Applied Acoustics 116 (2017) 364–374

Contents lists available at ScienceDirect

Applied Acoustics

journal homepage: www.elsevier.com/locate/apacoust

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

ARTICLE INFO

Article history:
Received 12 May 2016
Received in revised form 2 September 2016
Accepted 10 October 2016
Available online 20 October 2016

Keywords:
Active noise control
Impulsive noise
Stable distributions
FxRLS algorithm

ABSTRACT

Active noise control (ANC) systems employing adaptive filters suffer from stability issues in the presence of impulsive noise. New impulsive noise control algorithms based on filtered-x recursive least square (FxRLS) algorithm are presented. The FxRLS algorithm gives better convergence than the filtered-x least mean square (FxLMS) algorithm and its variants but lacks robustness in the presence of high impulsive noise. In order to improve the robustness of FxRLS algorithm for ANC of impulsive noise, two modifications are suggested. First proposed modification clips the reference and error signals while, the second modification incorporates energy of the error signal in the gain of FxRLS (MGFxRLS) algorithm. The results demonstrate improved stability and robustness of proposed modifications in the FxRLS algorithm. However, another limitation associated with the FxRLS algorithm is its computationally complex nature. In order to reduce the computational load, a hybrid algorithm based on proposed MGFxRLS and normalized step size FxLMS (NSS-FXLMS) is also developed in this paper. The proposed hybrid algorithm combines the stability of NSS-FxLMS algorithm with the fast convergence speed of the proposed MGFxRLS algorithm. The results of the proposed hybrid algorithm prove that its convergence speed is faster than that of NSS-FxLMS algorithm with computational complexity lesser than that of FxRLS algorithm.
© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Active noise control (ANC) is based on destructive interference between acoustic waves [1,2]. Basically, the primary noise is mitigated in the region of the error microphone by producing and combining a noise having same amplitude but opposite phase of the primary noise source. This anti-phase noise is provided by the controller which uses an adaptive algorithm such as the filtered-x least mean square (FxLMS) algorithm or filtered-x recursive least square (FxRLS) algorithm. Fig. 1 shows a filtered-x adaptive algorithm for single channel feed forward ANC system comprising of reference microphone to obtain the reference noise, a cancelling loudspeaker and an error microphone to perceive the error between the primary noise and the cancelling noise. In the figure, $P(z)$ represents the primary path between the reference microphone and the error microphone, $S'(z)$ is the estimate of the secondary path that lies between the cancelling speaker and the error microphone, which can be estimated using offline or online system identification approach [3,4]. Also $x_f(n)$ is the filtered reference input to the adaptive algorithm.

In this paper, we have focused on the active control of impulsive noise. A broad class of non-Gaussian noise can be characterized as impulsive noise such as infusion pump sounds in hospitals, power line communication interference, underwater acoustic signals, atmospheric noise and all types of man-made noise [5–7]. The statistics of impulsive noise can be modeled by symmetric alpha stable ($S\alpha S$) distribution [8]. $S\alpha S$ distribution has characteristic function of the form

$$\varphi(t) = e^{-\gamma|t|^{\alpha}} \tag{1}$$

where $\alpha (0 < \alpha < 2)$ controls the heaviness of its tail, called as characteristic exponent. If α is close to zero, then the distribution has a very heavy tail indicating high impulsive noise. For $\alpha = 2$ the distribution becomes Gaussian. The main difference between the $S\alpha S$ distribution and the Gaussian distribution is that the density function of α-stable distribution has a heavy tail as compared to Gaussian density function. In (1), γ ($\gamma > 0$) is defined as dispersion and for $\gamma = 1$, the $S\alpha S$ distribution is called standard $S\alpha S$ distribution. In this paper, we have modeled impulsive noise using standard $S\alpha S$ distribution. Fig. 2 shows the effect of varying α in its specified range.

For stable distributions, it has been proved that only those moments that are of order less than the characteristic exponent are finite[8]. Hence, the second order moment (variance) is infinite. So, the famous filtered-x LMS algorithm which minimizes the mean

* Corresponding author.
E-mail address: ayeshazeb79@ee.ceme.edu.pk (A. Zeb).

http://dx.doi.org/10.1016/j.apacoust.2016.10.011
0003-682X/© 2016 Elsevier Ltd. All rights reserved.

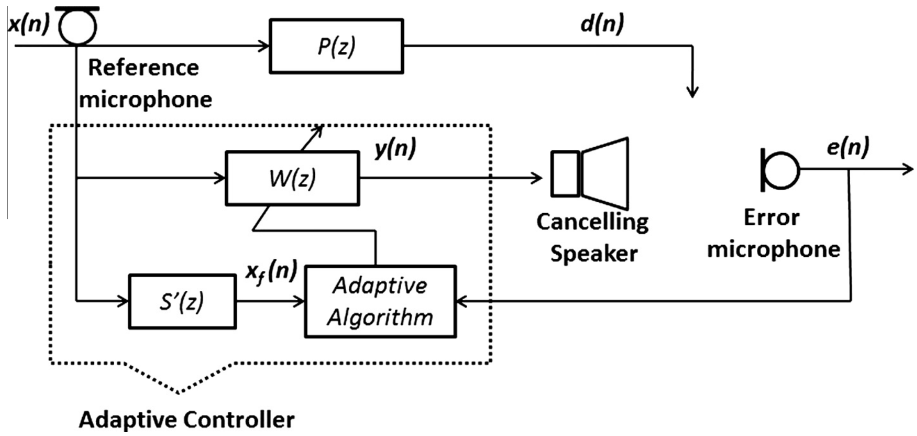

Fig. 1. Basic schematic of single channel feed forward ANC system.

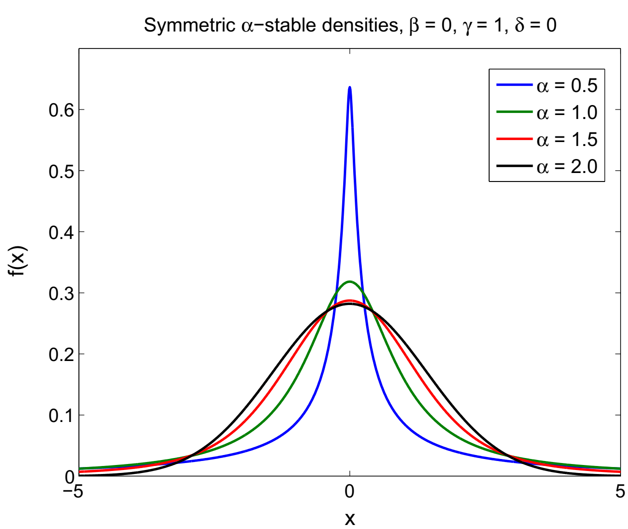

Fig. 2. PDFs of standard $S\alpha S$ process for various values of α.

square error of noise may become unstable and might not be the right choice for ANC of impulsive noise [8]. Some solutions are proposed in literature to tackle this problem. In [9], the filtered-x least mean p-power (FxLMP) algorithm is presented, based on reducing a lower order fractional moment of error i.e. $p(0 < p < \alpha)$ power of error, that exists for stable distributions. It gives better robustness as compared to FxLMS algorithm for active impulsive noise control. On the other hand, its convergence speed is very low especially when the noise is highly impulsive. Moreover, it requires prior estimation of p depending on α, which is a challenging task for good results. Sun et al. [10] proposed a modification of FxLMS algorithm for impulsive ANC. In this proposed technique the reference signal is modified on the basis of its statistical properties i.e. if the magnitude of any sample is above a certain threshold value, the sample is ignored. The threshold is decided by the statistics of the signal. This technique gives stable and better performance in comparison to the standard FxLMS algorithm. However, it's not robust enough to give stable results when α is small. In [11], Akhtar et al. proposed a modification of Sun's algorithm, which clips the samples of reference and error signals instead of ignoring them. The main issue with the threshold based algorithms [10,11] is their requirement of estimating appropriate thresholds, which is not possible during online ANC operation. In [12,13], two more algorithms are proposed by Akhtar et. al. One is normalized step size FxLMS (NSS-FxLMS) algorithm which gives improved stability by normalizing the step size, and the second one is data reusing based NSS-FxLMS (DRNSS-FxLMS) algorithm which provides better convergence speed by reusing the recent data. The noise cancellation performance of FxLMS algorithm based ANC system is improved further in [14], by applying a norm constraint to its weight update process. Furthermore, a robust filtered-s LMS algorithm is developed for nonlinear ANC [15]. Both the algorithms [14,15], are robust against interference to the error microphone.

