PERFORMANCE COMPARISON OF TWO FAST ALGORITHMS FOR ACTIVE CONTROL

Luis Vicente, Enrique Masgrau

Dept. of Electronics Engineering and Communication, University of Zaragoza
E50015 Zaragoza, SPAIN
E-mail: *lvicente@posta.unizar.es*

INTRODUCTION

Convergence of the Filtered-x Least Mean Square (FxLMS) algorithm is relatively slow and signal-dependent, since it is determined by the statistical properties of the filtered-x signal. When working in nonstationary environments slow convergence is a critical problem, since it means a reduction in the system cancellation capability. Applying some pre-processing to the reference signal and/or the filtered-x signal it is possible to speed up convergence. In the present work, we compare two systems that can achieve faster convergence than FxLMS at the expense of computational complexity.

In section I, we examine the convergence problems inherent in FxLMS systems and introduce a generic orthogonalizing system scheme to which the two systems presented fit. The first of these systems is the Filtered-x Gradient Adaptive Lattice (FxGAL), described in section II, which uses an Adaptive Lattice Predictor (ALP) to decompose the signal into orthogonal components. The other system, described in section III, is the ALE + FxLMS algorithm, where an Adaptive Line Enhancer (ALE) with a decorrelation delay of one sample is used to whiten the filtered-x signal. Extensions of both systems to the multiple channel case, where cross-correlation between the different reference signals slows down adaptation, are addressed in section IV. Comparative results between both systems and with the classical FxLMS are shown in section V.

Throughout the paper the notation proposed in [1] for Active Noise Control (ANC) systems will be used. Vectors and matrices are bold-typed.

I. FXLMS AND ORTHOGONALIZING SYSTEMS

The Filtered-x Least Mean Square (FxLMS) algorithm [1, 2] is the most widely used in the context of adaptive active control, due to its simplicity as well as robustness. However, the main drawback of this algorithm is its relatively slow and signal-dependent convergence, which is determined by the eigenvalue spread of the underlying correlation matrix of the input signal.

Due to the unavoidable presence of the secondary path transfer function, $S(z)$, after the adaptive filter, the input and adaptation signals for $W(z)$ in the FxLMS system are not the same (see Figure 1). A model of the secondary path, $\hat{S}(z)$, is required to obtain the filtered reference, $x'[n]$, to use for the adaptation process:

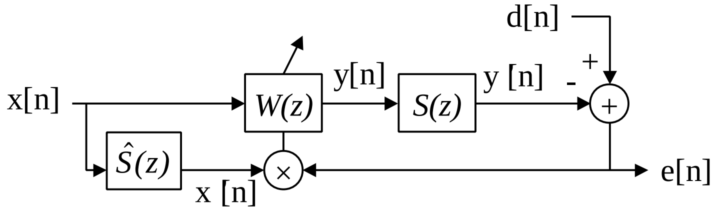


FIGURE 1. *Block diagram of Filtered-x Least Mean Square system.*

$$\mathbf{w}[n+1] = \mathbf{w}[n] + \mu e[n]\mathbf{x}'[n] \tag{1}$$

Under the assumptions of stationarity, slow convergence and perfect secondary path modelling, the statistical analysis of the weight vector convergence for the FxLMS system yields the well-known difference equation:

$$E\{\mathbf{w}[n]\} = (\mathbf{I} - \mu\mathbf{R})E\{\mathbf{w}[n-1]\} + \mu\mathbf{R}\mathbf{w}^* \tag{2}$$

where $\mathbf{R} = E\{\mathbf{x}'[n]\mathbf{x}'^{\mathrm{T}}[n]\}$ is the correlation matrix of the filtered-x signal, $\mathrm{x}'[n]$, and $\mathbf{w}^*$ is the optimum weight vector. This equation can be solved (made non-recursive) by translating the weight vector coordinate system. Using $\mathbf{w}'[n] = \mathbf{w}[n] - \mathbf{w}^*$, (2) becomes:

$$E\{\mathbf{w}'[n]\} = (\mathbf{I} - \mu\mathbf{R})E\{\mathbf{w}'[n-1]\} = (\mathbf{I} - \mu\mathbf{R})^n E\{\mathbf{w}'[0]\} \tag{3}$$

In general, the correlation matrix is not diagonal and the solutions for the various components of the weight vector are cross-coupled. In this case, the same adaptation step size, μ, should be used for the adaptation process of every filter coefficient. Every convergence mode will have its own time constant, inversely proportional to the corresponding correlation matrix eigenvalue, λ_l:

$$\tau_l \approx \frac{1}{\mu\lambda_l}, \qquad l = 1,\cdots,L \tag{4}$$

Using an adaptation step size, μ, normalized to the power of the filtered-x signal (Normalized FxLMS) can be seen as a first attempt to make convergence independent from the input signal. Nevertheless, there will still be faster and slower converging modes since the adaptation step size used is unique. Thus, the eigenvalue spread of the correlation matrix will determine the system convergence.

The fastest convergence can be obtained when the correlation matrix is diagonal, that is to say, when convergence modes are not coupled:

$$\mathbf{R} = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_L \end{bmatrix} = \mathbf{\Lambda} \tag{5}$$

In this case, the whole adaptive filter of order L can be seen as L adaptive filters of just one coefficient converging independently:

$$E\{\mathrm{w}_l[n]\} = \mathrm{w}_l^* + (1 - m_l l_l)^n \{E\{\mathrm{w}_l[0]\} - \mathrm{w}_l^*\} \qquad\qquad l = 1, \cdots, L \tag{6}$$

Therefore, by properly choosing the adaptation step sizes of these independent systems, m_l, it is possible to make every mode converge at the same speed, what is equivalent to minimizing the eigenvalue spread.

