Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 3601802, 15 pages
https://doi.org/10.1155/2017/3601802

Research Article

Control Effort Strategies for Acoustically Coupled Distributed Acoustic Nodes

Christian Antoñanzas, Miguel Ferrer, Maria de Diego, and Alberto Gonzalez

Institute of Telecommunication and Multimedia Applications, Universitat Politecnica de Valencia, Camino de Vera s/n, 46022 Valencia, Spain

Correspondence should be addressed to Christian Antoñanzas; chanma@iteam.upv.es

Received 19 May 2017; Revised 9 September 2017; Accepted 31 October 2017; Published 5 December 2017

Academic Editor: Athanasios Mouchtaris

This paper considers the effect of effort constraints on the behavior of an active noise control (ANC) system over a distributed network composed of acoustic nodes. A distributed implementation can be desirable in order to provide more flexible, versatile, and scalable ANC systems. In this regard, the distributed version of the multiple error filtered-x least mean square (DMEFxLMS) algorithm that allows collaboration between nodes has shown excellent properties. However, practical constraints need to be considered since, in real scenarios, the acoustic nodes are equipped with power constrained actuators. If these constraints are not considered within the adaptive algorithm, the control signals may increase and saturate the hardware devices, causing system instability. To avoid this drawback, a control effort weighting can be considered in the cost function of the distributed algorithm at each node. Therefore, a control effort strategy over the output signals at each node is used to keep them under a given threshold and ensuring the distributed ANC system stability. Experimental results show that, assuming ideal network communications, the proposed distributed algorithm achieves the same performance as the leaky centralized ANC system. A performance evaluation of several versions of the leaky DMEFxLMS algorithm in realistic scenarios is also included.

1. Introduction

Recent advances in electronics are enabling the development of high performance devices increasingly smaller, less expensive and with less power requirements. These electronic devices are usually equipped with electroacoustic transducers, such as sensors and actuators, as well as powerful and efficient processors with communication capability. As wireless communication technologies become affordable, the use of this kind of devices over wireless sensor networks (WSN) [1] has been growing during the last years. A WSN consists of a set of low-power, low-cost, and small-size sensor nodes specifically distributed in some area to perform a certain task. Some advantages of the WSNs compared to the traditional wired networks are scalability and low computational cost [2, 3] among others. In addition, many more sensors can be used to cover larger sound zones in order to get more information from the signals of interest. Different types of WSNs were developed for various applications, including military and security monitoring [4] or healthcare applications [5].

The wireless acoustic sensor networks (WASNs) are specifically designed for acoustic signal processing tasks [6, 7], such as environmental audio monitoring [8, 9], binaural hearing aids [10], and audio surveillance [11] as well as industrial monitoring and control [12]. It seems this kind of networks will be essential for future audio signal acquisition, control, and monitoring. In a WASN, the acoustic node usually consists of several microphones connected to a processing unit with some kind of communication and computation capability [13]. These passive nodes are interested in the estimation of the same network signal or parameter [14] or in solving node-specific estimation problems [15, 16]. The acoustic signals captured by the sensors are recorded and transmitted by the processing unit, doing eventually some processing before the transmission. However, if the WASN has to support a sound field control application, such as active noise control (ANC), nodes capable of generating signals via one or more actuators are required. Furthermore, the WASN should focus not only on the estimation of a certain parameter or signal, but also on the generation of the signals that will feed the actuators in order to control and modify the sound field [17]. Therefore, we consider an acoustic node as a device capable of measuring, processing, and generating signals individually as well as capable of exchanging the necessary information with other nodes using a suitable communication network. From a practical point of view, a WASN could be implemented using smartphones or tablets as acoustic nodes [18]. Nevertheless, the electroacoustic transducers of these electronic devices have limited capability, which may affect their frequency response and the power level of the captured and emitted sounds, and hence the audio application performance. On the other hand, note that, in real-time applications, the selection of the network topology can worsen the system performance, introducing communication delays [12] and requiring the use of synchronization mechanisms [19, 20]. Therefore, an ANC application over a distributed WASN should use a set of acoustic nodes placed strategically to reach the common objective of canceling an undesirable noise in some areas of interest. Every node should process signals independently and when there exists communication among them, it should be able to generate the proper output signal as a result of processing the signal captured by itself as well as the information received from other nodes.

In particular, the objective of a local ANC system is to create a zone of destructive interference by generating the appropriate acoustic waves in order to cancel an undesired noise [21]. To this end, the system makes use of loudspeakers devoted to emit the anti-noise signals to try to reduce the disturbance signal at specific spatial points monitored by microphones. As the global noise reduction in an entire enclosure is unfeasible, we can attempt to control the noise field within a certain area to create local zones of quiet [22]. The greater the wavelength of the signal to be canceled, the larger the zones of quiet. It has been shown [23] that a considerable attenuation can be achieved in an area around the control point with an approximate size of one-tenth of the wavelength of the signal to be canceled. Out of these quiet zones, noise level may even increase [24]. Since the characteristics of the acoustic noise source and the environment are time varying, ANC control system commonly uses adaptive filters [25] in order to deal with these variations [26]. The active noise controller adjusts the adaptive filters coefficients to minimize the noise signal picked up by the microphones. The most common form of adaptive filter for ANC systems is a finite impulse response (FIR) filter using the filtered-x least mean square (FxLMS) approach [27, 28] based on the well-known least mean square (LMS) algorithm [29].

Multichannel ANC systems are used to produce larger zones of quiet and to improve the system performance by adding multiple loudspeakers and microphones. Typically, multichannel ANC systems use a single centralized processor managed by a control algorithm that has access to all the signals generated by the loudspeakers and captured by the microphones. However, these systems require costly infrastructure and they can become unstable since a failure of the single controller means that no information is processed. Moreover, the addition of multiple transducers may increase drastically the computational cost required to capture, manage, and generate multiple signals. A distributed approach is often preferred, especially in terms of flexibility, versatility, and scalability. A distributed system consists of autonomous processors (or acoustic nodes) which control a subset of loudspeakers from the signals picked up by a subset of microphones. These independent nodes are devoted to collaborate to achieve the same solution as a single centralized system but distributing the computational burden as well as the acquisition and signal generation. One of the main problems of the distributed systems is how to share the information between the nodes in a controlled and efficient way.

In the case that there was no acoustic interaction among the nodes (uncoupled nodes), it is possible to achieve the centralized cancelation solution (and consequently, the system stability) by using a decentralized ANC system [30, 31] where the nodes process independently and they do not collaborate at all. But, in most of the multichannel ANC cases, the acoustic interaction among loudspeakers and microphones is present (coupled nodes). In that case, a distributed ANC system over a network of collaborative nodes must be used to reach results equivalent to those of the centralized method. A distributed ANC approach based on the multiple error FxLMS (MEFxLMS) algorithm [32] and using incremental communication strategies [14] was presented in [17] denoted as DMEFxLMS algorithm.

However, it should be noted that, in practical ANC systems, the hardware used to generate output signals at each node has power limitations. In case of saturation of loudspeakers or amplifiers outputs, the control signals generated by the adaptive filters may increase making the system unstable. Note that, in those cases, nonlinearities may cause the system divergence. A possible strategy is based on limiting the control signal power by minimizing the energy of the adaptive filters to avoid the fact that the signals emitted by the loudspeakers may increase unlimitedly. In this case, the objective is to control the signals generated by the adaptive filters at each node in order to limit the amplitude of the signals reproduced by the loudspeakers. Constraint techniques have been widely used in practical control systems [33, 34]. Some of them may be intended for use in real scenarios to improve the processing efficiency [35–37] or even to reduce nonlinearity effects of the system [38]. A common way is to use a leakage during the updating of the control filter coefficients in the LMS algorithm [29]. Because of the addition of bias to the coefficients' update, this leaky LMS algorithm suffers from a degradation in the steady-state error attenuation [25]. On the contrary, it is possible to stabilize the system by controlling the value of the leakage coefficient [30]. A possible solution to improve the performance of the leaky algorithm is to use the clipping and the rescaling strategies [34]. While clipping method just saturates the output signal, the rescaling method also scales the filter weights when the output is too large in order to avoid large oscillations in the coefficients' update. The behavior of these methods is analyzed in [34] for a single-channel ANC system. Regarding the proper value of the leakage coefficient, although it is possible to calculate a range to assure the system linearity [38] or introduce an uneven weight at each node [39], it is usually chosen by trial and error depending on the signal power

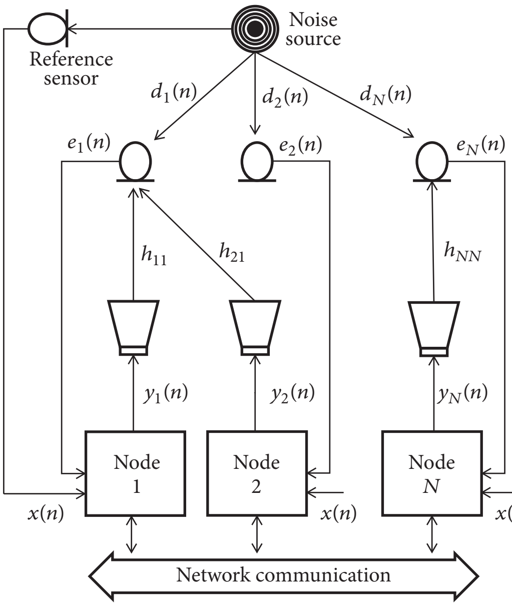


Figure 1: Distributed WASN of N nodes for an ANC system.

supported by the system loudspeakers. In order to provide a more realistic solution for practical implementations, this paper aims to analyze the effect of control effort weighting on the behavior of a distributed ANC system over a WASN. To this end, we analyze the performance of several constrained methods described in [34] when applied over a network with distributed nodes and incremental learning without communication constraints. A study of implementation aspects such as computational complexity and communication capabilities among the nodes in the network for the different control effort strategies is also presented. To our knowledge, no other analysis of this type has been already reported. In addition, we propose an intuitive strategy based on limiting the control signal power to avoid overdriving the loudspeakers that ensures distributed ANC system stability while reducing the communication demands of the applied constrained strategies.

The paper is organized as follows. In Section 2, we derive the distributed ANC solutions with effort constraints for a WASN composed of collaborative nodes. The experimental results carried out to compare the performance of different leaky strategies in realistic scenarios are shown in Section 3. Finally, Section 4 outlines the main conclusions of the present work.

Notation. For the sake of clarity, the following notation has been used throughout this work: letters in italics denote scalars (e.g., x), boldface lowercase letters denote vectors (e.g., $\mathbf{x}$), and boldface uppercase letters denote matrices (e.g., $\mathbf{X}$).