The main limitation of LMS based algorithms is their lower rate of convergence than RLS in general, and larger steady state mean square error as compared to RLS [16]. In [17], performance comparison of FxLMS, FxAPA and FxRLS ANC algorithms on fMRI noise validate the theoretical claims [16] that the FxRLS algorithm has a very high convergence rate. These findings served as a motivation to switch to least square family of adaptive filters. In [18], authors tested the FxRLS algorithm for active impulsive noise control which achieved faster convergence than that of FxLMS family algorithms. However, the FxRLS algorithm lacks robustness under non-stationary noise environment and stability is not guaranteed in case of high impulsive noise.

In this paper, two modifications in the FxRLS algorithm are presented to improve its robustness and stability. The first proposed modification is threshold based FxRLS (TFxRLS) algorithm which clips the reference and error signals based on certain thresholds before they are used in the weight update of the FxRLS algorithm. The second proposed modification is modified gain FxRLS (MGFxRLS) algorithm which incorporates the energy of error signal in the gain of FxRLS algorithm. Another drawback of the FxRLS algorithm is its enhanced computational complexity than that of FxLMS algorithm that limits its practical implementation in a real time ANC system. In [19], a hybrid FxRLS-FxNLMS adaptive algorithm is presented that combines stability of the FxNLMS algorithm with high convergence speed of the FxRLS algorithm, thus reducing the computational complication of the overall ANC system. To overcome the computational complexity of the FxRLS algorithm, a hybrid MGFxRLS-NSSFxLMS adaptive algorithm is presented in this paper that takes its fast convergence from the robust MGFxRLS and reduced computational complexity from the NSS-FxLMS algorithm.

This paper is organized as follows: Section 2 briefly describes the standard FxRLS algorithm. The proposed TFxRLS and MGFxRLS algorithms are presented in Sections 3 and 4, respectively. Section 5 presents the hybrid MGFxRLS-NSSFxLMS adaptive algorithm. The complexity analysis is given is Section 6 followed by the performance comparisons of the newly proposed algorithms with the standard FxRLS algorithm through simulation results in Section 7. Finally, Section 8 concludes the paper.

2. FxRLS algorithm

The block diagram of FxRLS algorithm based ANC system for impulsive noise is shown in Fig. 3.

In this figure, $W(z)$ is assumed to be a FIR filter of tap-weight length L given as $w(n) = [w(n), w(n-1), \ldots, w(n-L+1)]'$. The standard FxRLS algorithm is summarized below in (2)–(6)

Initialization:

$$\mathbf{w}(0) = 0 \qquad \mathbf{P}(0) = \delta^{-1}\mathbf{I} \tag{2}$$

Algorithm:

$$\boldsymbol{k}(n) = \frac{\boldsymbol{P}(n-1)\boldsymbol{x}_f(n)}{\boldsymbol{x}_f^H(n)\boldsymbol{P}(n-1)\boldsymbol{x}_f(n) + \lambda} \tag{3}$$

$$\boldsymbol{P}(n) = \lambda^{-1}\boldsymbol{P}(n-1) - \lambda^{-1}\boldsymbol{k}(n)\boldsymbol{x}_f^H(n)\boldsymbol{P}(n-1) \tag{4}$$

$$e(n) = d(n) - s(n) * (\mathbf{w}^{\mathrm{H}}(n-1)\boldsymbol{x}_f(n)) \tag{5}$$

$$\boldsymbol{w}(\mathrm{n}+1) = \boldsymbol{w}(\mathrm{n}) + \boldsymbol{k}(n)e(n) \tag{6}$$

where the reference noise signal vector is $x(n) = [x(n), x(n-1), \ldots, x(n-L+1)]'$ and $x'(n)$ is the reference noise signal filtered by the estimated secondary path filter $S'(z)$. The list of variables used in this ANC application is given in Table 1.

Initialization of the FxRLS algorithm is a critical point for the convergence of the algorithm. The regularization parameter delta (δ) in (2) depends on SNR or in other words, variance of the input signal $x'(n)$ [16]. When noise level in tap inputs is low, the FxRLS algorithm exhibits exceptionally high convergence rate, provided that δ is chosen small enough. However, if noise level is very high it is preferable to initialize the algorithm with large value of δ [16]. It has been revealed through simulation results in [18] that FxRLS algorithm outperforms all FxLMS family algorithms in terms of fast convergence rate for ANC of impulsive noise. But as delta depends on the noise level, the FxRLS algorithm based ANC system presented in [18] chooses different values of delta for different noise levels. Table 2 lists the values of delta chosen to initialize the FxRLS algorithm for different noise levels.

So, if the noise is non-stationary i.e. alpha is time-varying, the FxRLS algorithm needs to be re-initialized [16] which is very difficult or rather impossible during runtime application. In practice, if noise level changes during runtime application then the FxRLS algorithm lacks robustness and becomes unstable. Thus, modifications are required in the FxRLS algorithm to make it robust i.e. once it is initialized with a particular value of delta it should not become unstable if noise level changes. In the next section, we propose threshold based FxRLS (TFxRLS) algorithm. Compared to the standard FxRLS algorithm, for fixed value of delta it has much better convergence and stability characteristics in non-stationary noise environments.

Table 1
List of variables.

Variables	Description
$S(z)$	Secondary path of the cancelling signal
$S'(z)$	Estimated secondary path
$P(z)$	Primary path of the reference signal
$d(n)$	Desired signal to be cancelled
$y(n)$	Anti-phase secondary noise
$y'(n)$	Anti-phase noise following secondary path
$x(n)$	Reference noise signal
$x'(n)$	Filtered reference noise signal
$w(n)$	Adaptive filter coefficients
$P(n)$	Inverse of the correlation matrix
$k(n)$	Gain of FxRLS algorithm
δ	Regularization parameter
λ	Forgetting factor
$e(n)$	Residual noise

Table 2
Delta values chosen for different noise levels.

Alpha	Delta
1.65	1000
1.45	100,000
0.95	10,000,000

3. Threshold based FxRLS (TFxRLS) algorithm

In standard FxRLS algorithm, all the samples of reference noise vector, $x(n) = [x(n), x(n-1), \ldots, x(n-L+1)]'$, are treated "equally". A high amplitude noise sample may cause the FxRLS algorithm to become unstable. The main objective of this work is to improve the robustness of the FxRLS algorithm with same computational complexity as that of FxRLS algorithm. Based on extensive simulation studies in [18], a modified version of the FxRLS algorithm is

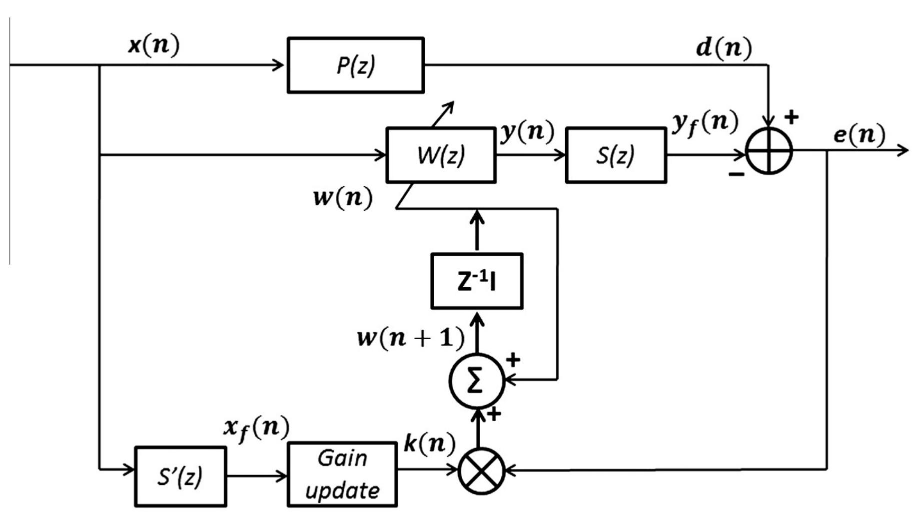

Fig. 3. Block diagram of FxRLS algorithm based ANC system.

