applied sciences

MDPI

Article

Developing a New Filtered-X Recursive Least Squares Adaptive Algorithm Based on a Robust Objective Function for Impulsive Active Noise Control Systems

Muhammad Tahir Akhtar

Department of Electrical and Computer Engineering, School of Engineering and Digital Sciences, Nazarbayev University, Astana 010000, Kazakhstan; muhammad.akhtar@nu.edu.kz or akhtar@ieee.org

Abstract: It is well-known that performance of the classical algorithms for active noise control (ANC) systems severely degrades when implemented for controlling the impulsive sources. The objective of this paper is to propose a new recursive least squares (RLS) algorithm (and its variant) for being implemented in the framework of ANC systems. The proposed RLS-based adaptive algorithm employs an objective function designed to achieve robustness against the impulse type sources. The derivation of the algorithm is quite straightforward; however, a few modifications have been incorporated to address the application at hand. In order to improve upon the numerical stability issue of RLS-based adaptation, it is suggested to employ smoothing while updating the inverse correlation matrix. Furthermore, it is proposed to introduce a step size in the update equation of the adaptive algorithm. This results in the fixed step-size modified filtered-x (MFx) robust RLS (FSS-MFxRRLS) algorithm. As expected, a fixed value step size results in a trade-off situation for convergence speed and steady-state misalignment. In order to address this issue of a trade-off situation, the idea of a convex combined step size (CCSS) is introduced into the adaptive procedure to develop the CCSS-MFxRRLS algorithm. When the ANC is started, the CCSS strategy (automatically) selects a large-valued step size to achieve a fast initial convergence. As the ANC system converges at the steady-state, the CCSS is automatically tuned to a small value which improves the steady-state performance of the proposed CCSS-MFxRRLS algorithm. Extensive simulations have been designed to mimic many scenarios for practical applications of ANC for impulsive sources. The simulation results demonstrate that the proposed CCSS-MFxRRLS algorithm is very effective in many practical scenarios involving ANC of impulsive sources.

Keywords: robust RLS adaptive filtering; active noise control; impulsive noise; convex-combined step size

Citation: Akhtar, M.T. Developing a New Filtered-X Recursive Least Squares Adaptive Algorithm Based on a Robust Objective Function for Impulsive Active Noise Control Systems. *Appl. Sci.* **2023**, *13*, 2715. https://doi.org/10.3390/app13042715

Academic Editors: Woon-Seng Gan, Dongyuan Shi and Jihui Aimee Zhang

Received: 12 December 2022
Revised: 15 February 2023
Accepted: 16 February 2023
Published: 20 February 2023

1. Introduction

The basic idea of active noise control (ANC), first conceptualized by P. Lueg in his US patent [1], is very simple: two out of phase acoustic waves would result in a destructive interference and hence cancel each other out. Therefore, any ANC system would essentially generate an anti-phase (acoustic) noise-cancelling signal, which is combined acoustically with the primary (disturbance or noise) signal generated from some noise source (for example, exhaust fan, vacuum cleaner, power transformer, etc.). These two signals, being anti-phase to each other, tend to cancel each other out [2]. Active noise and vibration control is a well-researched area that has spread from academic research in a tight laboratory settings to successful commercial products, thanks to great advancements in the modern semiconductor technology. Many successful applications of ANC technology have been reported in [3–8] (among others); however, the most popular is ANC headsets [9,10]. An excellent survey on recent research trends and interesting applications for ANC systems can be found in [11,12].

Broadly speaking, the ANC systems can be classified as single-channel ANC or multi-channel ANC, depending upon the number of sensors and actuators deployed to cover the area of interest. Here, only single-channel ANC has been considered, and the presented methods can be fairly easily extended for the multichannel ANC if needed. Furthermore, we consider acoustic noise sources without loss of generality, as the presented methods can be modified for application to active vibration control. In a basic setting of a single-channel scenario, the ANC system generally comprises one primary/reference sensor (e.g., microphone), one secondary actuator (e.g., loudspeaker), and one error sensor (e.g., microphone).

The objective of the reference microphone is to pick-up the reference/input noise prior to it propagating via the primary path and appearing around the error microphone—the location where the noise reduction is desired. An adaptive control filter generates the secondary (cancelling) signal, which is propagated via the secondary path between the loudspeaker and the error microphone. The ANC adaptive filter is adapted in such a way that the secondary (cancelling) signal is anti-phase to the primary noise appearing around the error microphone. This results in a destructive interference around the error microphone; hence, the error microphone records the residual noise, which, in turn, is used to guide adaptation of the control filter.

Due to the presence of a secondary (electro-acoustic) path between the cancelling loudspeaker and the error microphone, the classical least mean square (LMS) algorithm [13,14] cannot be directly implemented. In a variant of the LMS algorithm, the effect of the secondary path is compensated by employing a secondary path model to filter the reference signal and then use this filtered-reference signal in the LMS update equation, resulting in the so-called filtered-reference LMS, most commonly abbreviated as filtered-x (Fx) LMS (FxLMS) algorithm [3]. The classical FxLMS algorithm has been a first choice for practical ANC systems, mainly due to its reduced computational cost and simple implementation. However, the classical FxLMS algorithm suffers from slow convergence speed and shows poor performance (being a stochastic gradient-based algorithm) for non-Gaussian sources. In order to address slow convergence speed, different ANC algorithms have been proposed with improved convergence properties, viz., (1) lattice-ANC systems [15]; (2) infinite impulse response (IIR) filter-based LMS algorithms called Filtered-u Recursive LMS (FuRLMS) [16], and filtered-v algorithms [17]; (3) recursive least squares (RLS)-based algorithms called FxRLS [3] and Fx Fast-Transversal-Filter (FxFTF) [18]; and (4) frequency-domain-ANC systems (see [19] and references therein). Here, IIR-based structures have inherent stability problems, RLS-based ANCS exhibit numerical instability issues, and the other approaches mentioned above increase the computational complexity. These reasons make FxLMS still a good choice for ANC applications. However, the FxLMS algorithm does not appear as a viable choice for ANC of noise generated from non-Gaussian stable sources of an impulsive nature.

In many practical situations, viz., office, infant incubators, punching and cutting machines in industrial setups, traffic noise, gun shots and explosions, and noise in an MRI room, the target noise is non-Gaussian [20–23]. Such a noise signal exhibits an impulsive nature and hence can be better modeled by heavy tailed non-Gaussian (stable) distribution than Gaussian distribution. In order to address poor performance of the FxLMS algorithm for impulsive ANC systems, researchers have proposed many variants of the FxLMS algorithm, viz., Sun's algorithm [24], the thresholding FxLMS (Th-FxLMS) algorithm [25], the improved normalized step-size-based FxLMS (INSS-FxLMS) algorithm [26], the Fx logarithmic LMS (FxlogLMS) algorithm [27], and the Fx least mean M-estimate (FxLMM) algorithm [21,28,29]. Here, the INSS-FxLMS algorithm may exhibit poor performance for strongly impulsive sources, the FxlogLMS algorithm suffers from a problem of entering a dead zone during adaptation (due to characteristics of log operation), and the rest of the algorithms require estimating appropriate thresholding parameters such that reference and/or error signals are thresholded before being used in the update process. Obtaining appropriate threshold parameter values may pose a great challenge, especially if a suitable threshold needs to be estimated during the

online operation of impulsive ANC systems. An excellent review of threshold adaptive ANC algorithms for impulsive sources is found in [30].

In order to take into account the non-Gaussian nature of impulsive sources [31], the filtered-x least mean p-power (FxLMP) algorithm minimizes p-power of the error signal instead of the mean squared value [32]. Here, p is the fractional low-order moment which appears to be a robust choice compared with the second order moment considered in the derivation of the classical FxLMS algorithm [32]. The FxLMP algorithm does outperform the FxLMS algorithm for impulsive ANC systems; however, the convergence speed of the FxLMP algorithm is very slow, especially for strongly impulsive sources. Great effort has been devoted to improving the performance of the classical FxLMP algorithm, and many variants have been proposed [33–36]. It is worth mentioning that a suitable value of the parameter p must be found in relation to the characteristic of the noise source, which may be quite challenging in many situations. For several other robust cost-function-based algorithms, for example the least-mean-kurtosis-based adaptive algorithm, the interested reader is referred to [37].

This paper develops a robust adaptive algorithm (and its variant) for impulsive ANC systems. The key features of the developed algorithm(s) are summarized as follows:

- It is well-known that the RLS-based adaptation exhibits a far better convergence speed compared with the stochastic gradient-based adaptation (as in the LMS algorithm), where the price to be paid is increased computational complexity. The computational complexity may not be an issue with the recent advancements in semiconductor technology; therefore, in order to realize a fast convergence speed, the proposed algorithm is derived in the framework of an RLS-based adaptation.
- Instead of FxLMS-based ANC, the 'modified Fx' (MFx) LMS (MFxLMS)-based ANC structure [38] is used to implement the proposed algorithm. The MFxLMS ANC uses two copies of the ANC filter so that generation of the noise control signal and adaptation of coefficients of the ANC filter are treated as separate processes. This allows implementing advanced adaptive algorithms for ANC systems (for details, see [38] and the references therein).
- In the classical FxRLS algorithm [3], the cost function is (deterministic) least squares which is not well-suited for the application at hand, i.e., impulsive ANC systems. The RLS-based adaptive ANC algorithm developed in this paper employs an objective function expected to achieve robustness against impulsive sources. Furthermore, a fixed step size (FSS) is introduced in the proposed algorithm while computing the the increment vector, resulting in the FSS MFx robust RLS (FSS-MFxRRLS) algorithm for ANC of impulsive sources.
- In order to address the issue of the numerical instability of the FxRLS algorithm [18], a very simple solution of low-pass filtering is incorporated while computing the current estimate of the inverse correlation matrix.
- In previous work, a convex-combined step size (CCSS) based modified normalized FxLMS (CCSS-MNFxLMS) algorithm was developed [39,40]. Motivated by the result presented there, it is suggested to incorporate a similar strategy with the FSS-MFxRRLS ANC algorithm and develop the proposed CCSS-based MFx robust RLS (CCSS-MFxRRLS) algorithm.

A brief overview of the classical FxLMS and FxRLS algorithms is given in Section 2. The details on the development and derivation of the proposed RLS-based ANC algorithms are given in Section 3, and Section 4 presents results of the extensive numerical simulations. A few remarks on the issue of computational complexity are also presented in this section. Finally, Section 5 gives a few concluding remarks and directions for future work. A short version of this paper was presented at [41].

2. Classical ANC Algorithms

2.1. Classical FxLMS Algorithm

A block diagram of a single-channel ANC system [3] for duct applications is illustrated in Figure 1, where $P(z)$ denotes the primary path present between the noise source generating $x(n)$ and the location of the error microphone. Here, $S(z)$ is the secondary acoustic path present between the loudspeaker and the error microphone. As shown in Figure 1, the noise source is assumed to be of impulsive nature, further described while discussing the simulation results presented later. The reference signal $x(n)$ is propagated via $P(z)$ and appears as the disturbance $d(n)$ around the location of the error microphone.

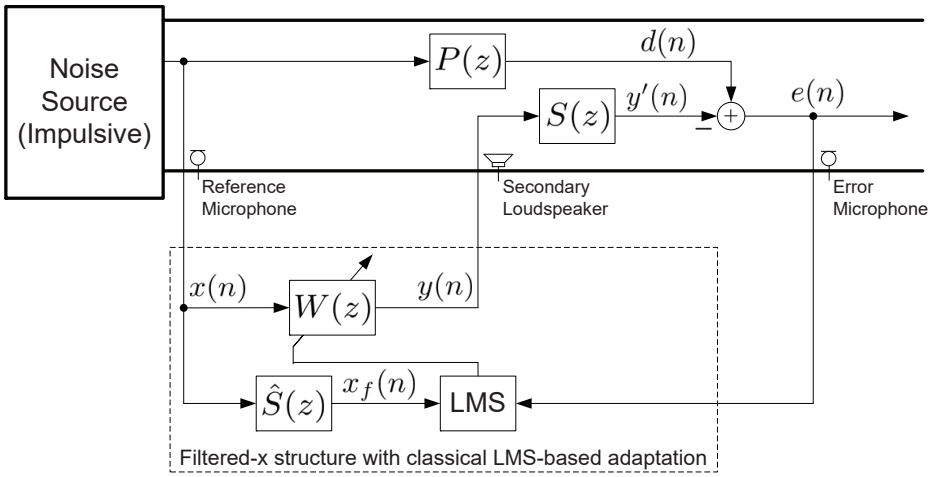

Figure 1. Block diagram of a filtered-x (Fx) LMS-based ANC structure.

Assuming that the ANC filter $w(n) \overset{z}{\longleftrightarrow} W(z)$ (having impulse response $w(n)$) is a finite impulse–response (FIR) filter of coefficient length L, the corresponding ANC filter output signal $y(n)$ can be expressed as

$$\begin{aligned} y(n) &= w(n) * x(n), \\ &= \boldsymbol{w}^T(n)\boldsymbol{x}(n), \end{aligned} \tag{1}$$

where $\boldsymbol{w}(n) = [w_0(n), w_1(n), \ldots, w_{L-1}(n)]^T$ and $\boldsymbol{x}(n) = [x(n), x(n-1), \ldots, x(n-L+1)]^T$ are the coefficient vector for $W(z)$ and the corresponding reference signal vector, respectively, and T and $*$ denote transposition and convolution, respectively. The ANC filter output signal $y(n)$ is filtered via $S(z)$ to give the cancellation signal $y'(n)$ which is acoustically combined with $d(n)$ to generate the residual error signal $e(n)$ being expressed as

$$\begin{aligned} e(n) &= d(n) - y'(n) \\ &= d(n) - s(n) * y(n) \\ &= d(n) - s(n) * w(n) * x(n). \end{aligned} \tag{2}$$

It is important to mention that neither the disturbance signal $d(n)$ nor the cancelling signal $y'(n)$ are directly accessible; $y(n)$ is generated from the ANC filter and then only $e(n)$ is picked-up by the error microphone. Assuming slow adaptation of the ANC filter $W(z)$ allows rearranging the convolutions in (2) and re-writing the expression for $e(n)$ as

$$\begin{aligned} e(n) &= d(n) - w(n) * x_f(n), \\ &= d(n) - \boldsymbol{w}^T(n)\boldsymbol{x}_f(n), \end{aligned} \tag{3}$$

where $\boldsymbol{x}_f(n) = [x_f(n), x_f(n-1), \ldots, x_f(n-L+1)]^T$ is the Fx signal vector where $x_f(n) = s(n) * x(n)$. The the classical FxLMS algorithm is based on minimizing the mean-squared-error (MSE)

$$\xi_{\text{MSE}}(n) = \mathbb{E}\{e^2(n)\} \approx e^2(n), \tag{4}$$

which results in the following updated equation for the FxLMS algorithm [3,4]

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu e(n)\boldsymbol{x}_f(n), \tag{5}$$

where μ is a FSS parameter. In (3) and (5), the Fx signal $x_f(n)$ can be computed using the secondary path modeling filter $\hat{s}(n) \xleftrightarrow{z} \hat{S}(z)$ as $x_f(n) = \hat{s}(n) * x(n)$ (see Figure 1).

