# Performance comparison between the FxLMS and the FxHLMS algorithms in an Active Noise Control system.

PEDRO RAMOS, ANA LÓPEZ, ANA SALINAS AND ENRIQUE MASGRAU [1]

Abstract. The aim of this work is to adapt the Hierarchical LMS algorithm to an Active Noise Control System and to compare its performance with the FxLMS algorithm.

Taking into account that the slightest change in the DSP code or in the electro-acoustical arrangement implies lots of experiments in the laboratory that can finish unsuccessfully, we have developed in MATLAB a simulated model of the system in order to test more easily in advance any variation of the basic algorithm. Apart from the cancelling algorithm chosen, the model includes the experimental estimate of the primary and secondary path transfer functions. The model has been tested with the Leaky FxLMS algorithm with shaping of the residual noise previously implemented in the TMS320C32 DSP and the achieved and simulated results have been compared.

Not only the effectiveness of the FxHLMS algorithm -in terms of convergence speed, excess mean squared error (MSE) and stability- but also the parameters that limit the viability of the implementation on the DSP such as computational costs and address pointers required have been considered.

**Key words**: Filtered x Hierarchical LMS (FxHLMS) algorithm.

## 1. Introduction

Active control of noise is a method for attenuating disturbances by the introduction of controllable secondary sources whose outputs are arranged to interfere destructively with the disturbance from the original primary source. The efficiency of active noise control systems in improving the performance of the passive systems in order to reduce noise level inside a road vehicle has been previously demonstrated by several researches [1].

With the advent of cheaper and more powerful Digital Signal Processing (DSP) hardware, the implementation of ANC systems based on adaptive digital signal processing algorithms has become practical. The most widely used of these adaptive algorithms is the well known Least Mean Square (LMS).

[1] Departamento de Ingeniería Electrónica y Comunicaciones. Universidad de Zaragoza (SPAIN).
pramos@posta.unizar.es, lopeztor@posta.unizar.es, salinas@posta.unizar.es, masgrau@posta.unizar.es

The fast Hierarchical Least Mean Square (HLMS) algorithm [2] presents several advantages over the basic LMS that can improve the performance on an Active Noise Control System by modifying the FxLMS including an adaptive filter organized into a hierarchy. The adaptation of this idea to an acoustical canceller requires certain changes due to the presence of a secondary path transfer function and the need to measure the undesired signal so as to adapt the intermediate levels of the hierarchy. The algorithm proposed has been called Filtered x Hierarchical LMS (FxHLMS).

Most of the parameters of any algorithm implemented on a DSP board must be empirically optimised in order to improved the results in terms of level of attenuation achieved, convergence speed or computational efficiency. However, a small change in the assembler code or in the physical arrangement of the system takes more time than it is worth in most of the cases. Therefore, we have been developing a computer simulation of the system that can be used as a first approach to the expected behaviour of the cancelling system.

Neither implementing in the DSP the new algorithm nor changing the disposal of the electro-acoustic elements is needed in order to predict the performance of an active noise control system as long as the software tool provides an accurate model of the system. The model has been used to simulate the Leaky FxLMS algorithm with shaping of the residual noise. Moreover the simulation tool allowed us to compare the performance of the basic LMS with the new and faster version of the adaptive algorithm known as FxHLMS algorithm.

## 2. The Filtered X Hierarchical LMS (FxHLMS) algorithm.

### 2.1 The Hierarchical LMS algorithm.

The basic LMS algorithm has been widely used because of the ease of implementation and the good results achieved. Nevertheless the convergence rate of the algorithm can be accelerated and the excess mean squared error reduced by means of the Hierarchical Least Mean Square algorithm.

In the proposed algorithm the N taps of the filter are organized into a logical β-ary hierarchy of α levels, where $\beta^{\alpha} = N$. The levels of the hierarchy are numbered from 1 (initial level) to α (final level). Let $\beta_i$ denote the number of taps of a subfilter for level i, then we have $\prod_{i=1}^{\alpha}\beta_i = N$. The idea is to minimise the MSE using the LMS algorithm within each subfilter (β taps) at each level. The output of the filter is equal to the weighted sum of the top level (level α). We use $r_{ij}^{l}$ and $w_{ij}^{l}$ to denote the input signal and the weight for jth tap of the ith subfilter at the lth level, respectively.