Diagonalizing a signal correlation matrix, orthogonalizing the signal and decorrelating the signal will be used as equivalent terms, since when the signal components are decorrelated, $\mathrm{R}_{j-i} = E\{\mathrm{x}'[n-i]\mathrm{x}'[n-j]\} = 0$, so, they are orthogonal to each other and the corresponding correlation matrix is diagonal.

There are various means of diagonalizing the autocorrelation matrix, each of which gives way to a different adaptive system. In this paper we study and compare two of such systems: the Filtered-x Gradient Adaptive Lattice (FxGAL) [3], described in section II, and the Adaptive Line Enhancer + FxLMS (ALE + FxLMS) [4], described in section III.

Both systems have an adaptive finite impulse response (FIR) filter, $H_D(z)$, whose input signal is the filtered reference, $\mathrm{x}'[n]$, and whose output is the orthogonalized version of the input, $\mathrm{x}'_{\mathrm{D}}[n]$. That is to say, $\mathrm{x}'_{\mathrm{D}}[n] = \mathbf{h}_D^{\mathrm{T}}[n]\mathbf{x}'[n]$ and $\mathbf{R}_D = E\{\mathbf{x}'_D[n]\mathbf{x}'^{\mathrm{T}}_D[n]\} = \Lambda_D$ is diagonal. The filter $H_D(z)$ is adaptive to track possible changes in the input statistics.

The decorrelated signal, $\mathrm{x}'_{\mathrm{D}}[n]$, is the one that we want to use in the adaptation of $W(z)$. Therefore, for the system to work properly the reference signal, $\mathrm{x}[n]$, needs also to be filtered by the same filter before feeding $W(z)$ (see block diagram in Figure 2.b). This can be understood when we consider the alternative system depicted in Figure 2.a. In this case, the orthogonalizing system is a pre-processing stage completely independent from the FxLMS system, being $\mathrm{x}_D[n]$ the reference input for the FxLMS. No convergence problem arises in any of the two adaptive systems, orthogonalizing and FxLMS, since they are completely independent. However, the

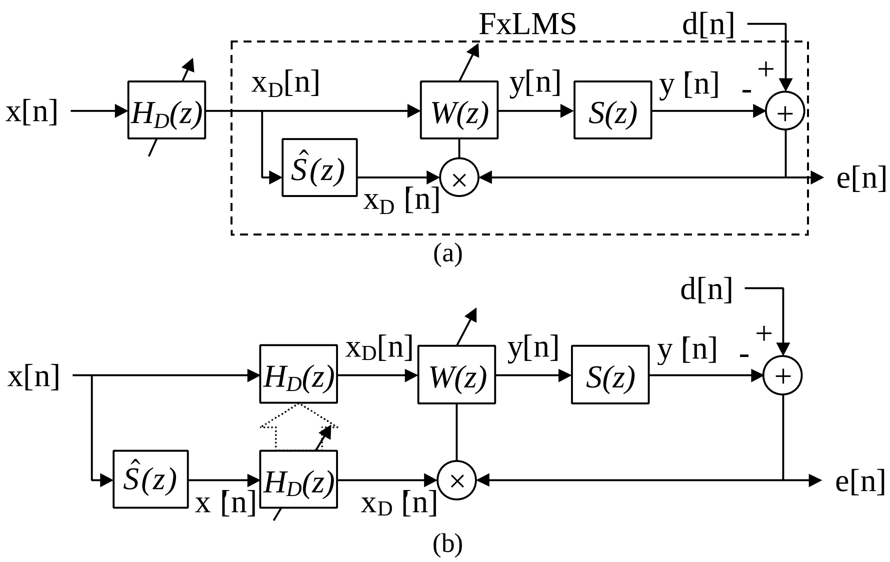


FIGURE 2. *Block diagrams of generic FxLMS systems with orthogonalizing stage: (a) Orthogonalizing filter as pre-processing stage. (b) Orthogonalizing filter as embedded stage.*

whole system does not really diagonalize the correlation matrix that determines the FxLMS convergence. This is so because the input to the orthogonalizing system is the reference signal itself, $\mathrm{x}[n]$, not the filtered reference, $\mathrm{x}'[n]$, and subsequent filtering by the secondary path model of the orthogonalized signal might correlate it again. Only in the particular case where the secondary path is a pure delay, this system configuration would diagonalize the FxLMS convergence correlation matrix.

Therefore, it is important to orthogonalize the filtered reference signal directly. To achieve this objective we can commute $H_D(z)$ and $\hat{S}(z)$ filters, under the assumption of slow convergence in the adaptation process of $H_D(z)$ (time-invariant approximation). Furthermore, the adaptation of $H_D(z)$ should be much slower than that of $W(z)$, so as not to cause coupling between them. Commuting $H_D(z)$ and $\hat{S}(z)$ correctly causes two identical $H_D(z)$ blocks to appear in each branch after the junction point (see Figure 2.b). The one that filters the reference signal (upper branch) is just a copy of the adaptive one (lower branch) and we will refer to it as the "slave filter". Since the block $H_D(z)$ with $\mathrm{x}'[n]$ as input is the one that is adaptive, this system will correctly diagonalize the correlation matrix of the FxLMS system. Therefore, the system configuration with embedded orthogonalizing filter is the one that we use for both systems, FxGAL and ALE + FxLMS.