2. Description of the Algorithms

Let us consider an ANC system working over a homogeneous WASN of N nodes spatially distributed in some area, as shown in Figure 1. We assume that all the nodes are composed of a single sensor and a single actuator, execute the same algorithm, and share the same reference signal, $x(n)$, captured by a reference sensor used to detect a single disturbance noise at the discrete time instant n. Our objective is to estimate an adaptive filter $\mathbf{w}_k(n)$ at every node to cancel the acoustic noise signal at the sensor locations, $d_k(n)$ (where $k = 1, 2, \ldots, N$). To that end, the control signals $y_j(n)$ (where $j = 1, 2, \ldots, N$), emitted by the actuators and filtered through the acoustic system, are designed to minimize the signals recorded at the sensors, called error signals and denoted by $e_k(n)$. The acoustic channel that links the actuator of the node j and the sensor of the node k ($\mathbf{h}_{jk}$) is usually estimated by means of a FIR filter of M coefficients denoted as $\mathbf{s}_{jk}$.

There exist several ways to achieve this objective depending on the selected strategy. Following [17], we start from a network centralized approach to derive into the contribution of every node in a distributed network using an incremental strategy of the data exchange and assuming practical constraints into the solution.

2.1. Centralized MEFxLMS Algorithm Using Control Effort.

Firstly, we consider a centralized strategy where a single processor receives and transmits all the information through the network (see Figure 2(a)). This central unit is required because all the error signals are necessary to calculate the coefficients of each filter [32]. Note that if we gather the signals involved in the ANC system depicted in Figure 1 into the following vectors

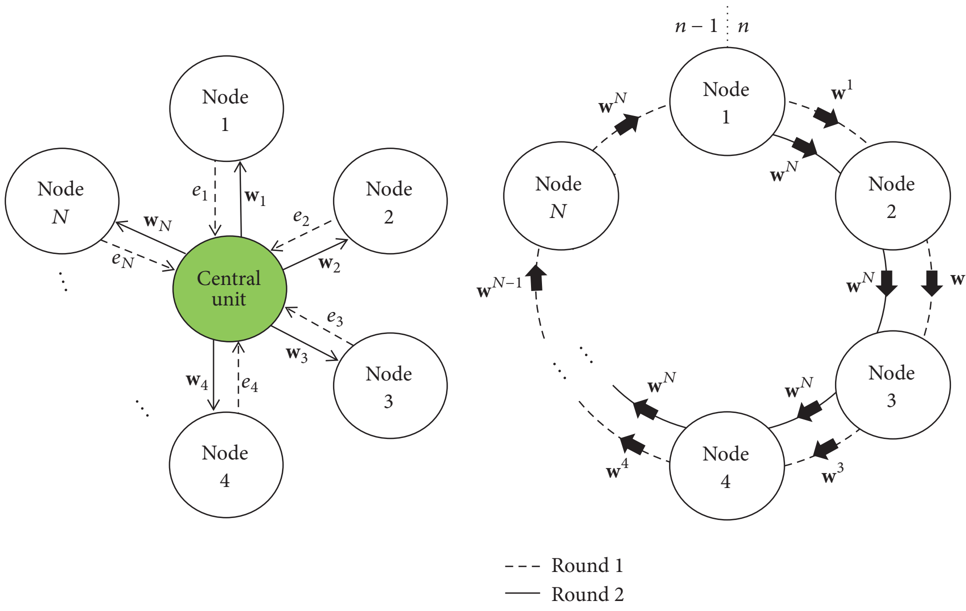
$$\begin{aligned}\mathbf{e}(n) &= \left[e_1(n)\ \ e_2(n)\ \ \cdots\ \ e_N(n)\right]^T, \\ \mathbf{d}(n) &= \left[d_1(n)\ \ d_2(n)\ \ \cdots\ \ d_N(n)\right]^T,\end{aligned} \tag{1}$$

the information captured by all the error sensors of the network is defined as

$$\mathbf{e}(n) = \mathbf{d}(n) + \mathbf{V}^T(n)\,\mathbf{w}(n), \tag{2}$$

where vector $\mathbf{w}(n) = \left[\mathbf{w}_1^T(n)\ \ \mathbf{w}_2^T(n)\ \ \cdots\ \ \mathbf{w}_N^T(n)\right]^T$ of size $[LN{\times}1]$ concatenates the N adaptive filters $\mathbf{w}_k(n)$ that contain the L filter coefficients of the kth node. Matrix $\mathbf{V}(n) = \left[\mathbf{v}_1(n)\ \ \mathbf{v}_2(n)\ \ \cdots\ \ \mathbf{v}_N(n)\right]$ is the concatenation of N vectors of size $LN \times 1$ defined as $\mathbf{v}_k(n) = \left[\mathbf{v}_{1k}^T(n)\ \ \mathbf{v}_{2k}^T(n)\ \ \cdots\ \ \mathbf{v}_{Nk}^T(n)\right]^T$ being $\mathbf{v}_{jk}(n)$ an L-length vector that contains the last L samples of reference signal $x(n)$ filtered through $\mathbf{s}_{jk}$. The ANC system tries to minimize a cost function that depends on the acoustic field to be controlled. The most commonly used centralized adaptive strategy for that purpose is the multiple error FxLMS algorithm (MEFxLMS) [32] which is devoted to minimize the sum of the power of the N instantaneous error signals. But, in order to minimize both error and control signals, an effort penalty may be applied over the adaptive filters minimizing the following cost function:

$$\begin{aligned}\mathbf{J}(n) &= \sum_{k=1}^{N} e_k^2(n) + \sum_{k=1}^{N} \beta_k \mathbf{w}_k^T(n)\,\mathbf{w}_k(n) \\ &= \mathbf{e}(n)^T\,\mathbf{e}(n) + \mathbf{w}^T(n)\,\boldsymbol{\beta}\mathbf{w}(n),\end{aligned} \tag{3}$$



(a) (b)

Figure 2: (a) A centralized WASN and (b) a ring topology distributed WASN with incremental communication. In (b), data transfer rounds are represented with different types of lines.

where $\boldsymbol{\beta}$ is a diagonal matrix of size $LN \times LN$ whose diagonal elements are the N values of β_k replicated L times. β_k is the leakage coefficient used to adjust the amplitude of the adaptive filter coefficients of the node k. This leak factor can be viewed as a regularization parameter which uses values less than 1 causing a gain reduction over the adaptive filter coefficients. Note that (3) is similar to the cost function of a multichannel centralized system composed of N loudspeakers and N microphones. As the central unit uses a gradient-descent method to estimate the adaptive filter coefficients of the network $\mathbf{w}(n)$, the global filter updating equation of the centralized leaky MEFxLMS (l-CMEFxLMS) algorithm is stated as follows:

$$\mathbf{w}(n) = \mathbf{w}(n-1) - \mu\left(\boldsymbol{\beta}\mathbf{w}(n-1) + \sum_{k=1}^{N}\mathbf{v}_k(n)\,e_k(n)\right), \quad (4)$$

where μ is the step size parameter. Given the well-known drawbacks of the centralized system, such as large computational and communication demands, the use of a distributed network is required. A distributed WASN involves not only the nodes which are physically distributed within a specific area, but also the processing which is divided among the nodes. Hence, the implementation of (4) over a network of distributed nodes will be presented in the following section.

2.2. Collaborative Distributed Algorithm Using Control Effort. Now, the goal is to distribute the calculation of $\mathbf{w}(n)$ among the N nodes of the WASN described in Figure 1 but considering a ring topology with an incremental strategy when there are no communication constraints in the network (see Figure 2(b)). Therefore, the data exchange is carried out in a consecutive order. To clarify the communications cycle, a differentiation between consecutive time indexes, $n-1$ and n, is represented in Figure 2(b). Note that, at each sample time n, all the necessary data transfer rounds, in which nodes interchange their information, must be completed. We must split up the sum of the global updating equation into the contributions of each node; that is, the kth term in the sum of (4) can be only calculated by the kth node. Defining $\mathbf{w}(n)$ as the global state of the network and considering $\mathbf{w}^k(n)$ as a local version of $\mathbf{w}(n)$ at the kth node, from node $k=1$ to node $k=N$, we can split up the contribution of each node in (4) as described in [17]:

$$\begin{aligned}
\mathbf{w}^1(n) &= \mathbf{w}^0(n) - \mu\left(\frac{\boldsymbol{\beta}}{N}\mathbf{w}(n-1) + \mathbf{v}_1(n)\,e_1(n)\right),\\
\mathbf{w}^2(n) &= \mathbf{w}^1(n) - \mu\left(\frac{\boldsymbol{\beta}}{N}\mathbf{w}(n-1) + \mathbf{v}_2(n)\,e_2(n)\right),\\
&\vdots\\
\mathbf{w}^N(n) &= \mathbf{w}^{N-1}(n) - \mu\left(\frac{\boldsymbol{\beta}}{N}\mathbf{w}(n-1) + \mathbf{v}_N(n)\,e_N(n)\right),
\end{aligned} \quad (5)$$

where $\mathbf{w}^0(n) = \mathbf{w}^N(n-1) = \mathbf{w}(n-1)$. Note that since every node calculates its portion of the sum and supplies to the following node its partial result, the last node, Nth node, contains the complete updated coefficients. Therefore, the updating filter equation of the kth node by using the leaky DMEFxLMS algorithm (l-DMEFxLMS) can be expressed as

$$\mathbf{w}^k(n) = \mathbf{w}^{k-1}(n) - \mu\left(\frac{\boldsymbol{\beta}}{N}\mathbf{w}(n-1) + \mathbf{v}_k(n)\,e_k(n)\right). \quad (6)$$

In order to achieve the same performance as the centralized strategy, the updated coefficients of the last node, $\mathbf{w}^N(n)$, must be disseminated to the rest of the nodes for the $(n+1)$th iteration. The reason is because every node needs the global state of the network at the previous iteration ($\mathbf{w}(n-1)$) to perform the partial updating of its coefficients. This means that $2(N-1)$ interchanges of the filter coefficients among the nodes per each sample are carried out (Figure 2(b)). In order to carry out a simple and transparent data exchange, we consider that every node can interchange only its local state of the network with its following node. Since $\mathbf{w}(n-1)$ is unavailable in (6), each node uses the local state of the last node, which contains the global state of the network at the previous iteration ($\mathbf{w}^0(n) = \mathbf{w}^N(n-1) = \mathbf{w}(n-1)$). It should be noted that only the local information, $\mathbf{w}_k(n) = \mathbf{w}^k(n-1)_{(1+L(k-1):Lk)}$, is needed to generate the kth node output signal $y_k(n)$.

As commented previously, we have considered homogeneous nodes to achieve the updating equation presented in (4). This implies that all nodes compute the same operations and execute the same algorithm. However, considering a WASN composed by nonhomogeneous nodes, (4) might be computed in a different manner. For instance, the first node, which updates its coefficients in the incremental sequence, could assume the full computation of the control effort releasing the remaining nodes to perform them. More specifically, the first node could calculate its local state as

$$\mathbf{w}^1(n) = \mathbf{w}^0(n) - \mu\left(\boldsymbol{\beta}\mathbf{w}(n-1) + \mathbf{v}_1(n)\, e_1(n)\right), \qquad (7)$$

while the rest of the nodes could update their coefficients as

$$\mathbf{w}^p(n) = \mathbf{w}^{p-1}(n) - \mu\mathbf{v}_p(n)\, e_p(n), \qquad (8)$$

where $p = 2, 3, \ldots, N$. Thus, the dissemination of the updated coefficients can be eliminated reducing the communications demands at the expenses of an increase in the computational cost of the first node.