proposed. In order to enhance the robustness of FxRLS algorithm in non-stationary noise environments, the samples of the reference and error signals are clipped if their magnitude is above a certain threshold set by statistics of the signal. This statistics can be estimated offline in practice. Block diagram of proposed TFxRLS algorithm is shown in Fig. 4. The reference and error signals are modified as

$$x'(n) = \begin{cases} c_1, & \text{if } x(n) \leqslant c_1 \\ c_2, & \text{if } x(n) \geqslant c_2 \\ x(n), & \text{otherwise} \end{cases} \tag{7}$$

$$e'(n) = \begin{cases} c_1, & \text{if } e(n) \leqslant c_1 \\ c_2, & \text{if } e(n) \geqslant c_2 \\ e(n), & \text{otherwise} \end{cases} \tag{8}$$

Using these modified reference and error signals, the proposed threshold based FxRLS (TFxRLS) algorithm for ANC of impulsive noise is given as

$$w(n+1) = w(n) + k(n)e'(n) \tag{9}$$

where

$$k(n) = \frac{P(n-1)[s(n) * x'(n)]}{[s(n) * x'^H(n)]P(n-1)[s(n) * x'(n)] + \lambda} \tag{10}$$

$$P(n) = \lambda^{-1}P(n-1) - \lambda^{-1}k(n)[s(n) * x'^H(n)]P(n-1) \tag{11}$$

Simulation results authenticate the improved robustness of TFxRLS algorithm. If the noise level changes, resulting in large amplitude impulses then the impulses are clipped to the threshold value ensuring stability of the algorithm. However, it is worth mentioning that all algorithms reported so far i.e. Sun's algorithm [10], modified Sun's algorithm [11], Akhtar's algorithm [11] and the proposed TFxRLS algorithm, require an appropriate selection of the threshold parameters $[c_1, c_2]$. In order to overcome this difficulty, in the next section we propose a modification in the gain of FxRLS algorithm that does not use modified reference and/or error signals and hence does not depends on the threshold parameters $[c_1, c_2]$.

4. Modified gain FxRLS (MGFxRLS) algorithm

The proposed algorithm is based on normalization of the gain parameter of the FxRLS algorithm. The gain of the standard FxRLS algorithm is given in (3). When the reference signal has a large peak, its energy would increase and this would in turn decrease the gain of the FxRLS algorithm. As stated earlier, the error signal is also peaky in nature and may cause the algorithm to become unstable. So, its effect must also be taken into account. We propose following modified gain for the FxRLS algorithm

$$\boldsymbol{k}(n) = \frac{\boldsymbol{P}(n-1)\boldsymbol{x}_f(n)}{\boldsymbol{x}_f^H(n)\boldsymbol{P}(n-1)\boldsymbol{x}_f(n) + E_e(n) + \lambda} \tag{12}$$

where $E_e(n)$ is the energy of the residual error signal $e(n)$ that is estimated online using a low pass estimator of the form

$$E_e(n) = \lambda E_e(n-1) + (1-\lambda)|e^2(n)| \tag{13}$$

where λ, called as forgetting factor, has value $0.9 < \lambda < 1$. The basic idea here is to freeze the adaptation if the noise amplitude is very high to prevent the algorithm from becoming unstable. It is worth mentioning that the proposed modified gain FxRLS (MGFxRLS) algorithm does not require estimation of threshold parameters $[c_1, c_2]$. Although the proposed MGFxRLS algorithm is based on an intuition based modification, the simulations suggest its improved robustness in comparison with other discussed algorithms.

The two proposed modifications i.e. TFxRLS algorithm and MGFxRLS algorithm have better robustness than that of FxRLS algorithm. The FxRLS family algorithms have high convergence speed however a major limitation is their enhanced computational complexity. To overcome this limitation while preserving high convergence speed and robustness, a hybrid algorithm is presented in the next section.

5. Hybrid MGFxRLS-NSSFxLMS algorithm

We provide here a new hybrid algorithm for impulsive ANC that influences the robustness and high convergence speed of MGFxRLS algorithm with the stability of NSSFxLMS algorithm. It gives a lower overall computational complexity than the MGFxRLS algorithm and faster convergence than NSSFxLMS algorithm. Another advantage is that different filter weight lengths for both the filters can be used. The hybrid algorithm allows the user to run the computationally extensive MGFxRLS algorithm with a filter size that is limited only by the performance of the hardware. Once quick convergence is achieved, the algorithm switches to lower computation NSSFxLMS algorithm with a larger filter length.

In order to estimate filter weights, $w(n) = [w(n), w(n-1)\ldots w(n-L+1)]'$, we use the MGFxRLS algorithm first until it converges to the lowest possible residual error. Once the algorithm stops converging we switch over to the NSSFxLMS algorithm with an optimal filter length, thus, continuing the convergence to the optimal filter weights at a slower rate. The proposed algorithm involves switching between 2 algorithms i.e. MGFxRLS and NSSFxLMS. In order to decide when to switch the algorithms, a switching threshold is defined on the basis of mean noise reduction (MNR) achieved at the location of error microphone expressed as

$$|\mathrm{MNR(n)}| \underset{\mathrm{NSS-FxLMS}}{\overset{\mathrm{MGFxRLS}}{\gtrless}} \Psi \tag{14}$$

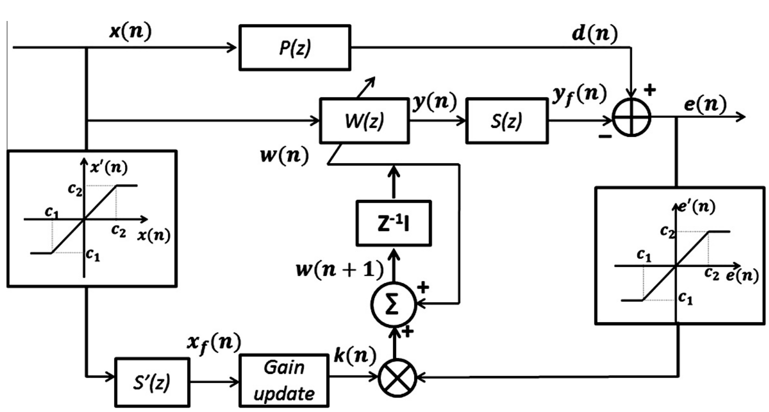

Fig. 4. Block diagram of proposed TFxRLS algorithm based ANC system.

where Ψ is found empirically. By setting the threshold on the absolute value of MNR rate of change, it switches the algorithm back to MGFxRLS algorithm for fast convergence under non stationary environments. The MNR is defined as

$$\mathrm{MNR(n)} = \mathbb{E}\left\{\frac{A_e(n)}{A_d(n)}\right\} \tag{15}$$

where $\mathbb{E}\{.\}$ represents ensemble averaging, $A_e(n)$ and $A_d(n)$ are estimated absolute values of residual error signal $e(n)$ and desired signal $d(n)$, respectively. These are estimated using low pass estimator of the form

$$A_r(n) = \lambda A_r(n-1) + (1-\lambda)|r(n)| \tag{16}$$

It has been verified through computer simulations that the proposed hybrid method has enhanced robustness, stability and convergence with reduced computational load as compared with that of MGFxRLS algorithm alone.

Table 3
Complexity analysis of proposed MGFxRLS algorithm.