II. FXGAL SYSTEM

The decorrelating system, $H_D(z)$, used in the FxGAL algorithm is an Adaptive Lattice Predictor (ALP) [5, 6]. It is a modular structure, in such a way that the predictor of order L consists of L – 1 identical cascaded stages.

The scheme of an ALP of order L = 4 is depicted in Figure 3. Modularity is evident in this figure. The inputs to the lattice stage l are the forward and backward prediction errors of the (l–1)-th order predictor, $\mathrm{f}_{l-1}[n]$ and $\mathrm{b}_{l-1}[n-1]$, and the outputs are the forward and backward prediction errors of the l-th order predictor, $\mathrm{f}_l[n]$ and $\mathrm{b}_l[n]$. The ALP system itself is an adaptive filter. A gradient search or steepest descent method is used to adjust the filter coefficients, $\{\kappa_l[n]\}$, independently at each stage, so as to minimize the mean square of the sum of forward and backward predictor errors at the output of that stage:

$$\kappa_l[n+1] = \kappa_l[n] - \frac{a}{\mathrm{P}_l[n]}\{\mathrm{f}_{l-1}[n]\mathrm{b}_l[n] + \mathrm{b}_{l-1}[n-1]\mathrm{f}_l[n]\} \tag{7}$$

where,

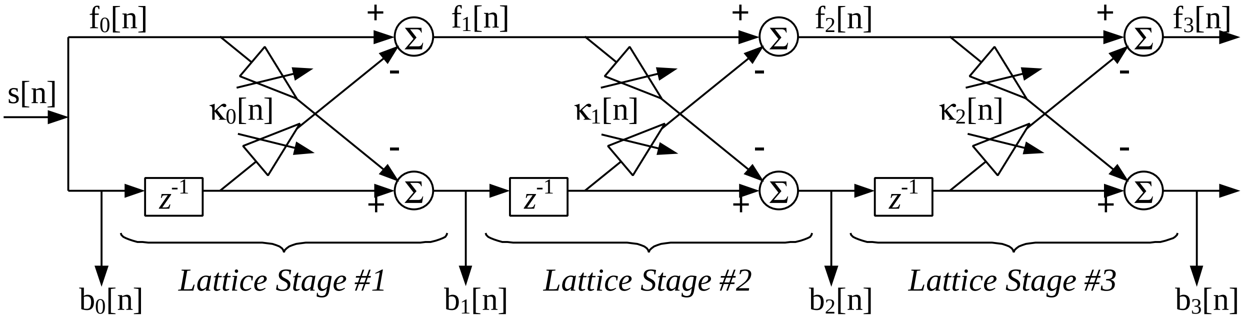


FIGURE 3. *Adaptive Lattice Predictor of order L = 4.*

$$\mathrm{P}_l[n]=(1-a)\mathrm{P}_l[n-1]+a\left\{\mathrm{f}_{l-1}^2[n]+\mathrm{b}_{l-1}^2[n-1]\right\} \tag{8}$$

is an adaptive estimation of the input power for the stage l. This adaptation algorithm is known as the Gradient Adaptive Lattice (GAL) algorithm.

The fundamental property of the ALP system we make use of is the orthogonality of the backward prediction errors. In every moment, the sequence of backward prediction errors, $\mathbf{b}[n]=\{\mathrm{b}_0[n],\mathrm{b}_1[n],\cdots,\mathrm{b}_{L-1}[n]\}$, are mutually uncorrelated and are a transformation of the input sequence, $\mathbf{s}[n]=\{\mathrm{s}[n],\mathrm{s}[n-1],\cdots,\mathrm{s}[n-L+1]\}$, without loss of information.

Therefore, in the FxGAL case, we transform the whole vector $\mathbf{x}'[n]$ into another vector, $\mathbf{x}'_{\mathrm{D}}[n]$, formed by the orthogonal components of it. This transformation needs also to be made with the reference signal vector, $\mathbf{x}[n]$, by means of a slave lattice filter. This filter is just a lattice structure equal to the one depicted in Figure 3, the coefficients $\{\kappa_l[n]\}$ not being adaptive but an instantaneous copy of the ones calculated in the ALP.

A lattice ANC system similar to FxGAL is proposed in [1]. However, the system proposed orthogonalizes the reference signal, $\mathbf{x}[n]$, and filters the filtered reference, $\mathbf{x}'[n]$, with a slave filter. As we said before, this system will only diagonalize the correlation matrix correctly in the particular case that the secondary path is a pure delay.

III. ALE + FXLMS SYSTEM

The correlation matrix $\mathbf{R}=E\{\mathbf{x}'[n]\mathbf{x}'^{\mathrm{T}}[n]\}$ can also be diagonalized by whitening the filtered-x signal, $\mathrm{x}'[n]$, since for a white signal, $\mathrm{R}_j=E\{\mathrm{v}[n]\mathrm{v}[n-j]\}=0$ for $j\neq 0$.

The Adaptive Line Enhancer (ALE) is a well-known adaptive structure [7, 8] which is able to whiten the signal when the decorrelation delay is just one sample, Δ=1 (Figure 4). It is a prediction error filter, whose transfer function is $H(z)=1-z^{-1}Pr(z)$, of order Q+1, being Q the order of $Pr(z)$. The filter $Pr(z)$ is a prediction filter, since the "desired" signal at the output of it is an anticipated version of the input signal. The output signal of the ALE system is the prediction error, that is the whitened version of the input signal. Any adaptive algorithm can be used for the prediction filter, $Pr(z)$. In this work, Least Mean Squares (LMS) [8, 9] and Gradient Adaptive Lattice (GAL) algorithms are considered. In the GAL case, the output signal would be the forward prediction error of the last lattice stage.