On the other hand, the clipping strategy described in [34] could be applied to the l-DMEFxLMS algorithm. That strategy addresses the problem of saturation in amplifiers or loudspeakers by limiting the output power. If the output signal power is greater than an upper threshold, a simple solution lies in limiting the output power to the threshold value. Defining $y_{k_{\max}}$ as the maximum allowed value of the output signal at each node k, the clipping l-DMEFxLMS algorithm is given by

$$\begin{aligned}
&\text{if } \left|\underline{y}_k(n)\right| > y_{k_{\max}}, \\
&\qquad \underline{y}_k(n) = \underline{y}_k(n)\,\alpha_k \frac{y_{k_{\max}}}{\left|\underline{y}_k(n)\right|}, \quad (0 < \alpha_k < 1) \\
&\text{end if} \\
&y_k(n) = \underline{y}_k(n),
\end{aligned} \qquad (9)$$

where $\underline{y}_k(n)$ is defined as the provisional output signal at kth node, calculated as

$$\underline{y}_k(n) = \mathbf{w}_k^T(n)\,\mathbf{X}(n), \qquad (10)$$

being $\mathbf{X}(n)$ a properly arranged matrix of size $[L \times M]$ defined as

$$\mathbf{X}(n) = \begin{bmatrix} x(n) & x(n-1) & \cdots & x(n-M+1) \\ x(n-1) & x(n-2) & \cdots & x(n-M+2) \\ \vdots & \vdots & \ddots & \vdots \\ x(n-L+1) & x(n-L+2) & \cdots & x(n-(L+M)+2) \end{bmatrix}. \qquad (11)$$

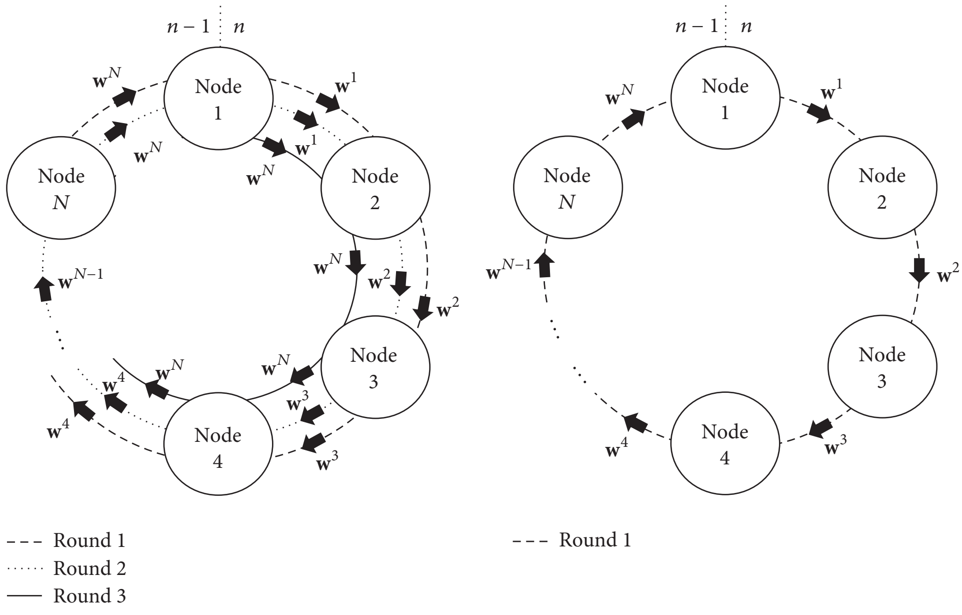
Depending on the value of the parameter α_k, a saturation ($\alpha_k = 1$) or compression ($0 < \alpha_k < 1$) effect on the dynamic range of the output may be applied at a certain threshold $y_{k_{\max}}$. To avoid large oscillations in the coefficients' update, the rescaling method could be applied to the clipping l-DMEFxLMS algorithm leading to the rescaling l-DMEFxLMS algorithm. To this end, it is only necessary to add to (9) the rescaling of the adaptive filters as follows:

$$\begin{aligned}
&\text{if } \left|\underline{y}_k(n)\right| > y_{k_{\max}}, \\
&\qquad \underline{y}_k(n) = \underline{y}_k(n)\,\alpha_k \frac{y_{k_{\max}}}{\left|\underline{y}_k(n)\right|}, \\
&\qquad \underline{\mathbf{w}}_k(n) = \mathbf{w}^k(n)_{(L(k-1)+1:Lk)}\;\; \alpha_k \frac{y_{k_{\max}}}{\left|\underline{y}_k(n)\right|}, \\
&\qquad \mathbf{w}^k(n)_{(1+L(k-1):Lk)} = \underline{\mathbf{w}}_k(n), \\
&\text{end if} \\
&y_k(n) = \underline{y}_k(n).
\end{aligned} \qquad (12)$$

Algorithm 1 illustrates a summary of the rescaling l-DMEFxLMS algorithm pseudocodes, which are executed per sample time at each node. Comparing (9) and (12), it can be seen that while the clipping l-DMEFxLMS algorithm only rescales the output, the rescaling l-DMEFxLMS algorithm rescales both the output and the filter coefficients. The dual rescaling prevents stability problems since the coefficients update is uncorrelated with the filter output when the clipping strategy is working [34, 36]. It is important to take into account the fact that the system stability is ensured by applying the suitable constraints over the output signal, although too restrictive saturation levels could result in performance impairments. Furthermore, the rescaling l-DMEFxLMS algorithm requires a higher data transfer speed compared to the l-DMEFxLMS algorithm. This result from the fact that $3(N-1)$ coefficients are exchanged among the nodes (see Figure 3(a)): a first round where each node passes the global state of the network at the previous iteration, $\mathbf{w}(n-1)$, to the following node; a second round where each node receives the information of the previous node, $\mathbf{w}^{k-1}(n)$, calculates its local version $\mathbf{w}^k(n)$ with the help of $\mathbf{w}(n-1)$, and supplies to the following node its partial result; and finally, a third round where each node rescales its portion within $\mathbf{w}^k(n)$, passes its local state to the following node, and generates its output signal with the rescaled coefficients.

```
for all node $1 \le k \le N$ do
    $\mathbf{w}^k(n) = \mathbf{w}^{k-1}(n)$   % Copy local state of previous node (at node $k = 1$, $\mathbf{w}^0(n) = \mathbf{w}(n-1)$)
    $\mathbf{w}_k(n) = \mathbf{w}^k(n)_{(1+L(k-1):Lk)}$   % Obtain local coefficients to generate the output signal
    $\underline{y}_k(n) = \mathbf{w}_k^T(n)\mathbf{X}(n)$   % Provisional output signal
    for all $1 \le j \le N$ do
        $\mathbf{v}_{jk}(n) = \mathbf{X}(n)\mathbf{s}_{jk}$   % Vector that contains reference signal filtered by estimated acoustic channels
    end for
    $\mathbf{w}^k(n) = \mathbf{w}^{k-1}(n) - \mu((\boldsymbol{\beta}/N)\mathbf{w}^k(n) + \mathbf{v}_k(n)e_k(n))$   % Update local state
    if $|\underline{y}_k(n)| > y_{k_{\max}}$   % If provisional output signal is greater than threshold
        $\underline{y}_k(n) = \underline{y}_k(n)\alpha_k(y_{k_{\max}}/|\underline{y}_k(n)|)$   % Rescale provisional output signal
        $\underline{\mathbf{w}}_k(n) = \mathbf{w}^k(n)_{(L(k-1)+1:Lk)}\ \alpha_k(y_{k_{\max}}/|\underline{y}_k(n)|)$   % Rescale its portion within its local state
        $\mathbf{w}^k(n)_{(1+L(k-1):Lk)} = \underline{\mathbf{w}}_k(n)$   % Updated local state with rescaled coefficients
    end if
    $y_k(n) = \underline{y}_k(n)$   % Generate output signal
end for
$\mathbf{w}(n) = \mathbf{w}^N(n)$   % Updated global state of the network properly rescaled
for all node $1 \le k \le (N-1)$ do
    $\mathbf{w}^k(n) = \mathbf{w}(n)$   % Disseminate global state of the network
end for
```

Algorithm 1: Rescaling l-DMEFxLMS algorithm for N-nodes WASN.



(a)

(b)

Figure 3: Two ring topology distributed WASNs with incremental communication using (a) the rescaling l-DMEFxLMS algorithm and (b) the 1r rescaling l-DMEFxLMS algorithm. Data transfer rounds are represented with different types of lines.

Thus, the last node will obtain the updated global state of the network properly rescaled which will be necessary at the time of dissemination to the rest of the nodes in the first round of the following iteration. Note that, in the last round, each node may supply only its rescaled portion directly to the last node. However, it would be necessary a different communication strategy to fulfill this solution.

With the aim of reducing the communication demands of this algorithm, a one-round strategy is proposed (1r rescaling l-DMEFxLMS). Using this method, in one round each node receives the information from its precedent node, $\mathbf{w}^{k-1}(n)$, calculates its local version $\mathbf{w}^k(n)$ with the help of its information at the previous iteration, $\mathbf{w}^k(n-1)$, and supplies to the following node its partial result with its portion previously rescaled. At the same time, each node generates its rescaled output signal. Note that, in this case, instead of using the global state of the network at the previous iteration in the updating equation, each node uses its local state at

```
for all node $1 \le k \le N$ do
    $\mathbf{w}_k(n) = \mathbf{w}^k(n)_{(1+L(k-1):Lk)}$   % Obtain local coefficients to generate the output signal
    $\underline{y}_k(n) = \mathbf{w}_k^T(n)\mathbf{X}(n)$   % Provisional output signal
    for all $1 \le j \le N$ do
        $\mathbf{v}_{jk}(n) = \mathbf{X}(n)\mathbf{s}_{jk}$   % Vector that contains reference signal filtered by estimated acoustic channels
    end for
    $\mathbf{w}^k(n) = \mathbf{w}^{k-1}(n) - \mu((\boldsymbol{\beta}/N)\mathbf{w}^k(n-1) + \mathbf{v}_k(n)e_k(n))$   % Update local state
    if $|\underline{y}_k(n)| > y_{k_{\max}}$   % If provisional output signal is greater than threshold
        $\underline{y}_k(n) = \underline{y}_k(n)\alpha_k(y_{k_{\max}}/|\underline{y}_k(n)|)$   % Rescale provisional output signal
        $\underline{\mathbf{w}}_k(n) = \mathbf{w}^k(n)_{(L(k-1)+1:Lk)}\ \alpha_k(y_{k_{\max}}/|\underline{y}_k(n)|)$   % Rescale its portion within its local state
        $\mathbf{w}^k(n)_{(1+L(k-1):Lk)} = \underline{\mathbf{w}}_k(n)$   % Updated local state with rescaled coefficients
    end if
    $y_k(n) = \underline{y}_k(n)$   % Generate output signal
end for
```

Algorithm 2: 1r rescaling l-DMEFxLMS algorithm for N-nodes WASN.

the previous iteration, $\mathbf{w}^k(n-1)$. Thus, the dissemination of the updated coefficients is avoided and it is possible to reduce the data transfer among the nodes making $(N-1)$ interchanges of the filter coefficients, as it can be seen in Figure 3(b). Algorithm 2 illustrates a summary of the 1r rescaling l-DMEFxLMS algorithm pseudocodes, which are executed per sample time at each node. Note that since the updated coefficients are not fully shared among the nodes, the behavior of the 1r rescaling l-DMEFxLMS algorithm is not exactly the same as the rescaling approach. Because the coefficients exchange is carried out in a consecutive order, those differences may result more relevant in larger WASN. This is a consequence of the incremental learning where each node computes a part of the global filter, aggregates it to the given filter, and passes it to the following node. In this way, nodes closer to the last node have a more accurate estimation of the global filter. This could deteriorate the convergence speed of the 1r rescaling l-DMEFxLMS algorithm in comparison to the rescaling l-DMEFxLMS strategy. However, in the case of WASN composed of few nodes, both methods will obtain a similar performance in terms of convergence speed and noise reduction. A performance comparison between all the proposed algorithms is presented in the following section.