2.2. Classical FxRLS Algorithm

The classical FxLRS algorithm considers the following deterministic least squares (LS)-based cost function [42]

$$\xi_{\text{LS}}(n) = \sum_{i=1}^{n} \lambda^{n-i} e^2(i), \tag{6}$$

where λ is the exponential forgetting factor. Using (3), $e(i)$ is the a posteriori error given as

$$e(i) = d(i) - \boldsymbol{x}_f^T(i)\boldsymbol{w}(n), \tag{7}$$

where $\boldsymbol{w}(n)$ is the currently available estimate of the ANC control filter. Computing the gradient of $\xi_{\text{LS}}(n)$ and equating it to zero

$$\frac{\partial \xi_{\text{LS}}(n)}{\partial \boldsymbol{w}(n)} = -2\sum_{i=1}^{n} \lambda^{n-i} \boldsymbol{x}_f(i)e(i) = 0, \tag{8}$$

and then following the derivation for the standard RLS algorithm as in [42], we arrive at the following computations for the classical FxRLS algorithm for ANC systems:

$$\boldsymbol{u}(n) = \boldsymbol{\Phi}(n-1)\boldsymbol{x}_f(n), \tag{9}$$

$$\boldsymbol{k}(n) = \frac{\boldsymbol{u}(n)}{1 + \boldsymbol{x}_f^T(n)\boldsymbol{u}(n)}, \tag{10}$$

$$\boldsymbol{\Phi}(n) = \lambda^{-1}\boldsymbol{\Phi}(n-1) - \lambda^{-1}\boldsymbol{k}(n)\boldsymbol{u}^T(n), \tag{11}$$

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \varepsilon(n)\boldsymbol{k}(n), \tag{12}$$

where $\boldsymbol{k}(n)$ is the Kalman gain vector which, in addition to (10), can be equivalently expressed as $\boldsymbol{k}(n) = \boldsymbol{\Phi}(n)\boldsymbol{x}_f(n)$ [42], and $\varepsilon(n)$ is the a priori estimation error expressed as

$$\varepsilon(n) = d(n) - \boldsymbol{x}_f^T(n)\boldsymbol{w}(n-1), \tag{13}$$

which requires the previous coefficient vector $\boldsymbol{w}(n-1)$ and information about $d(n)$. Unfortunately, the latter is present only in the acoustic domain and is not directly accessible. Another important issue to circumvent is the poor numerical stability of the classical FxRLS algorithm for ANC systems [3,18]. Nevertheless, it is well-known that RLS-based adaptive filters exhibit faster convergence compared to the stochastic-gradient-based LMS adaptive filters, though at a cost of increased computational complexity. Considering that modern

digital hardware allow implementing complex algorithms, it is worth developing FxRLS-based adaptive filtering solutions for practical applications. In the following- we explore developing a new FxRLS algorithm based an an objective function well suited for impulsive ANC systems.

3. Proposed Algorithm

3.1. Derivation of Robust FxRLS Algorithm

It is important to mention that the classical FxLMS and FxRLS algorithms are, respectively, based on MSE cost function $\xi_{\text{MSE}}(n)$ (4) and LS cost function $\xi_{\text{LS}}(n)$ (6) which are not robust for noise sources of impulsive nature. In order to develop a robust objective function for the impulsive ANC systems, we consider the following function [43]

$$f(n) = \mathbb{E}\left\{\frac{1}{1+\beta e^2(n)}\right\}, \tag{14}$$

where β is a constant selected empirically. It is noticed that $f(n) \to 0$ for impulsive sources and hence would be robust in mitigating the effect of impulsiveness. On the basis of function $f(n)$ in (14), following an LS-based objective function is proposed to derive the RLS-based algorithms presented in this paper

$$\xi_{\text{m}}(n) = \sum_{i=1}^{n} \lambda^{n-i}\left\{\frac{1}{1+\beta e^2(i)}\right\}, \tag{15}$$

where the subscript m signifies 'modified' objective function (in comparison with the standard LS cost function in (6)), and $e(i)$ is the same as expressed in (7). In order to minimize the error signal $e(n)$ so that reduced residual noise appears around the error microphone, the objective function $\xi_{\text{m}}(n)$ in (15) must be maximized. This can be achieved by computing the gradient of $\xi_{\text{m}}(n)$ and equating it to zero as

$$\frac{\partial \xi_{\text{m}}(n)}{\partial \boldsymbol{w}(n)} = 0, \tag{16}$$

which results in the following equation

$$\sum_{i=1}^{n} \lambda^{n-i} m(i) \boldsymbol{x}_f(i) e(i) = 0, \tag{17}$$

where the parameter $m(i)$ is given as

$$m(i) = \left\{\frac{1}{(1+\beta e^2(i))^2}\right\}. \tag{18}$$

Substituting (7) in (17) and performing some straightforward computations, we obtain the following normal equations

$$\boldsymbol{R}_{\text{m}}(n)\boldsymbol{w}(n) = \boldsymbol{p}_{\text{m}}(n) \quad \Rightarrow \quad \boldsymbol{w}(n) = \boldsymbol{\Phi}_{\text{m}}(n)\boldsymbol{p}_{\text{m}}(n), \tag{19}$$

where $\boldsymbol{R}_{\text{m}}(n)$ is the (modified) weighted correlation matrix given as

$$\boldsymbol{R}_{\text{m}}(n) = \sum_{i=1}^{n} \lambda^{n-i} m(i) \boldsymbol{x}_f(i) \boldsymbol{x}_f^T(i), \tag{20}$$

and $\boldsymbol{p}_{\text{m}}(n)$ is the (modified) weighted cross-correlation vector being expressed as

$$\boldsymbol{p}_{\text{m}}(n) = \sum_{i=1}^{n} \lambda^{n-i} m(i) \boldsymbol{x}_f(i) d(i), \tag{21}$$

and $\boldsymbol{\Phi}_{\mathrm{m}}(n) = \boldsymbol{R}_{\mathrm{m}}^{-1}(n)$ denotes the so-called inverse correlation matrix. Following the derivation of the classical RLS algorithm [42,44], we obtain

$$\boldsymbol{u}_{\mathrm{m}}(n) = \boldsymbol{\Phi}_{\mathrm{m}}(n-1)\boldsymbol{x}_f(n), \tag{22}$$

$$\boldsymbol{k}_{\mathrm{m}}(n) = \frac{\boldsymbol{u}_{\mathrm{m}}(n)}{\frac{\lambda}{m(n)} + \boldsymbol{x}_f^T(n)\boldsymbol{u}_{\mathrm{m}}(n)}, \tag{23}$$

$$\boldsymbol{\Phi}_{\mathrm{m}}(n) = \lambda^{-1}\boldsymbol{\Phi}_{\mathrm{m}}(n-1) - \lambda^{-1}\boldsymbol{k}_{\mathrm{m}}(n)\boldsymbol{u}_{\mathrm{m}}^T(n), \tag{24}$$

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \varepsilon(n)\boldsymbol{k}_{\mathrm{m}}(n), \tag{25}$$

where $\boldsymbol{k}_{\mathrm{m}}(n)$ is the 'modified' Kalman gain vector which, in addition to (23), can be equivalently expressed as $\boldsymbol{k}_{\mathrm{m}}(n) = m(n)\boldsymbol{\Phi}_{\mathrm{m}}(n)\boldsymbol{x}_f(n)$, $\varepsilon(n)$ is the a priori estimation error as given in (13), and $m(n)$ may be computed from (18). As noted from the above details, the derivation of the proposed algorithm is quite straightforward; however, there are many issues from the viewpoint of implementation for impulsive ANC systems and are tackled in the following subsections.

3.2. A Few Practical Considerations

With reference to the (modified) RLS algorithm given in (22)–(25), the a priori estimation error $\varepsilon(n)$ required in the coefficient update equation (25) would require the current value of $d(n)$ (see (13)). As stated earlier, $d(n)$ is not directly accessible in the ANC setup, and only the residual error signal $e(n)$ is available via the error microphone (see Figure 1). Therefore, it is proposed to implement the developed RLS-based algorithm in an MFx-based ANC structure instead of the classical Fx-based ANC structure shown in Figure 1. For details on the convergence properties of an MFx-based ANC structure, the interested reader may see [38] and references therein.

A block diagram for the proposed FxRLS algorithm-based ANC system implemented in an MFx structure is shown in Figure 2, which employs two copies of the secondary path modeling filer $\hat{S}(z)$. One copy of $\hat{S}(z)$ is used to 'filter' the reference signal $x(n)$ to obtain the Fx signal $x_f(n)$. The other copy of $\hat{S}(z)$ is used to filter the ANC filter output signal $y(n)$ to obtain an estimate of the canceling signal $\hat{y}'(n)$, which in turn is combined with $e(n)$ to obtain an estimate of the disturbance signal $\hat{d}(n)$ as

$$\begin{aligned}\hat{d}(n) &= e(n) + \hat{y}'(n),\\ &= e(n) + \hat{s}(n) * y(n),\end{aligned} \tag{26}$$

which is used to modify (13) as

$$\varepsilon(n) \equiv g(n) = \hat{d}(n) - \boldsymbol{x}_f^T(n)\boldsymbol{w}(n-1). \tag{27}$$

As shown in Figure 2, the MFx-based ANC structure employs two copies of ANC filter $W(z)$ where the coefficients of the adaptive one are updated using $\varepsilon(n) \equiv g(n)$ instead of $e(n)$. As in the update equation (12) of the classical FxRLS algorithm, the increment vector (to update the coefficients of the control filter) in (25) in the proposed algorithm is computed as a product of the (modified) Kalman gain vector (23) and 'estimated' a priori estimation error (27). Additionally, we suggest to introduce a step size μ to have a control over the adaptation, and hence, the adaptation in (25) is proposed to be modified as

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu g(n)\boldsymbol{k}_m(n). \tag{28}$$

After adaptation, the coefficients of the adaptive one are copied to the main ANC filter $W(z)$ responsible for generating $y(n)$ which will be used in performing noise cancellation. Equation (23) shows that computing the (modified) Kalman gain vector in the proposed algorithm requires computing $m(n)$ (as defined in (18)). Considering (18) and the above modification, the parameter $m(n)$ in the proposed algorithm is suggested to be computed as

$$m(n) = \left\{ \frac{1}{(1+\beta e^2(n))^2} \right\} \approx \left\{ \frac{1}{(1+\beta g^2(n))^2} \right\}. \tag{29}$$

Finally, considering the application at hand for ANC of impulsive noise sources, a smoothed version of the inverse correlation matrix in (24) is computed as

$$\mathbf{\Phi}_{\mathrm{m}}(n) = \zeta \mathbf{\Phi}_{\mathrm{m}}(n-1) + (1-\zeta)\left[\lambda^{-1}\mathbf{\Phi}_{\mathrm{m}}(n-1) - \lambda^{-1}\boldsymbol{k}_{\mathrm{m}}(n)\boldsymbol{u}_{\mathrm{m}}^T(n)\right], \tag{30}$$

where $0 \ll \zeta < 1$ is the forgetting factor. The resulting FSS-MFxRRLS ANC algorithm essentially replaces (24) and (25) with (30) and (28), respectively.

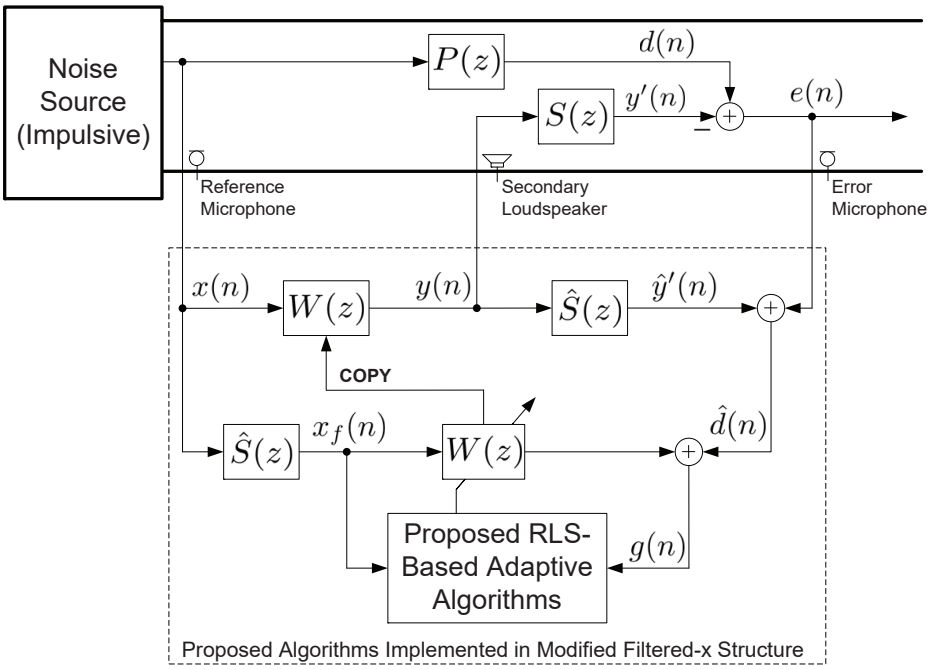

Figure 2. Block diagram of ANC structure employing proposed RLS-based adaptive algorithms.

3.3. Developing the Convex Combined Step Size Approach

It is no surprise that, like any FSS-based adaptive algorithm, the proposed FSS-MFxRRLS algorithm would also exhibit a trade-off situaiton for selecting the FSS: a large step size gives fast convergence but poor steady-state performance, and a small value would slow down the convergence speed while improving the performance at the steady state. Such a trade-off situation can be overcome by considering a variable step size (VSS) procedure which would automatically 'tune' the step size starting from a large value at the start to a small value as time progresses [45–48]. This would help in achieving both a fast initial convergence as well as a good steady-state performance. Recently, employing a convex combination of two adaptive filters has gained some popularity [49], where one adaptive filter is adapted using a large step size and the other using a small one. Their outputs are combined using a (convex) mixing parameter such that the overall output is decided by a fast converging adaptive filter (using a large step size) during the transient state and by a slow converging adaptive filter (using a small step size) at the steady state [50,51]. Since both adaptive filters are adapted simultaneously, it is obvious that this approach wound result in an increased computational complexity.

