and the scalar α_1 corresponds to the minimum mean-square error of the quadratic function $W^T Q_1 W - 2P_1^T W + 1$ and is given by

$$\alpha_1 = 1 - W_1^{oT} Q_1 W_1^o. \tag{7}$$

Hence, if there are enough number of degrees of freedom in the system, one would expect $\alpha_1 \cong 0$ and

$$W_1^{oT} Q_1 W_1^o \cong 1. \tag{8}$$

Hence, the constrained optimization problem can be reexpressed as

$$\min_W W^T R W \tag{9a}$$

$$\text{subject to } (W - W_1^o)^T Q_1 (W - W_1^o) \leq \xi. \tag{9b}$$

III. Limiting Solution as $\xi \to 1$

It can be verified [1] that the optimum weight vector that solves (9) is given by

$$\begin{aligned} \hat{W} &= \hat{\lambda}_o (R + \hat{\lambda}_o Q_1)^{-1} Q_1 W_1^o \\ &= W_1^o - (R + \hat{\lambda}_o Q_1)^{-1} R W_1^o \end{aligned} \tag{10}$$

where $\hat{\lambda}_o$ is the optimum Lagrange multiplier, which is the root of the following equation:

$$W_1^{oT} R (R + \hat{\lambda}_o Q_1)^{-1} Q_1 (R + \hat{\lambda}_o Q_1)^{-1} R W_1^o = \xi. \tag{11}$$

It is clear from (11) that as $\xi \to 0$, $\hat{\lambda}_o \to \infty$, and as $\xi \to 1$, $\hat{\lambda}_o \to 0$. Also

$$\begin{aligned} \frac{d\xi}{d\hat{\lambda}_o} = & -2 W_1^{oT} R (R + \hat{\lambda}_o Q_1)^{-1} Q_1 (R + \hat{\lambda}_o Q_1)^{-1} \\ & \cdot Q_1 (R + \hat{\lambda}_o Q_1)^{-1} R W_1^o \end{aligned} \tag{12}$$

Hence, ξ is a nonincreasing function of $\hat{\lambda}_o$ as $\hat{\lambda}_o$ increases.

Also, as $\hat{\lambda}_o \to 0$

$$(R + \hat{\lambda}_o Q_1)^{-1} \cong R^{-1} - \hat{\lambda}_o R^{-1} Q_1 R^{-1}. \tag{13}$$

Equation (11) can be reexpressed as

$$\begin{aligned} & \hat{\lambda}_o^2 W_1^{oT} Q_1 R^{-1} Q_1 R^{-1} Q_1 W_1^o - 2\hat{\lambda}_o W_1^{oT} Q_1 R^{-1} Q_1 W_1^o \\ & + W_1^{oT} Q_1 W_1^o = \xi. \end{aligned} \tag{14}$$

Substituting (13) into (10) gives

$$\lim_{\xi \to 1} W \cong \hat{\lambda}_o R^{-1} Q_1 W_1^o = \hat{\lambda}_o R^{-1} P_1. \tag{15}$$

Recently, Kikuma and Takao [5] proposed a technique based on the correlation-constrained minimization of power (CCMP) method for broad-band array design. The criterion is to minimize the output power under the constraint on the cross correlation between the desired signals at the input and output of the array. The constraint to protect the desired signal is determined by a prior knowledge of the characteristics of the desired signal in terms of its direction of arrival and frequency spectrum. It is shown in [4] that when the desired signal is modeled to have a flat spectrum over the frequency band of interest, the CCMP method is equivalent to solving the following constrained optimization problem:

$$\min_W W^T R W \tag{16a}$$

$$\text{subject to } P_1^T W = 1. \tag{16b}$$

The optimum weight vector that solves (16) is given by

$$\hat{W} = \frac{R^{-1} P_1}{P_1^T R^{-1} P_1}. \tag{17}$$

Comparing (15) and (17), it is interesting to note that the limiting solution of the quadratically constrained broad-band processor problem as $\xi \to 1$ is equivalent to that of the CCMP method. Hence, the following conclusions can be made.

1) It was reported in [5] that the CCMP method is sensitive to spectrum mismatch. Hence, one would expect that the soft constrained minimum variance beam-forming method as proposed in [6] would also be sensitive to spectrum mismatch if high levels of distortion are permitted.

