bigg(\mu\alpha_k\sigma_{Sk}^2[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)]$$

$$\prod_{i=0}^{j-1}\{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)\}\prod_{l=j+1}^{M-1}\{1 - (n + 1)\mu(\sigma_{Sk}^2 + \sigma_N^2)\}\Bigg)$$

$$= \sum_{n=0}^{p}\mu\alpha_k\sigma_{Sk}^2[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)]$$

$$\sum_{j=0}^{M-1} \{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)\}^j \{1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)\}^{M-1-j}$$

$$= \sum_{n=0}^{p} \mu\alpha_k\sigma_{Sk}^2[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)]\{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)\}^{M-1}$$

$$\sum_{j=0}^{M-1} \left[\frac{1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)}{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)}\right]^{M-1-j}$$

$$= \sum_{n=0}^{p} \mu\alpha_k\sigma_{Sk}^2[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)][1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)]^{M-1}$$

$$\frac{1 - \left[\dfrac{1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)}{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)}\right]^M}{1 - \left[\dfrac{1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)}{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)}\right]}$$

i.e.,

$$\overline{W_k(p, M)} = \sum_{n=0}^{p} \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)}[1 - 2n\mu(\sigma_{Sk}^2 + \sigma_N^2)]$$

$$\{[1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)]^M$$

$$- [1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)]^M\} \quad (A.8)$$

Now, $\mu(\sigma_{Sk}{}^2 + \sigma_N{}^2) \ll 1$

Therefore,

$$\overline{W_k(p, M)} \simeq \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} \sum_{n=0}^{p} [1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)]^2$$

$$\{[1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)]^M - [1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)]^M\}$$

$$\simeq \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} \sum_{n=0}^{p} \{[1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)]^{M+2}$$

$$- [1 - (n+1)\mu(\sigma_{Sk}^2 + \sigma_N^2)]^{M+2}\}$$

$$= \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)}\{1 - [1 - (p+1)\tilde{\mu}_k]^{M+2}\} \quad (A.9)$$

Note: if $p = 0$, then $\overline{W_k(p, M)}$ can be evaluated direct from Equation (A.8) and the last step of approximation in Equation (A.9) can be bypassed yielding

$$\overline{W_k(0, M)} \simeq \frac{\alpha_k\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)}[1 - (1 - \tilde{\mu}_k)^M] \quad (A.10)$$

Appendix 2

Here we present the evaluation of the mean-square value of the tap weight. From Equation (18), we have

$$\overline{|W_k(p, M)|^2} = \overline{\left|\sum_{n=0}^{p} B_k(M-1)A_k^n(M-1)\right|^2}$$

$$= \overline{|B_k(M-1)|^2} \; \overline{\left|\sum_{n=0}^{p} A_k^n(M-1)\right|^2} \quad (A.11)$$

Now, from Equation (17b)

$$|B_k(M-1)|^2 = \mu^2 \sum_{i=0}^{M-1}\sum_{j=0}^{M-1} \Bigg[D_k(i)D_k^*(j)X_k^*(i)X_k(j)$$

$$\prod_{l=i+1}^{M-1}\prod_{m=j+1}^{M-1} \{1 - \mu|X_k(l)|^2\}\{1 - \mu|X_k(m)|^2\}\Bigg]$$

$$= \Gamma_{1k}(M-1) + \Gamma_{2k}(M-1) + \Gamma_{3k}(M-1) \quad (A.12)$$

where

$$\Gamma_{1k}(M-1)$$

$$= \mu^2 \sum_{j=0}^{M-1} \left[|D_k(j)|^2|X_k(j)|^2 \prod_{l=j+1}^{M-1} \{1 - \mu|X_k(l)|^2\}^2\right] \quad (A.13)$$

$$\Gamma_{2k}(M-1)$$

$$= \mu^2 \sum_{i=0}^{M-1}\sum_{j=0}^{i-1} \Bigg[D_k(i)D_k^*(j)X_k^*(i)X_k(j) \prod_{l=i+1}^{M-1}\prod_{m=j+1}^{M-1}$$

$$\{1 - \mu|X_k(l)|^2\}\{1 - \mu|X_k(m)|^2\}\Bigg] \quad (A.14)$$

$$\Gamma_{3k}(M-1)$$

$$= \mu^2 \sum_{i=0}^{M-1}\sum_{j=0}^{i-1} \Bigg[D_k^*(i)D_k(j)X_k(i)X_k^*(j) \prod_{l=i+1}^{M-1}\prod_{m=j+1}^{M-1}$$