In the whitening process, the periodic information of the signal is lost. Therefore, whitening is only suitable for broadband active control, unless the periodic part of the signal is extracted, previously to the operation of the whitening system, and processed in parallel by

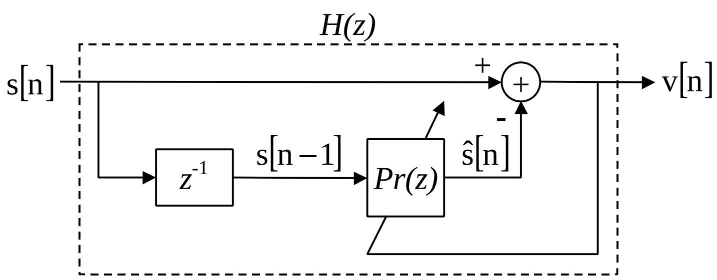


FIGURE 4. *Adaptive Line Enhancer with unit decorrelation delay: signal whitener.*

another system.

IV. MULTICHANNEL ORTHOGONALIZING SYSTEMS

In this section, we consider the multichannel extensions of the orthogonalizing systems previously introduced.

The correlation matrix that determines convergence of the multichannel FxLMS system (also known as Multiple Error LMS (MELMS), [2]) is $\mathbf{R} = E\{\mathbf{X}'[n]\mathbf{X}'^{\mathrm{T}}[n]\}$, but with $\mathbf{X}'[n]$ now being a $JKL \times M$ matrix:

$$\mathbf{X}'[n] = \begin{bmatrix} x'_{1,1,1}[n] & x'_{1,1,2}[n] & \cdots & x'_{1,1,M}[n] \\ x'_{1,1,1}[n-1] & x'_{1,1,2}[n-1] & \cdots & x'_{1,1,M}[n-1] \\ \vdots & \vdots & \ddots & \vdots \\ x'_{1,1,1}[n-L+1] & x'_{1,1,2}[n-L+1] & \cdots & x'_{1,1,M}[n-L+1] \\ x'_{2,1,1}[n] & x'_{2,1,2}[n] & \cdots & x'_{2,1,M}[n] \\ \vdots & \vdots & \ddots & \vdots \\ x'_{J,1,1}[n-L+1] & x'_{J,1,2}[n-L+1] & \cdots & x'_{J,1,M}[n-L+1] \\ x'_{1,2,1}[n] & x'_{1,2,2}[n] & \cdots & x'_{1,2,M}[n] \\ \vdots & \vdots & \ddots & \vdots \\ x'_{J,K,1}[n-L+1] & x'_{J,K,2}[n-L+1] & \cdots & x'_{J,K,M}[n-L+1] \end{bmatrix} \tag{9}$$

where J is the number of reference signals, K is the number of actuators of the system, M is the number of error sensors and L is the order of the adaptive filters, $W_{j,k}(z)$ (Figure 5).

Observe that there are JKM filtered reference signals, $x'_{j,k,m}[n]$, since every reference signal, $x_j[n]$, is filtered by each of the $K \times M$ secondary path filters, $\hat{S}_{k,m}(z)$. The update equation for the filter coefficients in the multichannel FxLMS system is:

$$\mathbf{w}_{j,k}[n+1] = \mathbf{w}_{j,k}[n] + m \sum_{m=1}^{M} e_m[n]\mathbf{x}'_{j,k,m}[n] \quad \text{for } j = 1,\cdots,J; k = 1,\cdots,K \tag{10}$$

being $W_{j,k}(z)$ the filter from reference input j to the actuator k.

The element (a,b) of the original correlation matrix is:

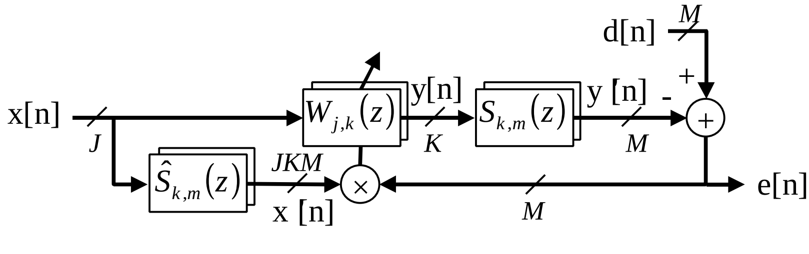


FIGURE 5. *Multiple-channel FxLMS system: J reference signals, K actuators and M error sensors.*

$$R_{(a,b)} = E\left\{\sum_{m=1}^{M} \mathrm{x}'_{j1,k1,m}[n-l1]\mathrm{x}'_{j2,k2,m}[n-l2]\right\} \tag{11}$$

where $a = l1 + (j1-1)L + (k1-1)JL$ and $b = l2 + (j2-1)L + (k2-1)JL$ ($0 \le l1, l2 \le L-1$, $1 \le j1, j2 \le J$ and $1 \le k1, k2 \le K$).