Eq.'s	Operations	$*$	+/−	÷
(1)	$\mathrm{x}_f(\mathrm{n})_{1\times1} = \hat{s}(n)_{1\times\mathrm{M}} * x(\mathrm{n})_{M\times1}$	M	M − 1	–
(2)	$\mathrm{y(n)}_{1\times1} = \mathrm{w^T(n)}_{1\times L} * x(\mathrm{n})_{L\times1}$	L	L − 1	–
(3)	$\mathrm{w(n+1)}_{L\times1} = \mathrm{w(n)}_{L\times1} + \mathrm{K(n)_{L\times L}} * \mathrm{e(n)}_{1\times1}$	L	L	–
(4)	$\mathrm{K(n)_{L\times1}} = \frac{\pi(\mathrm{n})_{\mathrm{L}\times}1}{\lambda+\mathrm{E}_e(\mathrm{n})+\mathrm{x}_f^T(\mathrm{n})_{1\times L}*\pi(\mathrm{n})_{\mathrm{L}\times1}}$	2L	L + 1	1
(5)	$\pi(\mathrm{n})_{\mathrm{L}\times1} = \mathrm{p(n-1)_{L\times L}} * \mathrm{x}_f(\mathrm{n})_{L\times1}$	L^2	$L^2 - L$	
(6)	$\mathrm{p(n)_{L\times L}} = \lambda^{-1} * \mathrm{p(n-1)_{L\times L}} - \lambda^{-1} * \mathrm{K(n)_{L\times1}} * \mathrm{x}_f(\mathrm{n})_{1\times L} * \mathrm{p(n-1)_{L\times L}}$	$3L^2$	$2L^2 - L$	1
(7)	$\mathrm{e(n)}_{1\times1} = \mathrm{d(n)}_{1\times1} - \mathrm{y_s(n)}_{1\times1}$	–	1	–
(8)	$\mathrm{y_s(n)}_{1\times1} = s(n)_{1\times M} * \mathrm{y(n)}_{M\times1}$	M	M − 1	–
(9)	$\mathrm{E}_e(\mathrm{n})_{1\times1} = \lambda E(n-1)_{1\times1} + (1-\lambda)\mathrm{E}^2(\mathrm{n})_{1\times1}$	3	2	
	Total	$4L^2 + 4L + 2M + 3$	$3L^2 + L + 2M + 1$	2

Table 4
Complexity analysis of FxLMS algorithm.

Eq.'s	Operations	$*$	+/−	÷
(1)	$\mathrm{x}_f(\mathrm{n})_{1\times1} = \hat{s}(n)_{1\times\mathrm{M}} * x(\mathrm{n})_{M\times1}$	M	M − 1	–
(2)	$\mathrm{y(n)}_{1\times1} = \mathrm{w^T(n)}_{1\times L} * x(\mathrm{n})_{L\times1}$	L	L − 1	–
(3)	$\mathrm{w(n+1)}_{L\times1} = \mathrm{w(n)}_{L\times1} - \mu_{1\times1} * \mathrm{e(n)}_{1\times1} * \mathrm{x}_f(\mathrm{n})_{1\times L}$	L + 1	L	–
(4)	$\mathrm{e(n)}_{1\times1} = \mathrm{d(n)}_{1\times1} - \mathrm{y_s(n)}_{1\times1}$	–	1	–
(5)	$\mathrm{y_s(n)}_{1\times1} = s(n)_{1xM} * \mathrm{y(n)}_{M\times1}$	M	M − 1	–
	Total	2L + 2M + 1	2L + 2M − 2	–

Table 5
Complexity analysis of NSS-FxLMS algorithm.

Eq.'s	Operations	$*$	+/−	÷
(1)	$\mathrm{x}_f(\mathrm{n})_{1\times1} = \hat{s}(n)_{1\times\mathrm{M}} * x(\mathrm{n})_{M\times1}$	M	M − 1	–
(2)	$\mathrm{y(n)}_{1\times1} = \mathrm{w^T(n)}_{1\times L} * x(\mathrm{n})_{L\times1}$	L	L − 1	–
(3)	$\mathrm{w(n+1)}_{L\times1} = \mathrm{w(n)}_{L\times1} - \mu(\mathrm{n})_{1\times1} * \mathrm{e(n)}_{1\times1} * \mathrm{x}_f(\mathrm{n})_{1\times L}$	L + 1	L	–
(4)	$\mu(\mathrm{n})_{1\times1} = \frac{\overline{\mu(\mathrm{n})_{1\times1}}}{\delta+\mathrm{x}_f^T(\mathrm{n})_{1\times L}*\mathrm{x}_f(\mathrm{n})_{L\times1}+\mathrm{E(n)}}$	L	L + 1	1
(5)	$\mathrm{e(n)}_{1\times1} = \mathrm{d(n)}_{1\times1} - \mathrm{y_s(n)}_{1\times1}$	–	1	–
(6)	$\mathrm{y_s(n)}_{1\times1} = s(n)_{1\times M} * \mathrm{y(n)}_{M\times1}$	M	M − 1	–
(7)	$\mathrm{E(n)}_{1\times1} = \lambda E(n-1)_{1\times1} + (1-\lambda)\mathrm{E}^2(\mathrm{n})_{1\times1}$	3	2	
	Total	3L + 2M + 4	3L + 2M + 1	1

Table 6
Complexity analysis of FxRLS algorithm.

Eq.'s	Operations	$*$	+/−	÷
(1)	$\mathrm{x}_f(\mathrm{n})_{1\times1} = \hat{s}(n)_{1\times\mathrm{M}} * x(\mathrm{n})_{M\times1}$	M	M − 1	–
(2)	$\mathrm{y(n)}_{1\times1} = \mathrm{w^T(n)}_{1\times L} * x(\mathrm{n})_{L\times1}$	L	L − 1	–
(3)	$\mathrm{w(n+1)}_{L\times1} = \mathrm{w(n)}_{L\times1} + \mathrm{K(n)_{L\times L}} * \mathrm{e(n)}_{1\times1}$	L	L	–
(4)	$\mathrm{K(n)_{L\times1}} = \frac{\pi(\mathrm{n})_{\mathrm{Lx1}}}{\lambda+\mathrm{x}_f(\mathrm{n})_{1\times L}^T*\pi(\mathrm{n})_{\mathrm{L}\times1}}$	2L	L	1
(5)	$\pi(\mathrm{n})_{\mathrm{L}\times1} = \mathrm{p(n-1)}_{\mathrm{L}\times L} * \mathrm{x}_f(\mathrm{n})_{L\times1}$	L^2	L^2-L	
(6)	$\mathrm{p(n)_{L\times L}} = \lambda^{-1} * \mathrm{p(n-1)_{L\times L}} - \lambda^{-1} * \mathrm{K(n)_{L\times1}} * \mathrm{x}_f(\mathrm{n})_{1\times L} * \mathrm{p(n-1)_{L\times L}}$	$3L^2$	$2L^2 - L$	1
(7)	$\mathrm{e(n)}_{1\times1} = \mathrm{d(n)}_{1\times1} - \mathrm{y_s(n)}_{1\times1}$	–	1	–
(8)	$\mathrm{y_s(n)}_{1\times1} = s(n)_{1\times M} * \mathrm{y(n)}_{M\times1}$	M	M − 1	–
	Total	$4L^2 + 4L + 2M$	$3L^2 + L + 2M - 2$	2

Table 7
Complexity analysis of DR-NSSFxLMS algorithm.

Eq.'s	Operations	*	+/−	÷
(1)	$x_f(n)_{1\times 1} = \hat{s}(n)_{1\times M} * x(n)_{M\times 1}$	M	M − 1	–
(2)	$y(n)_{1\times 1} = w^T(n)_{1\times L} * x(n)_{L\times 1}$	L	L − 1	–
(3)	$d_1(n)_{1\times 1} = e(n)_{1\times 1} + s(n)_{1\times m} * y(n)_{m\times 1}$	M	M	–
(4)	$e_1(n)_{1\times 1} = d_1(n)_{1\times 1} - w_1^T(n)_{1\times L} * x_f(n)_{L\times 1}$	L	L	–
(5)	Compute $w_1(n+1)_{L\times 1}$ using DR algorithm in Table 2 from [13]	N(3L + 4)	N(3L + 2)	–
	Total	2L + 2M + N(3L + 4)	2L + 2M − 2 + N(3L + 2)	–

6. Complexity analysis

Computational complexity of an algorithm is usually of significant importance particularly in real-time applications. The complexity of individual equations of newly proposed MGFxRLS algorithm is given in Table 3. Followed by the complexity analysis of other investigated algorithms in Tables 4–7 [20].

Table 8
Performance analysis of investigated algorithms.