As in the previous work [39,40] for FxLMS-based ANC, a CCSS is developed for the FxRLS-based ANC algorithm described above. It is worth mentioning that the CCSS approach borrows the concepts of the convex combination of adaptive filters, yet uses only ONE adaptive filter to perform the adaptation [52,53] as explained below. The key idea is to develop a time-varying step size, as in the classical VSS-based algorithms, however by employing the concepts of a convex combination of adaptive filters. With reference to Figure 2 for the proposed method, it is 'assumed' that two copies of $W(z)$ (adaptive one) are available to be updated according to (28) as

$$\boldsymbol{w}_1(n) = \boldsymbol{w}_1(n-1) + \mu_1 g(n)\boldsymbol{k}_{\mathrm{m}}(n), \tag{31}$$

$$\boldsymbol{w}_2(n) = \boldsymbol{w}_2(n-1) + \mu_2 g(n)\boldsymbol{k}_{\mathrm{m}}(n). \tag{32}$$

By selecting $\mu_1 > \mu_2$, it is understood that $W_1(z)$ shows fast convergence speed and $W_2(z)$ would exhibit good performance at the steady-state. In order to combine these two desired properties, the adaptive ANC filter $W(z)$ can have a coefficient vector obtained via the convex combination of coefficient vectors given in (31) and (32) for $W_1(z)$ and $W_2(z)$, respectively,

$$\boldsymbol{w}(n) = \gamma(n)\boldsymbol{w}_1(n) + (1-\gamma(n))\boldsymbol{w}_2(n), \tag{33}$$

where $\gamma(n)$ is a mixing parameter. It is a time-varying parameter (as explained later) whose value must be selected in the range $0 \leq \gamma(n) \leq 1$. From (31)–(33), we obtain the following expression

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu(n)g(n)\boldsymbol{k}_{\mathrm{m}}(n), \tag{34}$$

for updating the coefficient vector of the adaptive ANC filter $W(z)$. In (34), $\boldsymbol{w}(n-1)$ is the convex combination of previous values of the coefficient vectors of $W_1(z)$ and $W_2(z)$ and can be expressed as

$$\boldsymbol{w}(n-1) = \gamma(n)\boldsymbol{w}_1(n-1) + (1-\gamma)\boldsymbol{w}_2(n-1), \tag{35}$$

and $\mu(n)$ is the CCSS which combines μ_1 and μ_2 via mixing parameter $\gamma(n)$ as

$$\mu(n) = \gamma(n)\mu_1 + (1-\gamma(n))\mu_2. \tag{36}$$

From (34) and (36), it is noticed that $\gamma(n) = 1$ ($\gamma(n) = 0$) effectively means as if $W(z)$ is adapted using a large (small)-valued step size μ_1 (μ_2). As in any VSS-based algorithm, we would like to have $\mu(n) \approx \mu_1$ at $n = 0$ to achieve fast convergence at the start-up and $\mu(n) \approx \mu_2$ to obtain improved steady-state performance when $n \to \infty$. This can be achieved by designing the mixing parameter $\gamma(n)$ in such a way that $\gamma(n) = 1$ at $n = 0$ and $\gamma(n) \to 0$ as the ANC system converges at the steady state ($n \to \infty$). One option to have such a mixing parameter $\gamma(n)$ is to consider the sigmoidal activation function as [51]

$$\gamma(n) = \frac{1}{1+\exp\{-b(n)\}}, \tag{37}$$

where $b(n)$ is another adaptive parameter. Following a similar approach as presented in [39,40], the parameter $b(n)$ in the proposed algorithm is adapted using a stochastic gradient algorithm to minimize the a priori estimation error $g(n) \equiv \varepsilon(n)$ as follows

$$b(n) = b(n-1) - \mu_b \frac{\partial |g(n)|}{\partial b(n-1)}, \tag{38}$$

where μ_b is a small-valued FSS for the adaptive parameter $b(n)$, employing $|g(n)|$ (instead of $g^2(n)$, for example) would result in a 'sign' adaptive algorithm which is expected

to achieve robustness for the sources of impulsive nature. Using $g(n)$ as given in (27), (38) becomes

$$b(n) = b(n-1) + \mu_b \text{sgn}(g(n)) \boldsymbol{x}_f^T(n) \frac{\partial \boldsymbol{w}(n-1)}{\partial b(n-1)}. \tag{39}$$

We consider the following chain rule to compute $\frac{\partial \boldsymbol{w}(n)}{\partial b(n)}$:

$$\frac{\partial \boldsymbol{w}(n)}{\partial b(n)} = \frac{\partial \boldsymbol{w}(n)}{\partial \mu(n)} \cdot \frac{\partial \mu(n)}{\partial \gamma(n)} \cdot \frac{\partial \gamma(n)}{\partial b(n)}, \tag{40}$$

where three partial derivatives on the right-hand side (R.H.S) are computed by considering (34), (36) and (37), respectively, as

$$\frac{\partial \boldsymbol{w}(n)}{\partial \mu(n)} = g(n) \boldsymbol{k}_{\text{m}}(n), \tag{41}$$

$$\frac{\partial \mu(n)}{\partial \gamma(n)} = (\mu_1 - \mu_2), \tag{42}$$

$$\frac{\partial \gamma(n)}{\partial b(n)} = [\gamma(n)(1-\gamma(n))]. \tag{43}$$

Combining (41)–(43) in (40) and replacing $n \to n-1$, the update rule in (39) becomes:

$$\begin{aligned} b(n) \quad = \quad & b(n-1) + \mu_b \text{sgn}(g(n))(\mu_1 - \mu_2) \\ & [\gamma(n-1)(1-\gamma(n-1))] \boldsymbol{x}_f^T(n) g(n-1) \boldsymbol{k}_{\text{m}}(n-1), \end{aligned} \tag{44}$$

where a few modifications are suggested as follows: $\text{sgn}(g(n))$ is replaced with $\text{sgn}(g(n-1))$ (in order to align with the error signal $g(n-1)$) and $(\mu_1 - \mu_2)$ (being a scalar constant) is assumed to be absorbed with the FSS μ_b, and (44) becomes:

$$b(n) = b(n-1) + \mu_b \text{sgn}(g(n-1)) g(n-1) [\gamma(n-1)(1-\gamma(n-1))] \boldsymbol{x}_f^T(n) \boldsymbol{k}_{\text{m}}(n-1). \tag{45}$$

Equation (45) dictates that the increment term (to be added to $b(n-1)$ from the previous iteration) includes a product term $[\gamma(n)(1-\gamma(n))]$. This shows that $b(n)$ would not be updated when $\gamma(n) \to 0$ or $\gamma(n) \to 1$. One idea could be to add/subtract a small positive number ϵ to/from $\gamma(n)$ when $\gamma(n) \to 0/1$. Yet another idea could be looking at the evolution of $\gamma(n)$ versus the adaptive parameter $b(n)$, as shown in Figure 3 which essentially plots the expression in (37). We observe that $\gamma(n) \to 0$ as $b(n) \to -\infty$ and $\gamma(n) \to 1$ as $b(n) \to +\infty$. Considering this observation, $b(n)$ is restricted between $[-2.49, 2.49]$, and the following simple approach is adopted [40]:

1. Update $b(n)$ using (45).
2. Restrict $b(n)$:
$$b(n) = \begin{cases} -2.49, & b(n) < -2.49 \\ +2.49, & b(n) > 2.49 \end{cases} \tag{46}$$
3. Compute $\gamma(n)$ using (37).

From Figure 3, it is clear that by restricting $b(n)$ within the range $[-2.49, 2.49]$, it is ensured that $\gamma(n)$ is restricted in the range $0 < \gamma(n) < 1$ (see vertical dashed lines shown in Figure 3). Thus, the proposed CCSS-MFxRRLS algorithm replaces (24) with (30) and (28) with (34), (36), (37), (45) and (46). A pseudo-code style summary of algorithms presented in this paper is given in Table 1.

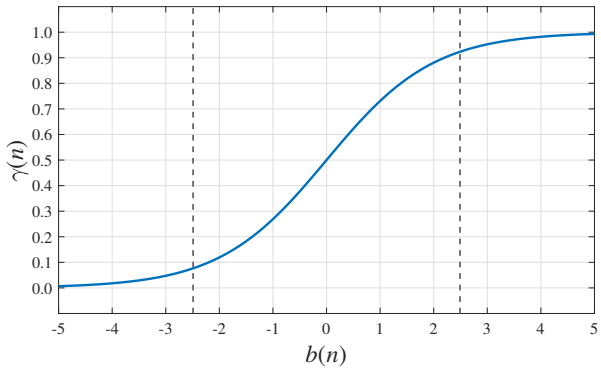

Figure 3. Variation of sigmoidal activation function $\gamma(n)$ vs. adaptive parameter $b(n)$ (and selected threshold values) in the proposed algorithm.

Table 1. The implementation summary of the proposed algorithms presented in this paper.

Initialization and Parameter Selection L, M, $\boldsymbol{w}(0)=\mathbf{0}$, $\boldsymbol{\Phi}_{\mathrm{m}}(0)=\delta^{-1}\boldsymbol{I}$ ($\delta=0.04$), $\gamma(0)=1$, $b(0)=0$, $\mu_b=0.1$, $\zeta=0.99$, $\lambda\approx 1.0$ μ_1, μ_2, β	
while $\{x(n),e(n)\}$ available **do**	
1. $\boldsymbol{x}(n)=[x(n),x(n-1),\cdots,x(n-L+1)]^T$;	% Update signal vector
2. $y(n)=\boldsymbol{x}^T(n)\boldsymbol{w}(n)$;	% ANC filter output
3. $x_f(n)=\hat{s}(n)*x(n)$;	% Filtered-x signal
4. $\hat{d}(n)=e(n)+\hat{s}(n)*y(n)$;	% Estimate of disturbance signal
5. $\boldsymbol{x}_f(n)=[x_f(n),x_f(n-1),\cdots,x_f(n-L+1)]^T$;	% Update filtered-x signal vector
6. $g(n)=\hat{d}(n)-\boldsymbol{x}_f^T(n)\boldsymbol{w}(n-1)$;	% Error signal for adaptation
7. $m(n)=\left\{\dfrac{1}{(1+\beta g^2(n))^2}\right\}$;	% Parameter in objective function
8. $\boldsymbol{u}_{\mathrm{m}}(n)=\boldsymbol{\Phi}_{\mathrm{m}}(n-1)\boldsymbol{x}_f(n)$;	% vector for Kalman gain computation
9. $\boldsymbol{k}_{\mathrm{m}}(n)=\dfrac{\boldsymbol{u}_{\mathrm{m}}(n)}{\frac{\lambda}{m(n)}+\boldsymbol{x}_f^T(n)\boldsymbol{u}_{\mathrm{m}}(n)}$;	% Kalman gain vector
10. $\boldsymbol{\Phi}_{\mathrm{m}}(n)=\zeta\boldsymbol{\Phi}_{\mathrm{m}}(n-1)+(1-\zeta)\left[\lambda^{-1}\boldsymbol{\Phi}_{\mathrm{m}}(n-1)-\lambda^{-1}\boldsymbol{k}_{\mathrm{m}}(n)\boldsymbol{u}_{\mathrm{m}}^T(n)\right]$;	% Inverse correlation matrix
11. $\boldsymbol{w}(n)=\boldsymbol{w}(n-1)+\mu g(n)\boldsymbol{k}_m(n)$;	% FSS-MFxRRLS update equation
12. $\mu(n)=\gamma(n)\mu_1+(1-\gamma(n))\mu_2$;	% Time-varying step-size
13. $\boldsymbol{w}(n)=\boldsymbol{w}(n-1)+\mu(n)g(n)\boldsymbol{k}_{\mathrm{m}}(n)$;	% CCSS-MFxRRLS update equation
14. $b(n)=b(n-1)+\mu_b\mathrm{sgn}(g(n-1))g(n-1)[\gamma(n-1)(1-\gamma(n-1))]\boldsymbol{x}_f^T(n)\boldsymbol{k}_{\mathrm{m}}(n-1)$;	% Adaptive parameter for $\gamma(n)$
15. $b(n)=\begin{cases}-2.49, & b(n)<-2.49\\ +2.49, & b(n)>2.49\end{cases}$;	% Threshold adaptive parameter $b(n)$
16. $\gamma(n)=\dfrac{1}{1+\exp\{-b(n)\}}$;	% Sigmoidal activation function
end while	

1 In 3 and 4, it has been assumed that $\hat{s}(n)\overset{z}{\longleftrightarrow}\hat{S}(z)$ is an FIR filter of length M.
2 Proposed FSS-MFxRRLS algorithm = 1–11.
3 Proposed CCSS-MFxRRLS algorithm = 1–10, 12–16.

4. Results and Discussion

This section provides detailed simulation results, where the objective is to demonstrate the performance of the proposed RLS-based ANC algorithms in comparison with the related existing algorithms: Sun's algorithm [24], Th-FxLMS [25], INSS-FxLMS [26], and the previous CCSS-NMFxLMS algorithm [39,40]. Additionally, a slightly modified variant of the FxLMK algorithm [37], as presented in [40] is also included in the performance comparison.

4.1. Simulation Conditions

The experimental data provided with [3] are used to obtain FIR models for the acoustic paths, and their characteristics are shown in Figure 4. It is worth mentioning that these acoustic paths were measured for a practical ANC setup and have been used in many previous works, including the seminal paper on impulsive ANC systems [24]. All adaptive and fixed filters in various methods are also considered to be of the FIR type.

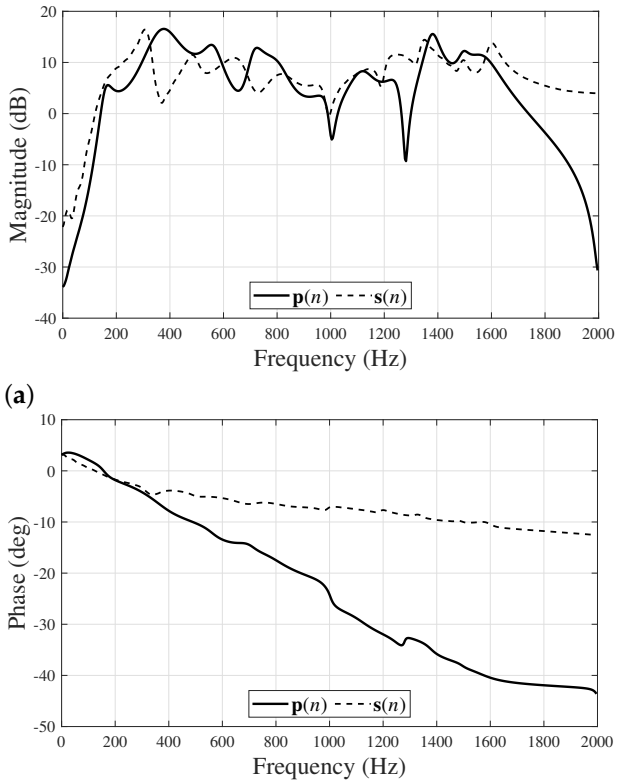

Figure 4. Frequency response characteristics of FIR models for acoustic paths used in computer simulations: (**a**) Magnitude response characteristics, and (**b**) phase response characteristics.

