Study on active noise cancellation

Ian C. Díaz, Matías Fogg, Lisandro Alvarez, Manuel Dieguez

***Abstract*— Today, active noise cancellation is applied in the most diverse contexts, from industry to medicine and consumer products. This research work refers to the different noise cancellation systems commonly used. In addition, it analyzes the different adaptive algorithms that can be used for the implementation of these systems. The paper emphasizes on the *Broadband Feedforward* cancellation system both in its practical aspects and in its implementation. Its implementation is explored with the FxLMS algorithms and its recursive variant FxRLS.**

I. Introduction

Active noise cancellation is a method for the elimination of one wave by the addition of another. It is implemented by means of an electroacoustic or electromechanical system that takes advantage of the destructive interference phenomenon. To achieve this, it uses a wave with equal amplitude but inverted phase with respect to the noise wave. Consequently, the combination of both waves results in the cancellation of both noises. As will be discussed later, noise cancellation systems or ANC for its acronym in English, are very effective in eliminating low frequency noise. This has caused a great deal of development in the industry since passive noise cancellation systems generally fail to eliminate noise in the low frequency range.

Noise can vary in frequency, amplitude, phase, speed of sound and can be caused by a myriad of sources. This very nature of noise implies that ANC systems must have an adaptive filter that allows modification to all the above mentioned variables. Adaptive filters adjust their coefficients, generally called "weights", to minimize the error signal. The coefficients can be determined by several algorithms, the most widely used is the least mean squares algorithm or LMS for its acronym in English. In addition, there are other algorithms such as NLMS, FxLMS, FxRLS, etc. These will be discussed later. This type of filters can be performed with finite response filters (FIR) and infinite response filters (IIR). As FIR filters are generally used, this is the type of filter that will be used in this work.

In this research work, noise is defined as any undesirable sound that you want to eliminate, noise generated by machines, ambient noise, conversations in a cafe, etc.. In other words, noise can be generated by any means.

ANC systems can be based on pre-feedback or *feedforward*, where the reference noise is sampled before it propagates to the second source, or on *feedback* where the noise is cancelled without a reference. That is, the signal is sampled after the injection of the canceling wave to correct the cancellation error.

Within the *feedforward* systems are 1) the narrowband systems or *Narrowband* that will be presented in the section II and 2) the broadband systems or *Broadband* that will be discussed in the section II. On the other hand, within the *feedback* systems is the adaptive narrowband system or *Narrowband adaptive* which will be introduced in section III.

In the IV section, the concept of Wiener adaptive filters and the criteria that must be taken into account to implement them correctly will be presented.

In the V section, the concepts and application of several algorithms to implement adaptive filters such as the LMS will be developed. In the VI section, a series of simulations of a *Broadband Feedforward* system are developed.

II. Feedforward systems

A. Narrowband Feedforward System

In many ANC applications noise signals are periodic. These signals can be generated by machines such as motors, fans, compressors, etc. This system uses the periodicity of these signals to cancel the noise. This type of system has the following advantages:

1) Feedback from the noise-canceling speaker to the reference microphone is avoided.
2) Alignments from the reference microphone are avoided.
3) As it is periodic noise, the causality limitation is eliminated.
4) The possibility of generating reference signals allows to control each harmonic independently.
5) It is only necessary to model the transfer function on frequencies close to that of the harmonics.

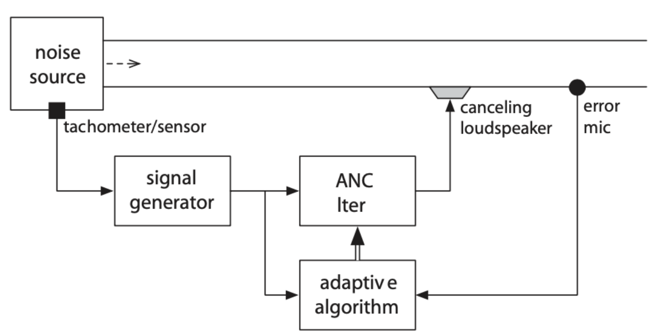

Fig. 1. Representation of the system *Narrowband Feedforward*

A basic diagram of such a system is shown in Fig. 1. The system has a sensor that acts as the *trigger* of the signal generator. The reference signal generated can be of two types: 1) a pulse train with period equal to the inverse of the fundamental frequency of the periodic noise and 2) sine waves with the same frequencies as the harmonics of the periodic noise. To implement the 1) type the method called wave synthesis is used while for the 2) type an adaptive *notch* filter is used. No matter what type of reference signal is generated, it is processed by an adaptive filter and sent to the driver. By means of an error signal the adaptive filter is modified (thanks to different algorithms) to minimize the error.

The synthesis method is explained in more detail below. The sensor captures the periodic noise signal and the noise samples are stored. We define N as the number of samples in one cycle of the noise wave. The samples represent the waveform to be generated to suppress the noise. The samples are sequentially sent to an analog digital converter to produce the signal to be reproduced in the noise cancelling driver. In this process is the adaptive FIR filter that has a length equal to one period of the noise signal or N samples. The residual sound is picked up by the error microphone which is synchronized with the pulses of the reference signal. Subsequently, the adaptive filter is modified to decrease the error. It can be proved that the LMS recursion is obtained as:

$$w(n+1) = w(n) - 2\mu e(n)x(n)$$

Its development will be discussed in more detail in the section V. After a series of mathematical steps [1], the transfer function of the system is obtained:

$$H(z) = \frac{E(z)}{D(z)} = \frac{1 - z^{-N}}{1 - (1 - 2\mu)z^{-N}} \tag{1}$$

(1) is an interesting equation as it relates the original noise, $d(n)$ and its suppressed version, $e(n)$. The zeros have a constant amplitude of ($|z| = 1$) and are equidistant at positions $0, \frac{2}{N}, \frac{4}{N}, ..., \frac{2(N-1)}{N}$ on the unit circle. It has N poles at the same angles as the zeros, but on the circumference $|Z| = (1 - 2mu)^{\frac{1}{N}}$. The transfer function describes a notch filter that attenuates at the fundamental and harmonic frequencies of the periodic noise. (1) shows that *mu* has a constraint regarding stability. *mu* must always satisfy $0 < \mu < 1$ for a train of unit amplitude pulses. The bandwidth for each notch this is $BW \approx 2(1-2\mu)^{\frac{1}{N}}$. Note that, the bandwidth is proportional to *mu*. On the other hand, the time constant of the decay of the response envelope is approximately $\tau \approx \frac{T}{\mu}$. Consequently, there is a *tradeoff* between the bandwidth and the duration of the transient response [1]. Finally, it should be clarified that (1) was calculated without considering the path between the noise-canceling driver and the error microphone, commonly called secondary path S_2. If the same is considered, the transfer function (1) undergoes the following modification:

$$H(z) = \frac{E(z)}{D(z)} = \frac{1 - z^{-N}}{1 - (1 - 2\mu S_2(z)S_2(z^-1))z^{-N}} \tag{2}$$

However, the results involving (2) will not be developed since this research work focuses on the *Broadband Feedforward ANC* system.