3. Simulation Results

In this section, we present the simulations carried out to evaluate the performance of the presented distributed algorithms over networks with no communication constraints. In a first stage, we have compared the performance of both the l-CMEFxLMS and l-DMEFxLMS algorithms in order to validate the theoretical solution outlined in Section 2.2. In a second stage, we have justified the use of the constrained techniques comparing the performance of the DMEFxLMS algorithm (described in [17]) and its leaky version (l-DMEFxLMS). Subsequently, we have validated the need of using the clipping and rescaling methods (analyzed in [34]) in order to fulfill the loudspeakers output constraint as well as the use of the proposed one-round strategy (1r rescaling l-DMEFxLMS) to reduce the communication demands of the network. To this end, we have compared the l-DMEFxLMS, the clipping l-DMEFxLMS, the rescaling l-DMEFxLMS, and the 1r rescaling l-DMEFxLMS algorithms in terms of final noise reduction, convergence behavior, computational complexity, and communication requirements. All the algorithms have been tested in a homogeneous WASN composed of four single-channel nodes, as shown in Figure 4. Only the nodes with the best and the worst performance are shown in the simulations in order to assess the behavior of the WASN. The performance of the other nodes remains within this range. For the designed WASN, we use real acoustic channels measured in a listening room of 9.36 meters long, 4.78 meters wide, and 2.63 meters high, located at the Audio Processing Laboratory of the Polytechnic University of Valencia [40], and modelled as FIR filters of $M = 256$ coefficients at a sampling rate of 2 kHz. The configuration of the simulated WASN is depicted as follows: four nodes composed of one loudspeaker and one microphone were considered. An equal separation of 20 cm between adjacent loudspeakers was selected. The microphones were placed

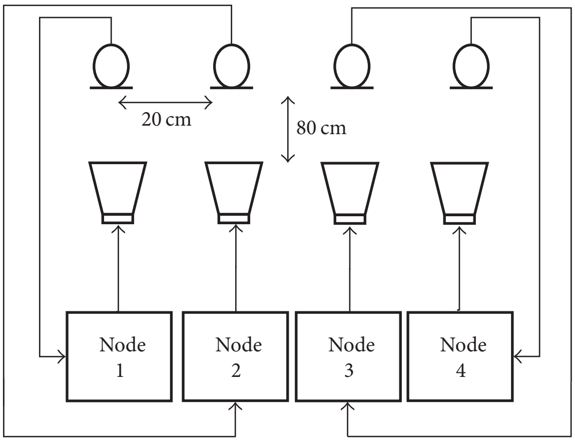


Figure 4: WASN of four nodes for an ANC system.

opposite to the loudspeakers and separated 80 cm away from them. The separation between the microphones was 20 cm. All microphones and loudspeakers involved were located at a height of 147 cm. The tested distribution emulates a real ANC application where we would seek to create local quiet zones in enclosures (such as a cabin of a public transport) using WASNs of acoustic nodes with similar separation as detailed above. We have considered a wideband zero-mean Gaussian white noise with unit variance as disturbance signal and an adaptive filter length of $L = 150$ coefficients. An initial step size parameter of $\mu = 0.0125$ as the highest value that ensures the stability of the algorithms has been used in the first simulations. Furthermore, we have considered a maximum allowed value of the output signals of $y_{k_{\max}} = 1.0$ and an attenuation parameter of $\alpha_k = 1$ for all the nodes and all the simulations. Initially, we use a leakage parameter of $\beta_k = 0.01$ for all the nodes. The reason of this leakage value is explained below.

In order to evaluate the performance of the different algorithms, we define the instantaneous noise reduction at node k, $\mathrm{NR}_k(n)$, as the ratio in dB between the estimated error powers with and without the application of the active noise controller:

$$\mathrm{NR}_k(n) = 10 \cdot \log_{10}\left[\frac{e_k^2(n)}{d_k^2(n)}\right], \tag{13}$$

where $d_k^2(n)$ is the signal power picked up at the kth microphone when the ANC system is inactive and $e_k^2(n)$ is the error signal power measured at the kth microphone when the ANC system works. Moreover, these signals powers have been estimated by applying an exponential windowing to the instantaneous signals.

3.1. Comparison between Centralized and Distributed Leaky Approaches. In the first simulation, we compare the noise reduction of a leaky centralized ANC system with one reference signal, four actuators, and four sensors (1 : 4 : 4 configuration) and a leaky distributed ANC system with 4 single-channel nodes. Figure 5 shows the time evolution of the $\mathrm{NR}_k(n)$ for both the l-CMEFxLMS and the l-DMEFxLMS algorithms for the microphone with best and worst performance in the centralized case and the node with the best and worst performance in the distributed case. As expected, the distributed implementation exhibits exactly the same results as the centralized version in terms of convergence speed and final residual noise, providing an attenuation up to 12 dB for the worst node and almost 14 dB for the best node.

3.2. Improving the Performance of the DMEFxLMs in Practical Scenarios. In the second simulation, we compare the performance of the DMEFxLMS algorithm with the l-DMEFxLMS strategy in order to justify its use in practical scenarios. As the previous simulation, an ANC system over a 4-single-channel-node WASN has been considered. Figure 6 shows the $\mathrm{NR}_k(n)$ for both algorithms at the nodes with the best and the worst performance. Two behaviors of the DMEFxLMS algorithm have been differentiated: its performance in a real scenario (denoted as real DMEFxLMS) and its performance in an ideal scenario (denoted as ideal DMEFxLMS). The real version emulates how the loudspeaker saturation influences the behavior of the distributed algorithm. On the other hand, the effect of the loudspeaker saturation is not considered in the ideal version. The time variations of the control signal for both algorithms at the worst node are shown in Figures 7(a) and 7(b), respectively. The amplitude of the control signals is normalized being +1 and −1 the maximum and the minimum signal amplitudes that can be produced by the loudspeakers. It can be shown that both the ideal and the real DMEFxLMS behaviors initially outperform the l-DMEFxLMS algorithm in terms of convergence speed and noise reduction but when they reach certain point, the real version turns unstable and does not converge. This is because the DMEFxLMS method fails to satisfy the control output constraint as shown in Figure 7(a). Consequently, the appearance of nonlinearities due to the loudspeakers saturation leads to the divergence of the DMEFxLMS algorithm. To avoid this, we use the l-DMEFxLMS algorithm with $\beta_k = 0.01$ for all the nodes selected by trial and error. Thus, the l-DMEFxLMS algorithm fulfills the constraint by limiting the maximum output signal (see Figure 7(b)), achieving a $\mathrm{NR}_k(n)$ around 10 dB for the best and the worst node, as it can be seen in Figure 6. For simplicity, and since the use of the leaky method in real scenarios has been justified, the effect of loudspeakers saturation on the performance of the leaky algorithms has not been considered in the next simulations.

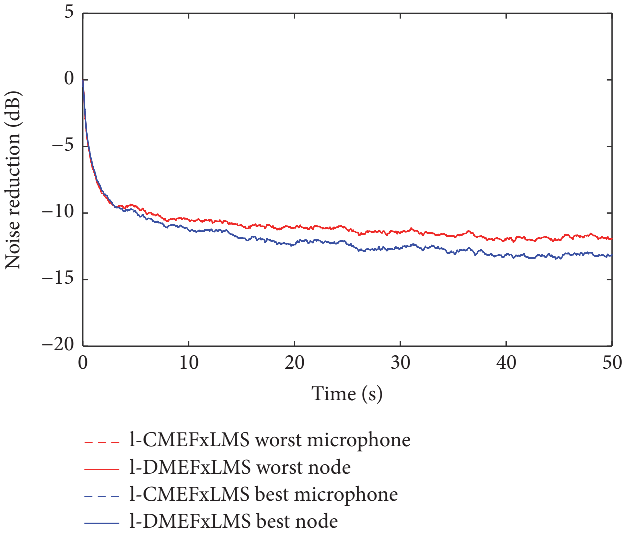


Figure 5: Noise reduction of the distributed system (solid line) with 4 single-channel nodes and the centralized system (dashed line) with a 1 : 4 : 4 configuration with $\beta_k = 0.01$ represented for the best and worst microphone.

3.3. Comparison between Distributed Leaky Approaches. Furthermore, it is possible to improve the performance of the l-DMEFxLMS algorithm, in terms of final attenuation, fulfilling the output signal constraint addressed by the clipping and the rescaling l-DMEFxLMS algorithms. This can be seen in Figures 8 and 9 where the leakage parameter β_k was selected as 0.001 for all nodes of a 4-node WASN, in order to

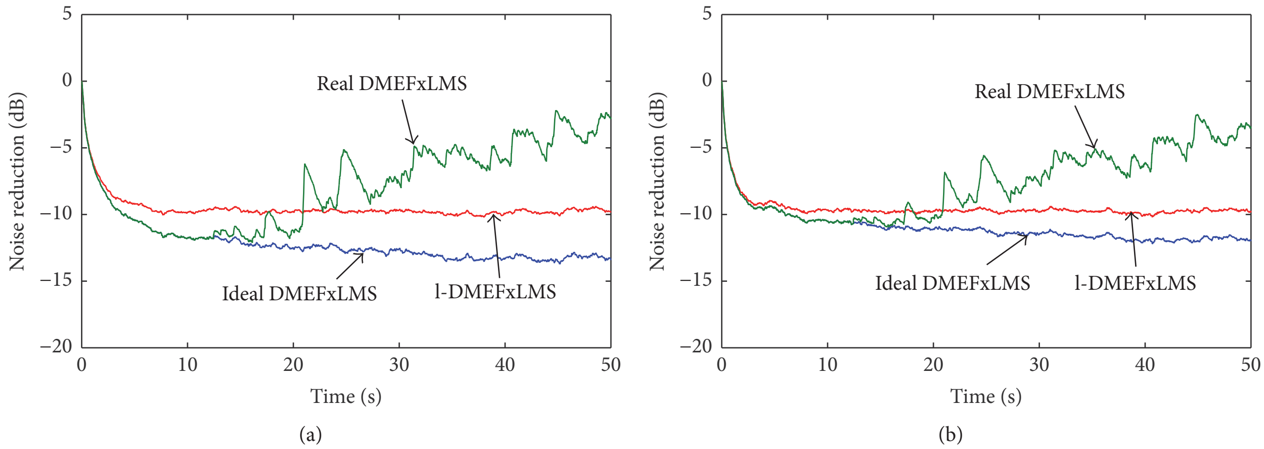


Figure 6: Noise reduction obtained for the ideal DMEFxLMS, the real DMEFxLMS, and the l-DMEFxLMS algorithms using a four-node WASN with $\beta_k = 0.01$ at the nodes (a) with the best performance and (b) with the worst performance.