The impulsive noise source is assumed to be modeled as a standard symmetric α-stable (SαS) distribution. The standard SαS distribution has no unique probability density distribution (PDF) and can be better described by using the characteristic function of the form [31]

$$\varphi(t) = e^{-|t|^{\alpha}}, \tag{47}$$

where α is called the characteristics exponent which signifies the 'impulsiveness' of the standard SαS distribution and dictates the corresponding PDF. In fact, α may take any value in the range $0 < \alpha \leq 2$, where $\alpha = 2$ corresponds to Gaussian PDF with zero impulsiveness, and increased impulsiveness (PDF with outliers) is expected if the value of α is small. Figure 5 presents one realization of the input signal $x(n)$ obtained from a standard SαS process with $\alpha = 1.1$, $\alpha = 1.4$, and $\alpha = 1.6$, which are termed as severely impulsive, strongly impulsive, and mildly impulsive for the results presented and explained in the following sections. The variances of typical signals shown in Figure 5 are found to be 2.6313×10^3, 0.2991×10^3, and 0.0168×10^3, for $\alpha = 1.1$, $\alpha = 1.4$, and $\alpha = 1.6$, respectively. It is observed that such reference signals are indeed of impulsive nature, and impulsiveness decreases as the value of α increases.

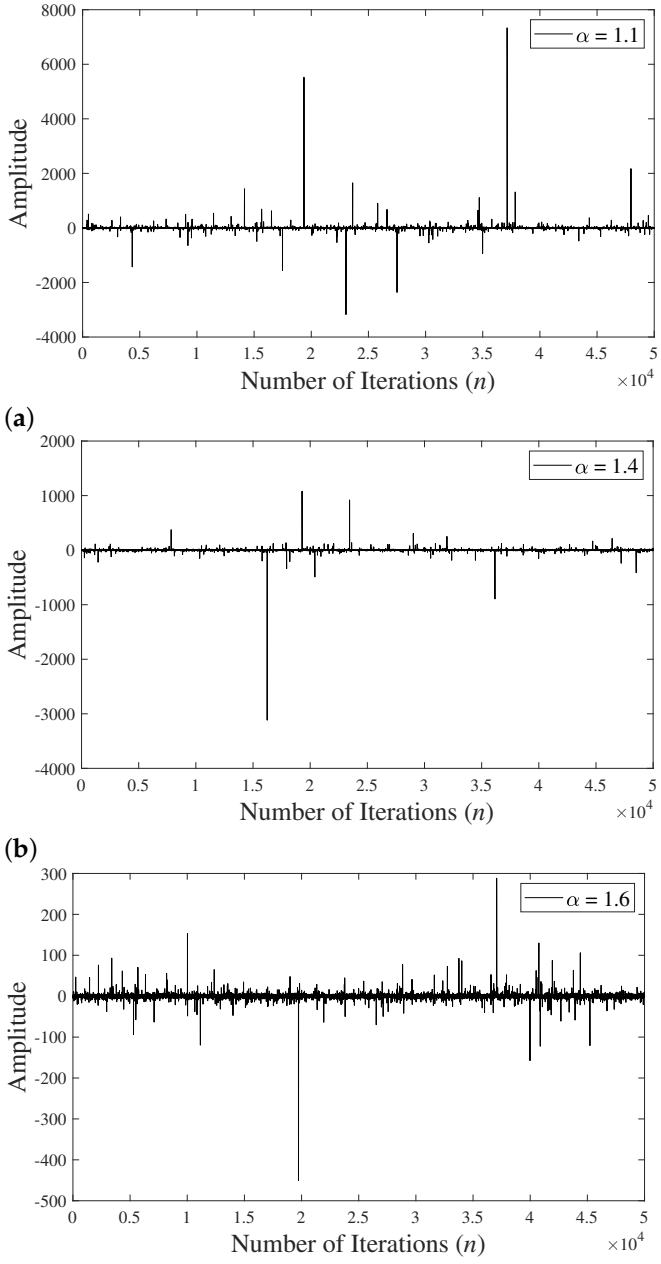

Figure 5. Typical alpha stable signals: (**a**) severely impulsive with $\alpha = 1.1$ and variance 2.6313×10^3; (**b**) strongly impulsive with $\alpha = 1.4$ and variance 0.2991×10^3; (**c**) mildly impulsive with $\alpha = 1.6$ and variance 0.0168×10^3.

Various case studies have been carefully designed to mimic practical scenarios involving application of ANC for impulsive sources. The performance metric to carry out the performance comparison between various algorithms is the mean noise reduction (MNR)

$$\text{MNR}(n) = \mathbb{E}\left\{\frac{\sigma_e(n)}{\sigma_d(n)}\right\}, \tag{48}$$

where $\sigma_e(n)$ and $\sigma_d(n)$ denote an estimate of absolute values of $e(n)$ and $d(n)$, respectively. Such estimates can be obtained via the low pass estimator as in (30). The results presented below have been ensemble averaged for 100 realizations. Considering the nature of a noise source, theoretical analysis to find values for various parameters is very challenging and laborious, if not impossible. Therefore, extensive simulations have been performed to tune

the parameters for various algorithms, as each algorithm adopts a different strategy and control mechanism for impulsive ANC systems.

4.2. Choosing the Parameter β

Before proceeding to the presentation and discussion of the main results, it is important to understand the impact of the parameter β. In this regard, the proposed FSS-MFxRRLS algorithm is employed for ANC of a standard SαS process with $\alpha = 1.1$ (see Figure 5a for a typical signal drawn from such a strongly impulsive process). The FSS step size $\mu = 0.1$ has been experimentally selected for fast convergence as well as good performance at the steady state. The value of the parameter β has been empirically adjusted from a very small value close to zero to a large value close to but less than unity, and the corresponding MNR curves are presented (in a butterfly pattern) in Figure 6. For clarity of presentation, the curves for $\beta = 0.01$ (small value close to zero) and $\beta = 0.995$ (a large value close to but less unity) are plotted using "thick-bold" lines. Furthermore, a vertical panel and a wide panel (plotted inside the main figure) are used to highlight the curves during transient and at the steady state, respectively. It is interesting to note that choosing an appropriate value for β poses a trade-off situation: too small a value may improve the convergence speed; however, it would degrade the steady-state performance, and vice versa. Considering such a behavior, the value of the parameter is selected as $\beta = 0.75$ for the rest of the experiments presented in this paper.

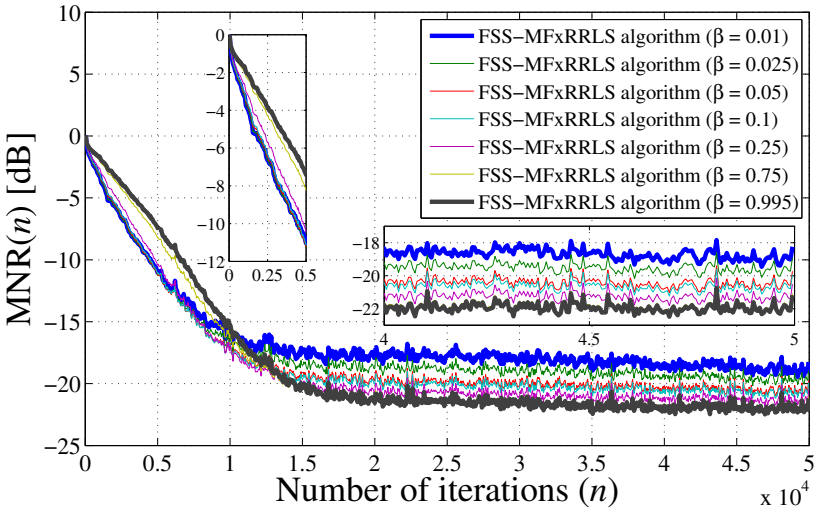

Figure 6. Effect of the parameter β on the noise reduction performance of the proposed FSS-MFxRRLS algorithm with $\mu = 0.1$.

4.3. Scenario 1: Impulsiveness of Noise Sources

The objective of this case study is to understand the performance of various adaptive algorithms for ANC of various impulsive sources with impulsiveness ranging from severe to mild (a few typical examples of such signals are shown in Figure 5). In this regard, Figure 7 shows detailed simulation results for the MNR performance of various algorithms for a severely impulsive standard SαS process with $\alpha = 1.1$. For clarity of presentation, the performance of individual algorithms is presented in separate sub-figures (the curves for the proposed algorithm are included in all sub-figures). Another objective of such a presentation is to have a clear picture about the variation of the step-size parameters (given in the legend of each sub-figure) for the respective algorithms.

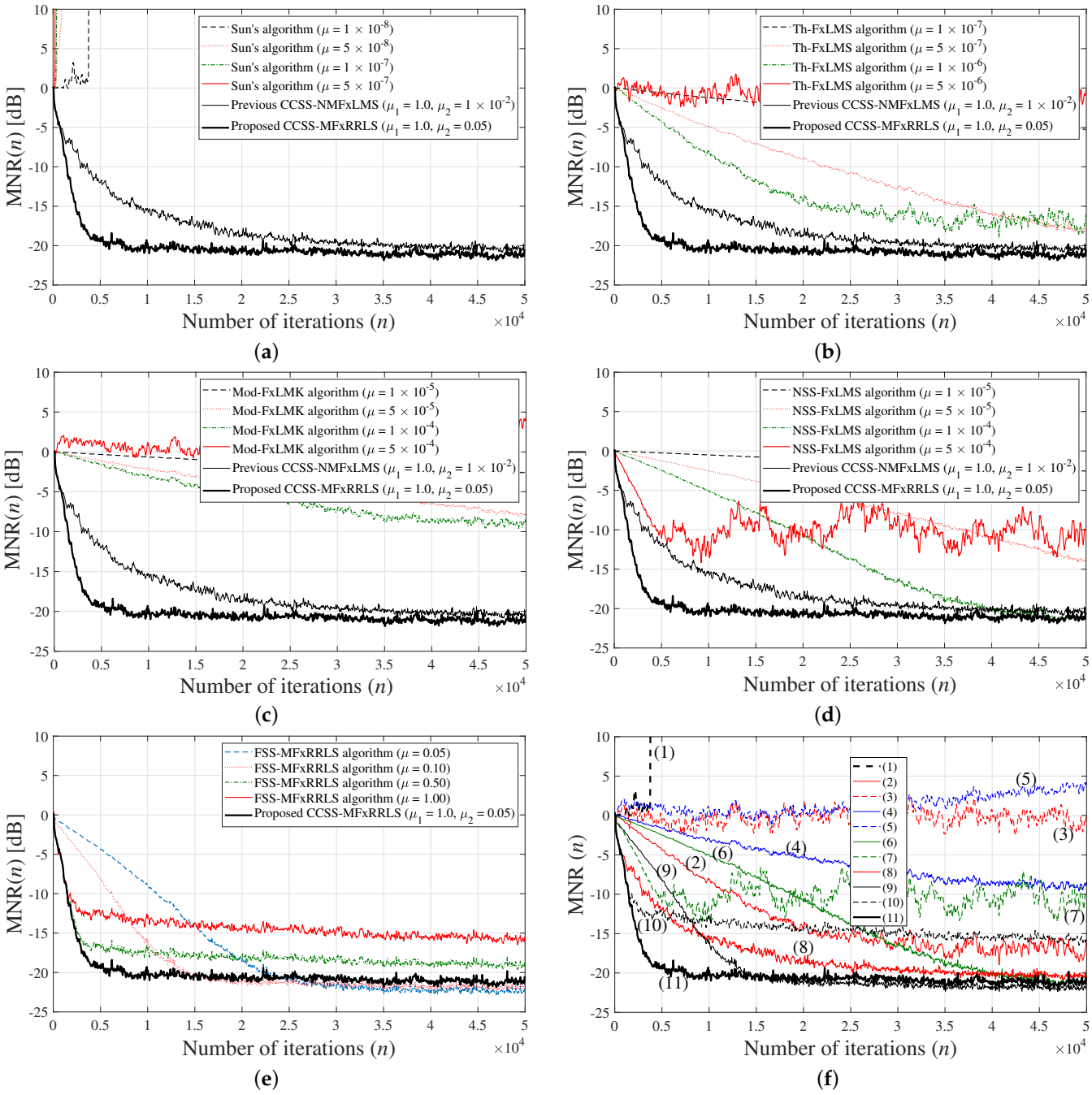

Figure 7. Detailed simulation results for severely impulsive noise source with $\alpha = 1.1$: (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-8}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-6}$); (3) Th-FxLMS algorithm ($\mu = 5 \times 10^{-6}$); (4) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (5) modified FxLMK algorithm ($\mu = 5 \times 10^{-4}$); (6) INSS-FxLMS algorithm ($\mu = 1 \times 10^{-4}$); (7) INSS-FxLMS algorithm ($\mu = 5 \times 10^{-4}$); (8) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (9) FSS-MFxRRLS algorithm ($\mu = 0.1$); (10) FSS-MFxRRLS algorithm ($\mu = 0.5$), (11) Proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

Figure 7a shows that Sun's algorithm is unstable even for a very small-valued step size. The reason lies in the fact that the noise source is of a severely impulsive nature, and the simple thresholding procedure in Sun's algorithm is unfortunately not able to handle it. Figure 7b–e show MNR results for FSS-based Th-FxLMS, modified FxLMK, INSS-FxLMS, and the proposed FSS-MFxRRLS algorithm, respectively. It is evident that an FSS indeed results in a trade-off situation as discussed earlier: it is almost impossible to realize both fast convergence speed and good steady-state performance. It is in fact noticed that the steady-state performance of these algorithms severely deteriorates for a large-valued step size being selected for a fast convergence speed. Finally, Figure 7f shows performance comparisons between various algorithms where two curves have been selected for each FSS-based algorithm, one for a small step size and one for a large step size. It is observed that the previous CCSS-NMFxLMS and the proposed CCSS-MFxRRLS algorithms achieve fast convergence speed and good steady-state performance, with the proposed CCSS-MFxRRLS algorithm outperforming the rest of the algorithms, the reason being the time-varying step size $\mu(n)$ as explained below.