As a final comment about this type of system, the time required must be considered. That is to say, the time it takes for the signal to be processed by the adaptive filter and then sent to the loudspeaker and also the time it takes for the signal to reach the error microphone must be taken into account. Taking these times into consideration, the algorithm that modifies the adaptive filter can be appropriately adapted to be able to use this type of systems in a real time implementation.

B. Broadband Feedforward System

The system is illustrated in Fig. 2. The system has a reference microphone that takes the input signal, the signal is processed by the ANC system that reproduces a signal in the cancellation speaker. It has an error microphone that modifies the behavior of the adaptive filter to minimize the error of the ANC system.

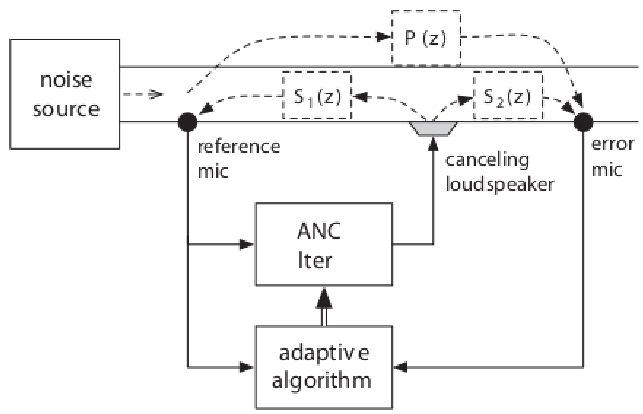

Fig. 2. Representación del sistema *Broadband Feedforward*

The primary path $P(z)$ is defined as the acoustic path between the coordinates near the reference microphone and the error microphone. There are also two secondary paths from the cancellation speaker to the reference signal $S_1(z)$ on one side and to the error microphone $S_2(z)$ on the other side. The adaptive filter $W(z)$ is used to estimate $P(z)$ to generate a noise cancelling signal.

To clarify the operation of the system, a block diagram of the system is illustrated in Fig. 3. The error microphone is depicted as the adder module whose input signals are $d(n)$ and $y(n)$. The system seeks to modify $W(z)$ so that $y(n)$ and $-d(n)$ have maximum correlation with each other. If this is achieved, when combining both signals, the residual error is $e(n) = d(n) + y(n) \approx 0$ since as they have maximum correlation the output will be white Gaussian noise.

Evaluating the system in Fig.3 yields the transfer function of the input signal between $x(n)$ and the error signal either $x(n)$:

$$\frac{E(z)}{X(z)} = P(z) + \frac{W(z)}{1 - S_1(z) \cdot W(z)} \cdot S_2(z) \tag{3}$$

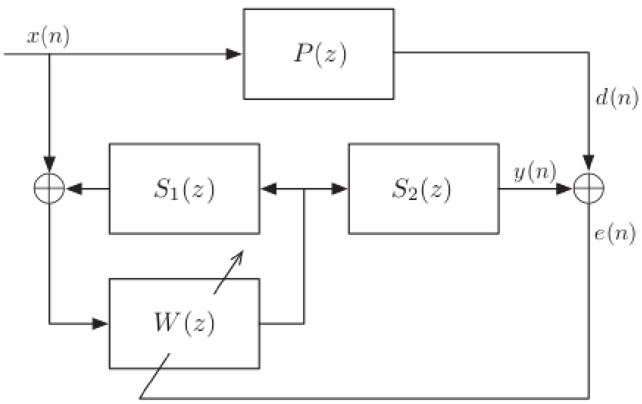

Fig. 3. Block representation of the *Broadband Feedforward*

Since perfect noise cancellation is desired, $e(n) = 0$ so $E(z) = 0$. Next, the transfer function that the adaptive filter must have is obtained:

$$W(z) = \frac{P(z)}{P(z)S_1(z) - S_2(z)} \tag{4}$$

As clarified above, the adaptive filter can be implemented with an IIR filter but its adaptation can be problematic because the poles close to the unit circle can move out of the unit circle generating the condition of instability to the system in the adaptation process. Ergo, one might think that the solution is to use a FIR filter. However, the use of a FIR filter as an alternative has difficulties in practice. Generally, the error microphone and the cancellation loudspeaker are placed at the same distance from the reference microphone. In addition, the error microphone is usually placed close to the cancellation speaker. This arrangement in the components results in high-order transfer functions for the primary path $P(z)$ and the first section of the secondary path $S_1(z)$, while the second section of the secondary path $S_2(z)$ has a low-order transfer. Therefore, (4) can only be approximated with a high-order FIR filter.

To solve this drawback, one could estimate $S_1(z)$ to remove its effect digitally. If an estimate of $S_1(z)$ defined as $\hat{S}_1(z)$ is introduced, the system in Fig. 3 undergoes the following modification:

One might think that since S_1 is estimated one could estimate S_2. This is what must inevitably be done since the path from the noise canceling speaker to the error microphone is not known in advance. Like $\hat{S_1}$, $\hat{S_2}$ is estimated using an adaptive filter driven by an adaptive algorithm such as LMS. A system with two modes of operation is presented in Fig. 5: 1) offline processing where the noise input comes from within the system thus obtaining the estimates of $\hat{S_1}$ and $\hat{S_2}$, and 2) online processing where the adaptive filter $W(z)$ comes into action to suppress an unknown noise.