2) In those applications, where the signal power spectrum is known, the improvement in SNR through the use of soft constraints can also be achieved through the constrained system defined by (16). In fact, the linearly constrained system defined in (16) is much easier to solve.

IV. Conclusions

The correspondence has derived the limiting solution of the quadratically constrained broad-band beam formers as $\xi \to 1$. It is shown that the limiting solution is equivalent to the CCMP method proposed in [5]. Hence, the improvement in SNR through the use of soft constraints [6] can also be achieved through the CCMP method, which is much easier to solve. However, like the CCMP method, one would expect that the soft constrained beam-forming method is sensitive to spectrum mismatch when high levels of distortion are permitted.

References

[1] M. H. Er and A. Cantoni, "A new approach to the design of broad-band element space antenna array processors," *IEEE J. Ocean. Eng.*, vol. OE-10, pp. 231-240, July 1985.
[2] ——, "An alternative formulation for an optimum beamformer with robustness capability," *Proc. Inst. Elec. Eng.*, vol. 132, part F, pp. 447-460, Oct. 1985.
[3] ——, "A unified approach to the design of robust narrow-band antenna array processors," *IEEE Trans. Antennas Propagat.*, vol. 38, pp. 17-23, Jan. 1990.
[4] M. H. Er, "An alternative implementation of quadratically constrained broadband beamformers," *Signal Processing*, vol. 21, pp. 117-127, Oct. 1990.
[5] N. Kikuma and T. Takao, "Broadband and robust adaptive antenna under correlation constraint," *Proc. Inst. Elec. Eng.*, vol. 136, part H, pp. 85-89, Apr. 1989.
[6] B. D. Van Veen, "Minimum variance beamforming with soft response constraints," *IEEE Trans. Signal Processing*, vol. 39, pp. 1964-1972, Sept. 1991.

On the Steady State Performance of Frequency Domain LMS Algorithms

Arie Feuer and Roberto Cristi

***Abstract*—The use of the fast Fourier transform (FFT) in the implementation of the least mean square (LMS) algorithm in the frequency**

Manuscript received May 24, 1990; revised May 4, 1992.
A. Feuer is with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel.
R. Cristi is with the Department of Electrical and Computer Engineering, Naval Postgraduate School, Monterey, CA 93943.
IEEE Log Number 9203376.

domain results in several types of algorithms, two of which can be classified as constrained and unconstrained. In this correspondence, we point out that, in general, especially with correlated data, the unconstrained algorithm may have a significant performance advantage in steady state if block mean square error is the criterion. Furthermore, we point out here that, in the constrained algorithms, the choice of different step sizes in different frequency bins (as is commonly done) will very likely result in a deterioration of steady state performance. This does not happen in the unconstrained algorithm.

I. Introduction

Adaptive filters are used in a wide variety of applications. The most common algorithm implemented in these filters is the least mean square (LMS) algorithm. However, two phenomena have adverse effects on the performance of the LMS. When there is a need for a high-order filter, the allowable step size is reduced, and the algorithm becomes very slow. Another problem occurs when the input signal to the filter is highly correlated. Then, typically, the result is a large spread of the eigenvalues of the autocorrelation matrix, which again causes the algorithm to slow down. In addition, with a large-order filter, the computational load can be very substantial

A number of researchers noted the possibility of implementing the adaptive filter in the frequency domain (see, e.g., [1]-[6], [9]). There, the use of the fast Fourier transform (FFT) provides a considerable reduction in computation, and the ability to control each frequency bin separately provides an improved convergence rate over the time domain LMS algorithm. There are several approaches described in the mentioned references, differing in the way the FFT is used. The common underlying denominator is that, in most cases, the frequency domain adaptive filter is used as an alternative to the time domain LMS, attempting to reach the Wiener optimal solution faster and in a computationally more efficient way. In [9] this aspect is emphasized. The simulation results presented there show the transient behavior of some of these approaches.

In this correspondence, we point out, however, that depending on the data used, the results may be quite different than implied in the various algorithms. We mainly concentrate on the optimal solutions and the steady state solutions, but also, discuss briefly some convergence aspects.

II. Constrained and Unconstrained Frequency Domain Adaptive Filters

The well-known least mean square (LMS) algorithm has the form

$$w(t+1) = w(t) + 2\mu e(t) X(t) \tag{1}$$

where $X(t)$ is the data vector (sometimes referred to as the regression vector), and in adaptive filters typically consists of

$$X(t) = [x(t), x(t-1), \cdots, x(t-N+1)]^T \tag{2}$$

and μ is the time domain LMS step size. The error at sample t is given by

$$\begin{aligned} e(t) &= d(t) - y(t) \\ &= d(t) - X(t)^T w(t) \end{aligned} \tag{3}$$

where $d(t)$ is the "training sequence" or the desired response of the adaptive filter.