The experiments for the proposed FSS-MFxRRLS algorithm help in deciding the step-size parameters μ_1 and μ_2 in the proposed CCSS-MFxRRLS algorithm. Note that FSS-MFxRRLS gives the fastest convergence speed for $\mu = 1.0$ and good steady-state performance for $\mu = 0.05$; the CCSS $\mu(n)$ in the proposed CCSS-MFxRRLS algorithm is computed using $\mu_1 = 1.0$ and $\mu_2 = 0.05$. For this choice of step-size parameters and the simulations results presented in Figure 7, the corresponding evolution of various parameters in computing the CCSS is shown in Figure 8. Here, Figure 8a–c plot variations of the adaptive parameters $b(n)$ (operations (14) and (15) in Table 1), the mixing parameter $\gamma(n)$ (operation (16) in Table 1), and the time-varying step size $\mu(n)$ (operation (12) in Table 1), respectively. As described earlier and as shown in Figure 8, the adaptive parameter $b(n)$ is close to $+2.49$ at the start-up which ensures $\gamma(n)$ is close to unity, and as the ANC system converges, the adaptive parameter $b(n)$ is automatically tuned to -2.49 which makes $\gamma(n) \to 0$. This ensures fast convergence speed by selecting $\mu(n)$ close to μ_1 at the start-up, and good steady-state performance by adjusting $\mu(n) \to \mu_2$.

The simulation results for the standard SαS process with $\alpha = 1.4$ (less impulsive as compared with the previous experiment for $\alpha = 1.1$) are presented in Figure 9. The noise source is strongly impulsive yet, as indicated by the signal shown in Figure 5b which is a typical signal drawn from such process. As in the previous experiment, detailed simulations have been performed to tune the step-size parameters. A similar performance comparison is observed as in the previous experiments: (1) Sun's algorithm becomes unstable even for a very small step size, (2) FSS-based (Th-FxLMS, modified FxLMK, INSS-FxLMS, and FSS-MFxRRLS) algorithms show slow convergence for a small valued step size, and their performance severely degrades for a large-valued step size, (3) the step size in the FSS-MFxRRLS algorithm can be tuned to give a better performance then the existing algorithms; however, the issue of trade-off situation for performance in transient-state and steady-state remains as it is, (4) both the previous CCSS-NMFxLMS and the proposed CCSS-MFxRRLS algorithms address the trade-off - thanks to being equipped with the CCSS approach, and (5) the proposed CCSS-MFxRRLS algorithm outperforms the rest of the algorithms considered in this simulation study.

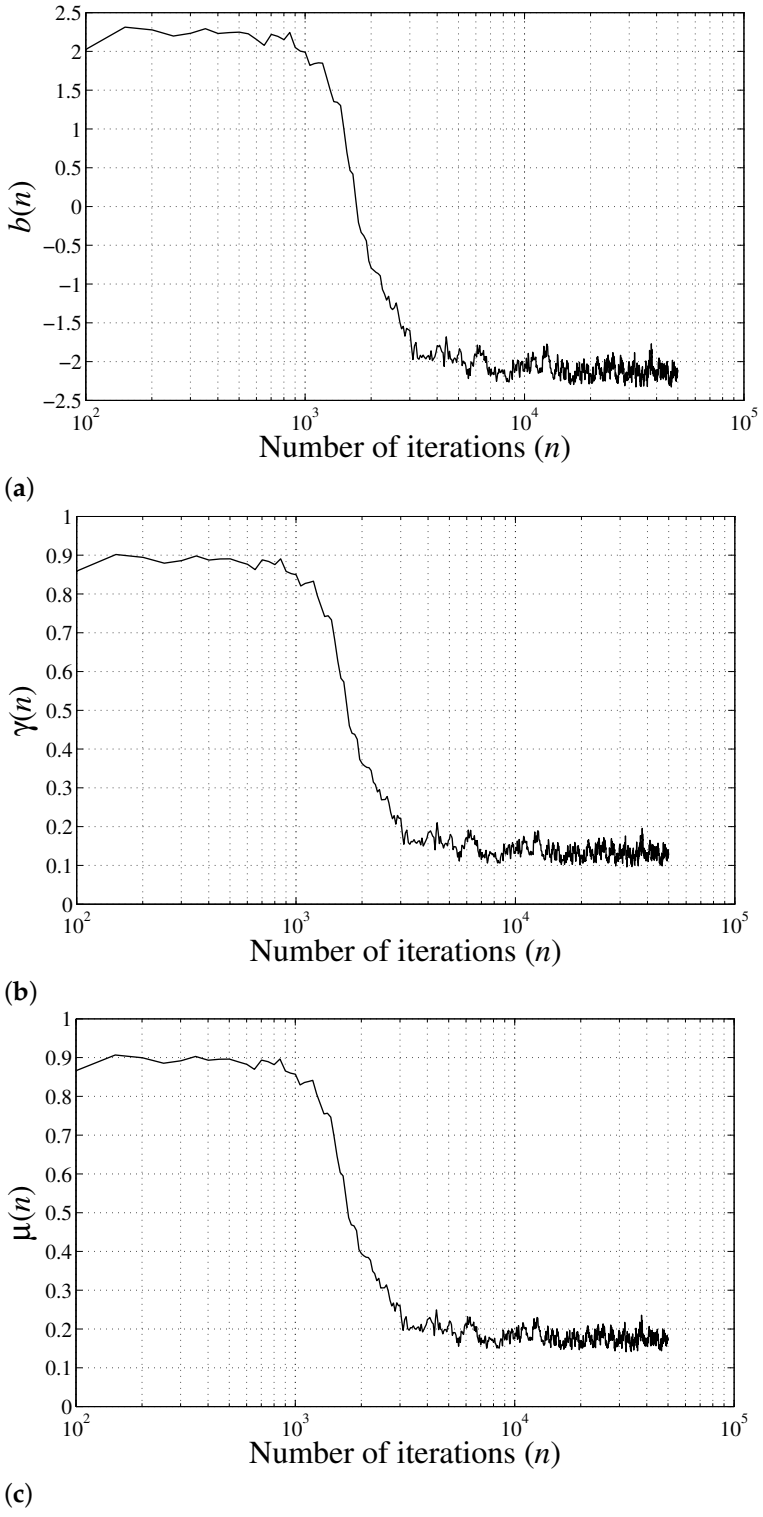

Figure 8. Evolution various parameters in the proposed CCSS-MFxRRLS for severely impulsive noise source with $\alpha = 1.1$: (**a**) the adaptive parameter $b(n)$; (**b**) the mixing parameter $\gamma(n)$; (**c**) the time-varying CCSS $\mu(n)$.

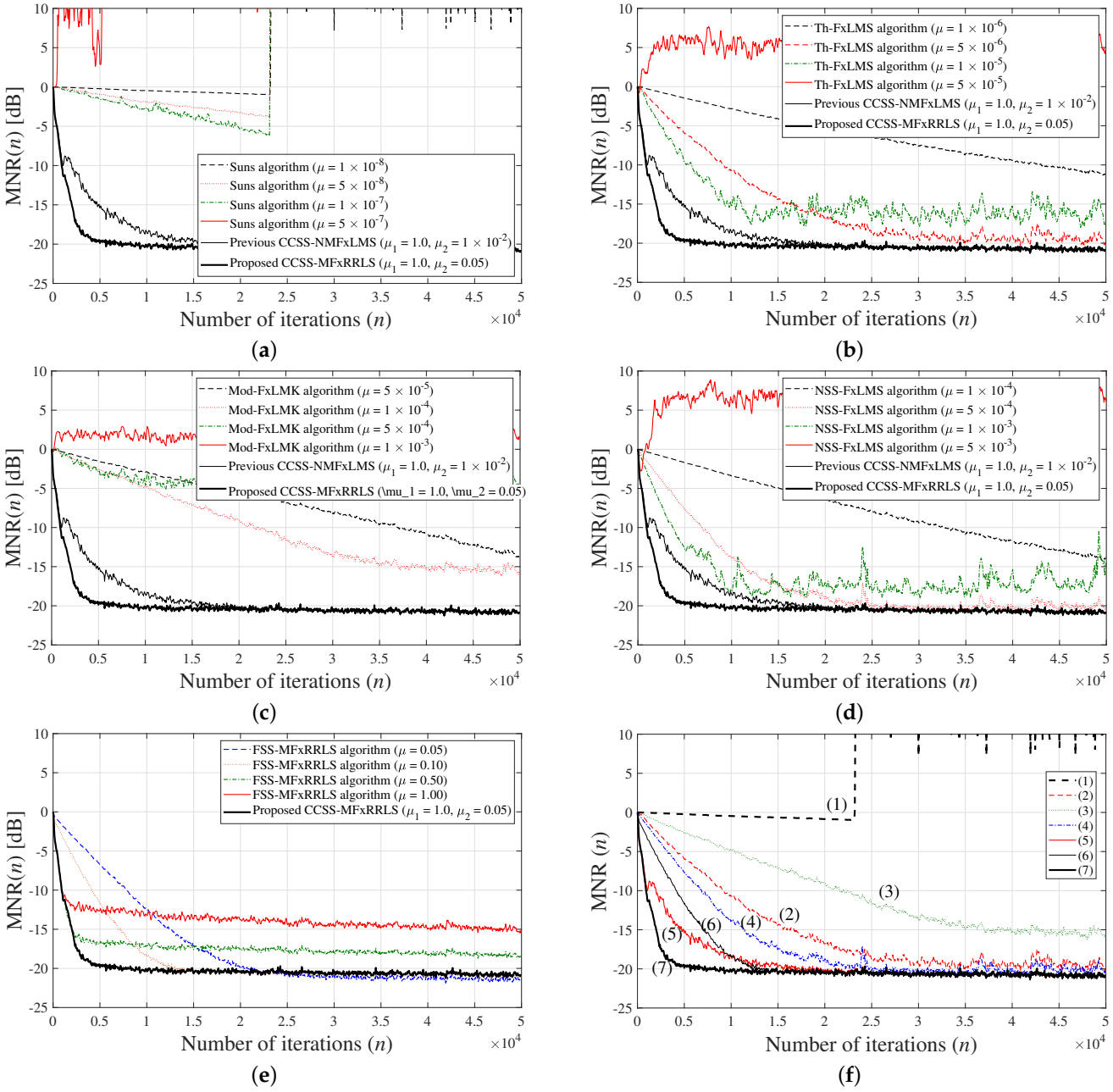

Figure 9. Detailed simulation results for strongly impulsive noise source with $\alpha = 1.4$: (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-8}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-6}$); (3) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (4) INSS-FxLMS algorithm ($\mu = 5 \times 10^{-4}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

Figure 10 presents the simulation results for various impulsive ANC algorithms for the standard SαS process with $\alpha = 1.6$ (mildly impulsive case). It is noticed that all FSS-based algorithms perform far better than the benchmark Sun's algorithm—in fact, Sun's algorithm is convergent only for a very small step size and hence exhibits extremely slow convergence speed. Furthermore, the proposed CCSS-MFxRRLS algorithm does not require any tuning for the parameter values, yet gives the best performance.

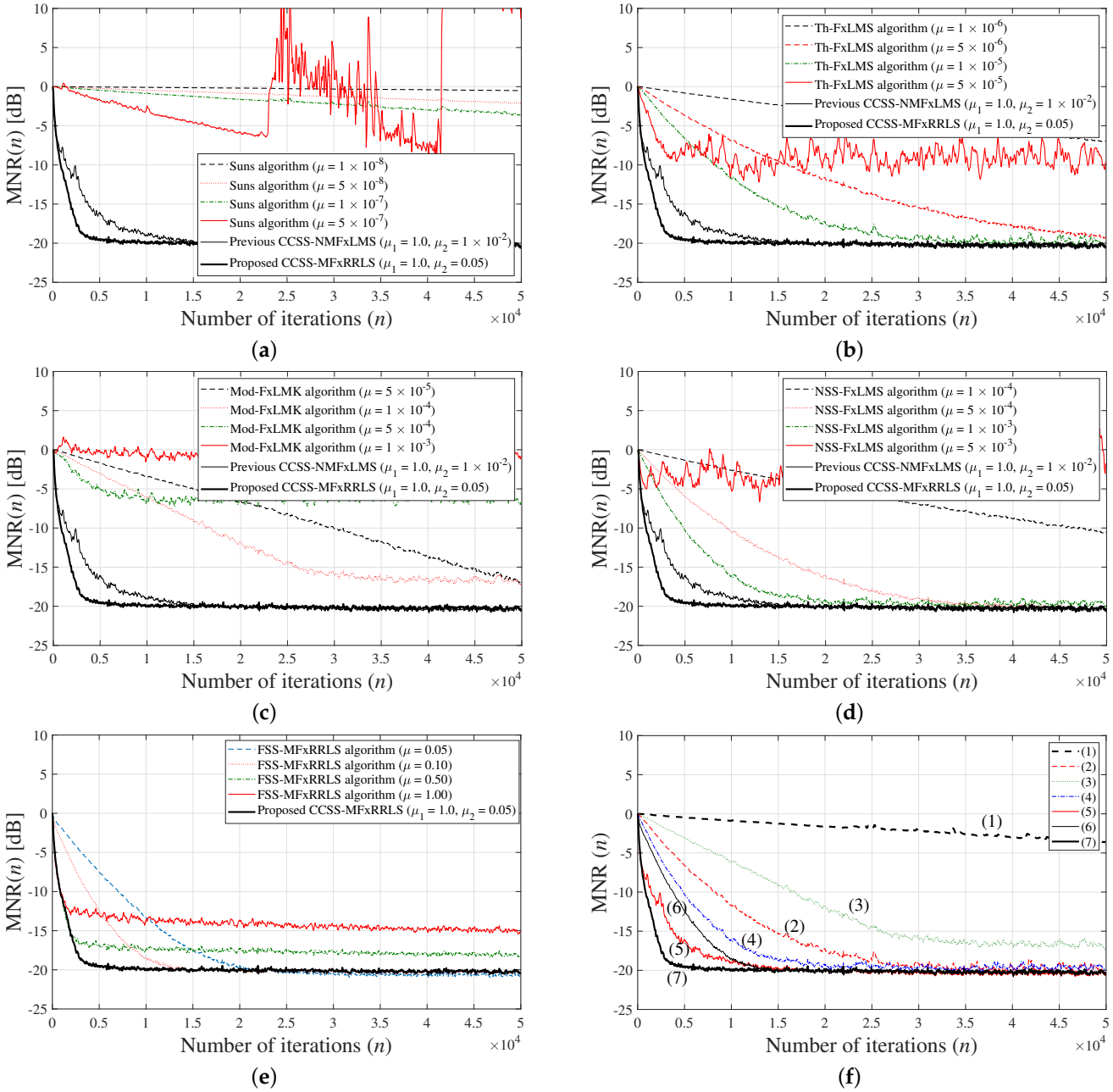

Figure 10. Detailed simulation results for a mildly impulsive noise source with $\alpha = 1.6$. (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-7}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-5}$); (3) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (4) INSS-FxLMS algorithm ($\mu = 1 \times 10^{-3}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

4.4. Scenario 2: Sources with Mixed Stable Distribution

Consider a mixed stable distribution $F(x)$ as

$$F(x) = \tau F_{\alpha_1}(x) + (1-\tau)F_{\alpha_2}(x), \tag{49}$$

where $F_{\alpha_1}(x)$ and $F_{\alpha_2}(x)$ denote stable distributions with characteristic exponents α_1 and α_2, respectively, and τ is the mixing parameter selected between $0 \leq \tau \leq 1$.