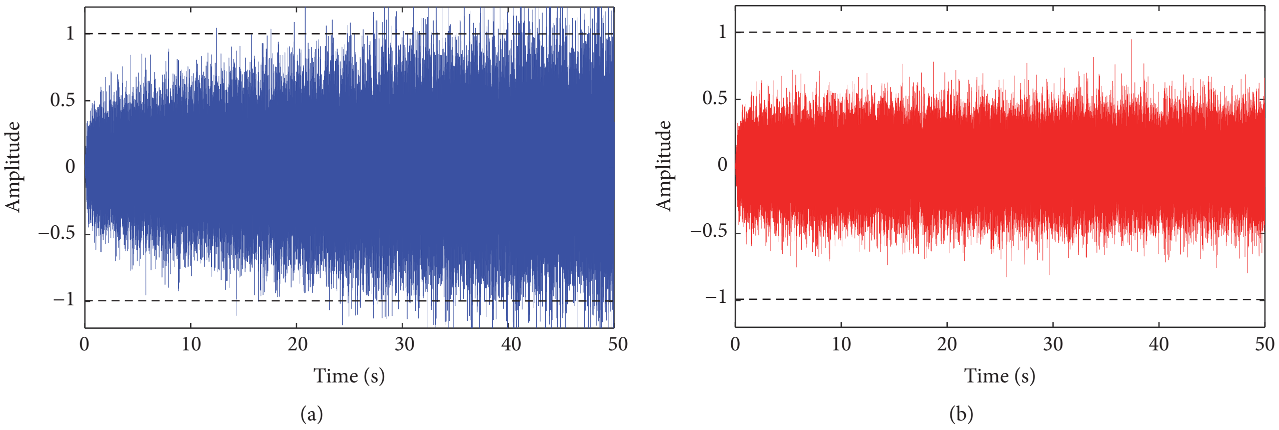


Figure 7: Time evolution of the control signals for both the (a) DMEFxLMS and (b) l-DMEFxLMS algorithms at the worst node. The threshold is represented by dashed lines.

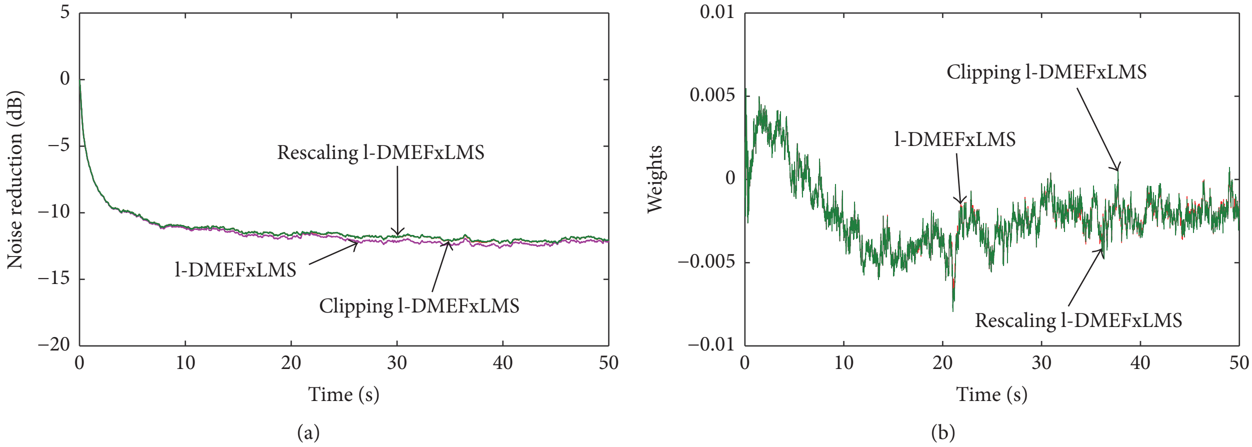


Figure 8: Behavior of the l-DMEFxLMS, the clipping l-DMEFxLMS, and the rescaling l-DMEFxLMS algorithms in a four-node WASN with $\beta_k = 0.001$ at the best node: (a) time evolution of the noise reduction obtained and (b) time evolution of the first filter coefficient.

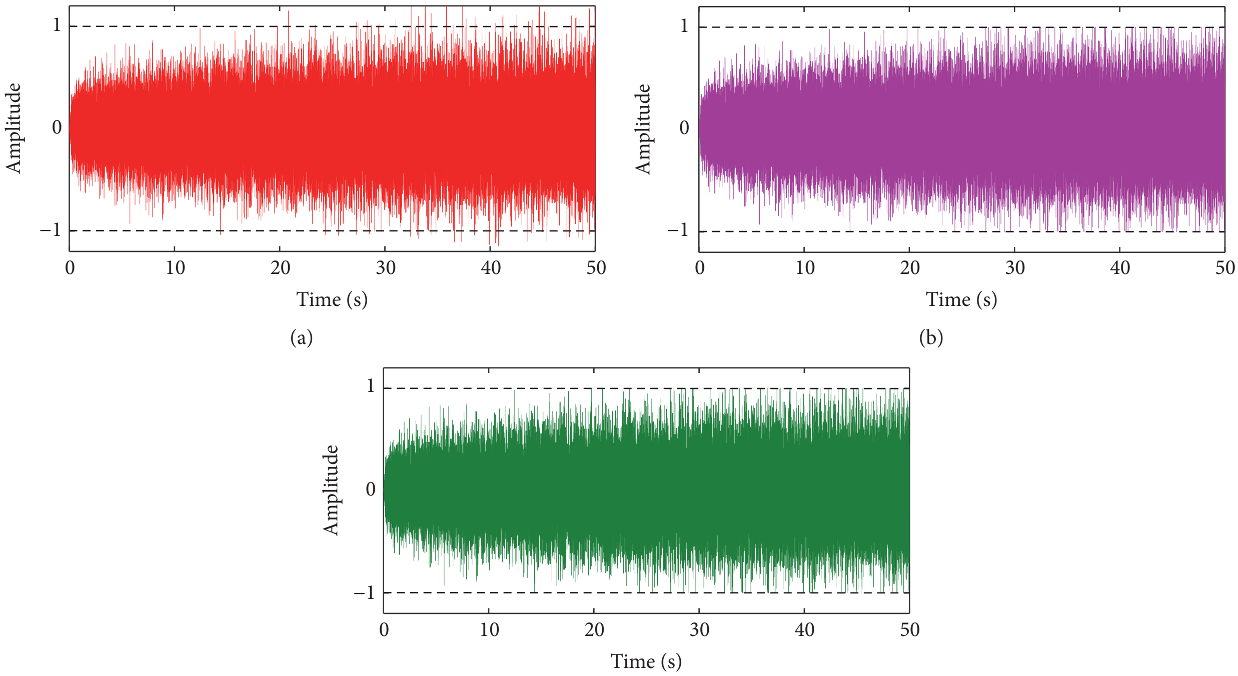


FIGURE 9: Time evolution of the control signals for the (a) l-DMEFxLMS algorithm, (b) clipping l-DMEFxLMS algorithm, and (c) rescaling l-DMEFxLMS algorithm at the best node. The threshold is represented by dashed lines.

increase the noise reduction of the leaky methods. Note that for $\beta_k = 0.01$, we will obtain the same results as the previous simulation since the leaky algorithms keep their output signal powers below the allowed threshold, $y_{k_{\max}}$, as it was described in (9) and (12). As the behavior of the four nodes of the network is similar, only the results obtained for the node with the best performance have been shown. As shown in Figure 8(a), all algorithms present similar noise attenuation levels providing 12 dB of noise reduction in the three cases. Regarding the convergence behavior of the first adaptive filter coefficient, plotted in Figure 8(b), it can be observed that it is almost identical for the three methods. However, the l-DMEFxLMS algorithm, as seen from Figure 9(a), fails to satisfy the output constraint requirement at certain time instants which may lead to system instability, as justified in the previous simulation. On the contrary, the output signal level of both clipping and rescaling l-DMEFxLMS algorithms is under the threshold fixed by the constraint, as it is illustrated by Figures 9(b) and 9(c).

3.4. Experiments with Periodic Noise as Input Signal. On the other hand, the clipping algorithm can lead to system instability as a result of the unpredictable behavior of its adaptive filter coefficients [34]. This can be observed in the next simulation where a periodic noise with 100 Hz, 200 Hz, 400 Hz, and 600 Hz components has been used as disturbance signal. As the previous case, Figures 10–12 show only the results obtained for the best node of a four-node WASN. In this case, the highest value of the step size parameter that ensures the stability of the algorithms was set to $\mu = 0.005$. Regarding the leakage parameter, a value of $\beta_k = 0.01$ for all the nodes was selected. Note that a higher value of β_k would maintain the output signal power of the l-DMEFxLMS algorithm below the threshold (see Figure 11(a)) obtaining a similar performance for the three algorithms. However, as it can be seen in the previous simulations, the smaller leakage parameter, the larger noise attenuation.

Under these conditions, as shown in Figure 10(a), both the rescaling l-DMEFxLMS and the l-DMEFxLMS algorithms show a stable behavior providing an attenuation up to 16 dB for the first one and almost 20 dB for the second one. However, as the previous simulation, note that the control output constraint of the l-DMEFxLMS algorithm was exceeded, as shown in Figure 11(a), while the rescaling l-DMEFxLMS strategy satisfies the output constraint (see Figure 11(c)). Although the clipping l-DMEFxLMS method fulfills the maximum output signal constraint (see Figure 11(b)), it achieves the worst performance in terms of final residual noise, obtaining a $\mathrm{NR}_k(n)$ of 10 dB for the best node. As it can be seen in Figure 10(b) and regarding the time evolution of the first filter coefficient, while both the l-DMEFxLMS and the rescaling l-DMEFxLMS algorithms exhibit a stable convergence, the coefficient of the clipping l-DMEFxLMS algorithm presents a significant oscillation. Although this behavior does not lead to system instability, it can result in the appearance of undesired frequency components at the spectrum of the error signal. The magnitude of the power spectral density of the error signal when the ANC system is off compared to the attenuation obtained by the three leaky strategies is represented in Figure 12.