Table 1. General features of the algorithm.

| Total number of taps at the lth level | Number of subfilters at the lth level | Output of the single filter at the top level | Number of levels |
|---|---|---|---|
| $N/\beta^{l-1}$ | $N/\beta^{l}$ | $output = \sum_{k=1}^{\beta} w_{1k}^{\alpha} \cdot r_{1k}^{\alpha}$ | $\log_{\beta}(N)$ |

The HLMS algorithm is given by:

Initialization: $\underline{w}_i^l = \left(w_{i1}^l,...,w_{i\beta}^l\right) = (0,...,0)$

for $l$ =1 to α do /* from level 1 to level α*/

for i=1 to $N/\beta^l$ do /* for each subfilter at level $l$ */

$e_i^l(k) = d(k) - \underline{w}_i^{l\,H}(k) \cdot \underline{r}_i^l(k)$ /* The i subfilter at level $l$ */

$\underline{w}_i^l(k+1) = \underline{w}_i^l(k) + \mu(k) \cdot e_i^l(k) \cdot \underline{r}_i^l(k)$

end of for;

end of for;

### 2.2. Advantages of the HLMS over the LMS algorithm.

The advantages of the HLMS algorithm can be summarized as follows:

- The subfilters can be executed in parallel to reduce the processing time. This characteristic will be extremely advantageous when the architecture of the DSP offers the possibility of working in parallel in different tasks.
- Each subfilter has a shorter length and therefore can converge faster because of the reduction in the eigenvalue spread of the autocorrelation matrix of the input signal.
- The HLMS algorithm demonstrates a lower MSE excess.

### 2.3. Adaptation of the HLMS algorithm to an ANC system. The FxHLMS algorithm.

The update of the coefficients of the adaptive filter in an ANC system based on the FxLMS algorithm requires the filtered version of the reference signal and the error measured -that in the practical implementation of the algorithm consist of the acoustical superposition of the undesired noise and the output of the system after having passed through the secondary path-.

One of the differences between the LMS and the HLMS algorithm is the necessity to have the undesired signal $d(n)$ in order to update the subfilters located at the intermediate levels of the hierarchy. Taking into account that during the current iteration it is only possible to measure the acoustical addition of the undesired noise and the filtered antinoise, that is $e(n) = d(n) - y'(n) = d(n) - y(n) * s(n)$ -where $s(n)$ represents the secondary path transfer function- it is necessary to obtain an estimate of the signal $d(n)$ to update the hierarchical filter. The estimate is calculated on the basis of the previous value of the undesired noise, that is:

$$\tilde{d}(n) = d(n-1) = e(n-1) + y'(n-1) \approx e(n-1) + \sum_{m=0}^{Ls-1} \tilde{s}_m \cdot y((n-1)-m) \qquad (1)$$

where $\{\tilde{s}_m\}$ are the Ls coefficients of the off-line estimate of the secondary path transfer function.

The second variation that the HLMS algorithm requires to be applied in an ANC system is due to the presence of a secondary path transfer function between the output of the filter and the error microphone. Because of that, it is necessary to filter the output of each subfilter at the intermediate levels with the estimate of the secondary path transfer function before being substracted from the undesired noise $d(n)$. We use $w_{ij}^l$ to denote the weight for jth tap of the ith subfilter at the lth level and $y_i^l$ and $e_i^l$ to denote the subfilter´s output and the error signal of the ith subfilter at the lth level respectively. The first hierarchical level of the output signals $\underline{y}^0$ corresponds to the reference signal of the ANC system. The FxHLMS algorithm can be expressed as follows:

Initialization: $\underline{w_i^l} = \left(w_{i1}^l, ..., w_{i\beta}^l\right) = (0,...,0)$

$\underline{y}^0 = \left(y_1^0, ..., y_N^0\right) = (Input)$ /* Shifting the (N-1) older samples of the reference signal and updating the last one*/

