Time-And Frequency-Domain Block LMS Adaptive Digital Filters: Part II - Performance Analysis

시간영역 및 주파수영역 분력적응 여파기에 관한 연구: 제2부 - 성능분석

Jae Chon Lee* Chong Kwan Un*

ABSTRACT

In Part I of the paper, we have developed various block least mean-square (BLMS) adaptive digital filters (ADF's) based on a unified matrix treatment. In Part II we analyze the convergence behaviors of the self-orthogonalizing frequency-domain BLMS (FBLMS) ADF and the unconstrained FBLMS (UFBLMS) ADF both for the overlap-save and overlap-add sectioning methods. We first show that, unlike the FBLMS ADF with a constant convergence factor, the convergence behavior of the self-orthogonalizing FBLMS ADF is governed by the same autocorrelation matrix as that of the UFBLMS ADF. We then show that the optimum solution of the UFBLMS ADF is the same as that of the constrained FBLMS ADF when the filter length is sufficiently long. The mean of the weight vector of the UFBLMS ADF is also shown to converge to the optimum Wiener weight vector under a proper condition. However, the steady-state mean-squared error (MSE) of the UFBLMS ADF turns out to be slightly worse than that of the constrained algorithm if the same convergence constant is used in both cases. On the other hand, when the filter length is not sufficiently long, while the constrained FBLMS ADF yields poor performance, the performance of the UFBLMS ADF can be improved to some extent by utilizing its extended filter-length capability. As for the self-orthogonalizing FBLMS ADF, we study how we can approximate the autocorrelation matrix by a diagonal matrix in the frequency domain. We also analyze the steady-state MSE's of the self-orthogonalizing FBLMS ADF's with and without the constraint. Finally, we present various simulation results to verify our analytical results.
I. INTRODUCTION

In Part I of the paper [1], we have developed various block least mean-square (BLMS) adaptive digital filters (ADF’s) realized using the fast Fourier transform (FFT) and the overlap-save or overlap-add sectioning method. Among those BLMS ADF’s, the self-orthogonalizing frequency-domain BLMS (FBLMS) ADF and the unconstrained FBLMS (UFBLMS) ADF have some attractive features. For example, the former has fast convergence speed, and the latter has reduced computational complexity.

The convergence properties of the BLMS ADF were studied by Clark, Mitra and Parker [2,3]. They obtained the optimum weight vector, the condition for convergence, the time constant (or convergence speed), and the steady-state mean-squared error (MSE) [2]. These results were also compared with those of the least mean-square (LMS) ADF [4], thereby making it possible to replace easily the existing LMS ADF by the BLMS ADF which can be implemented efficiently.

The UFBLMS ADF using the overlap-save sectioning method was introduced by Mansour and Gray [5]. Based on the almost sure asymptotic exponential stability of control theory, they proved the convergence of the UFBLMS algorithm in the context of system identification. However, no analytical results were presented on the steady-state MSE. Also, by computer simulation, the self-orthogonalizing UFBLMS algorithm was shown to have fast convergence speed. In another paper [6], the convergence behaviors of the two FBLMS ADF’s with and without the constraint were compared by computer simulation when the number of the filter weights is not sufficiently large.

The self-orthogonalizing FBLMS ADF based on the overlap-save sectioning method was studied by Picchi and Prati [7]. They derived the weight adjustment algorithm by minimizing the frequency-domain block MSE (BMSE) with the constraint on the frequency-domain weight vector. In order to realize the constraint, they applied the Rosen’s gradient projection method [8]. However, the convergence behavior of the developed algorithm was not fully analyzed.

In Part II of the paper, we analyze the convergence behaviors of the UFBLMS ADF and the self-orthogonalizing FBLMS ADF both for the overlap-save and overlap-add sectioning methods. In doing so, we inverse-

---

1 We believe that the inclusion of the overlap-add case will enhance the overall understanding of the convergence behaviors of the FBLMS ADF’s. However, for a coherent presentation of the results on the convergence analyses, the overlap-add case will be discussed separately in Appendix.
transform the weight vector from the frequency domain into the time domain and then follow the analysis procedure used in the LMS or BLMS case [2], [4], [9]. More specifically, the approach taken in our paper is as follows. We first compute the optimum Wiener solution that minimizes an appropriate performance criterion. We then derive the difference equation for the mean of the weight vector in order to prove the convergence of the underlying algorithm. From this equation, we can obtain the convergence condition and the time constants and also calculate the steady-state MSE.

Prior to detailed analysis, we investigate the differences in the mean weight-vector equations of the BLMS, self-orthogonalizing FBLMS and UFBLMS ADF's. We then study the convergence characteristics of the UFBLMS ADF's, and extend our analysis to the self-orthogonalizing algorithms. According to the results of our analysis, the optimum solutions of the two FBLMS ADF's with and without the constraint turn out to be the same when the filter length is sufficiently long. The mean of the weight vector of the UFBLMS ADF is also shown to converge to the optimum Wiener weight vector under a proper condition. On the other hand, when the filter length is not sufficiently long, the original and unconstrained FBLMS ADF's are shown to reveal quite different convergence behaviors in the steady state. As for the self-orthogonalizing FBLMS ADF, we obtain the difference equation for the mean of the weight-error vector and discuss how we can approximate the autocorrelation matrix by a diagonal matrix in the frequency domain. One of the results indicates that the self-orthogonalizing FBLMS ADF can have superior convergence speed over the self-orthogonalizing frequency-domain LMS (FLMS) ADF which operates on a sample-by-sample basis. We also obtain the analytical results on the steady-state MSE's of the self-orthogonalizing FBLMS ADF's with and without the constraint.

Following this Introduction, in Section II we briefly discuss the convergence properties of the BLMS, UFBLMS and self-orthogonalizing FBLMS ADF's. In Section III, we analyze in detail the convergence behaviors of the UFBLMS ADF. In Section IV, we study how we can realize the self-orthogonalizing algorithm in the frequency domain. In Section V, we present various simulation results to verify our analytical results. Finally, we draw conclusions in Section VI. In addition, the results of the convergence analysis of the overlap-add realization are given in Appendix.

II. CHARACTERISTICS OF BLOCK LMS ADF's

In this section, we briefly describe the properties of the BLMS, UFBLMS and self-orthogonalizing FBLMS ADF's. In our discussion, all input data for these ADF's are assumed to be stationary and real-valued. As for the BLMS ADF, a detailed convergence analysis can be found in [2]. In Sections III and IV, we compare the analytical results of the UFBLMS and self-orthogonalizing FBLMS ADF's with those of the BLMS ADF, which is reviewed in this section.

A. Optimum Block Wiener Solution and BLMS Algorithm

Assume that an FIR ADF has M weights \( \hat{w}_m \) and that the filter produces its output \( y_n \) from the input \( x_n \) and the desired response \( d_n \). For our discussion here, we use the following basic equations which we used in Part I:

---

1 The reason is that, as will be seen in Sections II-IV, we can get the physical meaning of the autocorrelation matrix more easily from the time domain rather than the frequency domain viewpoint.