A block implementation of the above algorithm was proposed in [7] to gain computational efficiency. The idea is to accumulate a block of data and update the gain vector w once every block, namely, at the jth block

$$w(j+1) = w(j) + \frac{2\mu_B}{L} \tilde{\mathfrak{X}}_j^T \epsilon_j \tag{4}$$

where L represents the block length.

$$\tilde{\mathfrak{X}}_j = [X((j-1)L+1), X((j-1)L+2), \cdots, X(jL)]^T \tag{5}$$

$$\epsilon_j = [e((j-1)L+1), e((j-1)L+2), \cdots e(jL)]^T \tag{6}$$

$$e((j-1)L+i) = d((j-1)L+i) - X((j-1)L+i)^T w(j) \tag{7}$$

and μ_B is the step size. It has been shown in [7] and in [8] that the block LMS (BLMS) algorithm of (4) attempts to optimize the average block mean square error (BMSE), namely

$$J_L = \frac{1}{L} \mathcal{E}\{\epsilon_j^T \epsilon_j\} \tag{8}$$

which, for stationary signals, results in the same optimal solution as the LMS algorithm [of (1)], provided the number of gains [namely, the dimension of the vector $w(j)$] is the same in both algorithms.

Next came the frequency domain implementation of the adaptive filter. This, however, created the need for $2N$-point FFT, because of the known relationships between the DFT, the circular, and the linear convolutions. Hence, the frequency domain algorithm, resulting from N-dimensional time domain algorithm, is $2N$ dimensional. Around this point evolve the differences between the algorithms proposed in [3], [4], [6], and [9]. The algorithm proposed in [6] unifies and generalizes the ones in [3], [4], and [9]. It is given in the following form:

$$W(k+1) = W(k) + 2G\mu_F X(k)^H VE(k) \tag{9}$$

where

$$W(k) = [W_1(k), \cdots, W_{2N}(k)]^T$$

is the frequency domain gain vector at iteration (or block) k.

In the equation above, we denote by G and V

$$G = FgF^{-1} \tag{10a}$$

$$V = FvF^{-1} \tag{10b}$$

where

$$F = \{F_{pq}\}, \quad F_{pq} = \exp\left(-j\frac{2\pi}{2N}(p-1)(q-1)\right), \quad p, q = 1, 2, \cdots, 2N$$

is the $2N$-point FFT matrix; and g, v are diagonal matrices representing the weight vector constraint and the error vector constraint, respectively.

By similar notation

$$X(k) = F\mathfrak{X}_k F^{-1} = \mathrm{diag}\,(X_0(k), \cdots, X_{2N-1}(k)) \tag{11}$$

is the input signal matrix in the frequency domain,

$$\mathfrak{X}_k = \begin{bmatrix} x((k-1)N) & x((k-1)N+1) & \cdots & x(kN) & \cdots & x((k+1)N-1) \\ x((k+1)N-1) & x((k-1)N) & \cdots & x(kN-1) & \cdots & x((k+1)N-2) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ x((k+1)N+1) & x((k-1)N+2) & \cdots & x(kN+1) & \cdots & x((k-1)N) \end{bmatrix} \tag{12}$$

is the input signal matrix

$$E(k) = D(k) - X(k)W(k) \tag{13}$$

is the error vector in the frequency domain: $D(k)$ is the $2N$-point FFT of the desired signal and μ_F is a nonsingular matrix controlling the convergence rate of the algorithm (typically a diagonal matrix).

With the above notation, we can readily point out the main differences in [3], [4], and [6].

In [4]

$$g = \sum_{i=N+1}^{2N} e_i e_i^T, \quad v = \sum_{i=1}^{N} e_i e_i^T, \quad \mu_F = \frac{\mu_B}{2N} I$$

(e_i is the ith column of the $2N \times 2N$ identity matrix I).

In [3]

$$g = I, \quad v = \sum_{i=1}^{N} e_i e_i^T, \quad \mu_F = \alpha \operatorname{diag}(\hat{P}_k(X_i)^{-1})$$

$\hat{P}_k(X_i)$ being the average power in frequency bin i.