Algorithm	Complexity		Memory
	Additions	Multiplications	
FxLMS	2L + 2 M − 2	2L + 2 M + 1	2(L + M)
NSS-FxLMS	3L + 2 M + 1	3L + 2 M + 4	2(L + M)
DRNSS-FxLMS	2L + 2 M − 2 + N(3L + 2)	2L + 2 M + N(3L + 4)	(N + 2)L + 3 M
FxRLS	$3L^2$ + L + 2 M − 2	$4L^2$ + 4L + 2 M	3L + 2 M
Proposed MGFxRLS	$3L^2$ + L + 2 M + 2	$4L^2$ + 4L + 2 M + 3	3L + 2 M

Here, L represents the number of filter coefficients, M represents the secondary path and N represents the data reuse order for DR-NSSFxLMS algorithm. The computational complexities

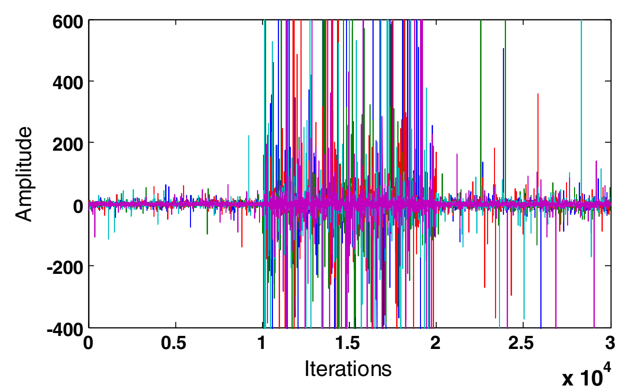

Fig. 6. Realizations of primary disturbance d(n) in non-stationary environment (case 1).

Table 9
Parameter set for proposed technique simulation.

ANC system			Impulsive noise		
Parameters	Symbol	Value	Parameters	Symbol	Value
Primary path tap size	L	256	Total samples	N	30,000
Secondary path tap size	M	128	Total realizations	Avg	5
Adaptive filter tap size	L_w	192	Characteristic exponent	α	1.45, 1.65, 0.95
FxRLS forgetting factor	λ	0.99	Scale parameter	γ	1
MGFxRLS forgetting factor	λ	0.99	Location parameter	C	0
			Skewness parameter	δ	0

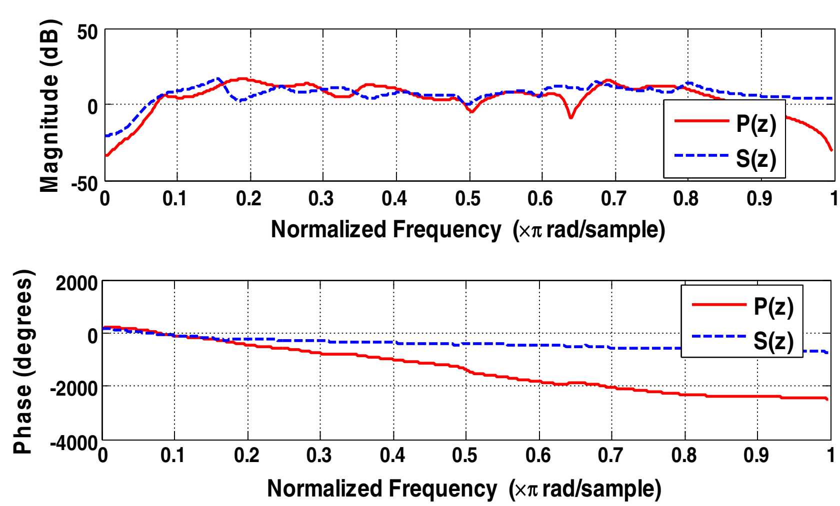

Fig. 5. Frequency response of acoustic primary and secondary path.

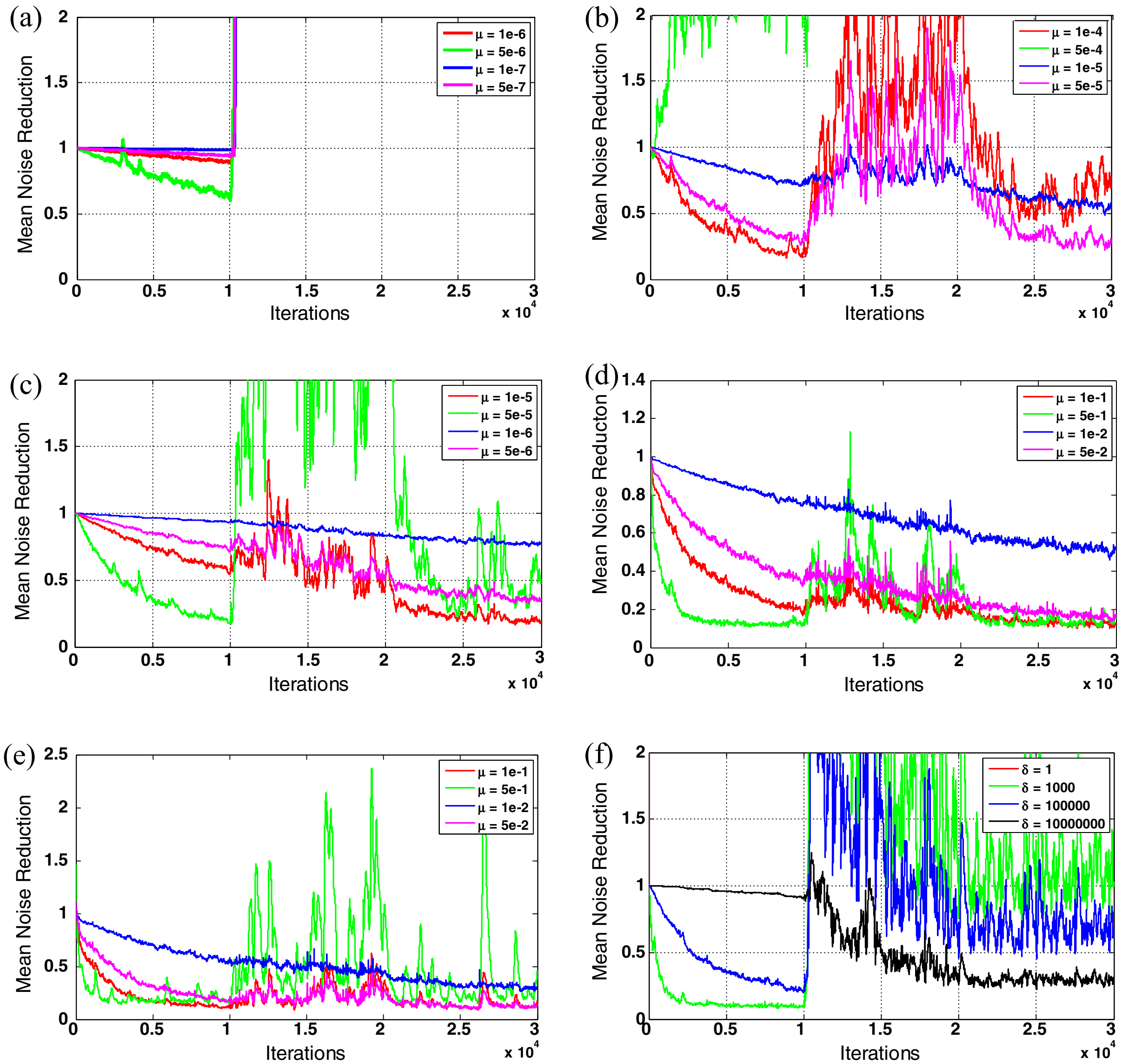

Fig. 7. MNR curves for (a) Sun's algorithm [10], (b) modified Sun's algorithm [11], (c) Akhtar's algorithm [11], (d) NSS-FxLMS algorithm [12], (e) DR-NSSFxLMS algorithm [13], and (f) FxRLS algorithm [18].

Table 10
Selected values of controlling parameter for investigated algorithms.

Algorithms	Selected value of controlling parameter	
	Case 1	Case 2
Sun's algorithm	$\mu = 5 \times 10^{-6}$	$\mu = 1 \times 10^{-4}$
Modified Sun's algorithm	$\mu = 1 \times 10^{-4}$	$\mu = 1 \times 10^{-4}$
Akhtar's algorithm	$\mu = 5 \times 10^{-5}$	$\mu = 1 \times 10^{-4}$
NSS-FxLMS algorithm	$\mu = 5 \times 10^{-1}$	$\mu = 5 \times 10^{-1}$
DR-NSSFxLMS algorithm	$\mu = 1 \times 10^{-1}$	$\mu = 5 \times 10^{-1}$
FxRLS algorithm	$\delta = 1000$	$\delta = 1000$

along with the memory requirements of the investigated algorithms are summarized in Table 8. The memory of investigated algorithms is calculated using the method given in [24].