2 All the notations used in Part II will be the same as those in Part I of the paper unless otherwise stated.
Block MSE

\[ s_{e,k} \triangleq \mathbb{E}\{ e_k e_k^T \} \]

Output vector:

\[ y_k = X_k w_k \]

Error vector:

\[ e_k = d_k - y_k \]

Weight vector:

\[ w_{k+1} = w_k + \mu X_k e_k \]

where

\[ \mu \text{ is a convergence factor.} \]

\[ X_k = \begin{bmatrix} x_{k-1} & x_{k-2} & \cdots & x_1 \end{bmatrix} \]

\[ d_k = \begin{bmatrix} d_{k-1} & d_{k-2} & \cdots & d_1 \end{bmatrix} \]

The optimum block Wiener solution that minimizes the BMSE defined in (1) can easily be obtained as the following:

\[ w_{opt} = \frac{R_k}{\lambda_{max}} s_{p_k} \]

where

\[ R_k \triangleq \mathbb{E}\{ X_k X_k^T \} \quad p_k \triangleq \mathbb{E}\{ d_k d_k^T \} \]

\[ R_k \triangleq \mathbb{E}\{ x_k x_k^T \} \quad p_k \triangleq \mathbb{E}\{ d_k d_k^T \} \]

and

\[ X_k = \begin{bmatrix} x_{k-1} & x_{k-2} & \cdots & x_1 \end{bmatrix} \]

\[ d_k = \begin{bmatrix} d_{k-1} & d_{k-2} & \cdots & d_1 \end{bmatrix} \]

Therefore, in the stationary case the block solutions are the same as the conventional Wiener solutions. That is,

\[ w_{opt} = \frac{R_k}{\lambda_{max}} s_{p_k} \text{ and } \epsilon_{min} \text{ is the minimum MSE.} \]

With the usual assumption that the signal matrix and the weight vector are uncorrelated, we can easily obtain the mean of the weight-error vector for the BLMS ADF given in (2) as the following:

\[ E\{ w_n w_n^T \} = \frac{1}{M} L R_k \cdot s_{p_k} \]

where \( v_k = w_k \cdot s_{wopt} \). Based on (6), we can obtain the convergence condition and the time constants of the MSE process, and also calculate the steady-state excess MSE as follows [2].

- Convergence condition:

\[ 0 < \mu < \frac{2}{\lambda_{max}} \quad \text{or} \quad 0 < \mu \leq \frac{2}{\text{tr}\{ R_k \}} \]

where \( \lambda_{max} \) is the largest value among the eigenvalues, \( \lambda_i \quad i = 1 \), of \( R_k \).

- Time constants:

\[ \tau = \frac{1}{2 \mu} \quad \text{(in blocks)} \]

\[ \tau = \frac{1}{2 \mu} \quad \text{(in samples)} \]

Excess MSE:

\[ \epsilon_{excess} = \frac{1}{2} \mu \text{tr}\{ R_k / L \} \epsilon_{min} \]

\[ \epsilon_{excess} = \frac{1}{2} \mu \text{tr}\{ R_k / L \} \epsilon_{min} \]

\[ \epsilon_{excess} = \frac{1}{2} \mu \text{tr}\{ R_k / L \} \epsilon_{min} \]
were $\text{tr}(R_k)$ denotes the sum of the diagonal elements of $R_k$ and $\alpha_k^2 \triangleq E[x_k^2]$

It is noted here that, since both the TBLMS and FBLMS ADF's using a constant convergence factor $\mu$ are exact implementations of the BLMS ADF, they have the same convergence characteristics as those of the BLMS ADF.

### B. Unconstrained FBLMS Algorithm

In Part I of the paper, we have discussed two FBLMS algorithms based on the overlap-save sectioning which are given as

$$s\omega_{k+1} = s\omega_k + \mu P_{n,0} \overline{X}_k \cdot x_k$$  \hspace{0.5cm} (10a)

and

$$s\omega_{k+1} = P_{n,0} (s\omega_k + \mu \overline{X}_k \cdot x_k)$$  \hspace{0.5cm} (10b)

It should be noted that the FBLMS algorithm in (10a) is an exact implementation of the BLMS algorithm in (2c), whereas the FBLMS algorithm in (10b) is not. Especially, they have different convergence behaviors in the sense that, unlike the latter algorithm, the former algorithm converges to the optimum solution only for a special initial condition (i.e., $P_{n,0} s\omega_0 = s\omega_0$). As will be seen in Sections III-V, when the filter length is sufficiently long, the UFBLMS algorithm has the same convergence characteristics as those of the FBLMS algorithm in (10b). For this reason, hereafter we represent the UFBLMS and FBLMS algorithms using new frequency-domain vectors $s\omega_k$ and $s\omega_k$ as follows.

**UFBLMS:**

$$s\omega_{k+1} = s\omega_k + \mu \overline{X}_k \cdot x_k$$  \hspace{0.5cm} (11)

**FBLMS** :

$$s\omega_{k+1} = P_{n,0} (s\omega_k + \mu \overline{X}_k \cdot x_k)$$  \hspace{0.5cm} (12)

In (11), the frequency-domain error vector of the UFBLMS ADF is given from (33) of Part I as

$$u_e_k \leftarrow d_k - P_{n,0} X_k \cdot w_k$$  \hspace{0.5cm} (13)

Let us discuss the convergence behavior of the UFBLMS algorithm of (11) in terms of the mean weight vector. Substituting (13) into (11) and taking expectation of both sides of (11) lead to

$$E[uw_{k+1} - uw_k] + \mu E[s\overline{X}_k \cdot d_k]$$

$$\cdots E[s\overline{X}_k P_{n,0} sX_k] = E[uw_k].$$  \hspace{0.5cm} (14)

In order to see the difference in the convergence behaviors between the BLMS and UFBLMS ADF's, we inverse-transform both sides of (14) and obtain

$$E[uw_{k+1} - uw_k] + \mu E[sX_k \cdot d_k] = E[uw_k].$$

where $u_w_k = P^{-1} u_b_k$. In (15), the LxN matrix $sX_k$ is a part of the NxN circulant matrix $X_k$ and they are defined from (5) of Part I as

$$sX_k = \begin{bmatrix}
X_k & X_{k-1} & \cdots & X_1 \\
X_N & X_k & \cdots & X_{N-1} \\
\vdots & \vdots & \ddots & \vdots \\
X_1 & X_N & \cdots & X_{N-1} \\
\end{bmatrix} \cdot \text{circulant}$$

It should be noted here that the size of the time-domain weight vector $u_w_k$ in (15) is N×1, while the size of $w_k$ in (2c) is M×1. As can be seen from (6) and (15), unlike the BLMS ADF, the time-domain autocorrelation matrix of the UFBLMS algorithm is given by $E[s\overline{X}_k \cdot X_k]$ whose size is N×N. The characteristics of this new autocorrelation matrix will be studied in detail in Sections III and IV.

### C. Self-orthogonalizing FBLMS Algorithm

Following the same point of view presented in the previous subsection, two self-orthogonalizing FBLMS algorithms using the overlap-save sectioning method can be considered as the following:

$$u_e_k \leftarrow d_k - P_{n,0} s\overline{X}_k \cdot x_k$$  \hspace{0.5cm} (15a)
and

\[ s_{\omega_{k+1}} = P_{u,v} \left( s_{\omega_k} + \gamma k \right) R_k^{-1} s_{X_k} + \xi_k, \]  

(17b)

where \( s_{R_k} \) is an \( NxN \) diagonal matrix which can be estimated using some appropriate method (for example, see (46) of Part I) in actual realization of (17). Here, we discuss the characteristics of the self-orthogonalizing version of the constrained algorithm in (17b). Inserting (13) into (17b), we can obtain the mean of the frequency-domain weight vector as

\[ E[s_{\omega_{k+1}}] = P_{u,v} E[s_{\omega_k}] - \gamma P_{u,v} s_{R_k} \]

\[ + E[s_{X_k} s_{d_k}] E[s_{X_k} P_{u,v} s_{X_k}] + P_{u,v} E[s_{\omega_k}]. \]  

(1b)

Noting that the last \( (N-M) \) elements of the inverse transform of \( s_{\omega_k} \) are zero, we can get from (18)

\[ E[w_{k+1}] = E[w_k] - \gamma P_{u,v} (F^{-1} s_{R_k}^{-1} F) \]

\[ + E[s_{X_k} d_k] E[s_{X_k} X_k] E[w_k]. \]  

(1f)

Comparing (15) and (19), one can see that the time-domain autocorrelation matrix of the self-orthogonalizing algorithm is not the same in its present form as that of the UFBLMS algorithm. However, we can show from (16) that, since \( s_{X_0} = [X_k : X_{k+1}] \) we have for the last term in (19)

\[ E[s_{X_k} X_k] E[w_k] = E[s_{X_k} X_k] E[w_k] = \left[ \frac{E[w_k]}{\sum_{n=0}^{N-1} \sum_{i=0}^{M-1} s_{n+i}}, \right. \]

(20)

Thus, the convergence behavior of the self-orthogonalizing FBLMS ADF is governed by the same \( N \times N \) matrix \( E[s_{X_k} X_k] \) as that of the UFBLMS ADF. For this reason, in next sections we analyze the convergence behavior of the UFBLMS ADF first and then extend our analysis to the self-orthogonalizing FBLMS ADF. It is noted that, when the matrix \( s_{R_k} \) is an identity matrix, the difference equation of (19) becomes that of the BLMS ADF in (6). Note also from (6) and (19) that the use of the diagonal matrix in the frequency domain introduces another input matrix \( \chi_{k} \) which does not appear in the weight vector equation of the BLMS ADF. The effect of \( \chi_{k} \) on the convergence behavior will be studied in Section IV. The convergence behavior of the self-orthogonalizing version for the UFBLMS ADF will also be analyzed in that section.