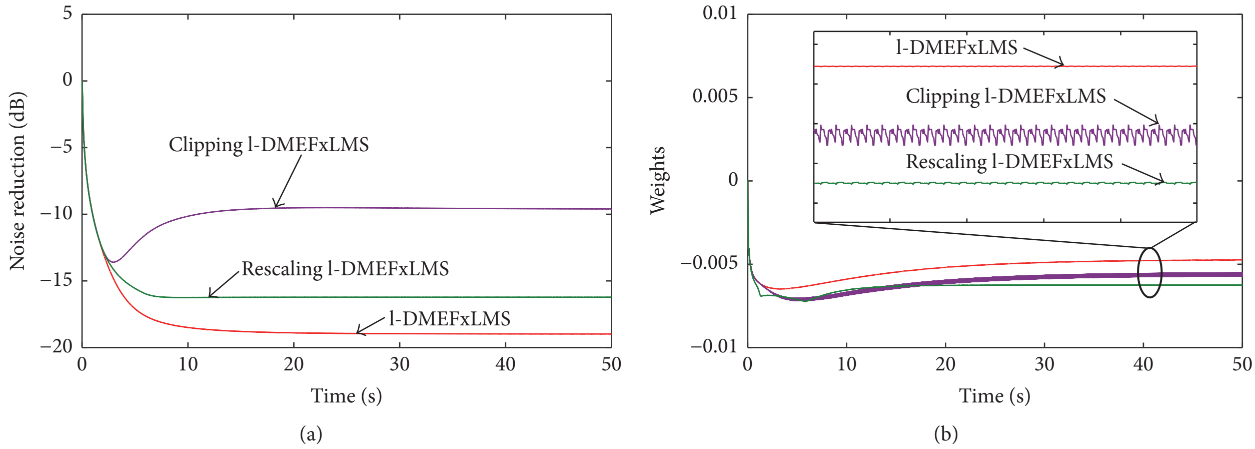


Figure 10: Behavior of the l-DMEFxLMS, the clipping l-DMEFxLMS, and the rescaling l-DMEFxLMS algorithms in a four-node WASN with $\beta_k = 0.01$ at the best node: (a) time evolution of the noise reduction obtained and (b) time evolution of the first filter coefficient.

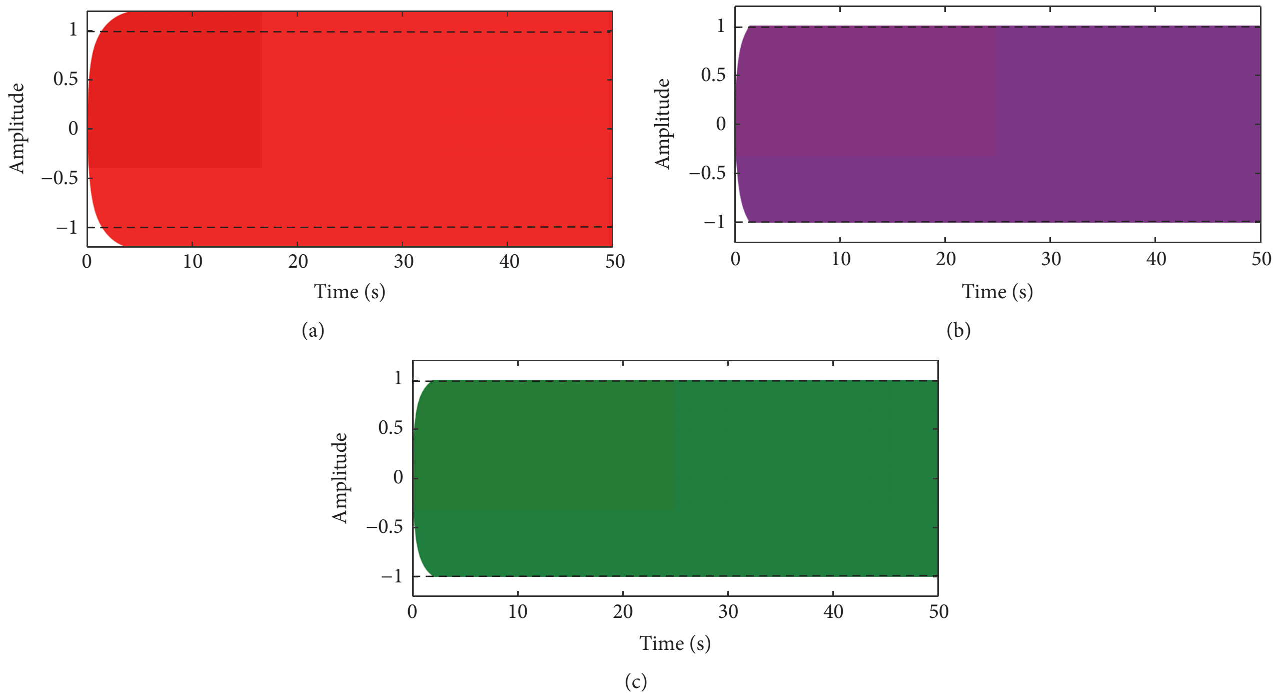
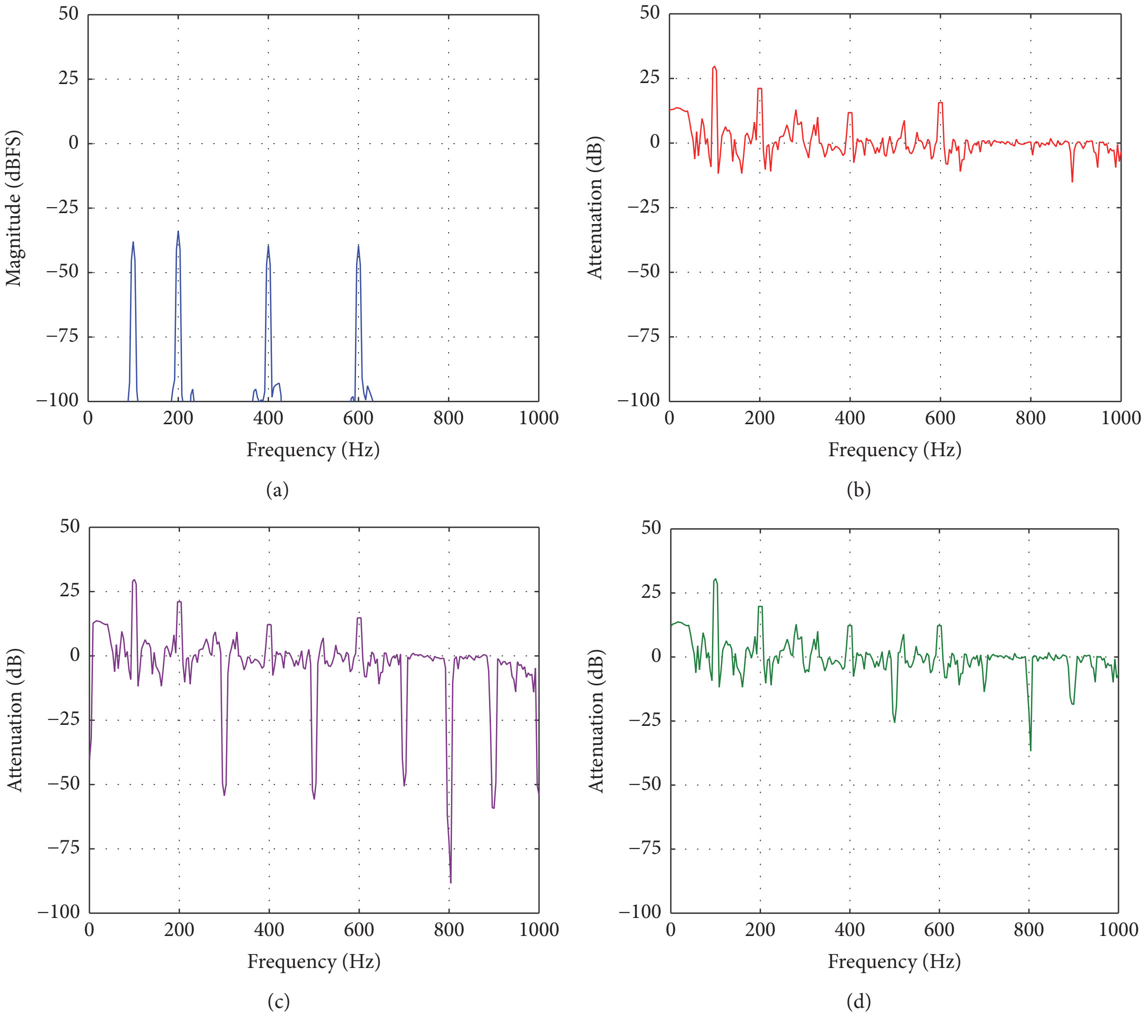


Figure 11: Time evolution of the control signals for the (a) l-DMEFxLMS algorithm, (b) clipping l-DMEFxLMS algorithm, and (c) rescaling l-DMEFxLMS algorithm at the best node. The threshold is represented by dashed lines.

It can be seen from Figures 12(b), 12(c), and 12(d) that all the methods, the l-DMEFxLMS algorithm, the clipping l-DMEFxLMS algorithm, and the rescaling l-DMEFxLMS algorithm, reduce the noise at the frequencies of interest, 100 Hz, 200 Hz, 400 Hz, and 600 Hz, obtaining an attenuation of almost 30 dB, 20 dB, 12 dB, and 14 dB, respectively. It can be observed that the rescaling l-DMEFxLMS algorithm introduces undesired noise at 500 Hz and 800 Hz, as can be seen in Figure 12(d). However, the clipping l-DMEFxLMS algorithm introduces much more additional harmonics, significantly at high frequencies, as shown in Figure 12(c). At least, five new harmonics appear at the frequencies 300 Hz, 500 Hz, 700 Hz, 800 Hz, and 900 Hz, achieving more than 50 dB in all of them. This may lead to the instability of the ANC system in real scenarios, probably caused by the strong saturation applied to the control signal (see Figure 11(b)). Since the l-DMEFxLMS algorithm does not ensure fulfillment of the constraint and the clipping l-DMEFxLMS algorithm may present some potential problems of stability, the rescaling l-DMEFxLMS algorithm exhibits the best overall performance, providing both the higher noise reduction and the system stability and fulfilling the output signal constraint.

3.5. Computational Complexity and Communication Requirements. Table 1 compares the computational complexity (in



(a)

(b)

(c)

(d)

Figure 12: (a) Power spectrum of the error signal for the ANC off and attenuation reached by the (b) l-DMEFxLMS algorithm, (c) clipping l-DMEFxLMS algorithm, and (d) rescaling l-DMEFxLMS algorithm compared to (a) at the best node of the four-node WASN.