In [6], an additional possibility is proposed

$$g = \frac{1}{2} \sum_{i=1}^{2N} \left(1 + \cos\left(\frac{\pi}{N} i\right)\right) e_i e_i^T,$$

$$v = \sum_{i=1}^{N} e_i e_i^T, \qquad \mu_F \text{ as in [3].}$$

III. The Optimal Solutions

It is well known that the LMS algorithm (1) is motivated by the MSE criterion and converges, with some misadjustment, to its optimal value. Similarly, the BLMS converges to the optimal value of the BMSE in (8). The question is how can the algorithm in (9) be motivated and what does it converge to. To answer this question, let us define the following problem.

Find $W \in C^{2N}$, which minimizes

$$J_N = \frac{1}{2N^2} \mathcal{E}\{E(k)^H V E(k)\} \tag{14}$$

subject to the constraint

$$W \in \text{range } [G] \tag{15}$$

(range $[G]$ is the subspace spanned by the columns of G). Note that J_N as defined in (14) is, in fact, the BMSE (with v of [3], [4], and [6]), and that in the algorithm given in (9), $W(k) \in$ range $[G]$ for all k provided $W(0) \in$ range $[G]$ (otherwise $W(k) \rightarrow$ range $[G]$).

To solve the above problem, let

$$n = \text{rank } [G]$$

(note that n is also the dimension of range $[G]$) then, since $G = G^H$, there exists a full column rank matrix $\tilde{G} \in C^{2N \times n}$ such that

$$G = \tilde{G}\tilde{G}^H.$$

Then for every $W \in$ range $[G]$, there exists a unique $\tilde{W} \in C^n$, such that

$$W = \tilde{G}\tilde{W}. \tag{16}$$

Substituting (16) into (14), the optimization problem can be recast in terms of $\tilde{W}$. Namely, find $\tilde{W}$ that minimizes

$$\begin{aligned} J_N &= \frac{1}{2N^2} \mathcal{E}\{E(k)^H V E(k)\} \\ &= \frac{1}{2N^2} [\mathcal{E}\{D(k)^H V D(k)\} \\ &\quad - 2\tilde{W}^H \tilde{G}^H \mathcal{E}\{X(k)^H V D(k)\} \\ &\quad + \tilde{W}^H \tilde{G}^H \mathcal{E}\{X(k)^H V X(k)\} \tilde{G}\tilde{W}]. \end{aligned} \tag{17}$$

Assuming $\mathcal{E}\{X(k)^H V X(k)\} > 0$, a unique optimal solution exists, which is given by

$$\tilde{W}_0 = [\tilde{G}^H \mathcal{E}\{X(k)^H V X(k)\} \tilde{G}]^{-1} \tilde{G}^H \mathcal{E}\{X(k)^H V D(k)\} \tag{18}$$

and the optimal value for J_N will then be

$$J_N^0 = \frac{1}{2N^2} [\mathcal{E}\{D(k)^H V D(k)\} - \tilde{W}_0^H \tilde{G}^H \mathcal{E}\{X(k)^H V X(k)\} \tilde{G}\tilde{W}_0]. \tag{19}$$

Going back to the algorithm in (9), and recasting it in terms of $\tilde{W}(k)$, we get

$$\tilde{W}(k+1) = \tilde{W}(k) + 2\tilde{G}^H \mu_F X(k)^H V E(k) \tag{20}$$

where

$$E(k) = D(k) - X(k)\tilde{G}\tilde{W}(k). \tag{21}$$

Taking the expected value of both sides of (20), and assuming that $X(k)$ and $\tilde{W}(k)$ are independent, it can readily be shown that if the algorithm converges then

$$\lim_{k \to \infty} \mathcal{E}\{\tilde{W}(k)\} = \tilde{W}_\infty$$

and $\tilde{W}_\infty$ will satisfy

$$\tilde{G}^H \mu_F \mathcal{E}\{X(k)^H V D(k)\} = \tilde{G}^H \mu_F \mathcal{E}\{X(k)^H V X(k)\} \tilde{G}\tilde{W}_\infty. \tag{22}$$

The discussion so far enables us to make the following observations.

Observation 1: In the unconstrained frequency-domain adaptive filter (UFLMS), where $n = 2N$, G, and μ_F have no effect on the optimal solution and $\tilde{W}_0 = \tilde{W}_\infty$. Namely, the algorithm *converges to the optimal solution*. However, in general, the optimal solution of the UFLMS does not correspond to a time-invariant filter (it will result in *circular* rather than *linear* convolution).