Three situations have been considered. In the first experiment, the values for characteristic exponents have been selected as $\alpha_1 = 2.0$ and $\alpha_2 = 1.1$. For $\alpha \equiv \alpha_1 = 2.0$, the corresponding standard SαS distribution is Gaussian, and for $\alpha \equiv \alpha_1 = 1.1$ the corresponding distribution is severely impulsive. The mixing parameters has been selected as $\tau = 0.5$, which indicated that 50% of (randomly selected) samples have been generated from the standard SαS distribution with $\alpha \equiv \alpha_1 = 2.0$ (Gaussian), and the rest of 50% (randomly selected) samples have been drawn from the standard SαS distribution with $\alpha \equiv \alpha_1 = 1.1$ (severely impulsive). Figure 11a shows one realizations of a typical reference signal obtained from such a mixed distribution (49). It is noticed that the generated signal is indeed severely impulsive with variance being estimated to be 5.4953×10^3. In the second experiment, the parameters in (49) have been selected as $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.1$. This choice of parameter values shows that the 75% of (randomly selected) samples have been generated from standard SαS distribution with $\alpha \equiv \alpha_1 = 2.0$ (Gaussian), and merely 25% (randomly selected) samples are from standard SαS distribution with $\alpha \equiv \alpha_1 = 1.1$ (severely impulsive). Since majority (75%) of samples have been (randomly) drawn from the Gaussian distribution, this situation would correspond to less impulsive nature as compared with the previous case for $\tau = 0.5$. A typical signal drawn from such a mixed distribution is shown in Figure 11b with the corresponding variance being estimated to be 0.5527×10^3. In the third experiment, the parameters in (49) have been selected $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.6$. In this case, the samples from non-Gaussian stable distribution are from a less impulsive source compared with the previous examples. A typical signal drawn from such a mixed distribution is shown in Figure 11c, which has a variance of 0.0042×10^3 and looks more or less a Gaussian signal as impulses are less frequent as compared with the signals shown in Figure 11a,b. The simulation parameters have been adjusted experimentally as in the previous experiments.

The simulation results for noise sources having a mixed characteristics stable distribution (49) with $\tau = 0.5$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.1$ are presented in Figure 12. It is interesting to compare these results with those presented in Figure 7 which considered 100% samples being drawn from the standard SαS distribution with $\alpha = 1.1$. In the current experiment, though 50% of the samples have been (randomly) drawn from the Gaussian distribution, the performance of the existing algorithms is still very poor compared with that of the proposed algorithm. The sole reason is the severely impulsive nature of the source signals. The proposed CCSS-MFxRRLS algorithm can handle the strongly impulsive source well and demonstrates the best performance among the algorithms considered in this paper.

The simulation results for noise sources having a mixed characteristics standard SαS distribution (49) with $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.1$ are presented in Figure 13. Though most of the samples (75%) come from Gaussian distribution, it is observed that Sun's algorithm still fails to show any convergence. Furthermore, the rest of the FSS-based algorithms require tuning the step parameters as well as exhibit a trade-off situation for the fast convergence and good steady-state performance. It is noticed that the proposed CCSS-MFxRRLS algorithm clearly demonstrates the best performance among the algorithms considered in this paper.

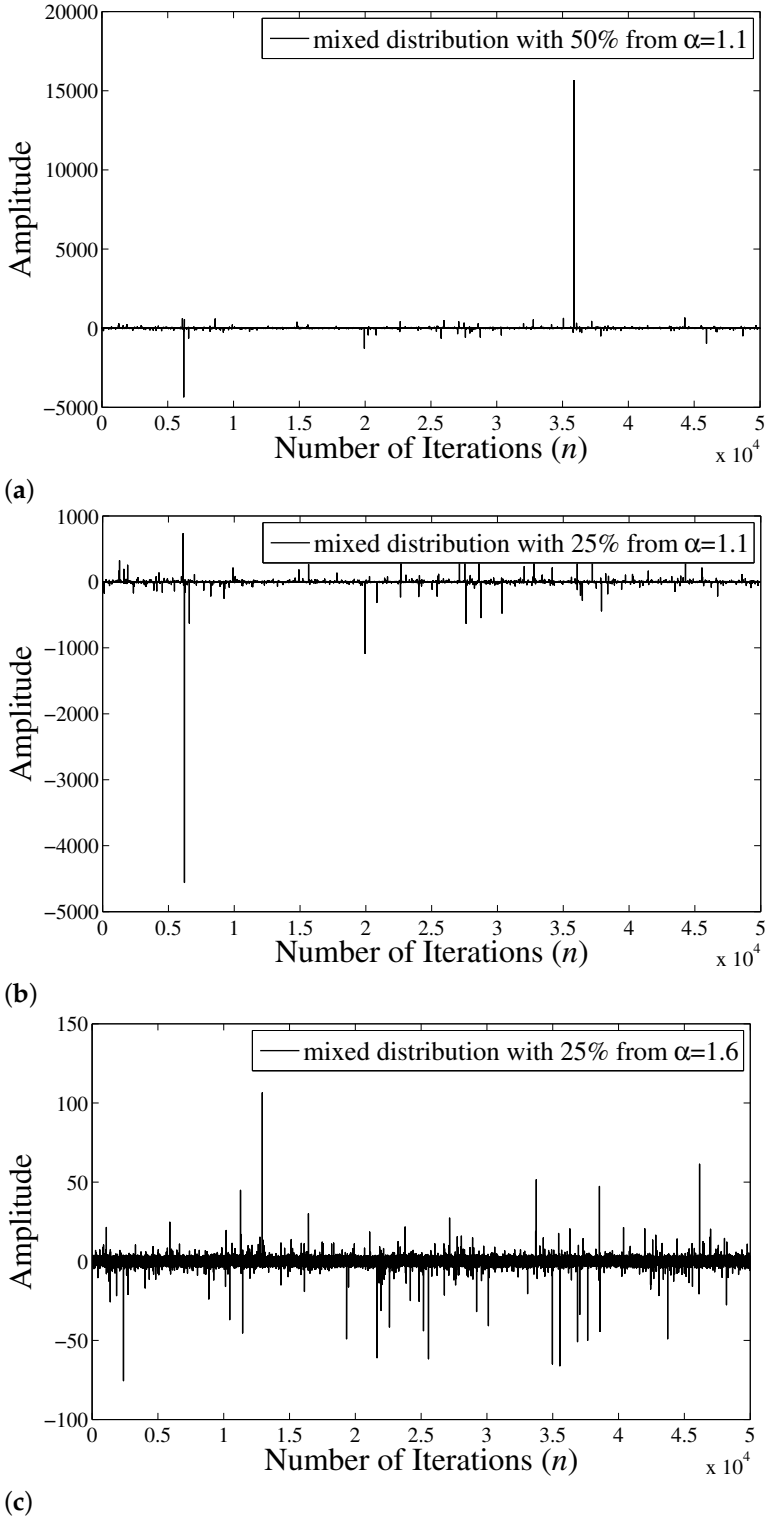

Figure 11. Typical alpha stable signals generated from mixed distribution (49): (**a**) 50% samples from $\alpha = 2.0$ (Gaussian) and 50% samples from $\alpha-$stable distribution with $\alpha = 1.1$ (variance of shown signal is found to be 5.4953×10^3); (**b**) 75% samples from $\alpha = 2.0$ (Gaussian) and 25% samples from $\alpha-$stable distribution with $\alpha = 1.1$ (variance of shown signal is found to be 0.5527×10^3); (**c**) 75% samples from $\alpha = 2.0$ (Gaussian) and 25% samples from $\alpha-$stable distribution with $\alpha = 1.6$ (variance of shown signal is found to be 0.0042×10^3).

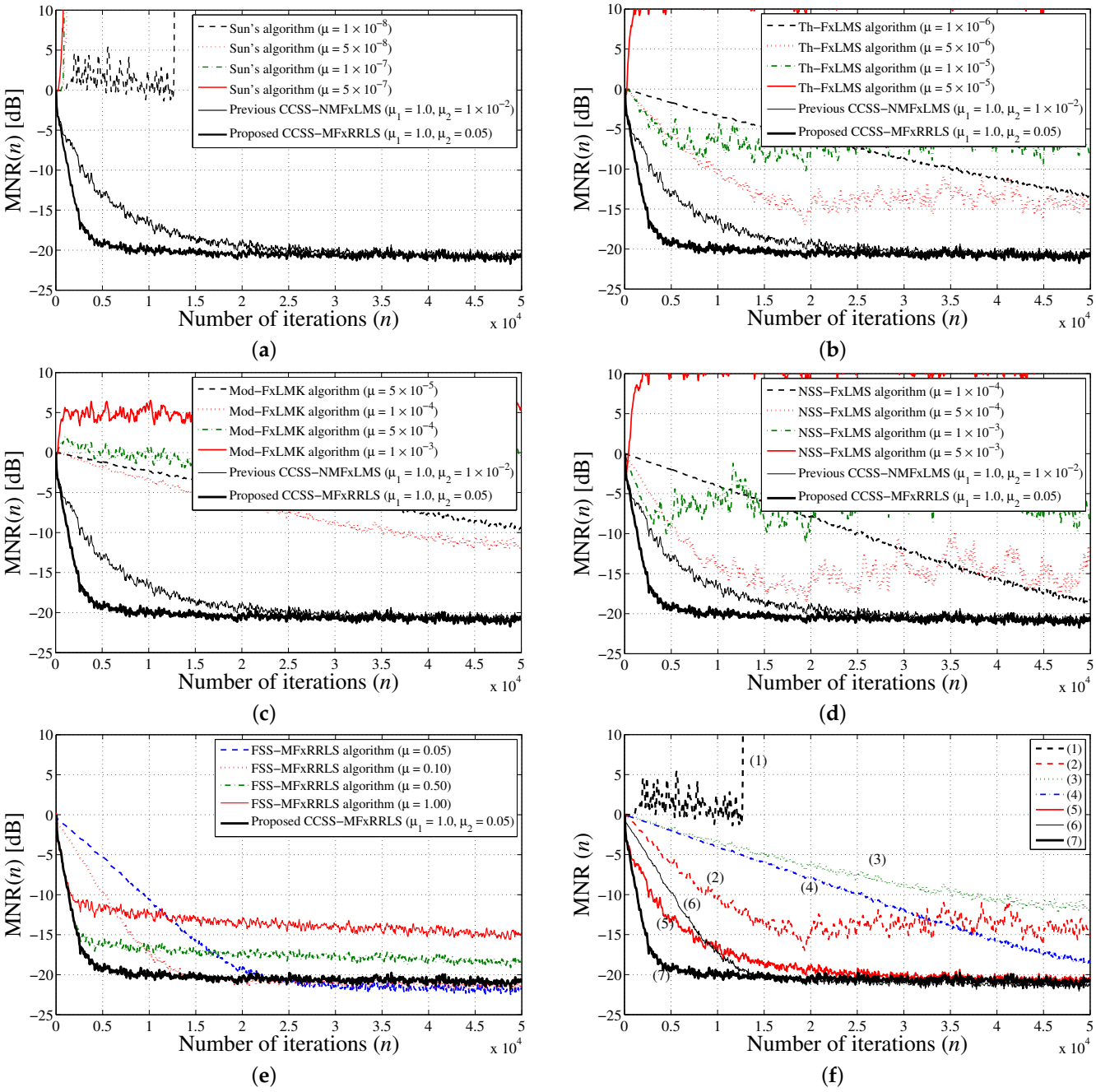

Figure 12. Detailed simulation results for reference signals from mixed distribution (49) with $\tau = 0.5$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.1$: (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-8}$); (2) Th-FxLMS algorithm ($\mu = 5 \times 10^{-6}$); (3) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (4) INSS-FxLMS algorithm ($\mu = 1 \times 10^{-4}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

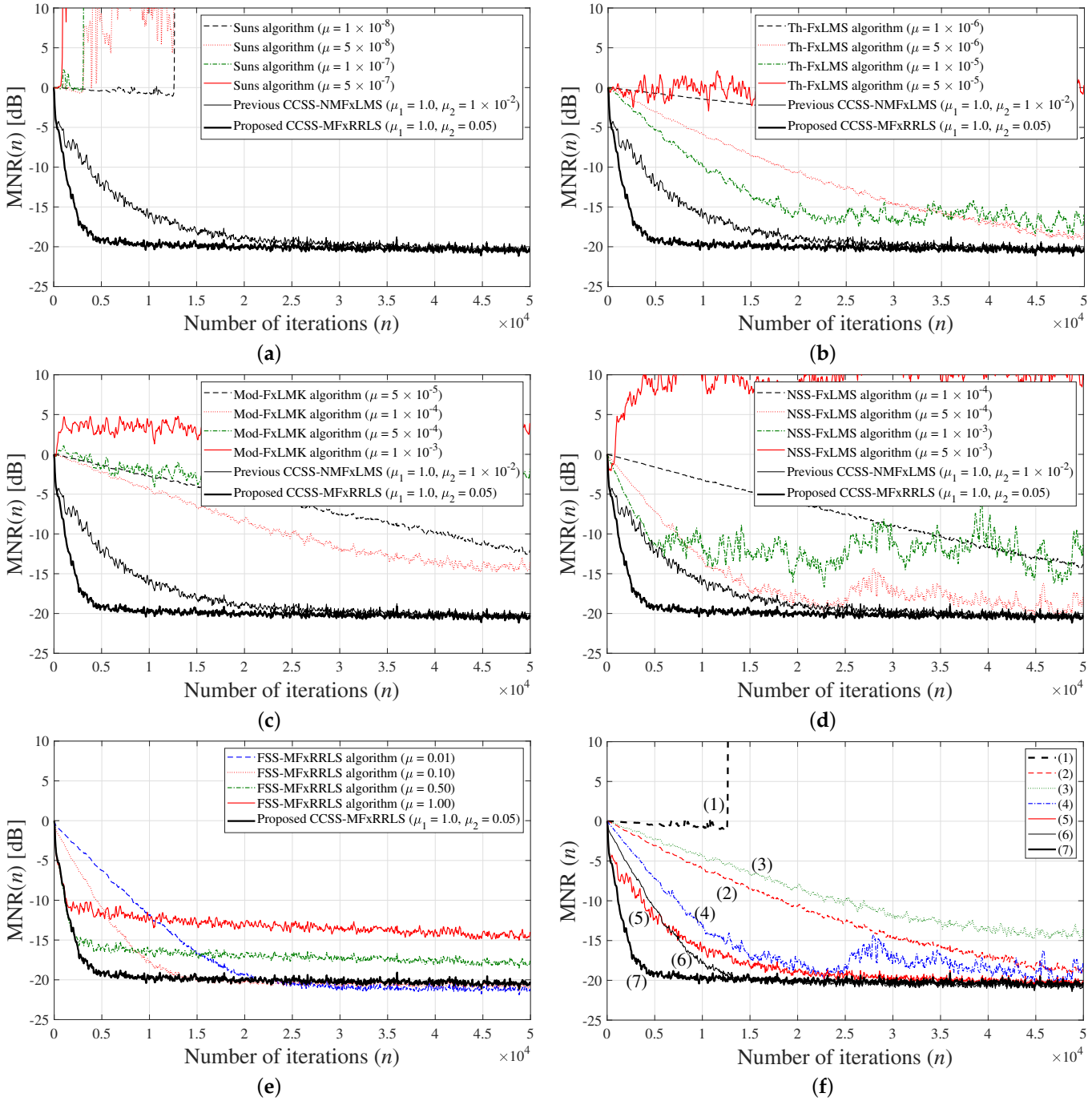

Figure 13. Detailed simulation results for reference signals from mixed distribution (49) with $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.1$: (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-8}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-5}$); (3) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (4) INSS-FxLMS algorithm ($\mu = 5 \times 10^{-4}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