7. Comparison with existing techniques and simulation results

The ANC system for impulsive noise is implemented using the MATLAB platform. The performance of proposed algorithm is compared with the performance of following algorithms:

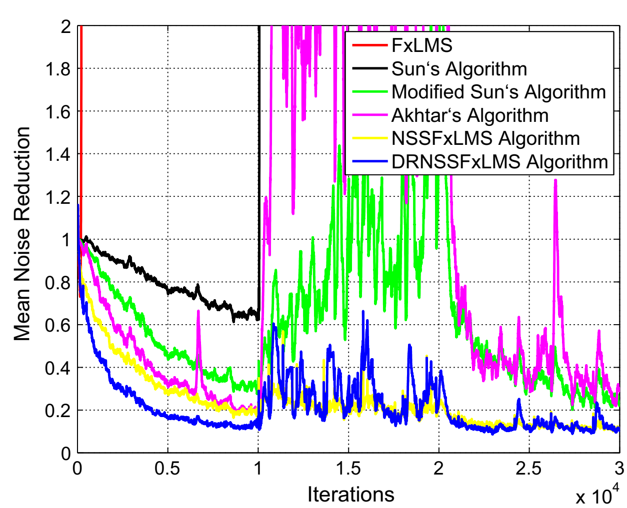

Fig. 8. MNR curves comparison of various algorithms for case 1.

- Sun's algorithm [10]
- Modified Sun's algorithm [11]
- Akhtar's algorithm [11]
- NSS-FxLMS algorithm [12]
- DR-NSSFxLMS algorithm [13]
- FxRLS algorithm [18]

The parameters used in simulating the ANC system are tabularized below in Table 9.

For the simplicity of our simulations, we have made an assumption that estimated secondary path model $\hat{s}(z)$ is same as $s(z)$ [13,21–23]. The numeric values of coefficients of primary and secondary acoustic paths are taken from data set given in [25]. The frequency response comprising of magnitude and phase of both path filters are depicted in Fig. 5.

The performance metric used for comparison of studied algorithms is mean noise reduction. It is calculated as

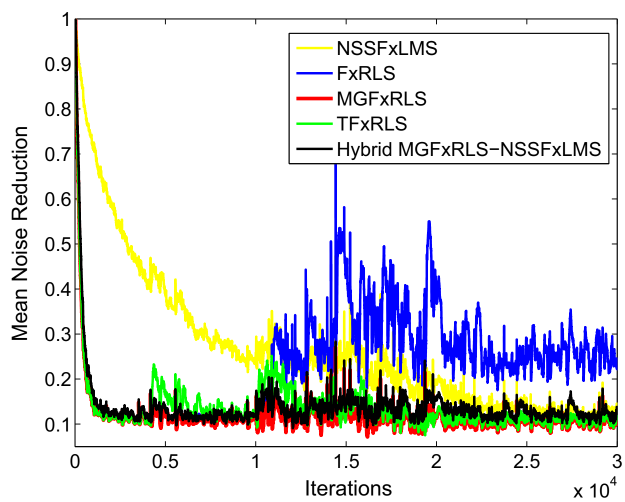

Fig. 9. MNR curves comparison of proposed algorithms for case 1.

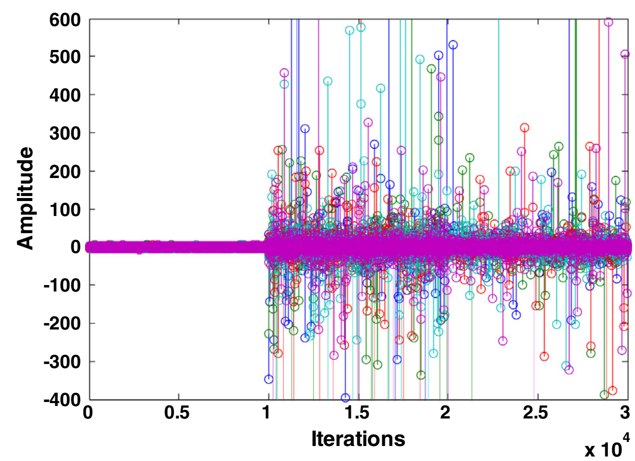

Fig. 11. Realizations of primary disturbance d(n) in case 2.

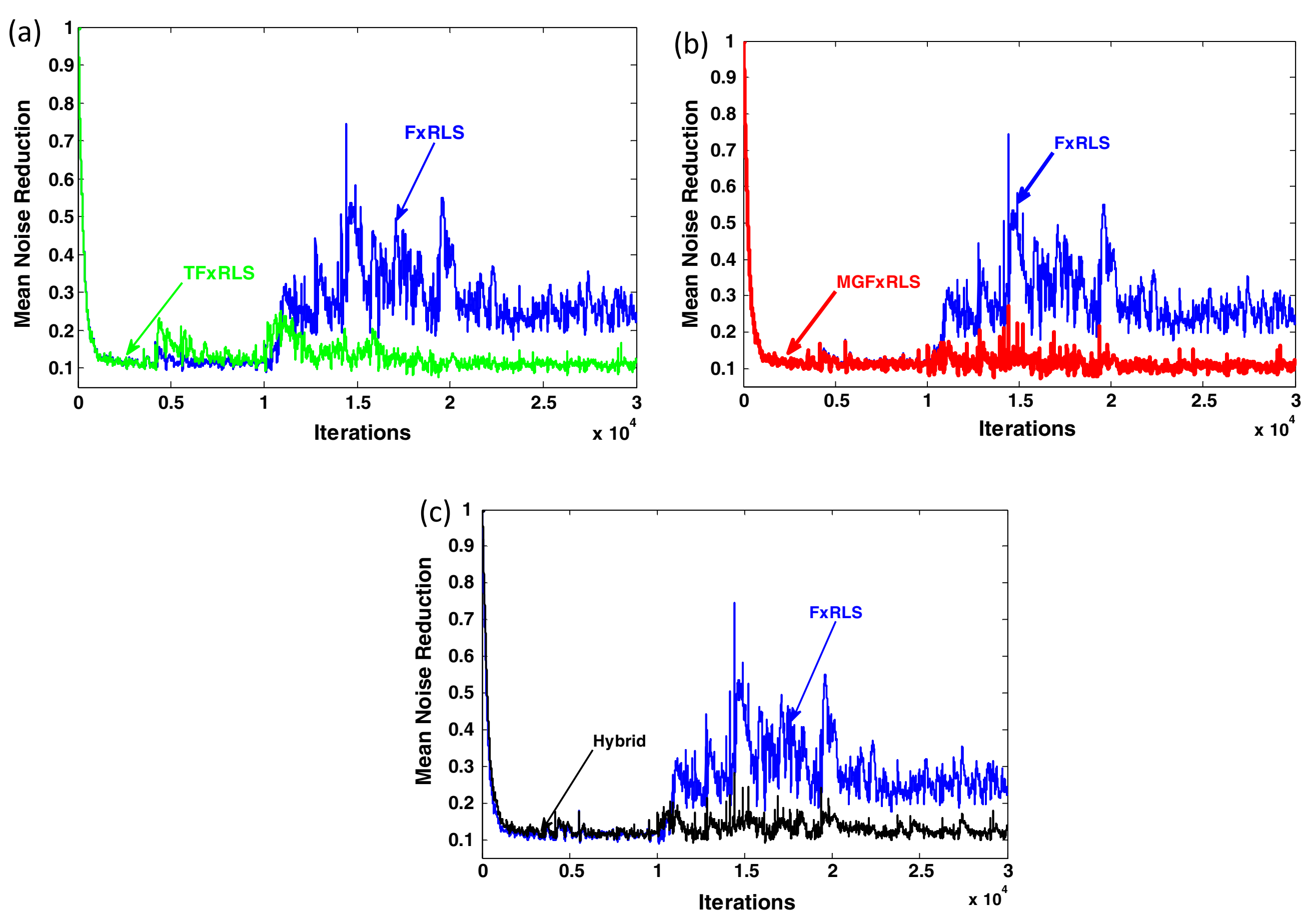

Fig. 10. MNR curves comparison of (a) TFxRLS, (b) MGFxRLS, and (c) hybrid algorithm with FxRLS algorithm. (Case 1).

$$MNR(n) = E\left\{\frac{A_e(n)}{A_d(n)}\right\} \tag{17}$$

$$A_e(n) = \lambda A_e(n-1) + (1-\lambda)|e(n)| \tag{18}$$

$$A_d(n) = \lambda A_d(n-1) + (1-\lambda)|d(n)| \tag{19}$$

where $A_e(n)$ and $A_d(n)$ are the estimate of the absolute value of residual error and disturbance signal.