### III. CONVERGENCE-ANALYSIS OF UNCENSTRADED FBLMS ADF's

As can be seen in (3), the optimum Wiener weights are completely determined in the time domain by the autocorrelation values of the input signal and the crosscorrelation values between the desired and input signals. Alternatively, they are determined in the frequency domain by the power spectra of the input and desired signals. It is known that, unless the bandwidth of the frequency spectrum is extremely narrow like a tone, the effective duration of the inverse transform of the spectrum which is of finite bandwidth in the frequency domain is also finite. Therefore, in most applications of the ADF in which the signal spectra are of finite bandwidth, both the correlation values and optimum filter weights approach zero when they are sufficiently far away from the center of the time origin. In this case, the optimum Wiener filter can be approximated by a finite impulse response (FIR) filter. Based on this FIR approximation, we can represent the desired signal as

\[ \xi_0 = \sum_{i=0}^{N_0} w_{i} d_{i} + \xi_0 \]  

(21)

where \( M_0 \) is the number of the model filter weights, \( w_{i} \) and \( \xi_0 \) is a zero-mean white noise process that is uncorrelated with \( \xi_0 \). It is noted that the accuracy of the approximation can be arbitrarily improved by increasing \( M_0 \). Consequently, in realization of the system whose signals are modeled by (21), the number of the weights of the ADF, \( M \), must be greater than or equal to \( M_0 \) to achieve the best per-
formance. It can easily be shown that under the signal model of (21) with \( M = M_o \), the optimum Wiener weight vector becomes the model weight vector \( w_{opt} = w_d = [w_{d,1}, w_{d,2}, \ldots, w_{d,n-1}] \). In the following, we analyze the convergence behavior of the UFBLMS ADF first when \( M = M_o \) and then discuss how the convergence behavior will be changed when \( M < M_o \).

In the UFBLMS algorithm, there is no constraint on the weight vector. Therefore, unlike in the constrained algorithm, all the \( N \) elements of the time-domain weight vector are used in computing the output and adjusting the weights. To see this aspect more clearly, from (13) and (15) we represent the output and adjustment algorithm of the UFBLMS ADF in the time domain as

\[
y_k = s X_k u w_k
\]

and

\[
a w_k = a w_k + u X_k e_k.
\]

By splitting into

\[
a w_k \big| a w_k \big| u X_k e_k,
\]

we decompose (22) as the following:

\[
y_k = X_n w_k : X_n w_k,
\]

\[
a w_k = a w_k + u X_k e_k,
\]

\[
a w_k = a w_k + u X_k e_k.
\]

One can see from (24) that removing the constraint on the weights introduces additional terms, which are related with \( X_n \) in the output and adjustment algorithm. Since \( X_n \) is generated by circular extension of \( X_k \), the additional terms incurred above have been called the circular convolution effect. In the following, we investigate how the circular convolution terms affect the convergence behavior. We first compute the optimum solution and the mean of the weight vector of the UFBLMS algorithm. We then obtain various results on the convergence behavior.

A. UFBLMS Algorithm and Its Optimum Solution

The optimum weight vector of the UFBLMS ADF is derived by minimizing the unconstrained frequency-domain BMSE. Alternatively, we can obtain the same solution based on the block orthogonality principle [2]. According to this principle and from (11), the optimum weight vector, \( w_{opt} \), must satisfy

\[
E[ s X_k u e_k ] = 0.
\]

We get from (25) the equation for the optimum weight vector as

\[
E[ s X_k u d_k ] = E[ s X_k u w_{opt} ].
\]

Or

\[
E[ s X_k u d_k ] = E[ s X_k s X_k u w_{opt} ],
\]

Also, combining (26b) and (15), we obtain the expression for the mean of the weight-error vector as

\[
E[ s X_k u e_k ] = 1 \times w_{opt} E[ s X_k u ]
\]

where \( w_{opt} \) and \( E[ s X_k u ] \) depend on the properties of the matrix \( R_k \); the UFBLMS ADF can have a unique optimum weight vector and also the weight vector of the filter can converge in the mean to the optimum solution independently of the initial value of the weight vector \( w_0 \) or \( \overline{w}_0 \).

In the next subsection, we investigate the properties of \( R_k \).
B. Properties of the Autocorrelation Matrix $\mathbf{R}_x$

Based on (16), we can decompose $\mathbf{R}_x$ as

$$
\mathbf{R}_x = \mathbf{E}[\mathbf{X}_n \mathbf{X}_n^H] = \mathbf{E}[\mathbf{X}_n^H \mathbf{X}_n]
$$

It is noted from (28) that $\mathbf{R}_x$ is an NxN symmetric matrix. To evaluate $\mathbf{R}_x$, we have to specify the elements of the $N \times (N-M)$ matrix which is the lower-right part of the NxN circulant matrix $\mathbf{X}_n$. For the ease of analysis, $\mathbf{X}_n$ is assumed to be zero for which case $N=L+M-1$. In this case, $\mathbf{X}_n$ is given as

$$
\begin{bmatrix}
\mathbf{X}_{11} & \mathbf{X}_{12} & \cdots & \mathbf{X}_{1N} \\
\mathbf{X}_{21} & \mathbf{X}_{22} & \cdots & \mathbf{X}_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{X}_{N1} & \mathbf{X}_{N2} & \cdots & \mathbf{X}_{NN}
\end{bmatrix}
$$

Defining the correlation of the stationary input process as $\rho_{ij} = \mathbf{E}[x_n x_{n+i-j}]$, from (2) and (29) we can compute all the elements of the submatrices $\mathbf{R}_x, \mathbf{R}_c$ and $\mathbf{R}_f$ as the following:

$$
\mathbf{R}_x = \mathbf{X}_n \mathbf{X}_n^H
$$

where $1 < i,j < M$.  \hfill (30a)

$$
\mathbf{R}_c = \mathbf{X}_n \mathbf{X}_n^H - \mathbf{I}
$$

where $1 < i,j < N-M$.  \hfill (30b)

$$
\mathbf{R}_f = \mathbf{X}_n \mathbf{X}_n^H - \mathbf{I}
$$

where $1 < i,j < N$ and $M < i,j < M+1$.  \hfill (30c)

It is noted from (30) that, unlike the matrix $\mathbf{R}_f$, both matrices $\mathbf{R}_x$ and $\mathbf{R}_c$ are symmetric Toeplitz. When $\rho_{ij}$ is very small for $i$ greater than a positive integer $m$, we can neglect those terms in the above matrices. When $M$ is sufficiently greater than $m$, $\mathbf{R}_x$ reduces to

$$
\mathbf{R}_x = \begin{bmatrix}
\rho_{00} & \rho_{01} & \cdots & \rho_{0m} \\
\rho_{10} & \rho_{11} & \cdots & \rho_{1m} \\
\vdots & \vdots & \ddots & \vdots \\
\rho_{m0} & \rho_{m1} & \cdots & \rho_{mm}
\end{bmatrix}
$$

when $M > m$. $N$ is also far greater than $m$. In that case, one of two correlation terms both in (30b) and (30c) can be neglected. Thus, we get from (30b) and (30c)

$$
\mathbf{R}_c = \mathbf{X}_n \mathbf{X}_n^H - \mathbf{I}
$$

where $1 < i,j < N$ and $M < i,j < M+1$.  \hfill (30d)