After the orthogonalization filter, we have $\mathrm{x}'_{\mathrm{D}l,j,k,m}[n]$ instead of $\mathrm{x}'_{j,k,m}[n-l]$. In the FxGAL system we have $\mathrm{x}'_{\mathrm{D}l,j,k,m}[n] = \mathrm{b}_{l,j,k,m}[n]$ and in the ALE + FxLMS system, $\mathrm{x}'_{\mathrm{D}l,j,k,m}[n] = \mathrm{v}_{j,k,m}[n-l]$. To diagonalize de matrix, we have to minimize $\left|R_{D(a,b)}\right|$ when $a \ne b$, that is to say, when at least one of the following holds: $l1 \ne l2$, $j1 \ne j2$ or $k1 \ne k2$. The optimum solution will be the one that achieves

$$R_{\mathrm{D}(a,b)} = E\left\{\sum_{m=1}^{M} \mathrm{x}'_{\mathrm{D}l1,j1,k1,m}[n]\mathrm{x}'_{\mathrm{D}l2,j2,k2,m}[n]\right\} = 0 \text{ for } a \ne b \tag{12}$$

There are *JKM* input signals for the multichannel orthogonalizing system: the *JKM* filtered reference signals, $\mathrm{x}'_{j,k,m}[n]$. However, no more than *JK* filters are required, since each filter, $H_{D\,j,k}(z)$, filters *M* different signals, $\mathrm{x}'_{j,k,m}[n]$, for $m = 1, \cdots, M$, obtaining *M* error signal outputs, $\mathrm{x}'_{\mathrm{D}l,j,k,m}[n]$, for $m = 1, \cdots, M$. The sum of the *M* products of errors by inputs is used for the adaptation process of the filter $H_{D\,j,k}(z)$, in a similar way to the multichannel FxLMS case, in (10).

Next, we are going to inspect the different possibilities in the minimization of $\left|R_{(a,b)}\right|$: $l1 \ne l2$, $j1 \ne j2$ and $k1 \ne k2$.

When $l1 \ne l2$ (different time index), orthogonality is obtained in both systems in the same way as in the monochannel case: by whitening, in the ALE + FxLMS case, or due to the orthogonality of the backward prediction errors, in the FxGAL system.

To obtain orthogonality when $j1 \ne j2$ (different reference signals) the orthogonalization has to be vectorial in *j*, that is, every component of the vector $[\mathrm{x}'_{1,k,m}[n-l1], \mathrm{x}'_{2,k,m}[n-l1], \cdots, \mathrm{x}'_{J,k,m}[n\text{-}l1]]$, is directly involved in the computation of each one of the outputs $\mathrm{x}'_{\mathrm{D}l2,j,k,m}[n]$ ($l1 < l2$ for FxGAL and $l1 > l2$ for ALE + FxLMS). Therefore, the possible correlation between different reference signals is removed, that is, each output, $\mathrm{x}'_{\mathrm{D}l2,j,k,m}[n]$, will be decorrelated from every output with a different time index, $\{\mathrm{x}'_{\mathrm{D}l1,1,k,m}[n], \mathrm{x}'_{\mathrm{D}l1,2,k,m}[n], \cdots, \mathrm{x}'_{\mathrm{D}l1,J,k,m}[n]\}$.

There is a special case when $j1 \ne j2$ and $l1 = l2$, since $\mathrm{x}'_{j1,k,m}[n-l]$ is involved in the computation of $\mathrm{x}'_{\mathrm{D}l,j1,k,m}[n]$, but $\mathrm{x}'_{j2,k,m}[n-l]$ is not. Therefore, some correlation between the different output components in the same time instant, $\{\mathrm{x}'_{\mathrm{D}l,1,k,m}[n], \mathrm{x}'_{\mathrm{D}l,2,k,m}[n], \cdots, \mathrm{x}'_{\mathrm{D}l,J,k,m}[n]\}$, could remain. In this case, some additional processing is required to obtain intravectorial decorrelation. This topic will be addressed later in this section.

Orthogonality when $k1 \ne k2$ (different actuators) could be obtained in the same way as when $j1 \ne j2$, that is, with vectorial calculation in *k*. Nevertheless, the physical meaning of this solution makes it undesirable: different actuators would be driven by completely uncorrelated signals. Several actuators are required in many applications due to the complexity of the primary field and/or the number of excited modes. However, if we orthogonalize the inputs to the different actuators we might limit the system cancellation capability, due to the misuse of the multiple actuators configuration. Moreover, cancellation in points different from where the error sensors are placed would be compromised. For that reason, it is not advisable to completely diagonalize the correlation matrix when there are several actuators, and, in this case, only a block-diagonalization is possible.

Therefore, there are *KM* identical vectorial filters, $H_D(z)$, with *J* inputs and *J* outputs, one for each (*k*, *m*) pair. A copy of this vectorial filter will also be used as slave filter, for filtering the reference signals.

Intravectorial decorrelation can be obtained by means of any Principal Component Analysis (PCA) algorithm [10] applied to the outputs of the vectorial filter $H_D(z)$. In the present work, we have used an iterative algorithm, where the correlation between signals is adaptively estimated (Figure 6):

$$x_j[n] = \mathrm{s}_j[n] + \sum_{i=1}^{j-1} x_i[n] g_{i,j}[n] = \mathrm{s}_j[n] - \sum_{i=1}^{j-1} x_i[n] \frac{R_{x_i \mathrm{s}_j}(0)}{R_{x_i x_i}(0)} \tag{13}$$

Although this algorithm is similar to APEX or Rubner's models [10], it is not really a PCA algorithm, but simply a "correlation removing" algorithm. The decorrelation network is clearly asymmetric (or hierarchical) but it is conceptually very simple. However, other methods, more computationally efficient for DSP implementation, are under study and could be subject of future work.