It is useful to analyze how the system would be without considering S_1. The system without S_1 is illustrated in Fig. 6. Since it is desired that $e(n) = d(n) + y(n) \approx 0$ one must

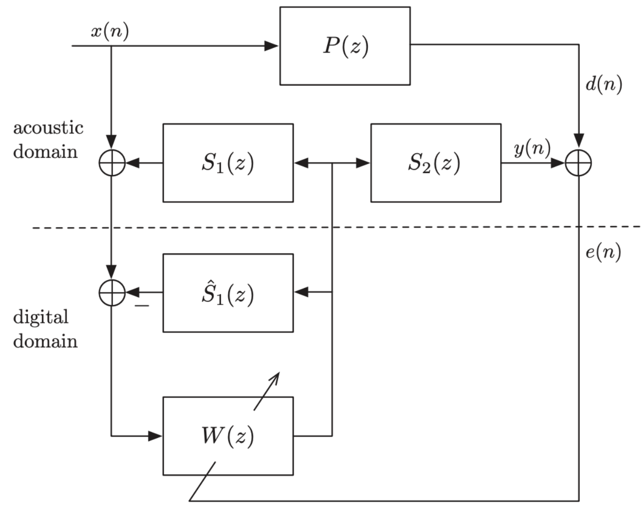

Fig. 4. Block representation of the *Broadband Feedforward* system using $\hat{S}_1(z)$.

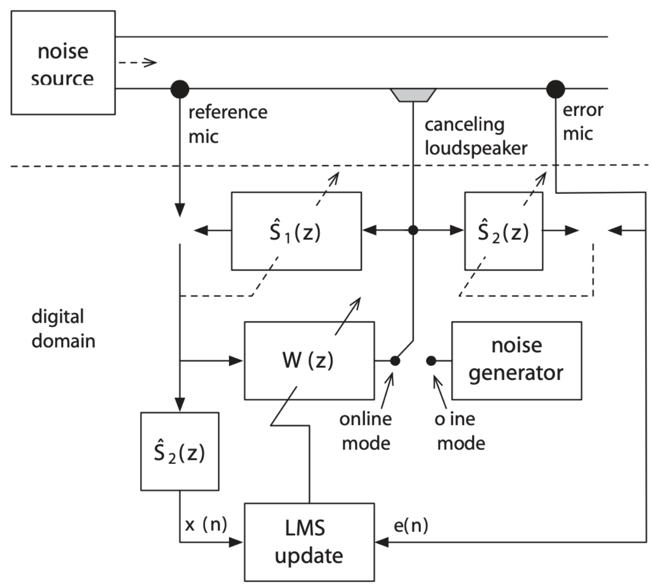

Fig. 5. Block diagram of the Broadband Feedforward System with online and offline modes of operation.

take $W(z) + S_2(z)P(z) = 0$. Therefore, the transfer function of the adaptive filter is :

$$W(z) = -\frac{P(z)}{S_2(z)} \tag{5}$$

Clearly (5) is simpler than (4). Like (4) the adaptive filter can be approximated by a FIR filter driven by an adaptive algorithm.

Finalizing the study of this type of systems it is not superfluous to mention the importance of the processing time. After the reference microphone takes the input signal, the system has a certain time to generate the cancellation speaker signal. If the time it takes to generate this signal (electronic delay) is longer than the time it takes for the noise

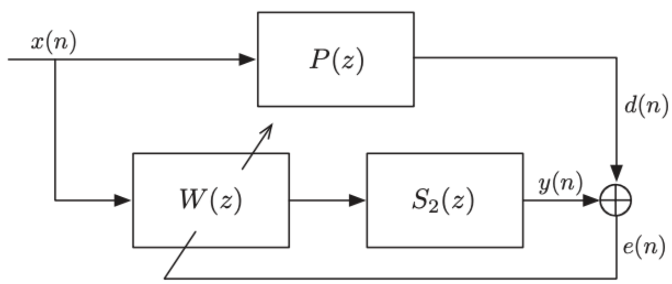

Fig. 6. Block representation of the system *Broadband Feedforward* without S_1.

to get from the reference microphone to the driver (acoustic delay), the efficiency of the system is severely impaired. This is because the system response is non-causal when the electronic *delay* is greater than the acoustic *delay*. When the causality condition is achieved the system is able to suppress random noises. On the contrary, if the causality condition is not achieved, the system can only eliminate periodic noises [4]. As a last clarification note that this system is not limited to periodic noises, this makes it somehow superior to the previous system. That is why this system has more applications than the *Narrowband Feedforward* system.

In conclusion, it can be said that the difficulty in this type of systems lies in the realization of the adaptive filter $W(z)$ and consequently, the algorithm to be used to implement it. For this reason, in the section V its implementation in this type of noise cancellation systems is studied in more detail. In the same way, it is determined that this system is the one that will be simulated in the VI section, since in comparison with the rest of the systems developed in this work, it is the one that has more practical applications and therefore the one of more interest.

III. Narrowband Feedback System

This section introduces the *NarrowBand Feedback* systems. A one-channel block diagram of a *Feedback* system is illustrated in Fig. 7. The signal acquired by the error microphone is processed by the adaptive filter to generate a signal at the noise-cancelling driver. This type of system synthesizes its own reference signal based only on the input from the adaptive filter and the error signal. That is why the idea is to estimate the primary noise and use it as a reference for the adaptive filter.

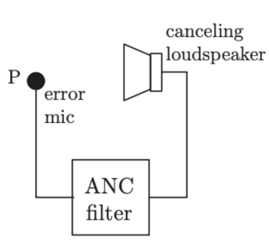

Fig. 7. Basic system representation Narrowband feedback

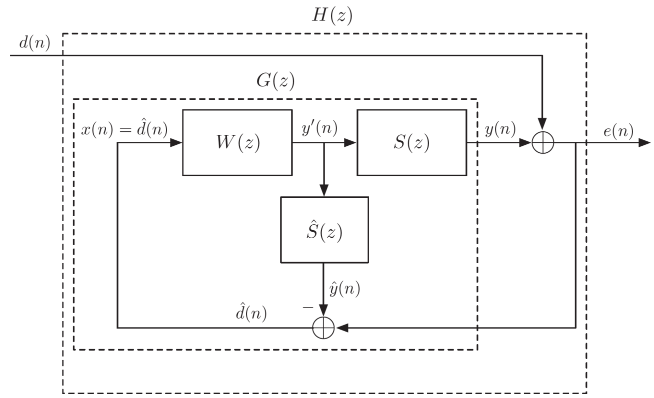

Fig. 8. System representation *Narrowband Feedback*

If $S(z)$ is defined as the acoustic path between the noise-canceling speaker and the error microphone, the system can be diagrammed as shown in Fig. 8. In such a configuration it is assumed that there is an estimate of $S(z)$ called $\hat{S}(z)$. Through it passes the output of $W(z)$, $y'(n)$ and outputs $\hat{y}(n)$. Then the estimate of $d(n)$ is obtained as:

$$\hat{d}(n) = e(n) - \hat{y}(n) \tag{6}$$

Then, when $\hat{d}(n)$ is a good estimate of $d(n)$, to reduce the error $e(n)$ the adaptive filter transfer must be:

$$W(z) = \frac{-1}{S(z)} \tag{7}$$

This scenario entails that $y(n) = -\hat{d}(n)$ and hence $e(n) = d(n) - \hat{d}(n)$ will be a small error signal when $\hat{d}(n)$ is a good estimate. The $W(z)$ configuration may not be implementable in practice due to the possibility that $\frac{1}{S(z)}$ is a noncausal transfer. It is because of this that in practice one has to resort to an approximation $W(z) \approx -1\frac{1}{S(z)}$.