$\tilde{d}(k) \approx e_1^\alpha(k-1) + \sum_{m=0}^{Ls-1} \tilde{s}_m \cdot y_1^{\alpha}((k-1)-m)$ /* Estimate of the undesired noise*/

for $l$=1 to α do /* from level 1 to level α*/

for i=1 to $N/\beta^l$ do /* for each subfilter at level $l$ */

$y_i^l(k) = \sum_{m=1}^{\beta_l} y_{(i-1)\cdot\beta_l+m}^{l-1}(k) \cdot w_{i,m}^l(k)$ /* Output of each sub filter*/

$e_i^l(k) = \tilde{d}(k) - y_i^l(k)$ /* Error of each subfilter. At the level α the acoustical error is measurable */

$\hat{y}_i^l(k) = \sum_{m=0}^{Ls-1} \tilde{s}_m \cdot y_i^l(k-m)$ /* Filtering the reference signal */

$\underline{w}_i^l(k+1) = \underline{w}_i^l(k) + \mu \cdot e_i^l(k) \cdot \underline{\hat{y}}_i^l(k)$ /* Filters´ update*/

end of for;

end of for;

Figure 1 shows the block diagram of the ANC system when the FxHLMS algorithm is implemented with two levels ($\alpha = 2$).

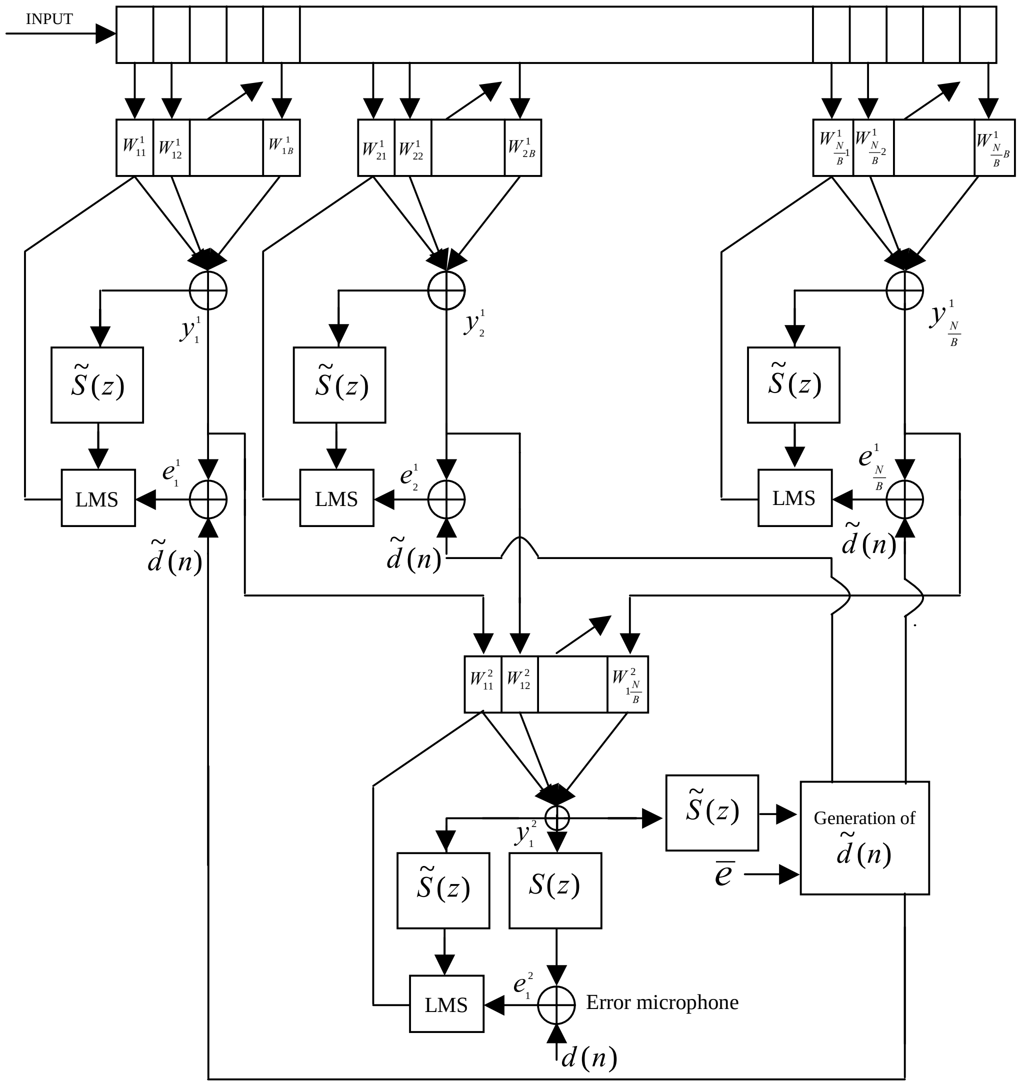


Figure 1. Block diagram of the ANC system based on a 2 level FxHLMS algorithm.