$$\{1 - \mu|X_k(l)|^2\}\{1 - \mu|X_k(m)|^2\}\Bigg] \quad (A.15)$$

Therefore,

$$\overline{|W_k(p, M)|^2} = \overline{[\Gamma_{1k}(M-1) + \Gamma_{2k}(M-1) + \Gamma_{3k}(M-1)]}$$

$$\overline{\left|\sum_{n=0}^{p} A_k^n(M-1)\right|^2}$$

$$= \overline{C_{1k}}(p, M-1) + \overline{C_{2k}}(p, M-1)$$

$$+ \overline{C_{3k}}(p, M-1) \quad (A.16)$$

where

$$\overline{C_{1k}}(p, M-1) = \overline{\Gamma_{1k}(M-1)\left|\sum_{n=0}^{p} A_k^n(M-1)\right|^2}$$

$$\overline{C_{2k}}(p, M-1) = \overline{\Gamma_{2k}(M-1)\left|\sum_{n=0}^{p} A_k^n(M-1)\right|^2}$$

and

$$\overline{C_{3k}}(p, M-1) = \overline{\Gamma_{3k}(M-1)\left|\sum_{n=0}^{p} A_k^n(M-1)\right|^2} \quad (A.17)$$

Now,

$$\overline{C_{1k}}(p, M-1)$$

$$= \overline{\left[\Gamma_{1k}(M-1) \sum_{n=0}^{p} \sum_{v=0}^{p} A_k^n(M-1) A_{kv}^*(M-1)\right]}$$

$$= \mu^2 \overline{\left[\sum_{j=0}^{M-1} |D_k(j)|^2 |X_k(j)|^2 \prod_{l=j+1}^{M-1} \{1 - \mu|X_k(l)|^2\}^2\right]}$$

$$\overline{\left[\sum_{n=0}^{p} \sum_{v=0}^{p} \prod_{i=0}^{M-1} \{1 - \mu|X_k(i)|^2\}^n \prod_{\lambda=0}^{M-1} \{1 - \mu|X_k(\lambda)|^2\}\right]^v} \quad (A.18)$$

Using the same technique as in Equation (19), we can separate Equation (A.15) into correlated and uncorrelated parts as that

$$\overline{C_{1k}}(p, M-1)$$

$$= \mu^2 \sum_{n=0}^{p} \sum_{v=0}^{p} \overline{\sum_{j=0}^{M-1} |D_k(j) X_k^*(j)|^2 [1 - \mu|X_k(j)|^2]^{n+v}}$$

$$\overline{\prod_{i=0}^{i-1} [1 - \mu|X_k(j)|^2]^{n+v}} \; \overline{\prod_{\lambda=j+1}^{M-1} [1 - \mu|X_k(\lambda)|^2]^{n+v+2}} \quad (A.19)$$

Using the moment theorem and the condition that $(n + v)\mu (\sigma^2{}_{Sk} + \sigma^2{}_N) \ll 1$, we have, after simplifying,

$$\overline{C_{1k}}(p, M-1) \simeq \frac{\mu}{2}\left\{|\alpha_k|^2 \sigma_{Sk}^2 + \sigma_N^2 + \frac{|\alpha_k|^2 \sigma_{Sk}^4}{(\sigma_{Sk}^2 + \sigma_N^2)}\right\}$$

$$\left[\{1 - (1 - 2(p+1)\mu(\sigma_{Sk}^2 + \sigma_N^2))^{M+2}\}\right.$$

$$+ 2 \sum_{n=1}^{p} \{1 - n\mu(\sigma_{Sk}^2 + \sigma_N^2)\}^{M+2}$$

$$\left. - \{1 - (p + n + 1)\mu(\sigma_{Sk}^2 + \sigma_N^2)\}^{M+2}\right] \quad (A.20)$$

Writing $\tilde{\mu}_k = \mu(\sigma_{Sk}{}^2 + \sigma_N{}^2)$, and simplifying, we have

$$\overline{C_{1k}}(p, M-1)$$

$$\simeq \frac{\tilde{\mu}_k}{2}\left\{1 + (|\alpha_k|^2 - 1)\frac{\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} + \frac{|\alpha_k|^2 \sigma_{Sk}^4}{(\sigma_{Sk}^2 + \sigma_N^2)^2}\right\}$$

$$[\{1 - (1 - 2(p+1)\tilde{\mu}_k)^{M+2}\}$$

$$+ 2 \sum_{n=1}^{p} \{1 - n\tilde{\mu}_k\}^{M+2}$$

$$\left. - \{1 - (p + n + 1)\tilde{\mu}_k\}^{M+2}\right] \quad (A.21)$$

By using a similar approach, we obtain

$$\overline{C_{2k}}(p, M-1)$$

$$\simeq \frac{|\alpha_k|^2 \sigma_{Sk}^4}{2(\sigma_{Sk}^2 + \sigma_N^2)^2}[1 - 2\{1 - (p+1)\tilde{\mu}_k\}^{M+4}$$