The transfer $H(z)$ of the system is given by:

$$\begin{aligned} H(z) &= \frac{1}{1-G(z)} \\ &= \frac{1+W(z)\hat{S}(z)}{1+W(z)\left(\hat{S}(z)-S(z)\right)} \end{aligned} \tag{8}$$

It is observed that if a good estimate $\hat{S}(z)$ is implemented such that $\hat{S}(z) = S(z)$, then (8) reduces to.

$$H(z) = 1 + W(z)S(z) \tag{9}$$

Furthermore, if $\frac{1}{S(z)}$ is stable and causal, and $W(z)$ is set to $-1/S(z)$, perfect suppression of $d(n)$ will occur. Since the stability and causality of $1/S(z)$ cannot be guaranteed in practice, a compromise decision must be made in the determination of the transfer $W(z)$ to minimize the error $e(n)$, this is where the FX-LMS algorithm developed in section V comes into play.

Fig 9 shows the system including the module implementing the adaptive algorithm.

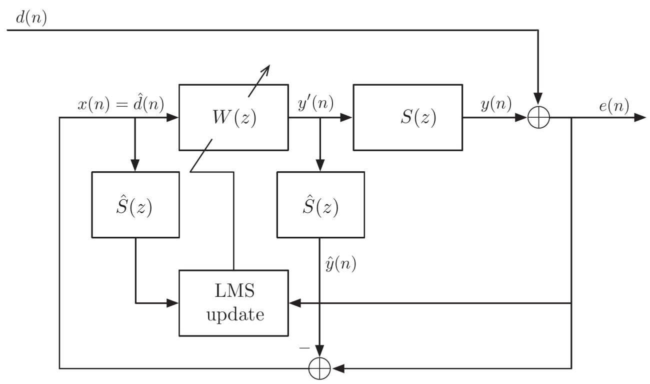

Fig. 9. *Narrowband Feedback* System

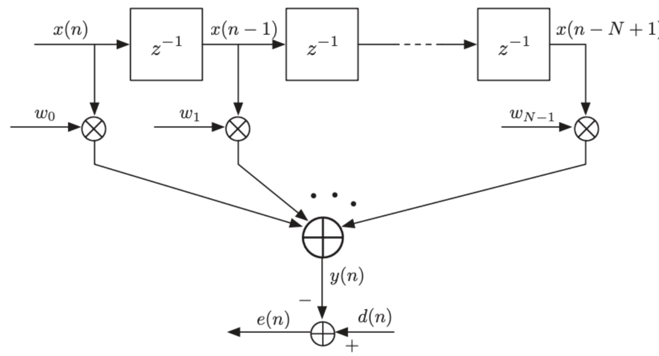

Fig. 11. Cross-sectional filter representation

IV. Wiener adaptive filter

In this section the concept of the Wiener filter is developed since, as seen in the previous sections, it is the adaptive filter most commonly used in noise cancellation systems.

A block diagram of a Wiener filter $W(z)$ is shown in Fig. 10. Here we seek to estimate the signal $d(n)$ based on the signal $x(n)$. It is assumed that both $x(n)$ and $d(n)$ are finite-length samples of random processes. The signal $e(n)$ determines how good the estimate is, where, if the correlation between the $y(n)$ and $d(n)$ signals is maximized, the $e(n)$ signal will be composed purely and exclusively of white Gaussian noise. Using $e(n)$ the filter modifies its coefficients to minimize $e(n)$. The coefficients are modified with an adaptive algorithm which are discussed in the section V.

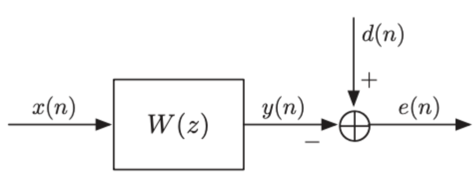

Fig. 10. Block representation of a filtering problem

Before starting with a more detailed explanation of this type of filters, the mean square error criterion is introduced. It arises from the need to determine which filter coefficients produce the smallest error. It is defined as:

$$\varepsilon = E[|e(n|)^2] \tag{10}$$

This concept is mentioned here because it will be very useful in the following.

Continuing with a more detailed explanation of this type of filters, a transversal filter is illustrated in Fig. 11. Both the input $x(n)$ and the output $d(n)$ are assumed to be real values. The filter coefficients named $w_0, w_1, w_2, ...w_{N-1}$ are also real. The coefficient vector and the input vector are defined as follows:

$$\vec{w} = [w_0, w_1, w_2, ...w_{N-1}]^T \tag{11}$$

$$\vec{x(n)} = [x(n), x(n-1), x(n-2), ...x(n-(N-1))]^T \tag{12}$$

The output $y(n)$ is given by:

$$y(n) = \vec{x(n)}^T \cdot \vec{w} \tag{13}$$

The error is given by:

$$\begin{aligned} e(n) &= d(n) - y(n) \\ e(n) &= d(n) - \vec{x(n)}^T \cdot \vec{w} \end{aligned} \tag{14}$$

Using (10) it can be shown that:

$$\begin{aligned} \varepsilon = E[|e(n|)^2] = E[d^2(n)] - \vec{w}^T E[\vec{x(n) \cdot} d(n)] \\ - E[d(n) \cdot \vec{x}^T(n)] \cdot \vec{w} + \vec{w}^T e[\vec{x}(n) \cdot \vec{x}^T(n)] \cdot \vec{W} \end{aligned} \tag{15}$$