## 3. Simulation tool.

### 3.1. Building and testing the model of the ANC system.

In order to obtain a complete and accurate model of the system it is necessary to estimate previously the transfer functions of the primary and the secondary paths. The secondary paths are obtained according to the LMS algorithm that updates a digital FIR filter whereas the primary paths are estimated by means of deconvolution techniques based on the complex cepstrum to obtain an estimate of the primary paths.

In previous works [3], [4] we showed the experimental results achieved in the cancellation of engine noise inside a van by means of a Leaky FxLMS algorithm with shaping of the residual noise implemented on a TMS320C32 DSP. The algorithm has been programmed in MATLAB following the structure of the assembler program that runs in the DSP, that is, generating one sample of every signal in each iteration of the main loop that is executed according to the timing imposed by the sampling frequency. Therefore we can extrapolate more easily the results of the simulation to new versions of the DSP code.

Different examples of the MATLAB tool´s performance have been carried out in order to compare the performance of the real implementation with simulated results and to show how the simulation program can help to predict the behaviour of the real system implemented in the van.

### 3.2. Performance comparison.

In order to test the FxHLMS and to compare the results with the FxLMS we have chosen two different configurations for the hierarchical structure. Taking into account that the length of the filters that are already implemented in the DSP routines is typically 64, we have chosen two different set of parameters of equal length to be tested by the simulation tool. The results achieved in MATLAB lead us to extract some conclusion about future implementations of this control strategy in the real system.

Table 2. First case: 2-level hierarchy, N=64, $\alpha$=2, $\beta$=8

| **LEVEL 1** | **LEVEL 2** |
|---|---|
| Total number of taps at the first level = 64 | Total number of taps at the second level = 8 |
| Number of sub-filters = 8 | Number of sub-filters = 1 |

Table 3. Second case: 3-level hierarchy N=64, $\alpha$=3, $\beta$=4

| **LEVEL 1** | **LEVEL 2** | **LEVEL 3** |
|---|---|---|
| Total number of taps at the first level = 64 | Total number of taps at the second level = 16 | Total number of taps at the third level = 4 |
| Number of sub-filters = 16 | Number of sub-filters = 4 | Number of sub-filters = 1 |

Figure 2 represents the evolution of the error´s power when the model simulates the attenuation of engine noise.

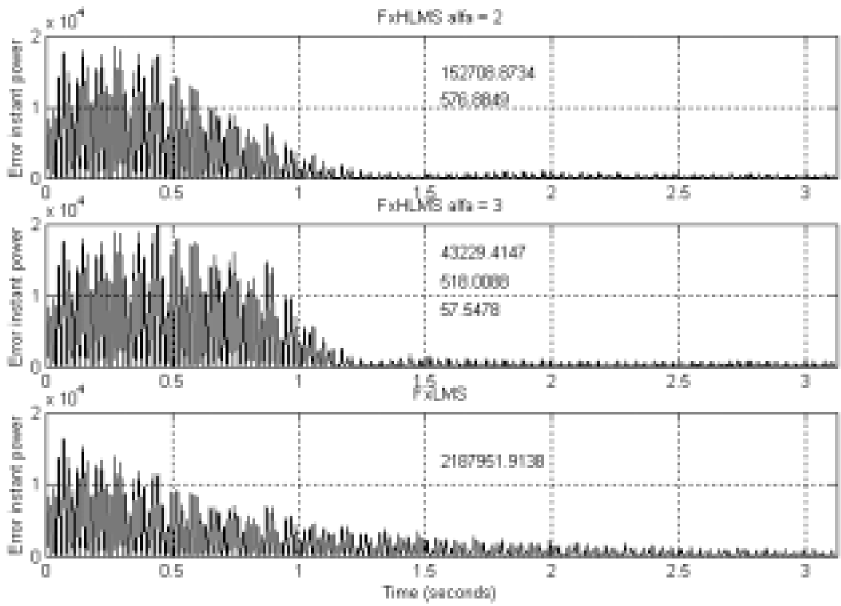


Figure 2. Comparison between the FxLMS and the FxHLMS algorithms.