$$+ \{1 - 2(p+1)\tilde{\mu}_k\}^{M+4}] \quad (A.22)$$

Since, Equations (A.12) and (A.13) are symmetric, then

$$\overline{C_{3k}}(p, M-1) = \overline{C_{2k}}(p, M-1) \quad (A.23)$$

If $p = 0$, then $\overline{|W_k(0, M)|^2}$ can be evaluated direct from Equation (A.16) without going through the last step of approximation. This gives

$$\overline{|W_k(0, M)|^2} = \overline{[\Gamma_{1k}(M-1) + \Gamma_{2k}(M-1) + \Gamma_{3k}(M-1)]}$$

$$= \frac{\tilde{\mu}_k}{2}\left\{1 + (|\alpha_k|^2 - 1)\frac{\sigma_{Sk}^2}{(\sigma_{Sk}^2 + \sigma_N^2)} + \frac{|\alpha_k|^2 \sigma_{Sk}^4}{(\sigma_{Sk}^2 + \sigma_N^2)^2}\right\}$$

$$(1 - \tilde{\mu}_k)^{-1}\{1 - (1 - 2\tilde{\mu}_k + 2\tilde{\mu}_k^2)^M\}$$

$$+ \frac{|\alpha_k|^2 \sigma_{Sk}^4}{(\sigma_{Sk}^2 + \sigma_N^2)^2}(1 - \tilde{\mu}_k)^{-1}$$

$$\{(1 - 2\tilde{\mu}_k) - (1 - \tilde{\mu}_k)^{M+1} + (1 - 2\tilde{\mu}_k + 2\tilde{\mu}_k^2)^M\} \quad (A.24)$$

References

1. Dentino, M., McCool, J. and Widrow, B., "Adaptive filtering in the frequency-domain," *Proc. IEEE*, Vol. 66, pp. 1658-1659, Dec., 1978.
2. Bershad, N.J. and Feintuch, P.L., "Analysis of the frequency-domain adaptive filter," *Proc. IEEE*, Vol. 67, pp. 1658-1659, Dec., 1979.
3. Reed, F.A. and Feintuch, P.L., "A comparison of LMS adaptive cancellers implemented in the frequency-domain and time domain," *IEEE Trans.*, Vol. ASSP-29, pp. 770-775, June, 1981.
4. Clark, G.A., Mitra, S.K. and Parker, S.R., "Block implementation of adaptive digital filters," *IEEE Trans.*, Vol. ASSP-29, pp. 744-752, June, 1981.
5. Morgul, A., Grant, P.M. and Cowan, C.N., "Wide-band hybrid analog/digital frequency-domain adaptive filter," *IEEE Trans.*, Vol. ASSP-32, pp. 762-769, Aug., 1984.
6. Gersho, A., "Adaptive equalization of highly dispersive channels for data transmission," *Bell System Technical Journal*, Vol. 48, pp. 55-70, Jan., 1969.
7. Widrow, B., McCool, J. and Ball, M., "The complex LMS algorithm," *Proc. IEEE*, Vol. 63, pp. 719-720, April, 1975.
8. Reed, I.S., "On a moment theorem for complex Gaussian processes," *IEEE Trans. on Information Theory*, pp. 194-195, April, 1962.
9. Widrow, B., McCool, J. and Ball, M., "The complex LMS algorithm," *Proc. IEEE*, Vol. 63, pp. 719-720, April, 1975.
10. Ljung, L., "Recursive identification methods for off-line identification problems," 6th IFAC Symposium on Identification and System Parameter Estimation, 1982.
11. Ljung, L. and Soderstorm, T., *Theory and practice of recursive identification*, MIT Press, Cambridge, 1983.
12. Young, P.C., "Some observations on instrumental variable methods of time series analysis," *International Journal of Control*, Vol. 23, 1976, pp. 593-612.
13. Mansour, D. and Gray, A.H., "Unconstrainted frequency-domain adaptive filter," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-30, No. 5, Oct. 1982.
14. Ogue, J.C., Saito, T. and Hashiko, Y., "A fast convergence frequency-domain adaptive filter," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-31, No. 5, Oct. 1983.
15. Narayan, S.S., Peterson, A.M. and Narasimha, M.J., "Transform domain LMS algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-31, No. 3, June, 1983.
16. Ferrara, E.R., "Frequency domain adaptive filtering," Chapter 6, *Adaptive Filter*, Edited by C.F.N. Cowan and P.M. Grant, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1985.