\hfill 4

\footnote{When $\mathbf{X}_n$ is not zero, though (30b) and (30c) must be slightly modified, the general structure will still be maintained. Also, in that case, for a better approximation of $\mathbf{R}_x$, it may be preferable to use $\mathbf{X}_n$ previous input data instead of zero-valued data (see the footnote 3 of Part I).}
\[ R_x = L \]

where

\[ a_l = 1 - \frac{l}{L}, \quad 0 \leq l \leq m. \]

When \( L \) is sufficiently larger than \( m \), all \( a_l \)'s can be approximated as unity. In that case the \( NxN \) matrix \( R_x \) becomes from (31)

\[ uR_x = F^{-1} u \Lambda_x F = (F/\sqrt{N})^{-1} u \Lambda_x (F/\sqrt{N}) \quad (33) \]

where \( F \) is the \( NxN \) DFT matrix and the \( NxN \) diagonal matrix \( u \Lambda_x \) is defined by \( u \Lambda_x \triangleq \text{diag}(u \lambda_1, u \lambda_2, \ldots, u \lambda_M) \). In (33), \( u \lambda_i \) are determined by the DFT of the first column of \( uR_x \) as the following:

\[ u \lambda_i = \sum_{l=0}^{m} \rho_l \exp[-j 2 \pi (i-1)/N]; \quad 0 \leq i \leq N-1 \]

\[ = \rho_0 + 2 \sum_{l=1}^{m} \rho_l \cos[2 \pi (i-1)/N], \quad 1 \leq i \leq N. \]

On the other hand, when \( m \ll M \), the \( MxM \) matrix \( R_x \) can be approximated by an \( MxM \) circulant matrix as was done in [9]. In that case, we get the eigenvalues \( \lambda_i \) of \( R_x / L \) as the following:

\[ \lambda_i = \sum_{l=0}^{m} \rho_l \exp[-j 2 \pi (i-1)/M]; \quad 0 \leq i \leq M-1 \]

\[ = \rho_0 + 2 \sum_{l=1}^{m} \rho_l \cos[2 \pi (i-1)/M], \quad 1 \leq i \leq M. \]

It is noted from (34) and (35) that \( |\lambda_i|^2 \) and \( |\lambda_1|^2 \) represent the M-point and N-point, respectively, power spectra of the same input random process \( x_n \). For example, when \( N=2M \), the even-numbered frequency components (or eigenvalues) of \( R_x \) are the same as those of \( R_x \), i.e., \( \lambda_{2i} = \lambda_i \), \( 1 \leq i \leq M \) [see (34) and (35)]. The odd-numbered ones are generated by interpolating the values \( \lambda_i \). Thus, if \( M>m \), the M-point power spectrum already has sufficient frequency resolution because the number of zeros appended is \( N-(2m+1) \). Hence, even in the N-point power spectrum which has increased frequency resolution, there will be no spurious peaks and valleys which did not appear in the M-point spectrum. Therefore, we can see that the eigenvalue distributions of the two matrices
\( uR_x \) and \( uRx^c \) are asymptotically the same except for the different number of eigenvectors. For example, \( u\lambda_{\max} = \lambda_{\max} \) and \( u\lambda_{\min} = \lambda_{\min} \).

C. Convergence Properties of UFBLMS Algorithm

We are now ready to solve the optimum solution and study the convergence behaviors of the UFBLMS ADF. According to the results obtained in the previous subsection, the matrix \( uR_x \) is nonsingular as \( uRx^c \) is nonsingular. Thus, we can solve (26b) for \( uw_{opt} \). However, our concern here is how the \( N\times1 \) weight vector \( uw_{opt} \) is related to the \( M\times1 \) weight vector \( v_{opt} \).

Based on the signal model in (21) and when \( M = M_0 \), we can show that

\[
d_k = X_k w_d + \xi_k
\]

where

\[
\xi_k = [\xi_{1k}, \xi_{2k}, \ldots, \xi_{nk-1}]
\]

Inserting (36) into (26b) yields

\[
uRx u_{w_{opt}} = E_{\{\Sigma X_k X_k^\top\}} w_d - E_{\{\Sigma X_k Y_k\}} v_{w_{opt}}
\]

Splitting the \( N\times1 \) vector \( v_{w_{opt}} \) into two subvectors \( v_{w_{opt}}^{\top} \) and \( w_{opt} \), we get from (37)

\[
uRx = \begin{bmatrix} a_{w_{opt}} & b_{w_{opt}} \\ a_{w_{opt}}^{\top} & b_{w_{opt}}^{\top} \end{bmatrix}
\]

Thus, we obtain from (38) the optimum solution of the UFBLMS ADF as

\[
uw_{opt} = a_{w_{opt}}^{-1} b_{w_{opt}}
\]

Also, one can see from (39) that the minimum MSE of the UFBLMS ADF, \( uw_{min} \), is the same as that of the BLMS ADF, \( v_{w_{min}} \). The optimum solution of (39) indicates that when \( M \geq M_0 \), no circular convolution effect occurs even without the constraining operation on the weights. This is another unique feature of the ADF compared to the fixed-coefficient digital filter. It has already been shown in [12] that, unlike in fixed-coefficient digital filtering, the overlap-save implementation requires less computation for the BLMS ADF than the overlap-add implementation. Adaptive realization of an ADF using appropriate desired signals enables the self-constraining operation.

We can get from (27) and Section III-B the convergence properties of the UFBLMS ADF as follows.

1. Convergence condition:

\[
0 < \mu < 2 \left[ \frac{1}{\lambda_{\max}} \right] \quad \text{or} \quad 0 < \mu < 2 \left[ \frac{1}{\lambda_{\min}} \right]
\]

2. Time constants:

\[
u_{\tau \min} = \frac{1}{2\mu \lambda_{\min}} \quad \text{in samples}, \quad 1 \leq i \leq N
\]

3. Excess MSE:

\[
u_{\tau \min} = \frac{1}{2} \mu \lambda_{\min} X_{\max}^2 \quad \text{as} \quad \frac{1}{2} \mu N \sigma^2 \tau_{\min}
\]

Comparing (9b) and (42), one can see that the UFBLMS ADF can save two FFT operations at the expense of the slightly increased steady-state MSE. When the same convergence constants are used in both cases. As will be seen in Section V, this increase is not significant. For example, according to (9b) and (42), the difference in the steady-state MSE's between the two cases is about 0.3 dB when \( \mu = 0.01 \), \( M = 16 \), \( N = 32 \), and \( \sigma^2 = 1 \).