Observation 2: Any constraint imposed on the FLMS, namely any case where $n < 2N$, will, in general, result in a deterioration of performance (the optimal BMSE J_N will be larger). This can easily be seen from the fact that we are optimizing over a subset of gains as stated in (15). Clearly, the smaller n is the smaller the set to which W is constrained and the larger the corresponding J_N (the worse the performance of the algorithm).

Observation 3: In the constrained case ($n < 2N$), the choice $\mu_F = \mu_0 I$, μ_0, a positive scalar, will guarantee (see (18) and (20))

$$\tilde{W}_\infty = \hat{W}_0.$$

Namely, the algorithm will converge to the corresponding optimal solution. However, any other choice of μ_F will result in $\tilde{W}_\infty \neq \tilde{W}_0$, which means that the steady state BMSE of the algorithm will be larger than the optimal (beside any misadjustment due to the stochastic nature of the algorithm).

Applying the above observations to the three specific cases discussed in [6] and detailed here earlier, we can conclude the following:

For $g = I$ ($n = 2N$) (as in [3]), we get the smallest optimal BMSE and can choose $\mu_F = \text{diag}(\mu_1, \cdots, \mu_{2N}) \neq \mu_0 I$ to improve the algorithms rate of convergence without affecting the corresponding optimal BMSE.

For $g = \Sigma_{i=N+1}^{2N} e_i e_i^T$ ($n = N$) (as in [4]), we get, in general, degradation in the possible optimal BMSE. Additional degradation is introduced by choosing $\mu_F \neq \mu_0 I$—again the typical tradeoff improved convergence rate on account of degraded steady state performance.

For $g = \Sigma_{i=1}^{2N} (1/2) + (1/2) \cos((\pi/N)i)) e_i e_i^T$ ($n = N - 1$) (as in [6]), again we get a possible degradation in the optimal BMSE with further degradation due to the choice $\mu_F \neq \mu_0 I$.

It should be pointed out that the above-mentioned phenomenas were not observed in the simulations described both in [4] and [6]. The reason for that is the special type of data used there. In both cases, the data used were generated by passing $x(t)$ through a finite impulse response (FIR) filter to get $d(t)$, and the order of this filter was assumed to be known and used to determine the adaptive filter order. We note that the simulations in [4] and [6] were done to test the algorithms for the ideal cases for which the analysis is tractable. However, since our purpose is to investigate a more realistic scenario, we choose to use the example described in [3] through which we demonstrate the points made here. This example is more realistic because we use correlated data, and we do not assume the order of the FIR filter.

IV. Numerical Example

To demonstrate the points made, we chose to use the same data used in [3]. These data are generated by passing a sequence $\{x(t)\}$ through an FIR filter of order 31. Namely, the desired signal $d(t)$ is given by

$$d(t) = \sum_{i=1}^{32} w_{0i} x(t - i + 1)$$

where two possibilities for $x(t)$ are considered:

Case 1 $x(t)$ white noise with $\mathcal{E}\{x(t)^2\} = 1$.

Case 2 $x(t)$ the output of an AR filter of order 12, namely

$$x(t) = \sum_{i=1}^{12} a_i x(t - i) + n(t)$$

with $\mathcal{E}\{n(t)^2\} = 1$.

The values of w_{0i} and a_i are given in Table I (they are the same as in [3]).

Differing from [3], we do not assume knowledge of the order of the FIR filter generating $d(t)$ instead choose $N = 16$. This means that the frequency domain implementations will require $2N = 32$ points FFT's. For each of the above cases, we calculate the BMSE

TABLE I

Values of w_{0i}	Values of a_i
$w_1 = 0.230847$	$a_1 = 1.46034$
$w_2 = -0.102171$	$a_2 = -1.26638$
$w_3 = -0.0383698$	$a_3 = 0.850541$
$w_4 = -0.295674$	$a_4 = -0.629542$
$w_5 = -0.00575074$	$a_5 = 0.497503$
$w_6 = -0.00693523$	$a_6 = -0.273701$
$w_7 = -0.029532$	$a_7 = 0.168227$
$w_8 = -0.00690188$	$a_8 = -0.257914$
$w_9 = -0.0503678$	$a_9 = 0.238396$
$w_{10} = 0.0102334$	$a_{10} = -0.508109$
$w_{11} = -0.0448287$	$a_{11} = 0.37944$
$w_{12} = -0.0354443$	$a_{12} = -0.204267$
$w_{13} = -0.331595$	
$w_{14} = 0.0051575$	
$w_{15} = -0.0419697$	
$w_{16} = 0.00518027$	
$w_{17} = -0.163817$	
$w_{18} = -0.00926803$	
$w_{19} = -0.000896433$	
$w_{20} = -0.00948155$	
$w_{21} = -0.00180632$	
$w_{22} = -0.000774926$	
$w_{23} = -0.00647487$	
$w_{24} = -0.00329426$	
$w_{25} = -0.00438749$	
$w_{26} = 0.000927637$	
$w_{27} = -0.0000200423$	
$w_{28} = -0.00152551$	
$w_{29} = 0.00340074$	
$w_{30} = -0.00310239$	
$w_{31} = 0.00551548$	
$w_{32} = 0.00336223$	

according to the following equation:

$$\begin{aligned} \text{BMSE} = \frac{1}{2N^2} [&\mathcal{E}\{D(k)^H VD(k)\} \\ &- 2\tilde{W}^H \tilde{G}^H \mathcal{E}\{X(k)^H VD(k)\} \\ &+ \tilde{W}^H \tilde{G}^H \mathcal{E}\{X(k)^H VX(k)\} \tilde{G}\tilde{W}] \end{aligned}$$

where we substitute for $\tilde{W}$ either $\tilde{W}_0$ or $\tilde{W}_\infty$ calculated from (18) and (22), respectively, according to the particular combination we are interested in.

In all the cases where $\mu_F \neq \mu_0 I$, we chose $\mu_F = \mathcal{E}\{X(k)^H X(k)\}$, which is clearly diagonal since $X(k)$ is a diagonal, and each value along the diagonal is the power in the corresponding frequency bin. Note that, as we pointed out earlier, when we refer to the steady state BMSE, we exclude the additional BMSE due to the stochastic nature if the algorithm (which corresponds to the misadjustment, a term commonly referred to in the literature). Let us denote by:

BMSE-1 Optimal BMSE in the constrained algorithm with $n = 16$ (as in [4]). This is identical to the steady state value for this algorithm when $\mu_F = \mu_0 I$, and to the optimal solution of the 16th-order LMS filter in the time domain.

BMSE-2 Steady state BMSE for the same algorithm as BMSE-1 when $\mu_F \neq \mu_0 I$.

BMSE-3 Optimal BMSE in the constrained algorithm with $n = 31$ (as in [6]).

BMSE-4 Steady state BMSE for the same algorithm as BMSE-3 when $\mu_F \neq \mu_0 I$.

BMSE-5 Optimal BMSE in the unconstrained algorithm [3] (as in [4]—the choice of μ_F has no effect and the steady state BMSE is equal to the optimal).

TABLE II

BMSE	Case 1	Case 2
BMSE-1	0.00059	0.0022
BMSE-2	0.00059	0.0088
BMSE-3	0.00042	0.001
BMSE-4	0.00042	0.0011
BMSE-5	0.00015	0.00069

Note that since no additive noise exists in the generation of $d(t)$, the minimal BMSE possible (which can be achieved if one takes $N = 32$) is 0.

The results are summarized in Table II.

From the results summarized in Table I we note that generally the differences between all the BMSE's are smaller in case I than in case 2. So the phenomena described here are more disturbing when the data are correlated (case 2 here). Since in case 1 it can be shown that the power is equally distributed in all the frequency bins [$x(t)$ is white noise], there are no differences in this case between BMSE-1 and BMSE-2, and similarly between BMSE-3 and BMSE-4. The only factor here is the constraint imposed through G—the larger n the smaller the BMSE. In case 2, on top of this phenomena, which can be observed by comparing BMSE-1, BMSE-3, and BMSE-5, we note the additional degradation due to choosing $\mu_F \neq \mu_0 I$. This can be seen by comparing BMSE-1 to BMSE-2 and BMSE-3 to BMSE-4. In the latter, since $n = 31$, which is very close to $n = N = 32$, the unconstrained algorithm, there is a very small difference between BMSE-3 and BMSE-4, and between both of them and BMSE-5. It is, however, clear that in both cases the unconstrained algorithm results in the smallest BMSE, as we claimed earlier.

V. Conclusion

In this correspondence, we have pointed out a potential problem in some of the frequency domain implementations of the LMS algorithm using the FFT. In general, using unconstrained algorithms will result in an improved BMSE at steady state. This is true whenever the adaptive filter order is smaller than the order of the data-generating filter. When these data are correlated, the above phenomenon seems to be more pronounced (case 2 of our simulations). Additional deterioration in the steady state performance will occur in the constrained algorithms when different step sizes are chosen for each frequency bin (namely, $\mu_F \neq \mu_0 I$). Our conclusion is that, from the point of view of our discussion here, the choice of the unconstrained algorithm is preferable since it could result in the best steady state performance, and the μ_F could be chosen so as to get the best convergence rate without affecting the steady state performance (we emphasize again that our reference to steady state BMSE does not include the additional BMSE due to the stochastic nature of the algorithm, the analysis of which is beyond the scope of this correspondence).

We would also like to refer to some of the comments made in [6]. There, the claim was that the constrained algorithms converge faster than the unconstrained. However, the modification proposed there is hardly a constrained algorithm (the resulting G is of rank 31), and we note that in the algorithm, (9), the product of a nonsingular G by a diagonal μ_F, can be replaced by a nondiagonal μ_F. Hence, the results in [6] seem to indicate that for the best convergence rate, one should consider a nondiagonal μ_F in the unconstrained algorithm.

References

[1] M. Dentino, J. McCool, and B. Widrow, "Adaptive filtering in the frequency domain," *Proc. IEEE*, vol. 66, pp. 1658–1659, Dec. 1978.

[2] E. R. Ferrara, "Fast implementation of LMS adaptive filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 474–475, Aug. 1980.

[3] D. Mansour and A. H. Gray, Jr., "Unconstrained frequency-domain adaptive filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 726–734, Oct. 1982.

[4] G. A. Clark, S. R. Parker, and S. K. Mitra, "A unified approach to time- and frequency-domain realization for FIR adaptive digital filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1073–1083, Oct. 1983.

[5] N. J. Bershad and P. L. Feintuch, "A normalized frequency domain LMS adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 452–461, June 1986.

[6] P. C. W. Sommen, P. J. Van Gerwen, H. J. Kotmas, and A. J. E. M. Janssen, "Convergence analysis of a frequency-domain adaptive filter with exponential power averaging and generalized window function," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 788–789, July 1987.

[7] G. A. Clark, S. K. Mitra, and S. R. Parker, "Block implementation of adaptive digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 584–592, June 1981.

[8] A. Feuer, "Performance analysis of the block least mean square algorithm," *IEEE Trans. Circuits and Systems*, vol. CAS-32, pp. 960–963, Sept. 1985.

[9] P. M. Grant and A. Alvarez, "Frequency domain adaptive filter designs," presented at the Saraga Colloq. Electron. Filters, London, U.K., 1984.

One-Bit Spectral-Correlation Algorithms

W. A. Gardner and R. S. Roberts

***Abstract*—A technique that greatly simplifies the computational complexity of digital cyclic spectral analysis algorithms is presented. The technique, which is based on Bussgang's theorem, replaces complex multiplications in spectral correlation operations with simple sign-change and data-multiplexing operations. Moreover, the technique is applicable to both time- and frequency-averaging algorithms. A simulation study that compares the computed results obtained using the new technique with results from standard time- and frequency-averaging algorithms shows that the new technique is very promising, particularly for frequency-averaging algorithms.**

I. Introduction

Most modulated signals encountered in communications and telemetry systems exhibit cyclostationarity. A fundamental tool in the study and exploitation of cyclostationarity is the cyclic spectrum analyzer; that is, an instrument that computes the cyclic spectrum from signal measurements and graphs this function (its magnitude and/or phase) as the height of a surface above the bifrequency

Manuscript received May 25, 1991; revised April 20, 1992. This work was supported in part by a grant from ESL, Inc., with partial matching support from the California State MICRO Program.

W. A. Gardner is with the Signal Processing Group, Department of Electrical and Computer Engineering, University of California, Davis, CA 95616.

R. S. Roberts was with the Signal Processing Group, Department of Electrical and Computer Engineering, University of California, Davis, CA 95616. He is now with the Los Alamos National Laboratory, Los Alamos, NM 87545.

IEEE Log Number 9203368.