Table 1: Total number of multiplications (MUX) and data transfer per iteration regarding (1) the computational complexity and (2) the communication requirements of the algorithms, respectively. L: length of the adaptive filters; M: length of the acoustic paths; N: number of nodes. As an example, some typical cases considering $L = 150$, $M = 256$, and $N = 2$, 4, and 8 nodes have been evaluated.

	Algorithms	Generic	$N = 2$	$N = 4$	$N = 8$
(1) Computational complexity (MUX)	l-DMEFxLMS	$LN(1 + LN) + MN + L + 1$	90963	361775	1443399
	Clipping l-DMEFxLMS	$LN(1 + LN) + MN + L + 2$	90964	361776	1443400
	Rescaling l-DMEFxLMS	$LN(2 + LN) + MN + L + 1$	91263	362376	1444600
	1r rescaling l-DMEFxLMS	$LN(2 + LN) + MN + L + 1$	91263	362376	1444600
(2) Communication requirement (data transfer)	l-DMEFxLMS	$2LN(N - 1)$	600	3600	16800
	Clipping l-DMEFxLMS	$2LN(N - 1)$	600	3600	16800
	Rescaling l-DMEFxLMS	$3LN(N - 1)$	900	5400	25200
	1r rescaling l-DMEFxLMS	$LN(N - 1)$	300	1800	8400

terms of multiplications per iteration) and the communication requirements (data transfer) of the distributed leaky ANC algorithms. The transmitted filter coefficients will be proportionally related to the transmitted bits depending on the used coding. The l-CMEFxLMS algorithm has not been included in the table because it has exactly the same computational complexity as the l-DMEFxLMS algorithm. To this end, we consider a network of N single-channel nodes. For simplicity, we assume that each node has access to $x(n)$ through an alternative broadcast channel. Therefore, the reference signal has not been considered in the calculation of the data transfer. Note that, for all the algorithms, the

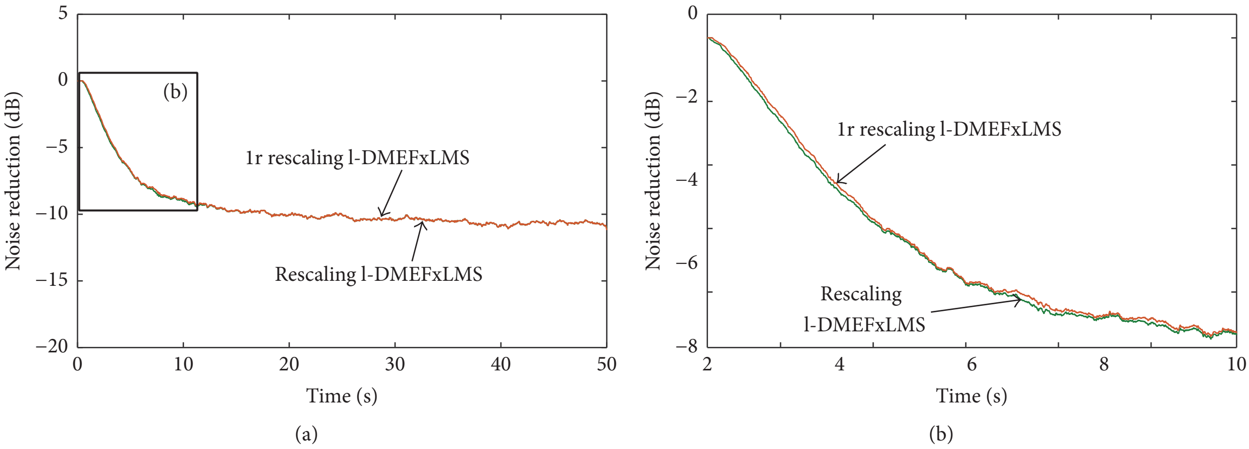


Figure 13: Noise reduction obtained for both the rescaling and the 1r rescaling l-DMEFxLMS algorithms using a four-node WASN at the node with the best performance with $\beta_k = 0.001$. To observe (a) both transient and steady state or (b) only the transient state.

computational complexity depends on L, M, and N while the communication requirements only depends on L and N. Then, both implementation aspects are particularized for $N = 2$, $N = 4$, and $N = 8$. Results show that the computational cost of the rescaling versions is slightly higher than the l-DMEFxLMS and clipping l-DMEFxLMS algorithms. Although there are no significant differences between them, it is important to take into account the fact that the computational complexity of all l-DMEFxLMS strategies increases significantly with the number of nodes. Regarding the communication needs, the rescaling l-DMEFxLMS strategy has higher requirements as shown in Table 1. For an incremental-learning N-node network, the rescaling l-DMEFxLMS method needs that every node transfers $LN \times 1$ coefficients to the following node $3(N-1)$ times in each iteration. However, using the 1r rescaling l-DMEFxLMS algorithm, a data stream of $LN(N-1)$ samples is propagated through the nodes, reducing the data transfer of the network. Note that, as expected, the same relation is maintained as the number of nodes increases. Following an example under the same configuration as described in [41], note that the clipping and rescaling l-DMEFxLMS algorithms would need a transfer rate of 16.1 and 24.2 megabytes per second (MBps), respectively, on an incremental four-node WASN. However, the 1r rescaling l-DMEFxLMS algorithm would need a transfer rate of at least 8.1 MBps. Therefore, in the last case, we could use a wireless network of 10 MBps to perform the required data transfer among the nodes.

3.6. Comparison between Rescaling Approaches. Finally, to compare the behavior of the two rescaling strategies, Figure 13 shows the time evolution of the $\mathrm{NR}_k(n)$ for the rescaling l-DMEFxLMS algorithm and the 1r rescaling l-DMEFxLMS algorithm for the node with the best performance of a four-node WASN. We have considered, as disturbance signal, the wideband noise used at the first simulations as well as a step size parameter of $\mu = 0.001$ and a leakage parameter of $\beta_k = 0.001$ for all the nodes. As it can be seen in Figure 13(a), the 1r rescaling l-DMEFxLMS algorithm presents similar results as the rescaling l-DMEFxLMS method in terms of final residual noise providing an attenuation up to 11 dB. As commented in Section 2, to avoid the dissemination of the update coefficients would affect the convergence speed of the 1r rescaling l-DMEFxLMS algorithm. However, differences between both strategies are negligible, as it can be seen from Figure 13(b). While in this case, the 1r rescaling l-DMEFxLMS strategy does not introduce strong degradation in the performance of the rescaling l-DMEFxLMS algorithm, it should be noted that, in other cases, such as larger WASN, the differences could be more relevant. To this end, in Figure 14, the time evolution of the $\mathrm{NR}_k(n)$ for the rescaling l-DMEFxLMS algorithm and the 1r rescaling l-DMEFxLMS algorithm for the node with the best performance of a sixteen-node WASN can be seen. In this case, differences between both algorithms, in terms of convergence speed, are slightly larger than the previous simulation, as it is shown in Figure 14(b). Differences in the result of the algorithms may become more evident if we use an increased numbers of nodes (maybe as many nodes as filter coefficients) or we work by block of samples. Since the 1r rescaling l-DMEFxLMS algorithm exhibits similar behavior to the rescaling l-DMEFxLMS algorithm, their results have not been included in the previous simulations. Note that all the results accomplished in this work depend on particular settings and parameter configuration.

4. Conclusions

In this paper, several control effort strategies have been implemented on a distributed ANC system over an ideal WASN using a collaborative incremental strategy. For this purpose, four new approaches have been derived from the DMEFxLMS to ensure ANC system stability in practical scenarios. Results show that the distributed version of the leaky MEFxLMS (l-DMEFxLMS) algorithm exhibits the same performance as its leaky centralized version (l-CMEFxLMS) when there are no communication constraints in the network.

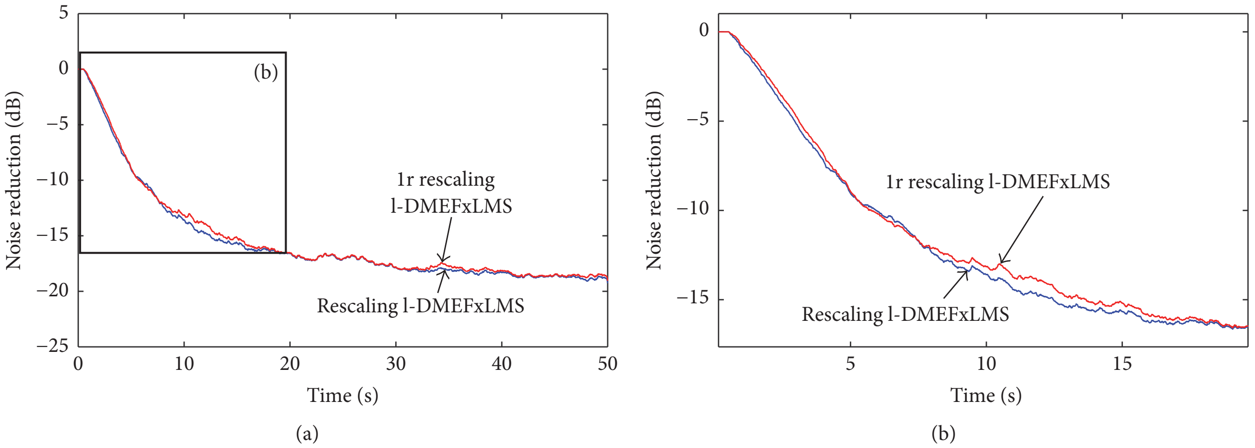


FIGURE 14: Noise reduction obtained for both the rescaling and the 1r rescaling l-DMEFxLMS algorithms using a sixteen-node WASN at the node with the best performance with $\beta_k = 0.001$. To observe (a) both transient and steady state or (b) only the transient state.

We have carried out simulations to show the performance of the distributed leaky approaches in different scenarios. The use of the l-DMEFxLMS algorithm in practical cases at the expense of worsening the system performance has been justified. In this regard, an appropriate leakage parameter value is required to reduce the performance degradation: high enough to fulfill the constraint but also low enough to obtain a good noise reduction. It has been demonstrated that both the clipping and the rescaling l-DMEFxLMS algorithms allow us to use low leakage values keeping the output constraint controlled. In addition, attenuation noise levels close to those obtained by the DMEFxLMS algorithm in an ideal scenario have been achieved for both methods. Due to the stability problems of the clipping l-DMEFxLMS strategy, the use of the rescaling l-DMEFxLMS algorithm is more promising in most cases. A simplified version of the rescaling l-DMEFxLMS algorithm is proposed, called 1r rescaling l-DMEFxLMS algorithm, which aims to reduce the communications demands of the rescaling l-DMEFxLMS method in WASN. Moreover, the computational complexity of the distributed leaky algorithms has been reported, bearing in mind that the computational capability of the nodes should be considered in practical scenarios. Regarding the communication requirements, the proposed 1r rescaling l-DMEFxLMS strategy achieves a noticeable data transfer saving maintaining similar performance to the rescaling l-DMEFxLMS algorithm. Therefore, it has been shown that the proposed 1r rescaling l-DMEFxLMS algorithm achieves a good control effort with a computational cost similar to the other strategies but reducing significantly the communication requirements of the network. However, the use of heterogeneous nodes, an analysis of the behavior of the proposed algorithms over constrained networks, and an automatic selection of all the parameters at each node are suggested for future works.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work has been supported by European Union ERDF together with Spanish Government through TEC2015-67387-C4-1-R project and Generalitat Valenciana through PROMETEOII/2014/003 project.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[2] J. Yick, B. Mukherjee, and D. Ghosal, “Wireless sensor network survey,” *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[3] D. Puccinelli and M. Haenggi, “Wireless sensor networks: applications and challenges of ubiquitous sensing,” *IEEE Circuits and Systems Magazine*, vol. 5, no. 3, pp. 19–31, 2005.