Observe that in the ALE + FxLMS case only one intravectorial decorrelator is required, since there is only one new *J*-vector output for each algorithm iteration. Nevertheless, in the FxGAL case, L intravectorial decorrelators are needed, one for each time index, since $\mathrm{x}'_{\mathrm{D}l,j,k,m}[n] \neq \mathrm{x}'_{\mathrm{D}l+1,j,k,m}[n-1]$.

V. PERFORMANCE COMPARISON

Considering that the convergence rate depends mainly on the input signal to the adaptive filter, the orthogonalizing stage will have to face convergence problems similar to those that we want to eliminate from the FxLMS. However, experimental results show that convergence for the systems proposed here can be much faster than for the FxLMS. The reason for this is that it is not the convergence rate of the orthogonalizing system that matters, but the "orthogonalizing

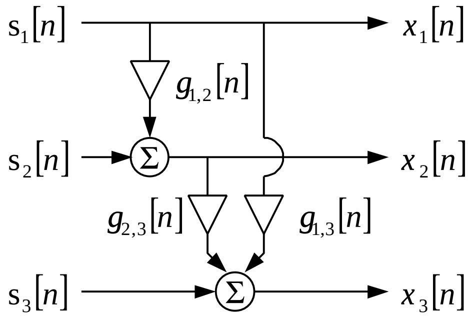


FIGURE 6. *Intravectorial decorrelation algorithm for 3 signals.*

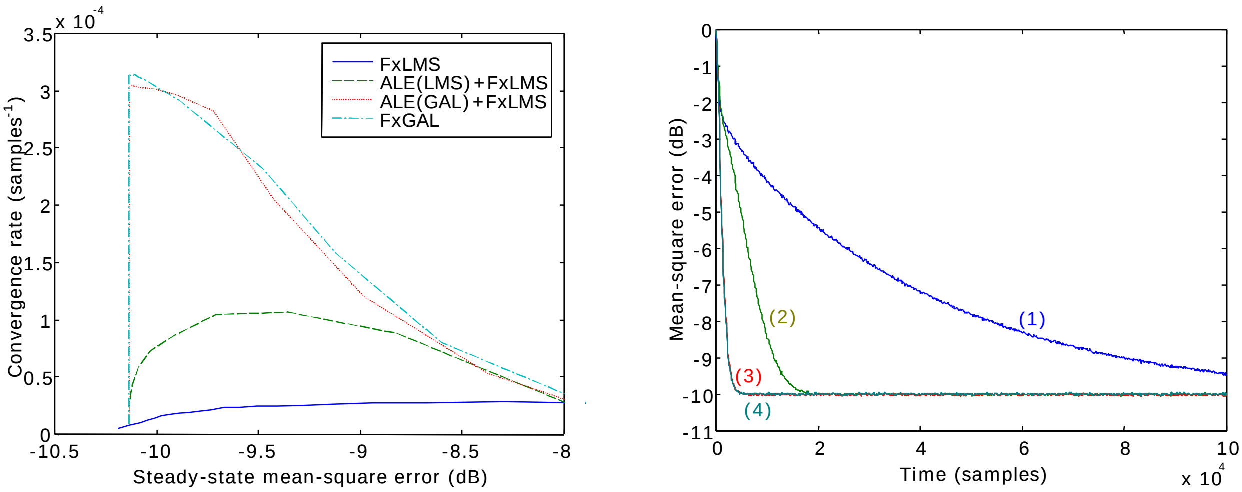
speed", since any improvement in its adaptation signal conditioning is seized immediately by the subsequent FxLMS system.

Due to the adaptive nature of the orthogonalizing system, there will always be a misadjustment error or noise added to the desired decorrelated signal. The effects of this noise are generally masked by the components in the primary noise that cannot be cancelled. However, there will be an upper limit in the adaptation step size for the orthogonalizing algorithm to ensure that the cancellation capability of the global system is not affected by that misadjustment noise.

Performance improvement for the proposed systems related to FxLMS depends on the correlation between reference signals and primary disturbances. Thus, when that correlation is high enough, the FxLMS is sufficiently fast, because the faster convergence modes (greater l_l) are also the more excited ones. In this case, orthogonalizing systems are not able to improve FxLMS performance. In fact, worse results are obtained due to the misadjustment noise introduced by the adaptation process. Nevertheless, this is not the common case, and, in general, orthogonalizing systems obtain better results than FxLMS.

The performance improvement can be observed in the diagram of Figure 7.a. The steady-state mean-square error (x_∞) has been plotted versus an estimation of the convergence rate ($1/t$). The FxLMS curve has been obtained for a fixed value of the filter length, L, and varying the adaptation step size parameter. This curve divides the $x_\infty - 1/t$ space in two regions: the systems that improve FxLMS performance (greater convergence rate for the same steady-state mean-square error) are placed above it while the systems with worse performance are placed below it. Figure 7.a was obtained by computer simulation with artificially generated broadband signals. Steady-state mean-square error was measured after all slow convergence modes had died out and convergence rate was estimated over one realization. The orthogonalizing systems curves were obtained for a fixed value of the adaptation step size of filter $W(z)$, the same number of coefficients L and varying the adaptation step size of the orthogonalizing system. When this adaptation step size tends to zero, the orthogonalizing systems proposed are equivalent to the FxLMS system. For that reason, all the curves in Figure 7.a converge in one point.