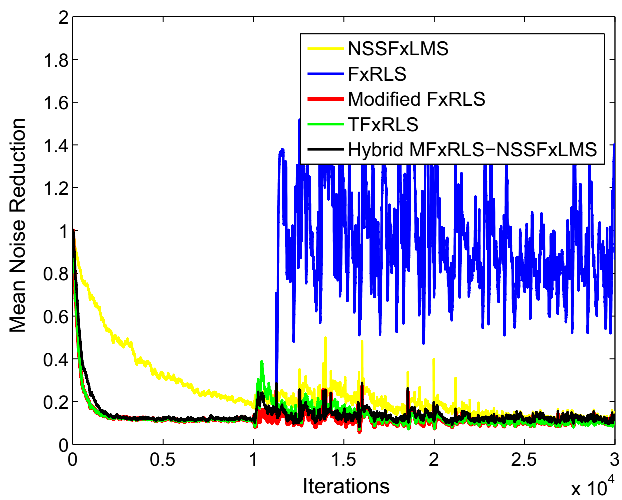

Fig. 12. MNR curves comparison of proposed algorithms for case 2.

In this section, we have validated the performance of our proposed algorithms for ANC of impulsive noise. The impulsive noise is generated by standard $S\alpha S$ model with non-stationary characteristics, by considering the following values for α:

- Case 1: $\alpha = \begin{cases} 1.65, & n \leqslant 10{,}000 \\ 0.95, & 10{,}000 < n \leqslant 20{,}000 \\ 1.45, & 20{,}000 < n \leqslant 30{,}000 \end{cases}$
- Case 2: $\alpha = \begin{cases} 2, & n \leqslant 10{,}000 \\ 1.1, & 10{,}000 < n \leqslant 20{,}000 \\ 1.3, & 20{,}000 < n \leqslant 30{,}000 \end{cases}$

7.1. Case 1

In case 1, the first 10,000 samples correspond to small impulsiveness ($\alpha = 1.65$), the next 10,000 samples depict an abrupt change to high impulsive environment ($\alpha = 0.95$) and finally, last 10,000 samples show moderate impulsiveness for ($\alpha = 1.45$). Fig. 6 shows butterfly plots for the corresponding primary disturbance $d(n)$ appearing at the error microphone in this case.

Extensive simulations are performed to find optimum values of controlling parameters of discussed algorithms. The detailed simulation results for effect of step size on the respective FxLMS family algorithms are presented in Fig. 7(a–e). Similarly, the effect of regularization parameter delta (δ) of FxRLS algorithm is shown in Fig. 7(f). The parameter δ depends on SNR [16], i.e. greater the value of SNR smaller value of delta is selected for better performance of algorithms and vice versa. The most suitable values of controlling parameters as observed from Fig. 7 are listed in Table 10.

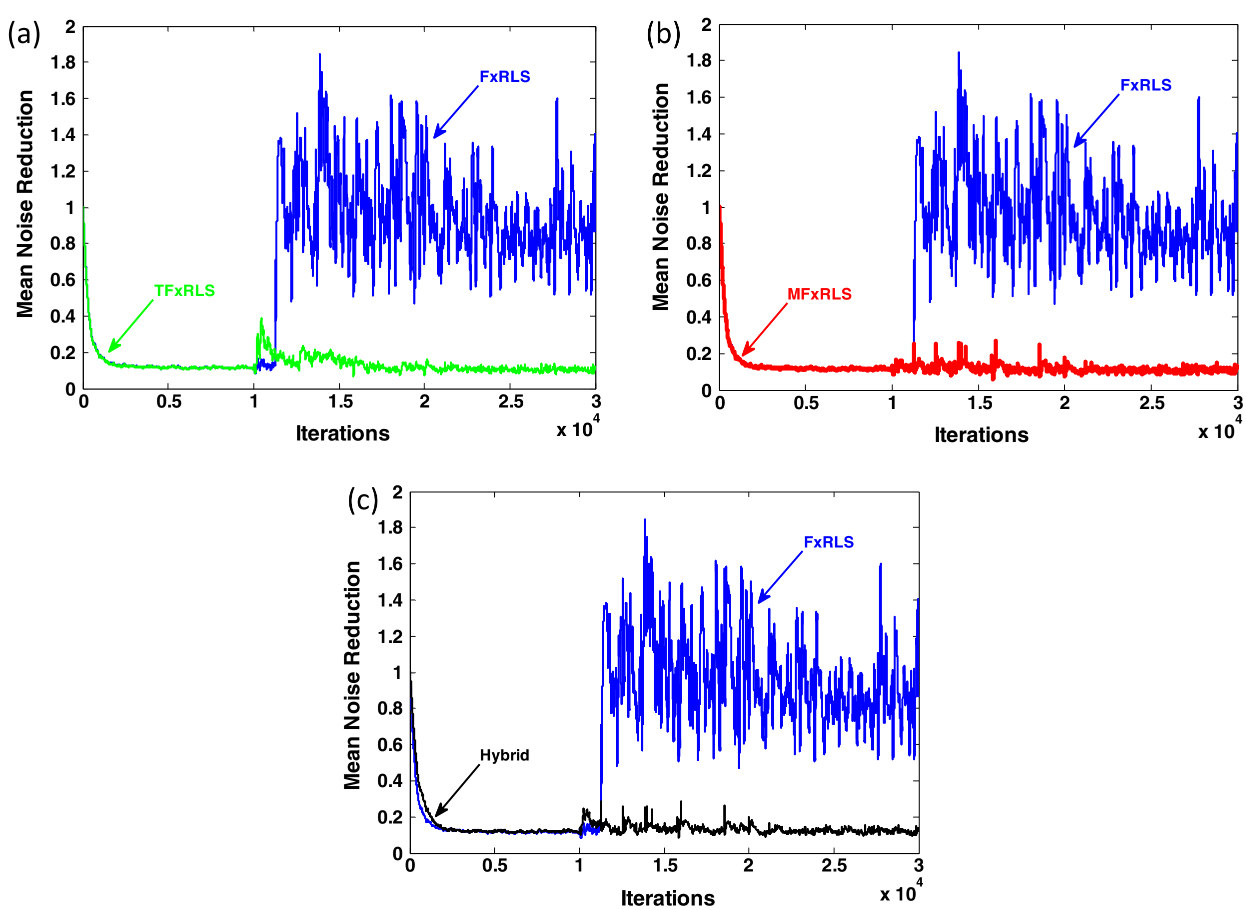

Fig. 13. MNR curves comparison of (a) TFxRLS, (b) MGFxRLS, and (c) hybrid algorithm with FxRLS algorithm. (Case 2).

Table 11
Simulation time comparison of investigated algorithms.

Algorithm	Simulation time (sec)	
	Case 1	Case 2
NSS-FxLMS algorithm	2.5598	2.5056
FxRLS algorithm	7.5574	8.1162
TFxRLS algorithm	7.4954	8.2148
MGFxRLS algorithm	7.5508	8.6314
Hybrid MGFxRLS-NSSFxRLS algorithm	4.4002	5.1526

Fig. 8 depicts the convergence curves of most widely used adaptive algorithms in ANC domain for the noise shown in Fig. 5. The optimum values of step size listed in Table 10 for the respective algorithms are used in this simulation. It can be seen that among the investigated algorithms of FxLMS family, the NSSFxLMS and DR-NSSFxLMS algorithms depict good convergence and stability. Furthermore, NSSFxLMS algorithm is more stable than DR-NSSFxLMS algorithm. Therefore, we have selected the NSS-FxLMS algorithm for further comparison with our proposed algorithms.