If the cross-correlation vector and the autocorrelation matrix are defined as:

$$p_n = E[\vec{x}(n) \cdot d(n)] \tag{16}$$

$$R_n = E[\vec{x}(n) \cdot \vec{x}^T(n)] \tag{17}$$

It is obtained that

$$\varepsilon = E[d^2(n)] - 2 \cdot \vec{w}^T \cdot p_n + \vec{w}^T \cdot R_n \cdot \vec{w} \tag{18}$$

To obtain the coefficients that minimize the function (18) one must solve the system of equations that result from making the partial derivatives of ε with respect to each coefficient equal to zero. It can be shown that one arrives at:

$$\vec{w_0} = R^{-1} \cdot p \tag{19}$$

(19) is known as the equation *wiener-Hopf* where the optimal filter coefficients can be obtained. Replacing the latter equation in 18 gives the minimum mean-squared error:

$$\varepsilon_{min} = E[d^2(n)] - p^T \cdot R^{-1} \cdot p \tag{20}$$

(20) allows to choose the optimal coefficients that minimize the error of the adaptive filter. This result will be used again in the V section.

V. Algorithms

For the implementation of adaptive signal filters, a distinction is made between two categories of algorithms used to minimize the quadratic error between the output and the desired result: those that estimate the filter parameters deterministically and those that do so by means of statistical methods.

The former are characterized by high convergence speeds, but at the same time imply high computational capacity requirements: the recursive least squares method (hereafter RLS) is a prominent example of this category. The latter are based on stochastic methods and therefore require fewer resources: the least mean squares method (LMS) and its variants (NMLS, FX-LMS, etc. are examples of this category).

A brief description of the various relevant algorithms is given below, with emphasis on the stochastic ones, since their computational complexity is relatively low.

A. Recursive Least Squares Algorithm: RLS

The RLS algorithm recursively finds the filter coefficients to minimize the cost function of a weighted linear regression of the input signals: it uses information from the input signals from the beginning of its operation, i.e. it has memory. Because of this, it operates with large information matrices, which implies great mathematical complexity. Although there are methods of this type designed to be more computationally efficient, they fail to compare with stochastic filters, which only deal with the current input. The family of least squares algorithms is known to have numerical conditioning problems. The implementation of this family of algorithms is very sensitive to rounding errors in finite precision systems [8].

B. Least Mean Squares Algorithm: LMS

Starting from the basis of Wiener's filter theory, we think of filters with a non-recursive structure. Therefore, the time expression of the filter output will be determined by (21).

$$y(n) = \sum_{i=0}^{N-1} a_i(n)x(n-i) = a_n^T x_n \tag{21}$$

Being,

$$x_n = [x(n), x(n-1), \dots, x(n-N+1)]^T \tag{22}$$

$$a_n = [a_0(n), a_1(n), \dots, a_{N-1}(n)]^T \tag{23}$$

If a cost function is defined as

$$\Psi(a_n) = E[e^2(n)] \tag{24}$$

$$e(n) = y(n) + d(n) \tag{25}$$

Where $e(n)$ is the error measured at the output of the system. Developing we arrive at the expression,

$$\Psi(a_n) = E[d^2(n)] - 2a_n^T p_n + a_n^T R_n a_n \tag{26}$$

Where,

$$p_n = E[d(n)x_n] \tag{27}$$

y

$$R_n = E[x_n x_n^T] \tag{28}$$

Being p_n the cross-correlation and R_n the auto-correlation. Having defined the cost function as seen in (26), we proceed to minimize it. For this, it is necessary to calculate the gradient of the cost function, since it will be useful to know the direction of motion to minimize the function. After some mathematical development [1], we arrive at the following expression:

$$\nabla\Psi(a_n) = -2p_n + 2R_n a_n \tag{29}$$

From this point on, it could be assumed that it would be sufficient to use the equation obtained from the gradient to apply the gradient descent algorithm and arrive at a minimum of the cost function. However, the dependence of the gradient on p_n and R_n generates problems, since in general those values are not available. It is for this very reason that the estimators seen below are used:

$$\hat{p_n} = d(n)x_n \tag{30}$$

$$\hat{R_n} = x_n x_n^T \tag{31}$$

Therefore, using these estimators and replacing in (29), we obtain:

$$\nabla\Psi(a_n) = -2e(n)x_n \tag{32}$$

Finally, after this mathematical development, and using the same logic in the derivation of the gradient descent algorithm, we obtain (33), which is the so-called LMS algorithm equation.

$$a_{n+1} = a_n + 2\alpha e(n)x(n) \tag{33}$$

1) F-X LMS Algorithm: As explained in section II, the filter transfer is given by $W(z) = -\frac{P(z)}{S_2(z)}$ and then this is approximated to a FIR filter using different algorithms. The *Filtered-X Least-Mean-Square*, better known as FX-LMS, is simply the version of the LMS algorithm applied to FIR-type transfers.

Therefore, if we define $x'(n)$ as the *filtered* version of $x(n)$ we obtain (34) which is the equation of the FX-LMS algorithm [1].

$$a_{n+1} = a_n + 2\alpha e(n)x'(n) \tag{34}$$

It is important to clarify that $x'(n)$ is the anti-transform of $X'(z) = X(z)\hat{S}_2(z)$. This can be clearly seen in Fig. 12.

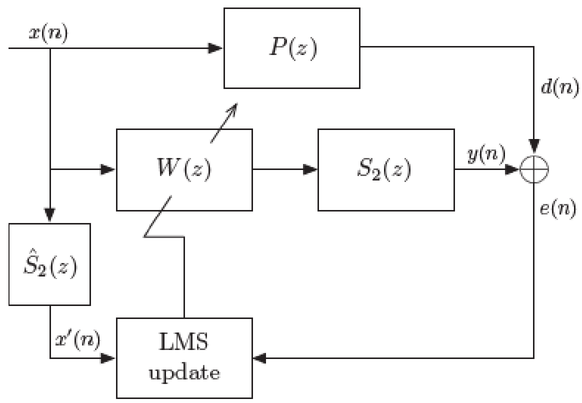

Fig. 12. Implementation of an FXLMS algorithm

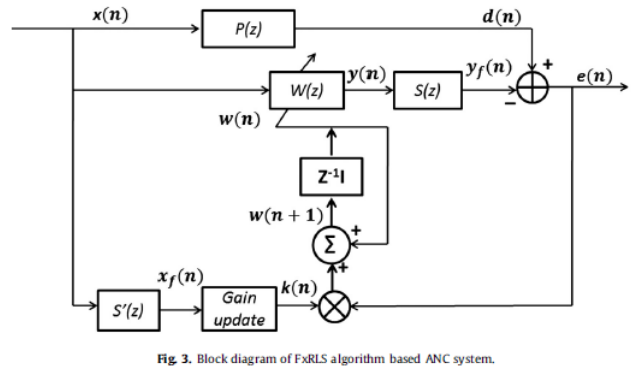

Fig. 13. Diagram of ANC system for FX RLS algorithm update

2) Convergence: Although there will not be a very detailed analysis on how to evaluate and define the parameters for the LMS algorithm to converge, the condition that must be fulfilled to be able to say a priori that the algorithm converges is given by (35).