[4] X. Du and H. H. Chen, “Security in wireless sensor networks,” *IEEE Wireless Communications Magazine*, vol. 15, no. 4, pp. 60–66, 2008.

[5] M. Al Ameen, J. Liu, and K. Kwak, “Security and privacy issues in wireless sensor networks for healthcare applications,” *Journal of Medical Systems*, vol. 36, no. 1, pp. 93–101, 2012.

[6] A. Bertrand, “Applications and trends in wireless acoustic sensor networks: a signal processing perspective,” in *Proceedings of the 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT ’11)*, pp. 1–6, IEEE, Ghent, Belgium, November 2011.

[7] Y. Guo and M. Hazas, “Acoustic source localization of everyday sounds using wireless sensor networks,” in *Proceedings of the 12th International Conference on Ubiquitous Computing (UbiComp ’10)*, pp. 411-412, Denmark, September 2010.

[8] K. Martinez, J. K. Hart, and R. Ong, “Environmental sensor networks,” *The Computer Journal*, vol. 37, no. 8, pp. 50–56, 2004.

[9] J. Segura-Garcia, S. Felici-Castell, J. J. Perez-Solano, M. Cobos, and J. M. Navarro, “Low-cost alternatives for urban noise nuisance monitoring using wireless sensor networks,” *IEEE Sensors Journal*, vol. 15, no. 2, pp. 836–844, 2015.

[10] A. Bertrand and M. Moonen, “Robust distributed noise reduction in hearing aids with external acoustic sensor nodes,”

EURASIP Journal on Advances in Signal Processing, vol. 2009, Article ID 530435, 14 pages, 2009.

[11] H. Wang, *Wireless sensor networks for acoustic monitoring [PhD thesis]*, University of California, 2006.

[12] A. Flammini, P. Ferrari, D. Marioli, E. Sisinni, and A. Taroni, "Wired and wireless sensor networks for industrial applications," *Microelectronics Journal*, vol. 40, no. 9, pp. 1322–1336, 2009.

[13] H. Kwon, V. Berisha, and A. Spanias, "Real-time sensing and acoustic scene characterization for security applications," in *Proceedings of the 3rd International Symposium on Wireless Pervasive Computing (ISWPC '08)*, pp. 755–758, Greece, May 2008.

[14] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4064–4077, 2007.

[15] N. Bogdanovic, J. Plata-Chaves, and K. Berberidis, "Distributed incremental-based LMS for node-specific adaptive parameter estimation," *IEEE Transactions on Signal Processing*, vol. 62, no. 20, pp. 5382–5397, 2014.

[16] A. Bertrand, *Signal processing algorithms for wireless acoustic sensor networks [PhD. thesis]*, Faculty of Engineering, KU Leuven (Leuven, Belgium), Belgium, May 2011.

[17] M. Ferrer, M. de Diego, G. Piñero, and A. Gonzalez, "Active noise control over adaptive distributed networks," *Signal Processing*, vol. 107, pp. 82–95, 2015.

[18] G. Pinero, J. Estreder, F. Martinez-Zaldivar, M. Ferrer, and M. de Diego, "Sound-field reproduction system over a two-node acoustic network of mobile devices," in *Proceedings of the 2nd IEEE World Forum on Internet of Things, WF-IoT 2015*, pp. 652–657, Italy, December 2015.

[19] M. Cobos, J. J. Perez-Solano, O. Belmonte, G. Ramos, and A. M. Torres, "Simultaneous ranging and self-positioning in unsynchronized wireless acoustic sensor networks," *IEEE Transactions on Signal Processing*, vol. 64, no. 22, pp. 5993–6004, 2016.

[20] C. Llerena-Aguilar, R. Gil-Pita, M. Rosa-Zurera, D. Ayllón, M. Utrilla-Manso, and F. Llerena, "Synchronization based on mixture alignment for sound source separation in wireless acoustic sensor networks," *Signal Processing*, vol. 118, pp. 177–187, 2016.

[21] S. J. Elliott and P. A. Nelson, "Active noise control," *IEEE Signal Processing Magazine*, vol. 10, no. 4, pp. 12–35, 1993.

[22] M. Pawełczyk, "Active noise control - A review of control-related problems," *Archives of Acoustics*, vol. 33, no. 4, pp. 509–520, 2008.

[23] S. J. Elliott, P. Joseph, A. J. Bullmore, and P. A. Nelson, "Active cancellation at a point in a pure tone diffuse sound field," *Journal of Sound and Vibration*, vol. 120, no. 1, pp. 183–189, 1988.

[24] P. Joseph, S. J. Elliott, and P. A. Nelson, "Near field zones of quiet," *Journal of Sound and Vibration*, vol. 172, no. 5, pp. 605–627, 1994.

[25] S. Haykin, *Adaptive filter theory*, Prentice-Hall, Upper Saddle River, NJ, USA, 2002.

[26] S. M. Kuo and D. R. Morgan, "Active noise control: a tutorial review," *Proceedings of the IEEE*, vol. 87, no. 6, pp. 943–973, 1999.

[27] J. C. Burgess, "Active adaptive sound control in a duct: a computer simulation," *The Journal of the Acoustical Society of America*, vol. 70, no. 3, pp. 715–726, 1981.

[28] B. Farhang-Boroujeny, *Adaptive filters: theory and applications*, John Wiley and Sons, Chicester, West Sussex, UK, 2013.

[29] B. Widrow and S. D. Stearns, *Adaptive signal processing*, Prentice Hall, Englewood Cliffs, NJ, USA, 1985.

[30] S. J. Elliott and C. C. Boucher, "Interaction between multiple feedforward active control systems," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 2, no. 4, pp. 521–530, 1994.

[31] P. Grosdidier and M. Morari, "Interaction measures for systems under decentralized control," *Automatica*, vol. 22, no. 3, pp. 309–319, 1986.

[32] S. J. Elliott, I. M. Stothers, and P. A. Nelson, "A multiple error LMS algorithm and its application to the active control of sound and vibration," *IEEE Transactions on Signal Processing*, vol. 35, no. 10, pp. 1423–1434, 1987.

[33] S. J. Elliott and K. H. Back, "Effort constraints in adaptive feedforward control," *IEEE Signal Processing Letters*, vol. 3, no. 1, pp. 7–9, 1996.

[34] X. Qiu and C. H. Hansen, "A study of time-domain FXLMS algorithms with control output constraint," *The Journal of the Acoustical Society of America*, vol. 109, no. 6, pp. 2815–2823, 2001.

[35] B. Rafaely, "A computationally efficient frequency-domain LMS algorithm with constraints on the adaptive filter," *IEEE Transactions on Signal Processing*, vol. 48, no. 6, pp. 1649–1655, 2000.

[36] W. J. Kozacky and T. Ogunfunmi, "An active noise control algorithm with gain and power constraints on the adaptive filter," *EURASIP Journal on Advances in Signal Processing*, vol. 2013, no. 1, article 17, 2013.

[37] J. A. Mosquera-Sánchez, W. Desmet, and L. P. R. de Oliveira, "A multichannel amplitude and relative-phase controller for active sound quality control," *Mechanical Systems and Signal Processing*, vol. 88, pp. 145–165, 2017.

[38] F. Taringoo, J. Poshtan, and M. H. Kahaei, "Analysis of effort constraint algorithm in active noise control systems," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 54649, 10 pages, 2006.

[39] D. J. Rossetti, M. R. Jolly, and S. C. Southward, "Control effort weighting in feedforward adaptive control systems," *The Journal of the Acoustical Society of America*, vol. 99, no. 5, pp. 2955–2964, 1996.

[40] "Audio and communications signal processing group (GTAC)," http://www.gtac.upv.es.

[41] C. Antoñanzas, M. Ferrer, M. de Diego, and A. Gonzalez, "Blockwise frequency domain active noise controller over distributed networks," *Applied Sciences*, vol. 6, no. 5, article no. 124, 2016.

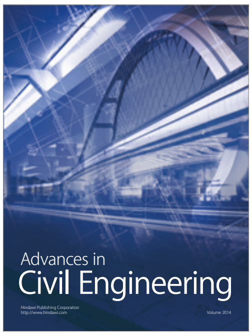
Journal of
Engineering

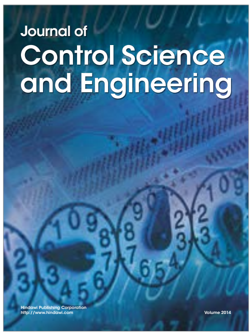
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

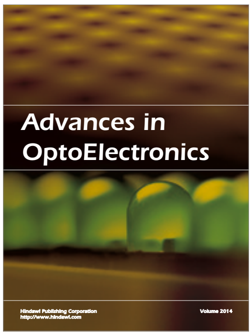
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

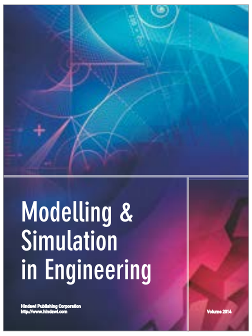
Journal of
Robotics

Journal of
Electrical and Computer Engineering

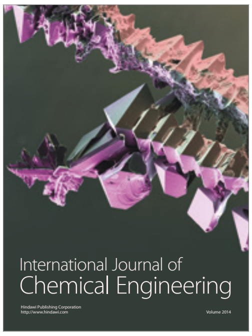

Advances in
OptoElectronics

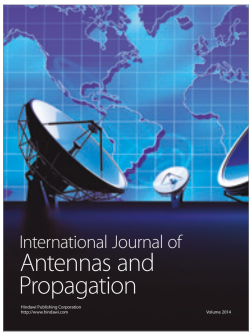
VLSI Design

International Journal of
Navigation and Observation

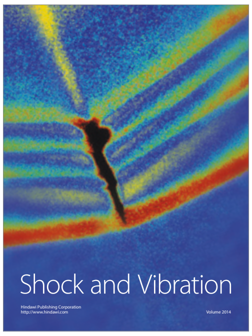

Modelling &
Simulation
in Engineering

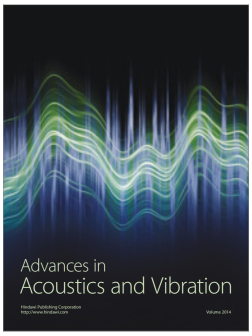
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration