Many conversion techniques operate on generalized moduli with the sacrifice of some speed and efficiency. The restrictions on the moduli values for this approach may not always be a significant drawback since the set of moduli, $(2^k - 1, 2^k, 2^k + 1)$, possesses certain efficiencies both in terms of simplicity of implementation and representation. Additionally this set is supported by recent results on specialized division by values of the form $2^n \pm 1$ [12], [13] that have been motivated by binary decimal conversion and decimal floating-point shifting [14]–[16].

References

[1] G. Alia and Martinelli, "A VLSI a algorithm for direct and reverse conversion from weighted binary number system to residue number system," *IEEE Trans. Circuits Systs.*, vol. CAS-31, pp. 1033–1039, Dec. 1984.

[2] K. M. Ibrahim and S. Saloum, "An efficient residue to binary converter design," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1156–1158, Sept. 1988.

[3] A. Shenoy and R. Kumaresan, " Residue to binary conversion for RNS arithmetic using only modular look-up tables," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1158–1162, Sept. 1988.

[4] R. M. Capocelli and R. Giancarlo, "Efficient VLSI networks for converting an integer from binary system to residue number system and vice versa," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1425–1431, Sept. 1988.

[5] S. Andraos and H. Ahmed, "A new efficient memoryless residue to binary converter," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1441–1444, Sept. 1988.

[6] S. Meehan, S. O'Neil, and J. Vaccaro, "A universal input and output RNS converter," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 799–803, June 1990.

[7] K. Elleithy and M. Bayoumi, "Fast and flexible architectures for RNS arithmetic decoding," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 226–234, Apr. 1992.

[8] F. Taylor and A. Ramnarayanan, "An efficient residue-to-decimal converter," *IEEE Trans. Circuits Syst.*, vol. 28, pp. 1164–1169, Dec. 1981.

[9] F. Taylor, "Residue arithmetic: A tutorial with examples," *IEEE Computer*, pp. 50–62, May 1984.

[10] A. Ramnarayanan, "Practical realization of $\mathrm{mod}\, p, p$ prime multiplier," *Electron. Lett.*, vol. 16, pp. 466–467, June 1980.

[11] A. B. Premkumar, "An RNS to binary converter in $2n+1, 2n, 2n-1$ moduli set," *IEEE Trans. Circuits Syst.*, vol. 39, pp. 480–482, July 1992.

[12] F. Petry and P. Srinivasan, "Division techniques by integers of the form $2^n \pm 1$," *Int. J. Electron.*, vol. 75, no. 5, pp. 659–670, 1993.

[13] P. Srinivasan and F. Petry, "Constant division algorithms," in *IEE Proc.—Comput. Digital Tech.*, 1994, vol. 141, no. 6, pp. 334–340.

[14] G. Bohlender, "Decimal floating-point arithmetic in binary representation," in *Computer Arithmetic, Scientific Computation and Mathematical*, E. Kaucher *et al.* Eds. Frankfurt, Germany: J. C. Balzer AG, 1991, pp. 13–27.

[15] P. Johnstone and F. Petry, "Design and analysis of nonbinary radix floating point representations," *Comput. Elect. Eng.*, vol. 20, no. 1, pp. 39–50, 1994.

[16] ——, "Rational number approximation in higher radix floating point systems," *Computers and Math. Applicat.*, vol. 25, no. 5, pp. 103–108, 1993.

Saving Complexity of Modified Filtered-X-LMS and Delayed Update LMS Algorithms

Markus Rupp

***Abstract*—In some applications, like in active noise control, the error signal cannot be obtained directly but only a filtered version of it. A gradient adaptive algorithm that solves the identification problem under this condition is the well known Filtered-x Least-Mean-Squares (FxLMS) algorithm. If only one coefficient of this error-filter function is nonzero, a special case of the FxLMS algorithm, the Delayed-update Least-Mean-Squares (DLMS) algorithm is obtained. The drawback of these algorithms is the increased dynamic order which, in turn, decreases the convergence rate. Recently, some modifications for these algorithms have been proposed, overcoming the drawbacks by additional computations of the same filter order as the filter length M. In this contribution, an improvement is shown yielding reduced complexity if the error path filter order P is much smaller than the filter order M, which is the case for many applications. Especially for the DLMS algorithm a strong saving can be obtained.**

***Index Terms*—Filtered-x LMS algorithm, delayed update LMS algorithm, reduced complexity.**

I. Introduction

Since the development of the Filtered-x Least-Mean-Squares (FxLMS) algorithm by Morgan in 1980 [1] and independently also by Widrow *et al*. [2] in 1981 the algorithm has been applied successfully in many situations. In contrast to the famed LMS algorithm, the error signal is now not directly available, but only a filtered version of it. The undesired algorithmic effect of slowing down the convergence rate can be removed by a modification that has been proposed recently by Bjarnason [3], and also Kim and Kim [4], and by Rupp and Frenzel for the case of the LMS algorithm with delayed update (DLMS) [5]. This modification assures the robust behavior of the LMS algorithm, however, for the additional cost of M operations, M being the number of the filter coefficients that are to be estimated. Many applications in noise control can be modeled by a pure delay in the error path (see [4], [6], [7]) or very few coefficients [8], [9]. In [9] measurements on a duct with 2 m length and 0.4 m $\times$ 0.4 m cross section are described where 15 coefficients for the error path were sufficient for running the FxLMS algorithm satisfactorily. Since the algorithm is very insensitive to estimation errors in the error path function, very often a shortened filter version is sufficient [7]. This effect of concentrating the error energy around a certain delay P is not only typical for applications in the noise control field but also in VLSI implementations of the LMS algorithm where a pure delay of the error occurs due to the synchronization of the various cells [10]. (If for example a tree structure is used the delay is given by $P = \log_2 M$ and thus $P \ll M$ for large enough M.) Due to the several inner vector products in the algorithm, it is possible to calculate the coefficient updates alternatively, thus saving complexity. The contribution of this correspondence is to propose an alternative method for calculating the coefficient update for the modified forms of FxLMS and DLMS algorithms that reduce the complexity of the algorithms if the error path filter length, or delay $P \ll M$.

Manuscript received December 1, 1993; revised April 29, 1996. This paper was recommended by Associate Editor B. Kim.

The author is with the Wireless Technology Research Laboratory, Lucent Technologies, Holmdel, NJ 07733 USA.

Publisher Item Identifier S 1057-7130(97)01161-0.

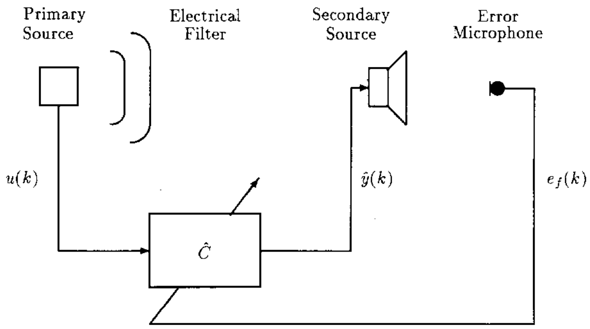

Fig. 1. Active noise cancellation in a duct.

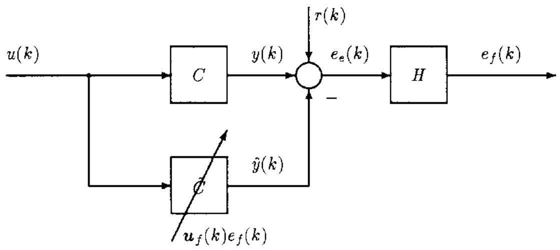

Fig. 2. Identification of an unknown plant C using the FxLMS algorithm.

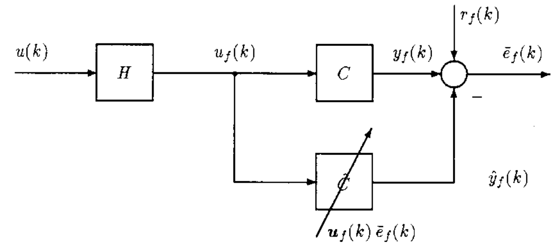

Fig. 3. Identification of an unknown plant C using the modified FxLMS algorithm.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE MODIFIED FxLMS USING (3)

Operation	Mult/Add	Storage place
$\hat{y}(k) = \hat{\mathbf{c}}^T(k)\mathbf{u}(k)$	M	$2M$
$\mathbf{u}_f(k)$	$P+1$	$M+P$
$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)\bar{e}_f(k)\mathbf{u}_f(k)$	$M+1$	
$\hat{y}(k-i) = \hat{\mathbf{c}}^T(k-i)\mathbf{u}(k-i)$ for $i = 0..P$		$P+1$
$\hat{C}(k)\,[H\,[u(k)]] = \hat{\mathbf{c}}^T(k)\mathbf{u}_f(k)$	M	1
$H\,[\hat{y}(k)] = H\left[\hat{C}(k)\,[u(k)]\right]$	$P+1$	
$e_f(k) + H\left[\hat{C}(k)\,[u(k)]\right] - \hat{C}(k)\,[H\,[u(k)]]$	2	
$\sum$:	$3M+2P+5$	$3M+2P+2$

II. THE FxLMS ALGORITHM

Fig. 1 depicts the typical situation for active noise control: a (noise) signal from the primary source is to be cancelled at the location of the error-microphone. The primary source signal is available either directly or by a second microphone. The idea is to find the optimal filter $\hat{C}$ such that the primary source signal filtered by this optimal filter and sent out at the loudspeaker (secondary source) interferes with the primary source signal at the location of the error-microphone in such a way that the resulting signal is diminished. The error signal is therefore fed back to the adaptive algorithm as a control signal in order to find the optimal parameters. Fig. 2 shows an equivalent model of the above situation. The transfer function from primary source to the secondary source is denoted by the unknown plant C. If the filter coefficients of $\hat{C}$ are chosen optimally, the filter output $\hat{y}(k)$ cancels the output of the system C. Additional noise $r(k)$ has been added in order to describe model errors and possible other sources. The resulting error signal $e_e(k)$, however, cannot be obtained directly but only a filtered version $e_f(k)$ of it due to the construction and the microphone. In order to obtain convenient expressions the filters are described by a linear operator notation

$$H(q^{-1}) = \sum_{i=0}^{P} h_i q^{-i},$$
$$C(q^{-1}) = \sum_{i=0}^{M-1} c_i q^{-i},$$
$$\hat{C}(k, q^{-1}) = \sum_{i=0}^{M-1} \hat{c}_i(k) q^{-i}$$

where q^{-1} denotes the unit delay operator. Applying these filter operators to a sequence, say $y(k)$, leads for example to

$$H(q^{-1})[y(k)] \triangleq H[y(k)] \triangleq \sum_{i=0}^{P} h_i y(k-i),$$

where we used the shorter notation $H[y(k)]$ for denoting the filter operation. By using this notation, the filtered error $e_f(k)$ can then be written as

$$\begin{aligned} e_f(k) &= H[e_e(k)] = H[y(k) + r(k) - \hat{y}(k)] \\ &= H[C[u(k)] + r(k) - \hat{C}(k)[u(k)]] \\ &= H[C[u(k)]] + r_f(k) - H[\hat{C}(k)[u(k)]]. \end{aligned}$$

Although the operator technique simplifies descriptions, vector notations are used as well

$$\begin{aligned} \mathbf{u}^T(k) &= [u(k), u(k-1), \ldots, u(k-M+1)], \\ \mathbf{u}_f^T(k) &= [u_f(k), u_f(k-1), \ldots, u_f(k-M+1)], \\ u_f(k) &= H[u(k)], \\ \mathbf{h}^T &= [h_0, h_1, \ldots, h_P], \\ \mathbf{c}^T &= [c_0, c_1, \ldots, c_{M-1}], \\ \hat{\mathbf{c}}^T(k) &= [\hat{c}_0(k), \hat{c}_1(k), \ldots, \hat{c}_{M-1}(k)]. \end{aligned}$$

With these definitions, the update of the FxLMS algorithm [11] can be given to

$$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k) e_f(k) \mathbf{u}_f(k). \quad (1)$$

In (1), the coefficients of the error path H are assumed to be known. The analysis of the update recursion (see [3]) (1) unfortunately leads to a dynamic system of higher order and therefore, it is difficult to calculate a stable range for the step-size $\mu(k)$, to give suitable normalizations, or even to find an optimal value in order to obtain maximal convergence rate (see also [12] and [13]).

III. THE MODIFIED FxLMS ALGORITHM

In [3] and [4] a modification of the FxLMS algorithm is given in such a way that the modified algorithm is again a system of order one and behaves basically like an LMS algorithm. A detailed stability analysis can be found in [4]. The modified system is depicted in Fig. 3, where the filter in the error path has been moved backward to the driving sequence $u(k)$. Although both systems are equivalent for a time invariant plant C, this is not longer true if $\hat{C}(k)$ is time-varying. In order to compensate for this difference a new error signal

TABLE II
COMPUTATIONAL COMPLEXITY OF THE MODIFIED FxLMS ALGORITHM USING (5)

Operation	Mult/Add	Storage place
$\hat{y}(k) = \hat{\mathbf{c}}^T(k)\mathbf{u}(k)$	M	$2M$
$\mathbf{u}_f(k)$	$P+1$	$M+P$
$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)\bar{e}_f(k)\mathbf{u}_f(k)$	$M+1$	
$\mu(k-j)\bar{e}_f(k-j)$ for $j = 0..P$		$P+1$
$\mathbf{u}_f^T(k-j)\mathbf{u}(k-i)$ for $i = 1..P, j = 1..i$	$2P$	$\frac{P(P-1)}{2}$
$\bar{e}_f(k) = e_f(k) + \sum_{i=0}^{P} h_i$		
$\sum_{j=1}^{i} \mu(k-j)\bar{e}_f(k-j)\mathbf{u}_f^T(k-j)\mathbf{u}(k-i)$	$\frac{P(P+1)}{2}$	
$\sum$:	$2M + 0.5P^2 + 3.5P + 2$	$3M + 0.5P^2 + 1.5P + 1$

$\bar{e}_f(k)$ has to be used. The new error signal $\bar{e}_f(k)$ can be given as

$$\begin{aligned}\bar{e}_f(k) &= C[H[u(k)]] + H[r(k)] - \hat{C}(k)[H[u(k)]] \\ &= C[u_f(k)] + r_f(k) - \hat{C}(k)[u_f(k)].\end{aligned}$$

Correspondingly, the update of the modified FxLMS algorithm can be written as

$$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)\bar{e}_f(k)\mathbf{u}_f(k) \tag{2}$$

a form that basically behaves like an LMS algorithm but with the filtered input sequence $u_f(k)$. However, (2) only makes sense, if the modified error signal can be calculated by known signals. A procedure, that calculates the modified error $\bar{e}_f(k)$ only by using observable variables, is shown next. Note that $\bar{e}_f(k)$ can be written in terms of the available error signal $e_f(k)$

$$\begin{aligned}\bar{e}_f(k) &= e_f(k) + H[\hat{C}(k)[u(k)]] - \hat{C}(k)[H[u(k)]] & (3)\\ &= e_f(k) + \sum_{i=0}^{P} h_i \hat{\mathbf{c}}^T(k-i)\mathbf{u}(k-i) - \sum_{i=0}^{P} h_i \hat{\mathbf{c}}^T(k)\mathbf{u}(k-i) \\ &= e_f(k) + \sum_{i=0}^{P} h_i(\hat{\mathbf{c}}^T(k-i) - \hat{\mathbf{c}}^T(k))\mathbf{u}(k-i) & (4)\\ &= e_f(k) + \sum_{i=1}^{P} h_i \sum_{j=1}^{i} \mu(k-j)\bar{e}_f(k-j)\mathbf{u}_f^T(k-j)\mathbf{u}(k-i). & (5)\end{aligned}$$

The first line (3) already shows how the modified error can be calculated. If, however, the update (2) is substituted into (4), the last line (5) can be obtained and complexity can be reduced. The computational and storage complexity for the modified FxLMS algorithm using (3) and (5), respectively, have been specified in the Tables I and II. By using (5) instead of (3) the complexity is reduced from $\approx 3M$ to $\approx 2M + \frac{P^2}{2}$, whereas the necessary storage remains basically $3M$ in both cases as long as P is not too large. Thus, if the filter order $P^2 \ll M$, the complexity is reduced to a $2M$-algorithm.

IV. SPECIAL CASE : THE MODIFIED DLMS ALGORITHM

The DLMS algorithm can be seen as a special case of the FxLMS algorithm, where only one coefficient of the filter path function H is nonzero

$$\mathbf{h}^T = (0, 0, \ldots, 0, h_P). \tag{6}$$

If this special filter is used, the signals can be rewritten as

$$\begin{aligned}\mathbf{u}_f(k) &= h_P \mathbf{u}(k-P) & (7)\\ e_f(k) &= h_P(C[u(k-P)] + r(k-P) \\ &\quad - \hat{C}(k-P)[u(k-P)]), & (8)\\ \bar{e}_f(k) &= h_P(C[u(k-P)] + r(k-P) - \hat{C}(k)[u(k-P)]). & (9)\end{aligned}$$

The update equation now reads

$$\hat{\mathbf{c}}(k+1) = \hat{\mathbf{c}}(k) + \mu(k)h_P\bar{e}_f(k)\mathbf{u}(k-P) \tag{10}$$

and the modified error $\bar{e}_f(k)$ reads as

$$\begin{aligned}\bar{e}_f(k) &= e_f(k) + h_p\hat{C}(k-P)[u(k-P)] \\ &\quad - h_p\hat{C}(k)[u(k-P)] & (11)\\ &= e_f(k) + h_P^2 \sum_{j=1}^{P} \mu(k-j)\bar{e}_f(k-j) \\ &\quad \times \mathbf{u}^T(k-P-j)\mathbf{u}(k-P). & (12)\end{aligned}$$

Equation (12) needs only $O(P)$ operations whereas (11) needs $O(M)$. If $M > 2P$ this modification has less complexity and for $P \ll M$ it results in a small additional complexity and a $2M$ algorithm is obtained again.

V. CONCLUSION

Modifications of the well-known FxLMS and DLMS algorithms are known to remedy the drawback of these algorithms, i.e., the order of the corresponding dynamic system is reduced. The cost of this improvement, however, is an additional complexity of order $O(M)$, where M is the length of the adaptive filter. In this paper a method has been presented in order to reduce complexity, while the behavior of the algorithms remains unchanged. If the filter order P of the error path is small in comparison to the filter length M, the complexity of the algorithms is approximately $2M$.

ACKNOWLEDGMENT

The author would like to thank the reviewers for many useful comments and suggestions. Also, A. Mathur and A. Keerthi for their careful reading and valuable suggestions that improved the quality of this brief.

References

[1] D. R. Morgan, “An analysis of multiple correlation cancellation loops with a filter in the auxiliary path,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 454–467, Aug. 1980.

[2] B. Widrow, D. Shur, and S. Shaffer, “On adaptive inverse control,” in *15. Asilomar Conf. Circuits, Syst. Components*, 1981, pp. 185–189.

[3] E. Bjarnason, “Active noise cancellation using a modified form of the filtered-XLMS algorithm,” in *Proc. Eusipco Signal Processing V, Brussels*, 1992.

[4] I. Kim, H. Na, K. Kim, and Y. Park, “Constraint filtered-x and filtered-u algorithms for the active control of noise in a duct,” *J. Acoust. Soc. Amer.*, vol. 95, no. 6, pp. 3397–3389, June 1994.

[5] M. Rupp and R. Frenzel, “The behavior of LMS and NLMS algorithms with delayed coefficient update in the presence of spherically invariant processes,” *IEEE Trans. Signal Processing*, vol. 42, pp. 668–672, Mar. 1994.

[6] S. D. Snyder and C. H. Hansen, “ Design considerations for active noise control systems implementing the multiple input, multiple output LMS algorithm, ” *J. Sound Vibration*, vol. 159, no. 1, pp. 157–174, 1992.

[7] M. Rupp, “An analog-digital echo canceller for hybrids,” in *Proc. ISCAS*, Chicago, IL, May 1993.

[8] A. C. Orgren, S. Dasgupta, C. E. Rohrs, and N. R. Malik, “Noise cancellation with improved residuals,” *IEEE Trans. Signal Processing*, vol. 39, pp. 2629–2639, Dec. 1991.

[9] E. Bjarnason, “Analysis and implementation of gradient algorithms for active noise control,” Ph.D. dissertation, TH Darmstadt, Germany, 1995.

[10] G. Long, F. Ling, and J. A. Proakis, “The LMS algorithm with delayed coefficient adaptation,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1397–1405, Sept. 1989.

[11] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, Signal Processing Series, 1985.

[12] M. Rupp and A. H. Sayed, “Two variants of the FxLMS algorithm,” *1995 Wkshp. Applicat.Signal Processing Audio and Acoustics*, New Paltz, NY, Oct. 1995.

[13] ——, “Modified FxLMS algorithms with improved convergence performance,” in *Proc. Asilomar Conf.*, Oct.–Nov. 1995.