D. Circular Convolution Effect

So far we have discussed the convergence behaviors of the UFBLMS ADF when \( M = M_0 \). Here, we consider the case when \( M < M_0 \). It is clear that, when \( M < M_0 \), the optimum solution of the BLMS ADF, \( v_{w_{opt}} \), is a truncated version of \( w_d \). Also, \( B_{min} \) increases because of the truncation of the optimum weight vector. In the following, we are
going to discuss how the UFBLMS ADF works in this situation. The output of the UFBLMS ADF is given from (22a) and (23) as

$$y_k - sX_u w_k = [X_k | X_c] \left[ \begin{array}{c} sW_k \\ v \end{array} \right]_{N-M}$$  \hspace{1cm} (29)

It is noted from (29) that, since $X_c$ is the circular extension of $X_k$, in each row of $X_c$ there is a jump in the time indices of the input data $x_n$. To see the effect of the assumption on $M < M_o$, we schematically draw the matrix $sX_u$ and the vector $w_k$ in Fig. 1. In Fig. 1 the jump is represented by the diagonal solid line in $X_c$. As can be seen in the previous subsections, the UFBLMS ADF in this situation will try to optimize its performance by utilizing the first $M_o > M$ elements of the vector, while the last $N - M_o$ elements will approach zero values. It is, therefore, important to see how the $M_o - M$ weights, $w_k$, in the middle of $w_k$ in Fig. 1 affects the convergence behavior of the UFBLMS ADF. One can see from Fig. 1 that the matrix $X_k$ with its size $(I \times M_o) \times M_o$, becomes the lower part of $X_k$ when $M \cdot M_o$. As for the matrix $X_k$ with its size $(M_o \times M_o)$ the time index of the data in the last $M_o$ columns is not continuous, and particularly in each row of $X_k$ two data being apart by $N$ sampling times are undesirably wrapped around. Therefore, the existence of non-zero $w_k$ will help improve the performance in the last $L - (M_o - M)$ output samples, whereas in the first $M_o - M$ output samples it will not. However, to reduce the overall block MSE, the $M_o - M$ weight vector $\bar{W}_k \Delta s w_k : e w_k$ will try to be close to the optimum weight vector $w_{opt}$ for the case of $M = M_o$. It can also be seen that, unlike for the first $M_o - M$ output samples, the MSE of the UFBLMS ADF for the last $L - (M_o - M)$output samples will be significantly improved compared to that of the BLMS ADF. All these phenomena will be demonstrated by computer simulation in Section V.

It can be seen from Fig. 1 that, given $M_o$, the undesirable wrap-around (or circular convolution) effect increases and propagates from the first output sample toward the last output sample in each block as $M$ decreases to 1. Thus.

---

**Fig. 1.** A schematic representation of the circular convolution effect when $M < M_o$.

(a) $sX_u \Delta [X_k | X_c] \Delta L$

(b) $sW_k \Delta w_k : e w_k$
unless the input has special characteristics (for example, periodicity) to avoid the wrap-around effect, the worst performance of the UFBLMS ADF yields when $M=1$. Assuming $N=1$ we can easily see that the case with $M=1$ leads to a special structure of the UFBLMS ADF in which $N=L$. In addition to the removed constraint in weight adjustment, the sectioning operation for computing the output is also eliminated in this structure since $P_{n,k}=I$, and thus $y_k=F^{-1}(sX_k,u_0)$. This UFBLMS ADF is the same as the structure studied in [13]-[18]. As discussed above, this structure cannot be expected to show good performance in most applications because of the serious wrap-around effect. One possible use of the UFBLMS ADF with $M=1$ would be in the adaptive line enhancement (ALE) application. According to our results of computer simulation, even in the ALE, this structure appears not to show good performance unless the time lag $N-1$ is some multiple of the period (in samples) of the input sinusoid and the signal-to-noise ratio (SNR) is high. In Section V, we will also present some results of computer simulation, illustrating the performance characteristics of the UFBLMS ADF in $\Delta$-step linear prediction applications. According to these results, the first $M$ elements of the time-domain weight vector of the UFBLMS ADF have the same information about the signals as that of the constrained algorithm.

IV. CONVERGENCE ANALYSIS OF SELF-ORTHOGONALIZING FBLMS ADF's

The self-orthogonalizing algorithms of the LMS-type ADF's have been known to have fast convergence speed without altering the optimum Wiener solution [19]. In this section, we first discuss how we can implement in the frequency domain some self-orthogonalizing algorithms for the UFBLMS ADF and then extend our discussion to the constrained algorithm.

A. Self-orthogonalizing Algorithms for UFBLMS ADF

In Section III, we have shown that the convergence behavior of the UFBLMS ADF is governed by the matrix $A_k$, and it can be approximated as a circulant matrix which is diagonalized by the DFT. Since the inverse of a circulant matrix can be computed relatively easily in the DFT domain, we can consider a self-orthogonalizing algorithm in which the mean of the weight-error vector is given as [9],[19]

$$\mathbf{w}_{k+1} = \mathbf{w}_{k} + \eta_L [R_x - uRx^\dagger] F^{-1}(sX_k,u_0)$$

when $\eta$ is a convergence constant. Referring to the approach taken in Section III-A, we can derive the UFBLMS algorithm which corresponds to (44) as the following:

$$\mathbf{w}_{k+1} = \mathbf{w}_{k} + \eta R_x uA_k sX_k$$

Combining (33) and (45), we finally get a self-orthogonalizing UFBLMS algorithm as

$$\mathbf{w}_{k+1} = \mathbf{w}_{k} + \eta A_k sX_k$$

As done in [9], the NxN diagonal matrix $A_k$ can be computed by first estimating the input autocorrelation values $\{a_l\}$, and then computing either the FFT of the N-point sequence $\{a_0, a_1, \ldots, a_{N-1}\}$ or the eigenvalues represented by (34).

Next we discuss another possible self-orthogonalizing algorithm which is given by

$$\mathbf{w}_{k+1} = \mathbf{w}_{k} + \eta sR_x sX_k$$

where $sR_x$ is an NxN diagonal matrix and defined by $sR_x = E[sX_k sX_k^\dagger]$. The motivation behind the self-orthogonalizing algorithm in (47) is that the self-orthogonalizing diagonal matrix $sR_x$ can easily be obtained using the
transformed input $sX_k$ which is already available independently of the self-orthogonalization process. To see how the above self-orthogonalizing algorithm works, we derive the mean of the weight-error vector from (17a), (27) and (47) as

$$E[s_{v_k}^2] = (1 - y^* R_k^{-1} s R_k) E[s_{v_k}]$$

where $s R_k \triangleq F^{-1} R_k F$. Since $s X_k - F s X_k F^*$ from (3a) of Part I, we can show from (47) and (48) that

$$s R_k - F^* E[s \overline{X} s X_k F - E[s X_k s X_k^*]]$$

Comparing (28) and (49), one cannot easily see whether the adjustment algorithm described by (47) and (48) is a self-orthogonalizing algorithm since $s R_k$ in its present form is not the same as $s R_k$. However, in the following it will be shown that $s R_k$ can be approximated and obtained by multiplying $s R_k$ with a scalar constant.

We can see from (49) that, since $s X_k$ is circulant, $s R_k$ is also circulant and symmetric. To compute $s R_k$ according to (49), we must specify all the elements of $s X_k$ which is represented by four matrices $X_k$, $X_k$, $X^T_k$, and $X_k^T$. Since $X_k$ and $X^T_k$ are described in (2) and (29) when $N_2 = 0$, we specify the elements of the $(N-L) \times M$ matrix $X_k$ and $(N-L) \times (N-M)$ matrix $X_k$ as the following:

$$X_k = \begin{bmatrix} X_k, & X_k, & \cdots & X_k, & X_k, & X_k, & \cdots & X_k, \end{bmatrix}$$

We can now compute the elements of $s R_k$ from (2), (29), (50) and (51) as

$$s R_{k1} = (N - |i - j|) s_{v_k}, \quad 0 \leq i, j \leq N - 1$$

Defining a new constant $b_r$ as

$$b_r \triangleq 1 - \frac{1}{N}, \quad 0 \leq f \leq m,$$

we can alternatively represent (52) as

When $m < N$, since $b_r \geq 1$ for $f = 1, 2, \ldots, m$, we can further approximate (55) as

Comparing (32) and (56), we obtain

$$s R_k \approx s R_k$$
It is noted here that the values of $M$ ($\leq M_k$) and $m$ are determined by the signal characteristics. Therefore, once $M$ and $m$ are given, better approximations of both matrices $uR_k$ and $sR_k$ are obtained for larger values of $L$. This is because, as $L$ increases, $N$ increases and thus the validity of the assumption of $m \ll N$ becomes better. Obviously, given $M$ and $m$, the FLMS case ($L=1$) results in the worst approximations of $uR_k$ and $sR_k$.