Fig. 9 shows that NSS-FxLMS algorithm gives slow convergence as compared to FxRLS family algorithms that achieves steady state value at about 20,000 iterations. The standard FxRLS algorithm gives fast convergence but is not robust enough to cater non-stationary noise. When noise value changes abruptly after 10,000 iterations, the performance of FxRLS algorithm is badly affected. The TFxRLS algorithm gives better performance in terms of stability and robustness than that of standard FxRLS algorithm under non-stationary noise environment. Furthermore, MGFxRLS algorithm is giving best performance among all investigated algorithms in terms of convergence, stability and robustness. However, in comparison with the FxLMS family algorithms, this improved performance is achieved at the cost of increased computational load. The newly proposed hybrid algorithm is giving almost same performance as that of MGFxRLS algorithm but with reduced computational load. The value of switching threshold for the simulation of this hybrid algorithm is selected to be 0.2. For more clarity, the individual comparison of proposed algorithms with the standard FxRLS algorithm is presented in Fig. 10(a–c).

7.2. Case 2

In case 2, the first part consisting of 10,000 samples corresponds to Gaussian noise ($\alpha = 2$) and in the next part high impulsive environment is encountered ($\alpha = 1.1$) which ends with moderate impulsiveness in the last 10,000 samples ($\alpha = 1.45$). As in case 1, extensive simulations are carried out (not shown in the paper) to find the most suitable step size parameter values for the investigated FxLMS family algorithms and delta value for the FxRLS algorithm. The selected values are shown in Table 10. Fig. 11 shows butterfly plots for the corresponding primary disturbance $d(n)$ appearing at the error microphone in this case.

The MNR performance is shown in Figs. 12 and 13, which again verify the enhanced performance of the proposed solutions as compared to standard FxRLS algorithm in terms of robustness. As seen from Fig. 12, the standard FxRLS algorithm is unstable under high impulsive environment, while the proposed TFxRLS algorithm is much stable as compared to FxRLS algorithm. The proposed MFxRLS and hybrid algorithms have even more better performance than that of TFxRLS algorithm in terms of stability and robustness.

In order to verify the reduced computational load of hybrid algorithm, the simulation time of the investigated algorithms is compared in Table 11.

It is concluded from Table 11 that the proposed hybrid MGFxRLS-NSSFxLMS algorithm has reduced computational load as compared to MGFxRLS algorithm and other investigated FxRLS family algorithms. So, the proposed hybrid MGFxRLS-NSSFxLMS algorithm gives enhanced robustness and stability with reduced complexity than that of standard FxRLS algorithm.

8. Conclusion

In this paper, new ANC algorithms based on filtered-x recursive least square (FxRLS) algorithm have been proposed to improve stability and robustness in the presence of impulsive noise. Simulation results depicted that the standard FxRLS algorithm gives better convergence than that of filtered-x least mean square (FxLMS) family algorithms but lacks robustness in presence of non-stationary impulsive noise. In order to enhance the robustness of FxRLS algorithm, two modifications were proposed. First one was threshold based FxRLS (TFxRLS) algorithm based on modifying the reference and error signals. In the second one the gain of standard FxRLS algorithm was modified to tackle high impulses, named as modified gain FxRLS (MGFxRLS) algorithm. Both modifications gave better performance in terms of stability and robustness than that of standard FxRLS algorithm. The computationally extensive nature of FxRLS algorithm is a major point of concern. In order to reduce the computational load of the overall ANC system based on FxRLS algorithm, a hybrid technique based on MGFxRLS algorithm and normalized step size FxLMS (NSS-FxLMS) algorithm was developed. The results proved the enhanced robustness and stability of proposed algorithms along with reduced computational load of hybrid algorithm than that of FxRLS family algorithms.

References

[1] Elliot SJ. Signal processing for active control. London, UK: Academic Press; 2001.
[2] Elliot SJ, Nelson PA. Active noise control. IEEE Signal Process Mag 1993;10(4):12–35.
[3] Kuo SM, Vijayan D. A secondary path modeling technique for active noise control systems. IEEE Trans Speech Audio Process 1997;5(4):374–7.
[4] Akhtar MT, Abe M, Kawamata M. A new variable step size LMS algorithm-based method for improved online secondary path modeling in active noise control systems. IEEE Trans Audio Speech Lang Process 2006;14(2):720–6.
[5] Zimmermann M, Dostert K. Analysis and modeling of impulsive noise in broad-band powerline communications. Electromagnet Compat IEEE Trans 2002;44(1):249–58.
[6] Lui L, Gujjula S, Thanigai P, Kuo SM. Still in womb: intrauterine acoustic embedded active noise control for infant incubators. Adv Acoust Vib 2008;2008:9.
[7] Zhou YI, Yin YX, Zhang QZ. An optical repetitive control algorithm for periodic impulsive noise attenuation in a non-minimum phase ANC system. Appl Acoust 2013;74:1175–81.
[8] Nikias CL, Shao M. Signal processing with alpha – stable distribution and applications. New York, USA: Wiley; 1995.
[9] Leahy R, Zhou Z, Hsu YC. Adaptive Filtering of Stable Processes for Active Attenuation of ImpulsiveNoise. In Proc IEEE ICASSP 1995, vol. 5; May 1995. p. 2983–6.
[10] Sun X, Kuo SM, Meng G. Adaptive algorithm for active conctrol of impulsive noise. J Sound Vib 2006;291:516–22.
[11] Akhtar MT, Mitsuhashi W. Improving performance of FxLMS algorithm for active noise control of impulsive noise. J Sound Vib 2009;327:647–56.
[12] Akhtar MT, Mitsuhashi W. A modified normalized FxLMS algorithm for active control of impulsive noise. In Proc EUSIPCO 2010; 23–27 August, 2010. p. 1–5.
[13] Akhtar MT, Nishihara A. Data reusing based filtered reference adaptive algorithms for active control of impulsive noise sources. Appl Acoust 2015;92:18–26.
[14] Lan H, Zhang M, Ser W. A weight constrained FxLMS algorithm for feedforward active noise control systems. IEEE Signal Procss Lett 2002;9(1):1–4.
[15] George NV, Panda G. A robust filtered-s LMS algorithm for nonlinear active noise control. Appl Acoust 2012;73(8):836–41.
[16] Haykin S. Adaptive filter theory. Upper Saddle River, NJ: Prentice-Hall; 2002.
[17] Reddy RM, Panahi IMS, Briggs R, Perez E. Performance comparison of FxRLS, FxAPA and FxLMS Active Noise Cancellation Algorithms on an fMRI Bore Test-Bed. In: Proc IEEE Dallas Engg Med Bio Workshop; 2007. p. 130.
[18] Zeb A, Mirza A, Sheikh SA. FxRLS algorithm based ANC of impulsive noise. In: Proc ICMIC; 2015. p. 18–20.
[19] Reddy RM, Panahi IMS, Briggs R. Hybrid FxRLS-FxNLMS adaptive algorithm for active noise control in fMRI application. IEEE Trans Control Sys Technol 2011;19(2):474–80.
[20] Mirza A, Zeb A, Sheikh SA. Robust adaptive algorithm for active control of impulsive noise. EURASIP J Adv Signal Process 2016;44:1–13.

[21] Akhtar MT. Binormalized data-reusing adaptive filtering algorithm for active control of impulsive sources. Digital Signal Process 2015;49:56–64. doi: http://dx.doi.org/10.1016/j.dsp.2015.11.002.
[22] Zhou Y, Zhang Q, Yin Y. Active control of impulsive noise with symmetric α-stable distribution based on an improved step-size normalized adaptive algorithm. Mech Syst Signal Process 2015;56:320–39. doi: http://dx.doi.org/10.1016/j.ymssp.2014.10.002.
[23] Sun G, Li M, Lim TC. A family of threshold based robust adaptive algorithms for active impulsive noise control. Appl Acoust 2015;97:30–6. doi: http://dx.doi.org/10.1016/j.apacoust.2015.04.003.
[24] Kuo SM, Panahi I, Chung KM, Horner T, Nadeski M, Chyan J. Design of active noise control systems with the TMS320 family. USA: Texas Instruments; 1996.
[25] Kuo SM, Morgan DR. Active noise control systems-algorithms and DSP implementation. New York: Wiley; 1996.