Observe that FxGAL and ALE(GAL) + FxLMS systems performances are very similar. This is due to two different reasons. The first one is that only broadband signals have been used,



(a) (b)

FIGURE 7. *(a) Steady-state mean-square error vs. convergence rate diagram. (b) Learning curves comparison: (1) FxLMS, (2) ALE(LMS) + FxLMS, (3) ALE(GAL) + FxLMS and (4) FxGAL.*

since ALE + FxLMS systems only work under this assumption. The second reason is that, in this test, the prediction order of the ALE system is high enough to whiten up the filtered-x signal, diagonalizing completely the correlation matrix. However, in general, better performance can be expected from the FxGAL system, since this system always diagonalizes the matrix completely, independently of the filter order or the signal nature (broadband or narrowband signals). The worse results of the ALE(LMS) + FxLMS system are only due to the better performance of the lattice structure when compared to a transversal structure.

Figure 7.b shows the learning curves obtained with the same signals of Figure 7.a when $x_\infty = 10\text{dB}$ for the four systems. Better performance of all of the orthogonalizing systems becomes evident in this figure. Observe that FxGAL and ALE(GAL) + FxLMS learning curves overlap, since, as was discussed previously, in this case both systems have very similar performance.

Real vibration signals recorded in the wheels of a medium-sized car were used in the following test. The secondary path transfer function was taken from the examples in [1] and the FIR model of it, necessary for the adaptive process, was obtained previously to the operation of the system. Figure 8 shows the learning curves obtained for the four systems. For viewing purposes, the curves were smoothed by moving average filtering. Adaptation step sizes for the four systems were determined independently, so as to maximize cancellation. As can be seen from the figure, the results obtained with artificially generated signals still hold with real nonstationary signals. The FxGAL system is the one that achieves the best performance, followed very closely by the ALE(GAL) + FxLMS. Also, the ALE(LMS) + FxLMS performance, although worse than that of FxGAL and ALE(GAL) + FxLMS, is clearly better than that of the FxLMS.

As we anticipated in the introduction, the expense of the improved performance of the systems proposed is the increase in computational complexity. Table 1 shows a summary of the number of operations (products, sums and divisions) for the FxLMS system and each one of the algorithms proposed in this paper. In [3], it was proposed an alternative to the FxGAL system that combines the lattice and transversal filter structures. This alternative, named here FxGAL-

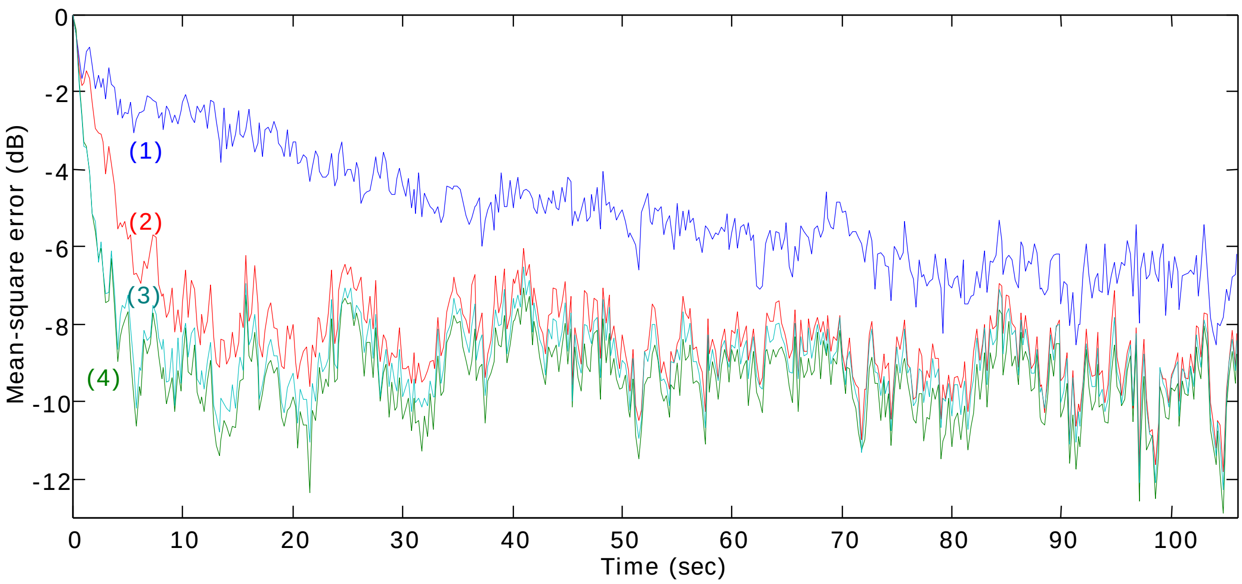


FIGURE 8. *Learning curves for real signals: (1) FxLMS, (2) ALE(LMS) + FxLMS, (3) ALE(GAL) + FxLMS and (4) FxGAL.*