Substituting (57) into (48), we finally get

$$E_l[v_{k+1}] - (I_k - \gamma L \frac{1}{N} uR_k^t sR_k) E_l[v_k]$$

Based on (44) and (58), we obtain the expressions for the excess MSE's of the two self-orthogonalizing UFBLMS algorithms as

$$\sigma - \frac{1}{2} \gamma \tau \sigma (uR_k^t uR_k) \omega_{\min} - \frac{1}{2} \gamma \omega_{\min}$$

It is noted here that the result on the excess MSE in (60) is independent of the block length $L$. Comparing (42),(59) and (60), we can have the relation of the convergence constants of the UFBLMS ADF's for the same steady-state MSE as

$$\gamma \approx \eta N - 2N\sigma_k^2$$

In an actual implementation of the self-orthogonalizing UFBLMS algorithm described in (47), $sR_k$ must be estimated in an appropriate way. It has been known that the self-orthogonalizing algorithm using a single-pole LPF for estimation of $sR_k$ significantly improves the convergence speed [11],[5],[9]. On the other hand, since the algorithm in (47) is based on the relation between $sR_k$ and $uR_k$ in (57), the convergence behavior depends on the accuracy of the approximations of $sR_k$ and $uR_k$. As noted earlier in this section, they can be approximated better by processing the signals on the block-by-block basis rather than on the sample-by-sample basis. The improved convergence speed of the self-orthogonalizing FBLMS ADF over the self-orthogonalizing FLMS ADF has been demonstrated in Part I of the paper. It is noted here that in the UFBLMS ADF with $M=1$ [13]-[18], the two matrices $sR_k$ and $uR_k$ are exactly the same. However, as discussed in Section III-D, this structure suffers from the circular convolution effect [20].

B. Self-orthogonalizing Algorithms for Constrained FBLMS ADF

We have shown in Section II-C that the constrained algorithm is derived from the UFBLMS algorithm by placing the constraint on the weight vector. Thus, constrained versions of the two self-orthogonalizing FBLMS ADF's shown in (46) and (47) can be considered as the following:

$$w_{k+1} = P_{k,s} (sR_k + \gamma uR_k + \gamma_1 sR_k X_k^t sR_k e_k)$$

$$w_{k+1} = P_{k,s} (sR_k + \gamma_1 sR_k X_k^t sR_k e_k)$$

We first discuss the convergence behavior of the algorithm in (63). The mean of the time-domain weight vector of the algorithm in (63) can be obtained from (19),(20),(26b),(39) and (48) as

$$E[w_{k+1}] = E[w_k] + \gamma P_{k,s}$$

$$E[w_{k+1}] = E[w_k] + \gamma P_{k,s} sR_k^t - uR_k^t E[w_k]$$

Thus, we get from (64) the mean of the weight-error vector as

$$E[w_{k+1}] = E[w_k] + \gamma P_{k,s} sR_k^t - uR_k^t E[w_k]$$

Substituting $sR_k$ of (57) into (65) leads to

$$E[w_{k+1}] = E[w_k] + \gamma P_{k,s} \frac{1}{N} uR_k^t sR_k E[v_k]$$

$$- (I_k - \gamma L \frac{1}{N} I_k) E[w_k]$$

$$E[w_{k+1}] = E[w_k] + \gamma P_{k,s} \frac{1}{N} uR_k^t sR_k E[v_k]$$

$$- (I_k - \gamma L \frac{1}{N} I_k) E[w_k]$$
Similarly, we can obtain the mean of the weight-error vector of the algorithm in (62) as

\[ E[v_{k+1}] = E[v_k] - \eta \mathbf{P}_u \mathbf{R}_z' \mathbf{R}_u \left[ \frac{E[v_k]}{0} \right] \]

\[ = (I_u \cdot \eta \mathbf{I}_u) \mathbf{E}[v_k]. \tag{67} \]

Thus, from (66) and (67) we obtain the excess MSE’s of the self-orthogonalizing constrained FBLMS algorithms in (62) and (63) as

\[ \varepsilon_{\Delta} = \frac{1}{2} \eta M \varepsilon_{\min} \tag{68} \]

and

\[ \varepsilon_{\Delta} = \frac{1}{2} \eta \frac{1}{N} M \varepsilon_{\min}. \tag{69} \]

Consequently, comparing (9b),(42),(59),(60), (68) and (69), we can see that all the constrained FBLMS algorithms have slightly less steady-state MSE’s compared to the corresponding UFBLMS ADF’s. The increased excess MSE of the UFBLMS ADF is attributed to the increased effective filter length N in comparison to M of the constrained ADF. Also, we can see the relation in (61) among the convergence constants applies to both cases of the FBLMS ADF’s with and without the constraint. The analytical results on the steady-state MSE’s of the self-orthogonalizing constrained FBLMS ADF obtained in this section have been demonstrated by computer simulation in Part I of the paper.

V. COMPUTER SIMULATION RESULTS AND DISCUSSION

In the previous sections (and appendix), we have analyzed the convergence behaviors of the unconstrained and self-orthogonalizing FBLMS ADF’s. In this section, we present the results of computer simulation to verify the analytical results obtained previously. For our computer simulation, we considered three application examples; adaptive channel equalization, adaptive echo cancellation, and adaptive spectral line enhancement. The adaptive equalizer used here is identical to that in Part I. For the adaptive data echo canceller, the adaptive structure shown in [21, Fig. 1] was chosen along with the echo canceller response in [21, Fig. 2]. The input of the echo canceller was random binary data and the variance of the noise added at the output was 0.001. On the other hand, the \( \Delta \)-step adaptive structure in [22, Fig. 1] was used for the simulation of the adaptive line enhancer (ALE). In our simulation, the decorrelation delay \( \Delta \) was set to be unity and the input SNR was -6 dB.

To see the effects of self-orthogonalization and removing the constraint on the convergence behaviors of the FBLMS ADF’s when the filter length is sufficiently long, we did simulation of the adaptive equalizer mentioned above. The results are shown in Figs. 2 and 3 for the cases with and without the constraint and the self-orthogonalization operation, respectively. We can see from all those figures that, as analyzed previously, the MSE’s of the four unconstrained algorithms increase slightly in comparison to the corresponding constrained algorithms. We can also note that the comparison of the FBLMS ADF’s with and without the self-orthogonalization in the frequency domain reveals two distinctive characteristics, one in the transient period and the other in the steady state. We used the same values chosen in Part I for the initial estimate of the self-orthogonalizing matrix for the cases shown in Fig. 3(a) and (b). Unlike the constrained algorithm, the unconstrained algorithm shows an overshoot in the transient period both for the cases of the overlap-save and overlap-add sectioning methods. Therefore, an increased initial estimate must be used in the unconstrained algorithm which is more likely to become unstable because of its increased effective filter length. On the other hand, unlike in the case of the constant convergence factor, the self-orthogonalizing algorithms realized using overlap-add sectioning show increased steady-state MSE’s as compared to those realized using overlap-save sectioning
To verify our discussion in Section III on the convergence characteristics of the unconstrained algorithm when the filter length is not sufficiently long, we did computer simulation of an adaptive echo canceller. In Fig. 4 we first show what values the weights of the FBLMS ADF’s with and without the constraint have eventually in the steady state. We can see from Fig. 4(a) and (b) that, unlike the constrained algorithm, some of the last \((N-M)\) weights of the unconstrained algorithm are not zero in the steady state, although the first \(M\) weights of the two filters are about the same. Comparing Fig. 4(b) with the original echo channel response, it turns out that they are very close to each other [21]. In Fig. 5 we show the MSE behaviors of the same ADF’s. We can see from this figure that, as discussed in Section III, the unconstrained algorithm reveals degradation in MSE in the beginning of the block due to the circular convolution effect. However, in the rest of the block, the MSE performance is significantly...
improved by having nonzero values after the $M_{th}$ weight. Consequently, in that way the unconstrained algorithm minimizes the overall block MSE which can be much less than that of the constrained algorithm.