$$0 < \alpha < \frac{1}{NE[x^2(n)]} \tag{35}$$

The upper bound $\frac{1}{NE[x^2(n)]}$ may not become sufficiently restrictive [1]. This may be due to the *delay* generated by the ANC filter to the input signal before reaching the error microphone, which may generate instability. Therefore, if the $S_2(z)$ filter generates a significant delay to the signal, the upper limit of $alpha$ should be reduced. Another alternative is to place the error microphone closer to the cancellation driver, thus reducing the delay due to the adaptation loop.

3) F-X RLS algorithm: On the other hand, implementing the RLS algorithm for filtered signal, we have the system shown in Fig. 13. Like the FX LMS the input signal is the signal $x'(s)$ and the error is $e(n)$. Therefore, the fitting of the coefficients are done with eqs:

$$k(n) = \frac{P(n-1)x'(n)}{y'^H(n)P(n-1)x'(n) + \lambda} \tag{36}$$

$$P(n) = \lambda^{-1}P(n-1) - \lambda^{-1}k(n)x'^H(n)P(n-1) \tag{37}$$

$$w(n+1) = w(n) + k(n)e(n) \tag{38}$$

As we know, using this algorithm will result in an extremely high convergence speed, but at the cost of computational complexity. This generates problems in our system, since implementing an algorithm of these characteristics in a compact device, such as a headset, would be insufficiently expensive. However, in case we have the capability to use it, we will see the advantages of it below.

VI. Broadband Feedforward ANC Simulation

To carry out the simulation of a *Broadband Feedforward* system we assume the following: 1) the path S_1 is not considered, 2) the path S_2 is defined beforehand but an estimate of it is made so $hatS_2$ or S_2 can be used in the simulation and 3) the adaptive FX-LMS algorithm discussed in section V is used. From 1) and 2) it follows that the result (5) seen in section II is used.

It should be noted that all simulations are performed using Python programming code.

A. Simulation of acoustic paths

In order to perform a simulation of everything discussed above, we must first have the acoustic paths $P(z)$ and $S_2(z)$. For this we make use of the Python library *pyroomacoustics* which provides a simple way to simulate our system. For the acoustics path $P(z)$ we define it as shown in Fig. 14, and its response has the form shown in Fig. 15.

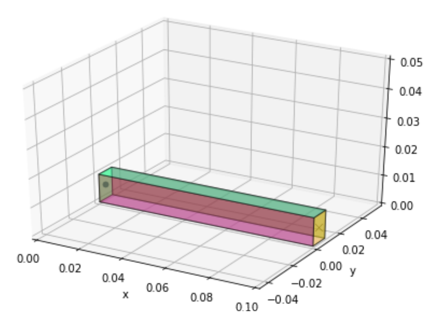

Fig. 14. Physical diagram of the path $P(z)$

On the other hand, for the acoustic path $S_2(z)$ an attempt is made to mimic a overear headphone by defining it as shown in Fig. 16, and obtaining an impulse response as shown in Fig. 17.

B. Simulation with FX LMS

First, $\hat{S}_2$ is simulated to study the FX-LMS algorithm. For the simulation, the LMS algorithm is used in order to

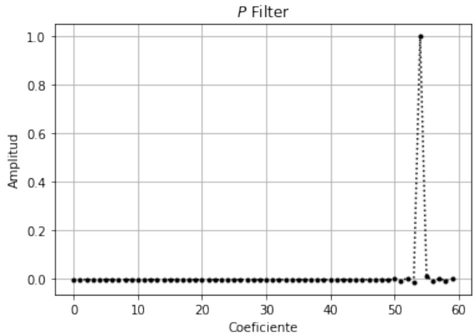

Fig. 15. Impulse response of the path $P(z)$

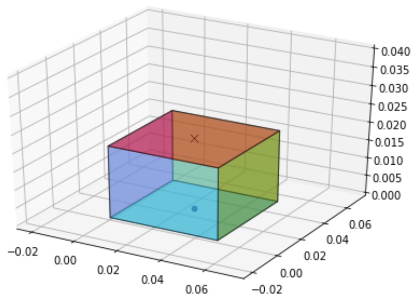

Fig. 16. Physical diagram of the path $S_2(z)$

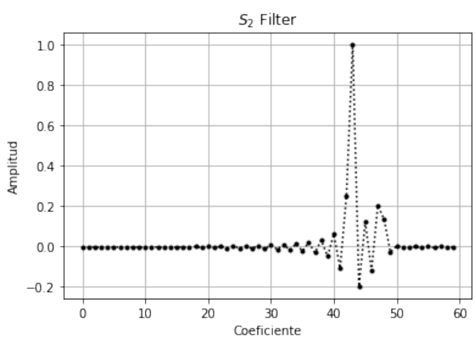

Fig. 17. Impulse response of the path $S_2(z)$

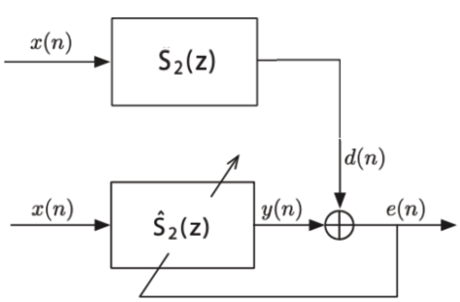

Fig. 18. Block representation of the system to estimate S_2.

approximate in offline mode as best as possible $\hat{S}_2(z)$ to $S_2(z)$.