		Monochannel	Multichannel
FxLMS	×	$2L + I + 4$	$(JKM + JK)L + JKMI + JKM + M + 2$
	+	$2L + I - 1$	$(JKM + JK)L + JKMI - K$
	÷	1	1
ALE(LMS) + FxLMS	×	$2L + 3Q + I + 8$	$(JKM + JK)L + (2J^2KM + J^2)Q + JKMI + 3JKM + M + 4$
	+	$2L + 3Q + I - 2$	$(JKM + JK)L + (2J^2KM + J^2)Q + JKMI - K - 1$
	÷	2	2
ALE(GAL) + FxLMS	×	$2L + 11Q + I + 4$	$(JKM + JK)L + (4J^2KM + 4J^2 + 2JKM + 4)Q + JKMI + JKM + M + 2$
	+	$2L + 8Q + I - 1$	$(JKM + JK)L + (4J^2KM + 2J^2 + 2JKM)Q + JKMI - K$
	÷	$Q + 1$	$2Q + 1$
FxGAL	×	$16L + I - 10$	$(4J^2KM + 4JKM + JK + 4J^2 + 2J + 4)L + JKMI - 4J^2KM - 2JKM - 4J^2 + M - 4$
	+	$11L + I - 10$	$(4J^2KM + 4JKM + 2J^2 + JK)L + JKMI - 4J^2KM - 3JKM - 2J^2 - K$
	÷	$2L - 1$	$(J + 2)L - 2$
FxGAL-LMS	×	$2L + 14Q + I + 4$	$(JKM + JK)L + (4J^2KM + 3JKM + 4J^2 + 2J + 4)Q + JKMI + JKM + 2J + M$
	+	$2L + 9Q + I - 1$	$(JKM + JK)L + (4J^2KM + 3JKM + 2J^2)Q + JKMI - K$
	÷	$2Q + 1$	$(J + 2)Q + J$
Intravectorial decorrelator (for one vector)	×	–	$(2J + JKM/2 + KM + 2)(J - 1)$
	+	–	$(J + JKM/2 + KM)\ (J - 1)$
	÷	–	$J(J - 1)/2$

TABLE 1. *Computational complexity comparison.*

LMS, aims to reduce FxGAL computational complexity while (almost) not affecting performance, and has also been included in Table 1.

In the table, L is the order of the filters $W(z)$ and I the number of coefficients of the models of the secondary path, $\hat{S}(z)$. Remember that in the multichannel case, J is the number of reference signals, K the number of actuators and M the number of error sensors. Q is the order of the ALE predictor in the ALE + FxLMS systems or the number of lattice stages in the FxGAL-LMS system ($0 \le Q \le L-1$).

Observe that the systems with lattice structures require greater computational effort. This fact is even more evident in the multichannel case, where lattice structures cannot have the same reflection coefficient for forward and backward prediction. Observe also the quadratic growth with the number of reference signals, J, for the multichannel orthogonalizing systems.

The computational complexity due to the intravectorial decorrelator is not included in the multichannel formulas of the orthogonalizing systems. As we mentioned above, alternative methods more computationally efficient are under study. However, the total amount of operations for the orthogonalizing systems can be computed just by adding the last file of the table, which corresponds to the intravectorial decorrelator we have used. Remember that in the FxGAL case the intravectorial decorrelator is used not one but L times in each iteration (Q times in the case of the FxGAL-LMS system). Therefore, the formulas of the intravectorial decorrelator should be multiplied by L in the FxGAL case (by Q in the FxGAL-LMS case).

VI. SUMMARY

In the present work, we have introduced the FxGAL and ALE + FxLMS algorithms for active control. Both, the monochannel and the multichannel versions of the systems, have been presented. These systems try to achieve faster convergence than FxLMS, while keeping the same steady-state mean-square error level. In most cases, this is possible, although it depends on the correlation between the references and the primary disturbances. The expense is an increase in computational complexity.

The FxGAL system is the more computationally demanding, but also the one that achieves better cancellation results. Furthermore, the FxGAL system is not limited to broadband cancellation as ALE + FxLMS systems are. When the order prediction in the ALE system is high enough, significant performance improvement can also be obtained by the ALE + FxLMS system. The GAL version of the ALE obtains better results than the LMS version, but also increases more the computational effort.

ACKNOWLEDGEMENTS

This work was supported by CONSI+D/DGA (grants B226/96 and P64/96) and CAI.

REFERENCES

1. Kuo, S. M., Morgan, D. R., *Active Noise Control Systems: Algorithms and DSP Implementations*, New York: John Wiley & Sons, 1996, ch. 3, 5, 8.
2. Elliott, S. J., Nelson, P. A., "Active Noise Control", *IEEE Signal Processing Magazine*, **10(4)**, pp. 12-35 (1993).
3. Vicente, L., Masgrau, E., "Fast Convergence Algorithms for Active Noise Control in Vehicles" in *Proceedings of Joint ASA/EAA/DEGA Meeting*, Berlin, 1999.
4. Vicente, L., Elliott, S. J., Masgrau, E., "Fast Active Noise Control for Robust Speech Acquisition" in *Proceedings of Eurospeech*, Budapest, 1999.
5. Haykin, S., *Modern Filters*, New York: Macmillan, 1989, ch. 6-8.
6. Friedlander, B., "Lattice Filters for Adaptive Processing", *Proceedings of the IEEE*, **70(8)**, pp. 829-867 (1982).
7. Widrow, B. et al, "Adaptive Noise Cancelling: Principles and Applications", *Proceedings of the IEEE*, **63(12**), pp. 1692-1716 (1975).
8. Zeidler, J. R., "Performance Analysis of LMS Adaptive Prediction Filters", *Proceedings of the IEEE*, **78(12)**, pp.1781-1806 (1990).
9. Widrow, B., Stearns S. D., *Adaptive Signal Processing*, New Jersey: Prentice-Hall, 1985, ch. 6.
10. Diamantaras, K. I., Kung, S. Y., *Principal Component Neural Networks: Theory and Applications*, New York: John Wiley & Sons, 1996, ch. 4.