Finally, we discuss the results of computer simulation of the ALE with $\Delta=1$. In Fig. 6(a) and (b), we show the time-domain weight values of the two FBLMS ALE's in the steady state. We see from this figure that the UFBLMS ALE has a very large peak in its time-domain response. The reason is as follows. Based on the representations of the output and data matrices shown in (43),(2) and (29), respectively, the desired response vector of the UFBLMS ALE with $\Delta=1$ is given as $d_k=[x_{k,L+1} x_{k,L+2} \cdots x_{k,L+L} L x_{k,L}]^T$. Thus, we can see from (29) that the last column of $X_c$ is almost the same as $d_k$. Consequently, we can see from (43) that, to minimize the MSE between $y_k$ and $d_k$, the last element of the weight vector $w_k$ must be very large. However, it is interesting to see from Fig. 6(a) and (b) that the first $M$ weights of the two ALE's appear to have similar information. Those weights are different only by a scale factor. This aspect can be seen more clearly in the frequency responses of Fig. 6(c). As an extreme case of the UFBLMS ALE, we can consider the
Fig. 6. Comparison of the steady-state performances of the FBLMS ALE's with and without the constraint (M=32, L=32, N=64, \( \mu =0.0001 \) and the normalized frequency of the sinusoid, \( \bar{f}_0 =0.1 \)).

(a) Overall N time-domain weight values

(b) First (N-1) time-domain weight values

(c) Magnitude frequency responses of the weights in (b).

VI. CONCLUSIONS
In Part I of the paper, both for the cases of the overlap-save and overlap-add sectioning, we derived the unconstrained algorithms from the alternative structures which cannot be obtained directly from the original BLMS ADF. The reason for doing so was as follows. According to our simulation result, unlike the conventional FBLMS ADF derived from the BLMS ADF, both the alternative and unconstrained FBLMS ADF's always converge to the optimum Wiener solution for any initial values of the frequency-domain weight vector. In Part II we have shown that the UFBLMS ADF is basically a frequency-domain problem for which case the optimum solution must be formulated directly in the frequency domain. Also, it has been shown that the autocorrelation matrix governing the convergence behavior of the UFBLMS ADF's is approximately a diagonal matrix in the frequency domain (or a circulant matrix in the time domain) both for the cases of the overlap-save and overlap-add sectioning. Furthermore, we have shown that the convergence behavior of the self-orthogonalizing FBLMS ADF with the constraint is governed by the same autocorrelation matrix as that of the FBLMS ADF without the constraint. Therefore, we believe that the UFBLMS ADF must be conceived as an independent problem and thereafter a constrained version of the UFBLMS ADF could be considered as a special case with the constraint on the weights if it is necessary such as in the $\Delta$-step linear prediction.

In this paper, we have analyzed extensively the convergence behaviors of the UFBLMS and self-orthogonalizing FBLMS ADF's realized based on overlap-save sectioning. According to the results of our analysis, the optimum solutions of the two FBLMS ADF's with and without the constraint are the same when $M \geq M_0$. The mean of the weight vector of the UFBLMS ADF has also shown to converge to the optimum solution regardless of the initial values of the frequency-domain weight vector. It has been shown by analysis that the steady-state MSE of the UFBLMS ADF, however, increases slightly in comparison to the constrained algorithm when the same convergence constant is used in both cases. On the other hand, when $M < M_0$, the original and unconstrained FBLMS ADF's have been shown to reveal quite different convergence behaviors in the steady state. It has been found that, when $M < M_0$, the UFBLMS ADF suffers from the circular convolution effect in the first $M - M_0$ output samples in each block. However, in the rest output samples in the block, while the constrained algorithm yields poor performance due to the insufficient filter length, the performance of the UFBLMS ADF is improved significantly by utilizing its extended filter-length capability. As another new result, we have shown by computer simulation that the UFBLMS ADF used even in the adaptive line enhancer application has the same information about the signals as the constrained algorithm does. Consequently, in most applications the use of the UFBLMS algorithms yields no significant degradation in performance.

As for the self-orthogonalizing FBLMS ADF, we have studied in detail the properties of the autocorrelation matrix and the self-orthogonalizing matrix. As a result, we have shown that the two matrices can be approximated such that they differ only by a constant scale factor. It has also been shown that the accuracy of the approximations of these matrices can be improved when the block length is sufficiently long. This result verifies why the self-orthogonalizing FBLMS ADF can have superior convergence speed over the self-orthogonalizing FLMS ADF which operates on a sample-by-sample basis. In addition, we have obtained the analytical results on the steady-state MSE's of the self-orthogonalizing FBLMS ADF's with and without the constraint and verified the relations among the convergence factors that were suggested in Part I. Finally, in Appendix we have analyzed the convergence behavior of the overlap-add implementation. According to the result obtained, it has been found that the
excess MSE of the self-orthogonalizing FBLMS ADF using overlap-add sectioning is larger than that of the overlap-save implementation for the same convergence constant.

APPENDIX

Convergence Analysis of FBLMS ADF’s Realized Using Overlap-add Sectioning

In this appendix, we analyze the convergence behavior of the overlap-add I implementation developed in Part I of the paper. This appendix together with Sections III and IV which dealt with the convergence behavior of the overlap-save implementation will provide a unified theory on the convergence behaviors of the FBLMS ADF’s realized based on the fast convolution.

A. Unconstrained Algorithm

The frequency-domain error vector of the overlap-add implementation is defined from (37) of Part I as

\[ w_k = \tilde{d}_k + (P_{l,0} a X_k + Q_{l,0} \tilde{X}_k - a w_{k-1}) \quad (A.1) \]

The pair of the FBLMS ADF’s based on overlap-add I sectioning is given as

\[ w_{k+1} = w_k + \mu (a \tilde{X}_k - a \tilde{X}_k - w_k) \quad (A.2) \]

and

\[ a w_{k+1} = P_{w,0} a w_k + \mu a \tilde{X}_k \tilde{r}_k + a \tilde{X}_k - a w_k \quad (A.3) \]

where \( w_k = Q_{w,0} a e_k \). The optimum weight vector \( w_{opt} \) of the UFLMS ADF in (A.2) must satisfy

\[ \begin{align*}
E \left[ a \tilde{X}_k \tilde{r}_k + a \tilde{X}_k - a w_{k-1} \right] = 0
\end{align*} \]

(A.4)

Substituting (A.1) into (A.4) yields

\[ \begin{align*}
E \left[ a \tilde{X}_k P_{l,0} a \tilde{X}_k + Q_{l,0} a \tilde{X}_k - a w_{k-1} \right]
\end{align*} \]

(A.5)

Converting all the variables from the frequency domain into the time domain, we get

\[ E \left[ a X_k P_{l,0} a X_k + S_{m-1,k} a d_k \right] \quad \text{(A.6)} \]

\[ E \left[ a X_k P_{l,0} a X_k + S_{m-1,k} (P_{l,0} a X_k - Q_{l,0} a X_k) \right] \]

\[ a w_{opt} \]

Using the definitions of \( a X_k, P_{l,0} \) and \( Q_{l,0} \), we can show that

\[ \begin{align*}
P_{l,0} a X_k + Q_{l,0} a X_k & = \left[ \frac{a X_k}{0} \right]_{k-1}^k \quad (A.7)
\end{align*} \]

Thus, we get from (A.6) and (A.7)

\[ \begin{align*}
E \left[ a X_k a d_k - E \left[ a X_k a X_k \right] a w_{opt} \right]
\end{align*} \]

(A.8)

Comparing (A.8) and (26b), we can see that the optimum weight vectors of the two UFLMS ADF’s are the same both for the overlap-save and overlap-add I implementations. In other words, the overlap-add I implementation in (A.2) is another exact implementation of the system described by (22a) and (22b). Consequently, the same results on the convergence behavior obtained in Section III apply to the overlap-add implementation with a constant convergence factor as well. However, it will be seen in the following that this is not the case when the weight-adjustment algorithm is modified using the frequency-domain self-orthogonalizing matrix.

B. Self-orthogonalizing Algorithm

We discuss the following self-orthogonalizing algorithms in order:

\[ \begin{align*}
w_k & \quad w_{opt} \quad R_k \quad \tilde{X}_k \quad a e_k \\
\end{align*} \]

(A.9)

and

\[ \begin{align*}
w_{k+1} & \quad P_{w,0} a w_k \quad a \tilde{r}_k \quad R_k \quad a \tilde{X}_k \quad a e_k
\end{align*} \]

(A.10)
where the \( \times N \times N \) diagonal matrix \( \mathbf{R}_k \) is defined as \( \mathbf{R}_k \triangleq \mathbb{E} [ \mathbf{X}_k \mathbf{X}_k^\top ] \). Following the same approach as used in the previous subsection, we can get the following for (A.9) as

\[
E [ u_{k+1} ] = E [ u_k ] + \gamma E [ \mathbf{R}_k^{-1} \mathbf{X}_k \mathbf{P}_L + \\
+ \mathbf{S}_{w,1} \mathbf{X}_{k-1} \mathbf{S}_{w,1}^\top ] + \mathbf{d}_k - ( \mathbf{P}_{L,0} \mathbf{X}_k + \mathbf{Q}_{L,0} \mathbf{X}_{k-1} ) \mathbf{w}_k | \]

(A.11)

where

\[
\mathbf{R}_k \triangleq \mathbb{F}^{-1} \mathbf{R}_k \mathbb{F}.
\]

When \( \mathbf{d}_k = \mathbf{X}_k \mathbf{w}_k + \xi_k \), we can show that

\[
\mathbf{d}_k = \mathbf{X}_k \left[ \begin{bmatrix} \mathbf{w}_k \\ \xi_k \end{bmatrix} \right] + \xi_k - \mathbf{X}_k \mathbf{w}_k \mathbf{w}_k^\top + \xi_k. \tag{A.12}
\]

Combining (A.7), (A.8), (A.11) and (A.12), we obtain

\[
E [ v_{k+1} ] = ( I_k - \gamma \mathbf{R}_v ) E [ v_k ] \tag{A.13}
\]

where

\[
\mathbf{R}_v \triangleq \mathbb{E} [ \mathbf{R}_k^{-1} \mathbf{X}_k \mathbf{P}_L + \mathbf{R}_k^{-1} \mathbf{X}_k \mathbf{X}_{k-1} \mathbf{S}_{w,1} \mathbf{S}_{w,1}^\top ] + ( \mathbf{P}_{L,0} \mathbf{X}_k + \mathbf{Q}_{L,0} \mathbf{X}_{k-1} ) \mathbf{w}_k | \]

(A.14)

Expanding \( \mathbf{R}_v \) into two terms leads to

\[
\mathbf{R}_v - \mathbf{R}_v^{-1} E [ \mathbf{X}_k \mathbf{P}_L ( \mathbf{P}_{L,0} \mathbf{X}_k + \mathbf{Q}_{L,0} \mathbf{X}_{k-1} )] \mathbf{X}_k \mathbf{P}_L^{-1} \mathbf{R}_v^{-1} E [ \mathbf{X}_k ( \mathbf{P}_{L,0} \mathbf{X}_k + \mathbf{Q}_{L,0} \mathbf{X}_{k-1} ) ] | \]

Noting that the correlation values between input samples of the different blocks are very small when \( L \gg M \), we can approximate \( \mathbf{R}_v \) in (A.14) as

\[
\mathbf{R}_v - \mathbf{R}_v^{-1} E [ ( \mathbf{P}_{L,0} \mathbf{X}_k )^\top ( \mathbf{P}_{L,0} \mathbf{X}_k ) ] - \mathbf{R}_v^{-1} \]

(A.15)

In the steady state, \( \mathbf{R}_v \) becomes

\[
\mathbf{R}_v - \mathbf{R}_v^{-1} E [ ( \mathbf{P}_{L,0} \mathbf{X}_k )^\top ( \mathbf{P}_{L,0} \mathbf{X}_k ) ] \mathbf{R}_v^{-1} | ( \mathbf{Q}_{L,0} \mathbf{X}_{k-1} )^\top | ( \mathbf{Q}_{L,0} \mathbf{X}_{k-1} ) \mathbf{R}_v^{-1} \]

(A.16)

Denoting \( \mathbf{X}_k \triangleq \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \mathbf{X}_k \), we can modify (A.16) as

\[
\mathbf{R}_v - \mathbf{R}_v^{-1} E [ \mathbf{A} \mathbf{A}^\top + \mathbf{B} \mathbf{B}^\top ] \]

(A.17)

We finally get from (A.13) and (A.17)

\[
E [ u_{k+1} ] = ( I_k - \gamma \mathbf{R}_v ) E [ u_k ] \tag{A.18}
\]

Consequently, the excess MSE of the self-orthogonalizing algorithm in (A.9) is given as

\[
\delta \mathbf{e} = \frac{1}{2} \gamma \frac{N}{L} \mathbf{e}_{\min} \tag{A.19}
\]

As for the constrained self-orthogonalizing FBLMS ADF of (A.10), we can get

\[
E [ v_{k+1} ] = E [ v_k ] - \gamma \mathbf{R}_{k,v} \mathbf{R}_v^{-1} \mathbf{R}_v \left[ \begin{bmatrix} E [ v_k ] \\ \xi_k \end{bmatrix} \right] \mathbf{w}_k \]

(A.20)

Thus, the excess MSE of (A.10) becomes from (A.20)

\[
\delta \mathbf{e} = \frac{1}{2} \gamma \frac{M}{L} \mathbf{e}_{\min} \tag{A.21}
\]

Comparing (A.19) and (A.21) with (60) and (69), respectively, we can see that the excess MSE's of the self-orthogonalizing FBLMS ADF's realized using overlap-save sectioning are larger than those of the overlap-save implementation when the same convergence constants are used in both cases.
REFERENCES


Jae Chon Lee was born in Seoul, Korea, in 1954. He received the B.S. degree with honors in electronics engineering from Seoul National University (SNU), Seoul, in 1977, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Seoul, in 1979 and 1983, respectively.

Since August 1983, he has been with KAIST as a senior research scientist. During the academic years 1984-1985 and 1985-1986, he was on leave at the Massachusetts Institute of Technology (MIT) and the University of California at Santa Barbara (UCSB), working with Prof. Jae Lim and Prof. Sanjit Mitra, respectively. His research interest is in the area of digital signal processing and digital communication systems including adaptive filtering and neural computing.

Dr. Chong Kwan Un received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Delaware, Newark, Delaware, in 1964, and 1969, respectively. From 1969 to 1973 he was Assistant Professor of Electrical Engineering at the University of Maine, Portland, where he taught communications and did research on synchronization problems. In May 1973, he joined the Staff of the Telecommunication Sciences Center, SRI International, Menlo Park, CA, where he did research on voice digitization and bandwidth compression systems. Since June 1977 he has been with KAIST, where he is Professor of Electrical Engineering and Head of the Communications Research Laboratory, teaching and doing research in the areas of digital communications and digital signal processing. He has authored or coauthored over 140 papers and 50 reports in speech coding and processing, packet voice/data transmission, synchronization, and digital filtering. Also, he holds 6 patents granted or pending. From February 1982 to June 1983 he served as Dean of Engineering at KAIST.

Dr. Un has received a number of awards including the 1976 Best Paper Award from the IEEE Communications Society, the National Order of Merits (Dong Baik Jang) from the Government of Korea, and Achievement Awards from KITE, KICS and ASK. He is a Fellow of IEEE and a member of Tau Beta Pi and Eta Kappa Nu Honor Societies.