A block diagram of the estimation procedure is illustrated in Fig. 18. The estimation is performed as follows: 1) place $x(n)$ white Gaussian noise at the input of the system in order to best estimate the filter so that there is no predominance at certain frequencies, 2) define S_2 as previously mentioned, and 3) place an adaptive Wiener filter on $\hat{S}_2$. When performing the simulation, S_2 will be gradually modified by minimizing $e(n)$. The simulation is shown in Fig. 19. As can be seen, the error J_s becomes smaller and smaller as time goes by, which means that the adaptive filter becomes more and more similar to what it wants to estimate, in this case S_2.The whole procedure detailed in this paragraph is what is called *offline* mode in section II.

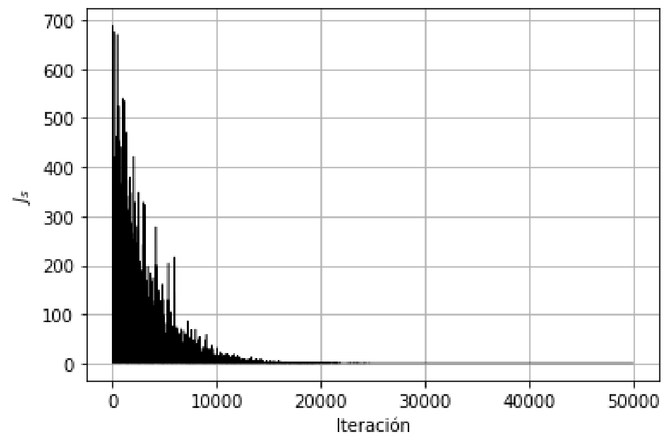

Fig. 19. Simulation of the estimation error of $\hat{S}_2$. Gaussian white noise is set as input $x(n)$.

Having the estimate $\hat{S}_2$ it is possible to simulate the complete system. The test starts with a periodic noise signal. The sound produced in the cockpit of an airplane was selected. The results are shown in Fig. 20. The *System input signal* is the signal that is injected to be nulled, in this case it is the ambient cockpit noise, the *Error signal* is the resulting error signal $e(n)$ which as stated repeatedly, is expected to tend to 0, and is what the pilot would hear in his ears. As can be seen the system works properly since the signal $e(n)$ tends to decrease the sound quickly, and this translates to the adaptive algorithm being quick to modify the adaptive filter, achieving relative silence in the pilot's ears. In Fig. 21 it can be observed in a random segment of the signal, how the signal generated by the adaptive filter tries to obtain the maximum correlation with respect to the desired inverted signal.

Fig. 22 shows the result of a simulation with a random noise. The noise to be suppressed is the noise produced by a fan in a factory. Note that although it seems that the algorithm takes longer to adapt than in the previous case, this is due to the shorter length of this audio.

Finally, Fig. 24 shows how the first 4 coefficients of the adaptive filter converge to their respective values.

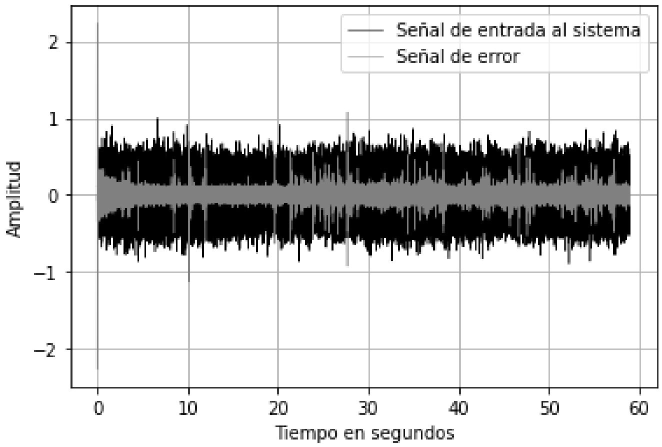

Fig. 20. Simulation of the system inside the aircraft cabin

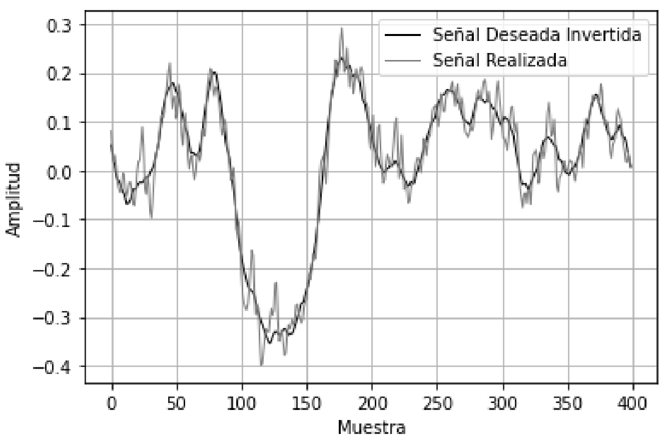

Fig. 21. Sample of the correlation between the desired inverted signal and the signal generated by our system for the aircraft cockpit case.

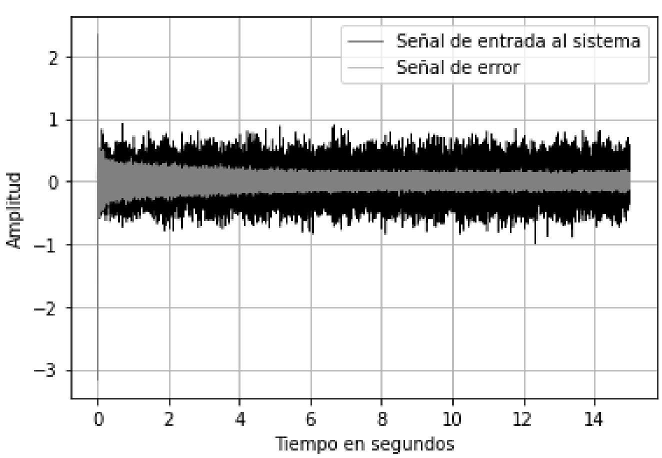

Fig. 22. Simulation of the system with noise from a factory fan

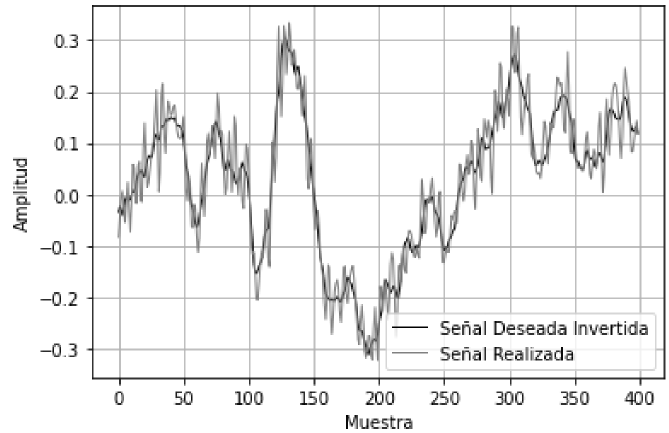

Fig. 23. Sample of the correlation between the desired inverted signal and the signal generated by our system for the case of the factory fan.