The simulation results for noise sources from a mixed distribution (49) with $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.6$ are presented in Figure 14. As described earlier, the reference signal is mostly (75%) Gaussian, with (25%) impulsive samples coming from a less impulsive distribution as compared with the previous experiments. Therefore, all algorithms considered

hereby show improved performance as compared with the previous experiments. Nevertheless, the FSS-based algorithms have a trade-off situation for the selection of appropriate step-size parameters. The proposed CCSS-MFxRRLS algorithm does not suffer from such a trade-off issue and achieves the best performance.

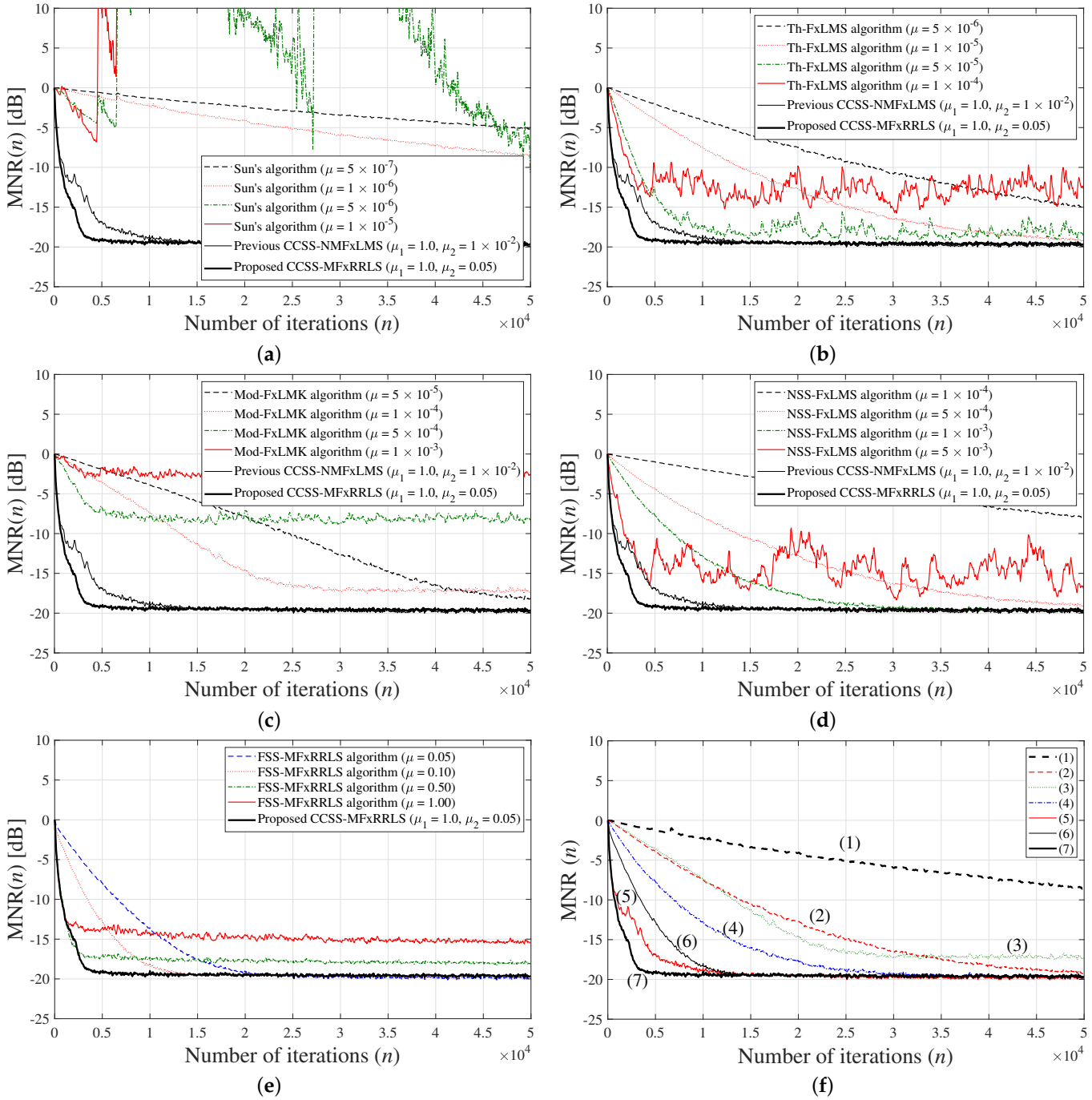

Figure 14. Detailed simulation results for reference signals from mixed distribution (49) with $\tau = 0.75$, $\alpha_1 = 2.0$, and $\alpha_2 = 1.6$; (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-6}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-5}$); (3) modified FxLMK algorithm ($\mu = 1 \times 10^{-4}$); (4) INSS-FxLMS algorithm ($\mu = 1 \times 10^{-3}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

4.5. Scenario 3: Impulsive Sources with Time-Varying Characteristics

This scenario considers noise sources having time-varying characteristics from the view point of the impulsiveness. At the start, the noise source is assumed to be modeled as a standard SαS process of strongly impulsive nature having $\alpha = 1.4$, exhibits severely impulsive characteristics with $\alpha = 1.1$ during the middle, and finally returns to strongly impulsive behavior with $\alpha = 1.4$ for the latter part of the simulation. A typical signal drawn from such a standard SαS process is shown in Figure 15, where the vertical axis is restricted to clearly show that the impulses are indeed more frequent in the middle part. The corresponding simulation results averaged over 100 realizations are presented in Figure 16. It is observed from the presented results that the existing algorithms work fine provided that the step size is adjusted to a small value, which surely results in an overall slow convergence speed. A large step size may improve the initial convergence speed; however, may result in a poor performance when impulsiveness increases during the middle portion of the simulation. The previous CCSS-NMFxLMS, the proposed FSS-MFxRRLS and proposed CCSS-MFxRRLS algorithms do not require any further tuning in the parameter values and keep the good performance for the whole duration of simulation. Furthermore, the proposed proposed CCSS-MFxRRLS algorithm demonstrates the best performance as compared with the rest of the algorithms.

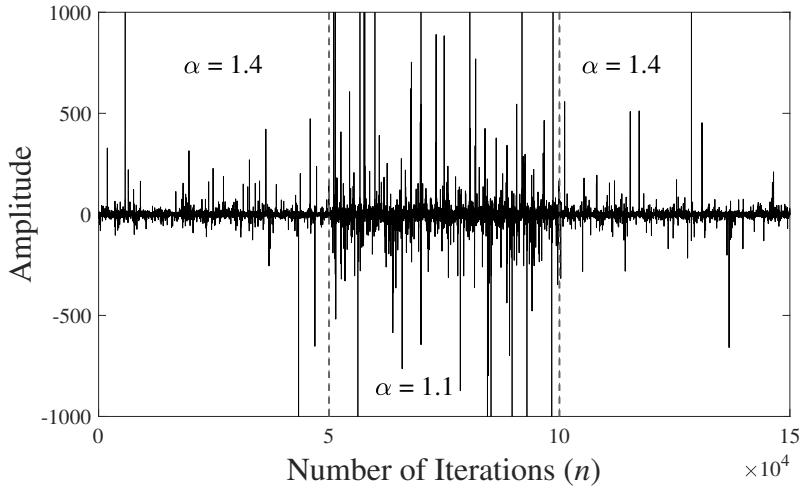

Figure 15. A typical alpha stable signal generated from changing characteristics.

4.6. A Few Remarks on Computational Complexity Comparison

Last but not least, it is worth giving a few remarks on the computational complexity of various algorithms considered in this paper. With reference to [40], the computation requirements for the existing algorithms can be summarized as follows:

- Sun's algorithm [24] requires $2L + M + 1$ multiplications per iteration, $2L + M - 2$ additions/subtractions per iteration, and 1 comparison operation.
- The Th-FxLMS algorithm [25] requires $2L + M + 1$ multiplications per iteration, $2L + M - 2$ additions/subtractions per iteration, and 2 comparison operation.
- The INSS-FxLMS algorithm [26] requires $3L + M + 4$ multiplications per iteration, $2L + M + 1$ additions/subtractions per iteration, and 1 division operation.
- The previous CCSS-NMFxLMS algorithm [39,40] requires $6L + 2M + 7$ multiplications per iteration, $4L + 2M + 4$ additions/subtractions per iteration, 1 comparison operation, and 2 division operations.

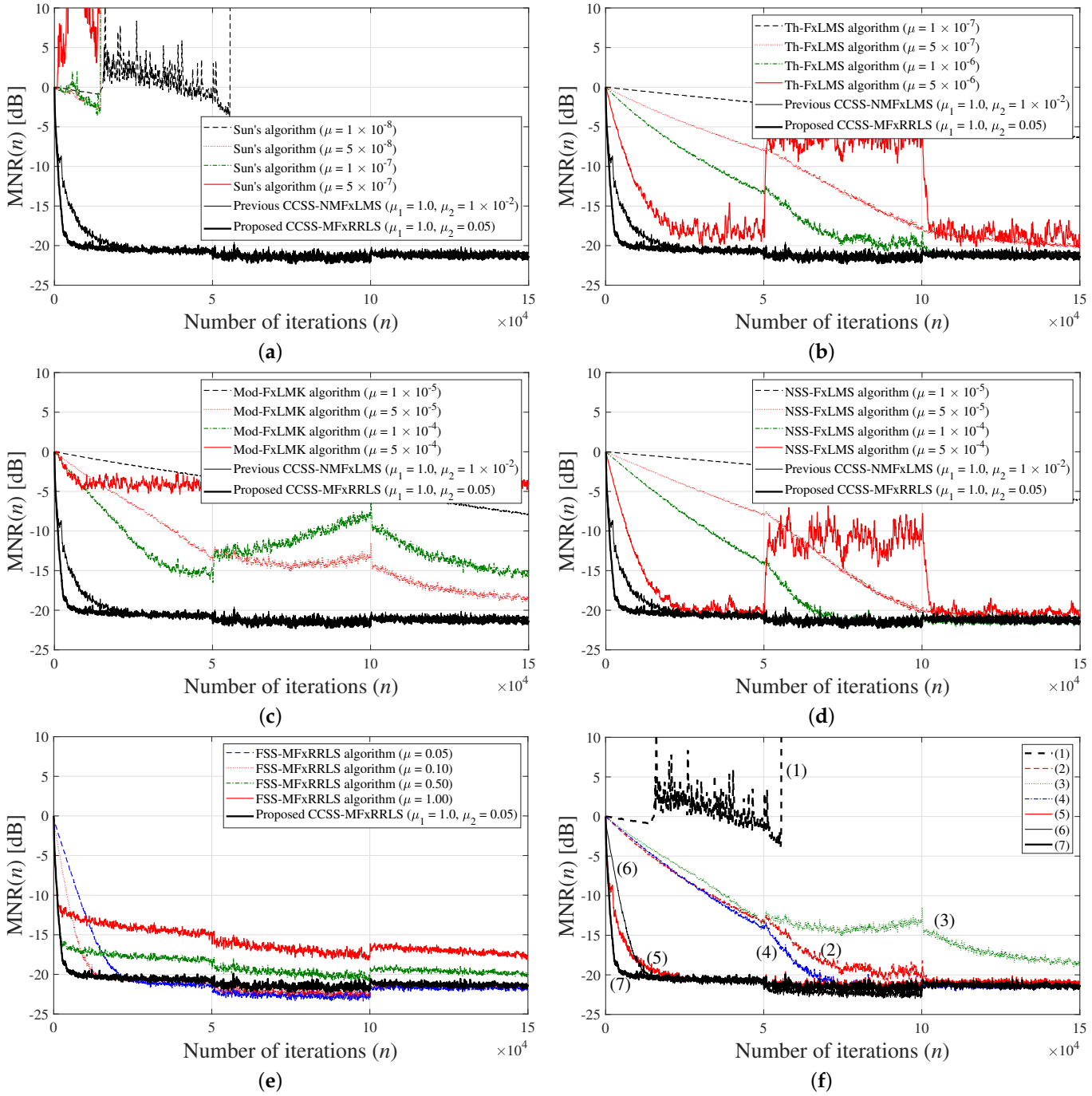

Figure 16. Detailed simulation results for reference signals from non-stationary SαS process with varying α: (**a**) Sun's algorithm [24]; (**b**) Th-FxLMS algorithm [25]; (**c**) modified FxLMK algorithm; (**d**) INSS-FxLMS algorithm [26]; (**e**) effect of step-size parameter in the proposed FxRLS-based impulsive ANC algorithms; (**f**) performance comparison between various algorithms: (1) Sun's algorithm ($\mu = 1 \times 10^{-8}$); (2) Th-FxLMS algorithm ($\mu = 1 \times 10^{-6}$); (3) modified FxLMK algorithm ($\mu = 5 \times 10^{-5}$); (4) INSS-FxLMS algorithm ($\mu = 1 \times 10^{-4}$); (5) previous CCSS-NMFxLMS algorithm ($\mu_1 = 1.0, \mu_2 = 1 \times 10^{-2}$); (6) FSS-MFxRRLS algorithm ($\mu = 0.1$); (7) proposed CCSS-MFxRRLS ($\mu_1 = 1.0, \mu_2 = 0.05$).