The numbers that appear over the graphics correspond to the eigenvalue spread at each level of the hierarchy, from the first to the final (α) level. As expected, at the higher levels of the structure, the eigenvalue spread is reduced, allowing a faster convergence of the cancellation process. When a larger number of levels is employed, the length of the filters is reduced, and the size of the autocorrelation matrix is shorter.

Nevertheless, the higher the number of hierarchical levels the more unstable becomes the system when the step size is increased -and a larger step size guarantees a faster convergence-. Therefore a trade-off is established in order to minimise the eigenvalue spread without endangering the system´s stability.

### 3.3. Hardware and software requirements.

As far as the computational advantages of the algorithm are concerned, the possibility of executing simultaneously all the filtering at the same level of the hierarchy can reduce the processing time by taking advantage of multiprocessor hardware. Nevertheless there is a limit imposed by the number of address registers that can by used simultaneously and the number of operations that can be made independently at the same time.

Table 4. Number of multiplies in each algorithm.

| ALGORITHM | FxLMS # Multiplies | FxHLMS # Multiplies | FxHLMS # Multiplies not simultaneous |
|---|---|---|---|
| Estimate of the undesired noise. | None | *Ls (length of the secondary path)* | *Ls* |

| | | | |
|---|---|---|---|
| Generation of the antinoise. | $N = \beta^{\alpha}$ | $\left[\sum_{l=1}^{\alpha} \frac{N}{\beta^{l}}\right] \cdot \beta$ | $\alpha \cdot \beta$ |
| Reference´s filtering with the estimate of the secondary path. | $Ls$ | $\left[\sum_{l=1}^{\alpha} \frac{N}{\beta^{l}}\right] \cdot Ls$ | $Ls$ |
| Updating the adaptive filters. | $2 \cdot N = 2 \cdot \beta^{\alpha}$ | $\left[\sum_{l=1}^{\alpha} \frac{N}{\beta^{l}}\right] \cdot 2 \cdot \beta$ | $2 \cdot \beta$ |

Comparing the FxHLMS and the FxLMS algorithms, if the hardware system could work independently for each subfilter, the processing time employed in multiplies by the CPU would be reduced in a factor $\frac{Ls + 3 \cdot \beta^{\alpha}}{2 \cdot Ls + (\alpha + 2) \cdot \beta}$, that becomes specially important when the length of $\tilde{S}(z)$ is reduced and the number of taps is large. On the other hand, such a reduction implies an increase in a factor up to $\frac{1}{2}\left[\sum_{l=1}^{\alpha} \frac{N}{\beta^{l}}\right]$ in the number of address pointers required and in a factor up to $\left[\sum_{l=1}^{\alpha} \frac{N}{\beta^{l}}\right]$ in the number of simultaneous operations that the system can support.

## 4. Conclusions

We have proposed a new Active Noise Control algorithm based on a hierarchical structure that can accelerate the convergence rate. The algorithm is specially useful when can be implemented in a multi processor hardware architecture. Simulation results have shown the effectiveness of the proposed scheme.

## 5. Acknowledgments

These researches are supported by CICYT (Comisión Interministerial de Ciencia y Tecnología) of Spanish Government under grant TIC-2002-04103-C03-01.

## References

[1] S.M. Kuo and D.R. Morgan, *Active Noise Control Systems: Algorithms and DSP implementations*, John Wiley & S, 1996.

[2] T.K. Woo, *Fast Hierarchical Least Mean Square Algorithm*, IEEE Signal Processing Letters, Vol. 8, NO. 11, November 2001.

[3] P. Ramos et al., *Practical implementation of a multiple-channel FxLMS Active Noise Control system with shaping of the residual noise inside a van*. Proc. ACTIVE 2002. University of Southampton, July 2002.

[4] A. López et al. *Acoustical analysis aimed at the characterisation of an Active Noise Control system inside a van and the improvement of its performance*. Proc. ACTIVE 2002.