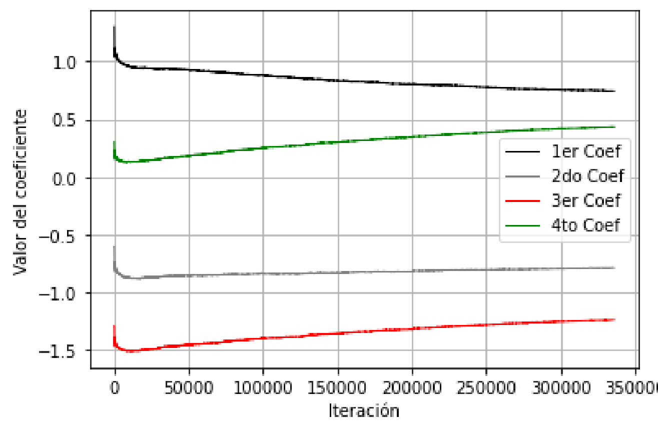

Fig. 24. Adjustment of the first 4 coefficients of the adaptive filter.

The system behaved appropriately under inputs of different nature. This shows that it is suitable for use in, for example, noise cancelling headphones.

C. Simulation with FX RLS algorithm

As with the FX LMS, we first estimated the $\hat{S}_2$ filter, giving a result of J as can be seen in Fig. 25

Finally, we simulated the same environments proposed for the LMS case and arrived at the results shown in Figures 26 and 27.

VII. Conclusion

In this research work, the theory and implementation of three types of noise cancellation systems were developed. It was found that the *Broadband Feedforward* system has the most practical applications and was the most detailed. In conjunction, the FX-LMS algorithm was found to be the most appropriate for the implementation of the adaptive filter. Although the system was successfully simulated, its physical implementation is suggested to corroborate the results obtained in the simulations.

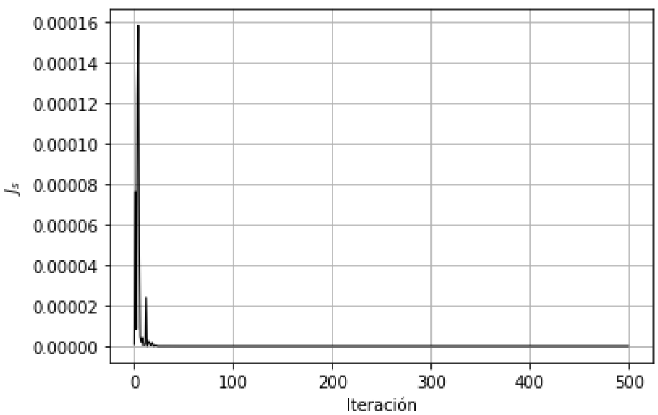

Fig. 25. J min for S_2 filter estimation

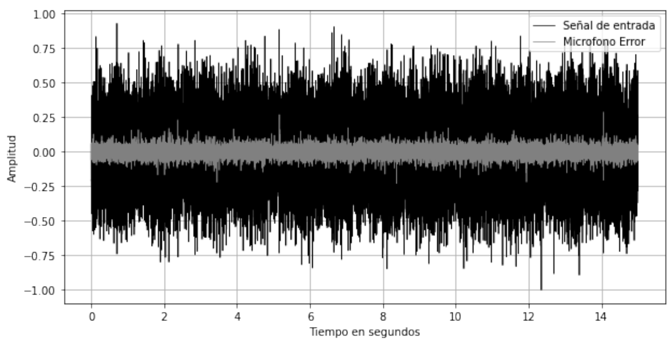

Fig. 26. Simulation of a fan in a factory with the FX RLS algorithm

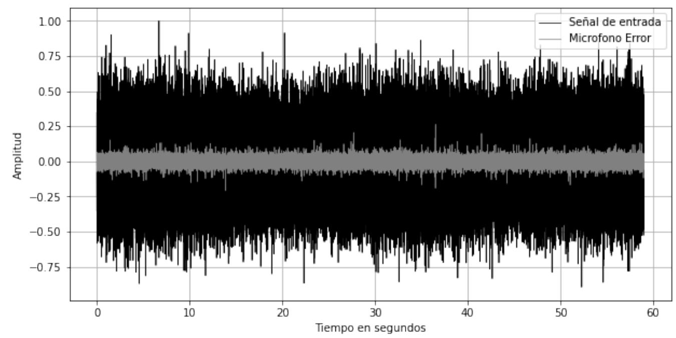

Fig. 27. Simulation of an aircraft cockpit with the FX RLS algorithm

References

[1] B. Farhang and B. Wiley, Adaptive Filters: Theory and Applications (Book style), 2nd ed, pp. 555.

[2] Simon Hayking, Adaptative Filter Theory, 5th ed

[3] D. G. Manolakis, V. K. Ingle and S. M. Kogon, Statstical and Adaptive Signal Processing.

[4] S. M. Kuo and D. R. Morgan, "Review of DSP ALgorithms for Active Noise Control (Presented Conference Paper style)," presented at the International Conference on Control Aplications, Anchorage, Alaska, September 25–27, 2000, Paper 90 SM 690-0 PWRS.

[5] A. Antoniou, Digital Signal Processing: Signals, Systems and Filters (Book style), 1st ed, pp 425

[6] Sen M. Kuo, Sohini Mitra, and Woon-Seng Gan "Active Noise Control System for Headphone Applications"

[7] Reshma B and Kiran K A "Active Noise Control System for Headphone Applications"

[8] On the Numerical Stability and Accuracy of the Conventional Recursive Least Squares Algorithm

[9] Improving performance of FxRLS algorithm for active noise control of impulsive noise Ayesha Zeb, Alina Mirza, Qasim Umar Khan, Shahzad A. Sheikh Department of Electrical Engineering, College of Electrical Mechanical Engineering (CEME), National University of Sciences and Technology (NUST), Rawalpindi 46000, Pakistan