Here, L denotes length of the FIR ANC filter $W(z)$, and M is the length of the secondary path modeling filter $\hat{S}(z)$. Considering the execution summary of the proposed algorithms presented in Table 1, it is straightforward to determine that the proposed FSS-MFxRRLS algorithm (operations 1–11 in Table 1) would require $4L^2 + 5L + 2M + 5$

multiplications per iteration, $3L^2 + 3L + 2M - 1$ additions/subtractions per iteration, and 4 division operations. In addition to these computations, the time-varying step size in the proposed CCSS-MFxRRLS algorithm (operations 12, 14-16 in Table 1) would require $L + 5$ multiplications per iteration, $L + 4$ additions/subtractions per iteration, and 1 division, 1 comparison, and 1 $\exp\{\cdot\}$ operation per iteration of execution. It is obvious that the proposed algorithms, being based on an RLS-based adaptation, have an increased computation complexity. However, the performance of the proposed algorithm is far superior compared with the other LMS-algorithms considered in this paper for impulsive ANC systems, as demonstrated by the extensive simulation results presented earlier. Therefore, the increased computational cost may be considered as the price paid for an improved performance.

5. Conclusions

This paper has presented robust RLS-based adaptive algorithms for impulsive ANC systems. Two variants of the proposed algorithm have been described. The first version is based on an FSS and signifies the importance of robustness of the objective function and further (ad hoc) modifications suggested in comparison with the classical FxRLS algorithm. In the second variant, the CCSS strategy has been incorporated to address the trade-off situation offered by the FSS-based one. Since theoretical analysis for non-Gaussian stable processes is extremely difficult, if not impossible, extensive simulations have been carried out. It is observed that the proposed CCSS-MFxRRLS algorithm achieves very fast convergence even for severely impulsive sources. Furthermore, it demonstrates robust performance for the non-stationary noise sources with time-varying characteristics from the viewpoint of impulsiveness.

The proposed algorithms have been derived in the framework of classical RLS adaptive filtering which has a higher computational complexity in comparison with the LMS-based counterparts. The increased computational cost may be considered as the price paid for robust performance. It is important to mention that 'RLS-like' adaptive algorithms can be found in the literature which have reduced computational complexity [54–56]. In future, it will be interesting to consider such RLS-like algorithms to develop robust algorithms for impulsive ANC. In addition, it is very important to carry out theoretical analysis of the developed algorithm. This poses a great challenge for the filtered-x structure as well as impulsive nature of noise source and is left as a task for future work.

Funding: This research was partially supported by the Nazarbayev University Collaborative Research Grants Program under the grant number 021220CRP0222.

Institutional Review Board Statement: The study did not require ethical approval.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Author would like to thank the anonymous reviewers for their many insightful comments, which have greatly helped in improving the quality of presentation of manuscript.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Lueg, P. Process of Silencing Sound Oscillations. U.S. Patent, 2043416, 9 June 1936.
2. Elliot, S.J. *Signal Processing for Active Control*; Academic Press: London, UK, 2001.
3. Kuo, S.M.; Morgan, D.R. *Active Noise Control Systems-Algorithms and DSP Implementations*; Wiley: New York, NY, USA, 1996.
4. Kuo, S.M.; Morgan, D.R. Active Noise Control: A tutorial review. *Proc. IEEE* **1999**, *87*, 943–973. [CrossRef]
5. Liu, L.; Kuo, S.M.; Raghuathan, K.P. An audio integrated motorcycle helmet. *J. Low Freq. Noise, Vib. Act. Control* **2010**, *29*, 161–170. [CrossRef]
6. Lei, C.; Xu, J.; Wang, J.; Zheng, C.; Li, X. Active headrest with robust performance against head movement. *J. Low Freq. Noise, Vib. Act. Control* **2015**, *34*, 233–250. [CrossRef]
7. Latos, M.; Stankiewicz, K. Studies on the effectiveness of noise protection for an enclosed industrial area using global active noise reduction systems. *J. Low Freq. Noise, Vib. Act. Control* **2015**, *34*, 9–20. [CrossRef]

8. Xu, Z.H.; Lee, C.M.; Qiu, Z. A Study of the virtual microphone algorithm for ANC system working in audio interference environment. *J. Low Freq. Noise, Vib. Act. Control* **2014**, *33*, 89–206. [CrossRef]
9. Gan, W.S.; Kuo, S.M. An integrated audio and active noise control headsets. *IEEE Trans. Consum. Electron.* **2002**, *48*, 242–247. [CrossRef]
10. Kuo, S.M.; Mitra, S.; Gan, W.S. Active noise control system for headphone applications. *IEEE Trans. Control Sys. Tech.* **2006**, *14*, 331–335. [CrossRef]
11. Lu, L.; Yin, K.L.; de Lamare, R.C.; Zheng, Z.; Yu, Y.; Yang, X.; Chen, B. A survey on active noise control in the past decade–Part I: Linear systems. *Signal Process.* **2021**, *183*, 108039. [CrossRef]
12. Lu, L.; Yin, K.L.; de Lamare, R.C.; Zheng, Z.; Yu, Y.; Yang, X.; Chen, B. A survey on active noise control in the past decade–II: Nonlinear systems. *Signal Process.* **2021**, *181*, 107929. [CrossRef]
13. Widrow, B.; Stearns, S.D. *Adaptive Signal Processing*; Prentice Hall: Hoboken, NJ, USA, 1985.
14. Douglas, S.C. A family of normalized LMS algorithms. *IEEE Signal Process. Lett.* **1994**, *1*, 49–51. [CrossRef]
15. Park, Y.C.; Sommerfeldt, S.D. A fast adaptive noise control algorithm based on lattice structure. *Appl. Acoust.* **1996**, *47*, 1–25. [CrossRef]
16. Eriksson, L.J.; Allie, M.C.; Greiner, R.A. The selection and application of an IIR adaptive filter for use in active sound attenuation. *IEEE Trans. Acoust. Speech Signal Process.* **1987**, *35*, 433–437. [CrossRef]
17. Crawford, D.H.; Stewart, R.W. Adaptive IIR filtered-v algorithms for active noise control. *J. Acoust. Soc. Am.* **1997**, *101*, 2097–2103. [CrossRef]
18. Bouchard, M.; Quednau, S. Multichannel recursive-least-squares algorithms and fast-transversal-filter algorithms for active noise control and sound reproduction systems. *IEEE Trans. Speech Audio Process.* **2000**, *8*, 606–618. [CrossRef]
19. Kuo, S.M.; Tahernezhadi, M. Frequency-domain periodic active noise control and equalization. *IEEE Trans. Speech Audio Process.* **1997**, *5*, 348–358. [CrossRef]
20. Bergamasco, M.; Rossa, F.D.; Piroddi, L. Active noise control with on-line estimation of non-Gaussian noise characteristics. *J. Sound Vibr.* **2012**, *33*, 27–40. [CrossRef]
21. Liu, L.; Gujjula, S.; Thanigai, P.; Kuo, S.M. Still in womb: Intrauterine acoustic embedded active noise control for infant incubators. *Adv. Acoust. Vibr.* **2008**, *2008*, 495317. [CrossRef]
22. Zhou, Y.L.; Yin, Y.X.; Zhang, Q.Z. An optimal repetitive control algorithm for periodic impulsive noise attenuation in a non-minimum phase ANC system. *Appl. Acoust.* **2013**, *74*, 1175–1181. [CrossRef]
23. Wierzchowski, W. Review of active noise control algorithms for impulsive noise control. *Pomiary Autom. Kontrola* **2014**, *60*, 358–361.
24. Sun, X.; Kuo, S.M.; Meng, G. Adaptive algorithm for active control of impulsive noise. *Jr. Sound Vibr.* **2006**, *291*, 516–522. [CrossRef]
25. Akhtar, M.T.; Mitsuhashi, W. Improving performance of FxLMS algorithm for active noise control of impulsive noise. *Jr. Sound Vibr.* **2009**, *327*, 647–656. [CrossRef]
26. Akhtar, M.T.; Mitsuhashi, W. A modified normalized FxLMS algorithm for active control of impulsive noise. In Proceedings of the EUSIPCO 2010, Aalborg, Denmark, 23–27 August 2010; pp. 1–5.
27. Wu, L.; He, H.; Qiu, X. An active impulsive noise control algorithm with logarithmic transformation. *IEEE Trans. Audio Speech Lang. Process.* **2010**, *19*, 1041–1044.
28. Li, P.; Yu, X. Active noise cancellation algorithms for impulsive noise. *Mech. Syst. Signal Process.* **2013**, *36*, 630–635. [CrossRef] [PubMed]
29. Sun, G.; Li, M.; Lim, T.C. Enhanced filtered-x least mean M-estimate algorithm for active impulsive noise control. *Appl. Acoust.* **2015**, *90*, 31–41. [CrossRef]
30. Sun, G.; Li, M.; Lim, T.C. A family of threshold based robust adaptive algorithms for active impulsive noise control. *Appl. Acoust.* **2015**, *97*, 30-36. [CrossRef]
31. Shao, M.; Nikias, C.L. Signal processing with fractional lower order moments: Stable processes and their applications. *Proc. IEEE* **1993**, *81*, 986–1010. [CrossRef]
32. Leahy, R.; Zhou, Z.; Hsu, Y.C. Adaptive filtering of stable processes for active attenuation of impulsive noise. In Proceedings of the IEEE ICASSP 1995, Detroit, MI, USA, 9–12 May 1995; Volume 5, pp. 2983–2986.
33. Akhtar, M.T.; Mitsuhashi, W. Improving robustness of filtered-x least mean p-power algorithm for active attenuation of standard symmetric-*alpha*-stable impulsive noise. *Appl. Acoust.* **2011**, *72*, 688–694. [CrossRef]
34. Akhtar, M.T. Fractional lower order moment based adaptive algorithms for active noise control of impulsive noise sources. *J. Acoust. Soc. Am.* **2012**, *132*, EL456–EL462. [CrossRef]
35. Akhtar, M.T. A time-varying normalized step size based generalized fractional moment adaptive algorithm and its application to ANC of impulsive sources. *Appl. Acoust.* **2019**, *155*, 240–249. [CrossRef]
36. Song, P.; Zhao, H. Filtered-x generalized mixed norm (FXGMN) algorithm for active noise control. *Mech. Syst. Signal Process.* **2018**, *107*, 93–104. [CrossRef]
37. Lu, L.; Zhao, H. Improved filtered-x least mean kurtosis algorithm for active noise control. *Circuits Syst. Signal. Process.* **2017**, *36*, 1586–1603. [CrossRef]
38. Akhtar, M.T.; 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*, 720–726. [CrossRef]

39. Akhtar, M.T. A convex-combined step-size-based normalized modified filtered-x least mean square algorithm for impulsive active noise control systems. In Proceedings of the EUSIPCO 2018, Roma, Italy, 3–7 September 2018; pp. 2468—2472.
40. Akhtar, M.T. On active impulsive noise control (AINC) systems: Developing a filtered-reference adaptive algorithm using convex-combined normalized step-size approach. *Circuits Syst. Signal Process.* **2020**, *39*, 4354–4377. [CrossRef]
41. Akhtar, M.T. Novel recursive least squares-based filtered-x adaptive algorithm developed for active control of impulsive noise sources. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC 2020), Toronto, ON, Canada, 11–14 October 2020; pp. 2359–2364.
42. Haykin, S. *Adaptive Filter Theory*, 4th ed.; Prentice Hall: Hoboken, NJ, USA, 2002.
43. Huang, F.; Zhang, J.; Zhang, S. Maximum versoria criterion-based robust adaptive filtering algorithm. *IEEE Trans. Circuits Sys.—II Express Briefs* **2017**, *64*, 1252–1256. [CrossRef]
44. Diniz, P.S.R. *Adaptive Filtering: Algorithms and Practical Implementation*; Springer: New York, NY, USA, 2013.
45. Benesty, J.; Rey, H.; Vega, L.R.; Tressens, S. A nonparametric VSS NLMS algorithm. *IEEE Signal Process. Lett.* **2006**, *13*, 581–584. [CrossRef]
46. Vega, L.R.; Rey, H.; Benesty, J.; Tressens, S. A new robust variable step-size NLMS algorithm. *IEEE Trans. Signal Process.* **2008**, *56*, 1878–1893. [CrossRef]
47. Farhang-Boroujeny, B. *Adaptive Filters: Theory and Applications*, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2013.
48. Zhang, S.; Zhang, J. New steady-state analysis of variable step-size LMS algorithm with different noise distributions. *IEEE Signal Lett.* **2014**, *21*, 653–657. [CrossRef]
49. Arenas-García, J.; Gómez-Verdejo, V.; Figueiras-Vidal, A.R. New algorithms for improved adaptive convex combination of LMS transversal filters. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 2239–2249. [CrossRef]
50. Ferrer, M.; Gonzalez, A.; deDiego, M.; Pi nero, G. Convex combination filtered-x algorithms for active noise control systems. *IEEE Trans. Audio Speech Lang. Process.* **2013**, *21*, 156–167. [CrossRef]
51. Azpicueta-Ruiz, L.A.; Silva, M.T.M.; Nascimento, V.H.; Sayed, A.H. Combinations of adaptive filters: Performance and convergence properties. *IEEE Signal Process. Mag.* **2016**, *33*, 120–140.
52. Zhang, S.; Zheng, W.X.; Zhang, J. A new combined-step-size normalized least mean square algorithm for cyclostationary inputs. *Signal Process.* **2017**, *141*, 261–272. [CrossRef]
53. Huang, F.; Zhang, J.; Pang, Y. A novel combination scheme of proportionate filter. *Signal Process.* **2018**, *143*, 222–231. [CrossRef]
54. Stanciu, C.; Anghel, C.; Udrea, M.; Stanciu, L. Variable-regularized low complexity RLS algorithm for stereophonic acoustic echo cancellation. In Proceedings of the 2017 International Symposium Signals, Circuits and Systems (ISSCS), Iasi, Romania, 13–14 July 2017.
55. Belge, M.; Miller, E.L. A Sliding window RLS-like adaptive algorithm for filtering alpha-stable noise. *IEEE Sig. Process. Lett.* **2000**, *7*, 86–89. [CrossRef]
56. Liang, J.; Ji, B.; Zhang, J.; Wang, S.; Zhao, F. Recursive least squares-like algorithms for the adaptive second-order lattice notch filter. *Digit. Signal Process.* **2008**, *18*, 291–306. [CrossRef]