

Article

Robust Bias Compensation Method for Sparse Normalized Quasi-Newton Least-Mean with Variable Mixing-Norm Adaptive Filtering

Ying-Ren Chien ^{1,*} , Han-En Hsieh ¹ and Guobing Qian ² ¹ Department of Electrical Engineering, National Ilan University, Yilan 260007, Taiwan; r1141009@niu.edu.tw² College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China; guobingqian@swu.edu.cn

* Correspondence: yrchien@niu.edu.tw; Tel.: +886-3931-7396

Abstract: Input noise causes inescapable bias to the weight vectors of the adaptive filters during the adaptation processes. Moreover, the impulse noise at the output of the unknown systems can prevent bias compensation from converging. This paper presents a robust bias compensation method for a sparse normalized quasi-Newton least-mean (BC-SNQNLN) adaptive filtering algorithm to address these issues. We have mathematically derived the biased-compensation terms in an impulse noisy environment. Inspired by the convex combination of adaptive filters' step sizes, we propose a novel variable mixing-norm method, BC-SNQNLN-VMN, to accelerate the convergence of our BC-SNQNLN algorithm. Simulation results confirm that the proposed method significantly outperforms other comparative works regarding normalized mean-squared deviation (NMSD) in the steady state.

Keywords: bias compensation; convex combination; impulse noise (IN); noisy inputs; process innovation; variable mixed-norm adaptive filtering algorithm

MSC: 93C73; 93E35



Citation: Chien, Y.-R.; Hsieh, H.-E.; Qian, G. Robust Bias Compensation Method for Sparse Normalized Quasi-Newton Least-Mean with Variable Mixing-Norm Adaptive Filtering. *Mathematics* **2024**, *12*, 1310. <https://doi.org/10.3390/math12091310>

Academic Editor: Davide Valenti

Received: 3 April 2024

Revised: 21 April 2024

Accepted: 23 April 2024

Published: 25 April 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Adaptive filtering algorithms play a pivotal role in signal processing, encompassing tasks such as system identification, channel estimation, feedback cancellation, and noise removal [1]. While literature commonly assumes a Gaussian distribution for system noise, real-world scenarios, including underwater acoustics [2–5], low-frequency atmospheric disturbances [6], and artificial interference [7–9], often exhibit sudden changes in signal or noise intensity [10]. These abrupt variations can disrupt algorithms, serving as external solid interference or outliers [11,12].

Recently, the sparse quasi-Newton least-mean mixed-norm (SQNLMMN) algorithm [13] has emerged as a potential solution to mitigate the impact of both Gaussian and non-Gaussian noises on the convergence behavior of adaptive algorithms [14]. This algorithm introduces a novel cost function incorporating a linear combination of L_2 and L_4 norms while promoting sparsity. Despite its promise, the SQNLMMN algorithm exhibits specific weaknesses. Firstly, it overlooks the presence of input noise at the adaptive filter inputs, leading to biased coefficient estimates [15,16]. Secondly, the fixed mixing parameter δ , governing the balance between the two norms, fails to adapt dynamically. This rigidity in parameter choice trades off convergence rate and mean squared deviation (MSD) concerning the weight coefficients. Notably, the approach of employing a mixed step size for least mean fourth (LMF) and normalized LMF algorithms [17] to address such trade-offs [18] differs from the concept of a variable mixing parameter [19,20].

Based on the unbiased criteria, several methods have been reported to compensate the biases caused by the noisy inputs, such as the bias compensation least mean square algorithm (BC-LMS) [15,21], bias compensation normalized LMS algorithm (BC-NLMS) [22],

bias compensation proportional normalized least mean square algorithm (BC-PNLMS) [23], bias compensation normalized least mean fourth algorithm (BC-NLMF) [24], bias compensated affine-projection-like (BC-APL) [25], and bias compensation least mean mixed norm algorithm (BC-LMMN) [26]. However, the BC-LMMN algorithm used a fixed mixing factor, which resulted in a higher misadjustment. In [27], the authors proposed using a biased-compensated generalized mixed norm algorithm and cooperating with correntropy-induced metric (CIM) as the sparse penalty constraint for sparse system identification problems. Unlike the conventional mixed norm approach, they mixed norm with $p = 1.1$ and $q = 1.2$ to better combat non-Gaussian noise as well as impulse noise. A modified cost function that considered the cost caused by the input noise was used to compensate for the bias [27]. Hereinafter, we refer to it as the BC-CIM-LGMN algorithm. Yet, estimating the input noise power might be adversely affected by impulse noise present at the output of the unidentified system. The same trick that adopted CIM as the sparse penalty constraint is applied in [23], which is referred to as BC-CIM-PNLMS hereinafter. On the other hand, an L_0 norm cost function was used to accelerate the convergence for the sparse systems. In [28], the authors combined it with an improved BC-NLMS, referred to as the BC-ZA-NLMS algorithm. Unfortunately, the BC-ZA-NLMS algorithm fails to consider the impact of the impulse noise. Researchers have proposed combining the variable step-size (VSS) method with BC-LMS to the direction of arrival (DOA) estimation problem [29].

However, few studies have comprehensively addressed all impairments, including noisy input, impulse noise in observations (measurements), and sparse unknown systems. Building upon the SQNLMMN algorithm, this paper introduces a robust bias compensation method for the sparse normalized quasi-Newton least-mean with variable mixing-norm (BC-SNQNLN-VMN) adaptive filtering algorithm. The key contributions of this research are as follows. Firstly, we introduce a normalized variant of the SQNLMMN algorithm and incorporate it with the Huber function to alleviate the impact of impulse noise. Secondly, we develop a bias compensation method to counteract the influence of noisy input on the weight coefficients of the adaptive filter. Thirdly, we introduce a convex combination approach concerning the mixing parameter, enabling the utilization of the variable mixing parameter in the mixed norm approach. Consequently, our proposed method can simultaneously achieve rapid convergence and low misadjustment.

The rest of this paper is organized as follows. Section 2 describes the system model we considered in this work. Section 3 briefly reviews the SQNLMMN algorithm and outlines the proposed BC-SNQNLN-VMN adaptive filtering algorithm. Section 4 validates the effectiveness of our proposed BC-SNQNLN-VMN algorithm by using computer simulations. Conclusions and future prospects are drawn in Section 5.

2. System Models

The system with finite impulse represented by the vector $\mathbf{w} \in \mathbb{R}^{M \times 1}$ to be identified that considers both input noise and observation noise is depicted in Figure 1. The outputs from this system are subject to corruption by two types of noise. The observable desired signal $d(n)$ can be mathematically defined as follows:

$$d(n) = y(n) + [v(n) + \Omega(n)] = \mathbf{u}^T(n)\mathbf{w}(n) + [v(n) + \Omega(n)], \quad (1)$$

where $(\cdot)^T$ is the transpose symbol; the weight vector $\mathbf{w}(n)$, with components $w_0(n), w_1(n), \dots, w_{M-1}(n)$, arranged in column form, represents the unknown system to be identified. $\mathbf{w}(n) = [w_0(n), w_1(n), \dots, w_{M-1}(n)]^T$ represents the weight vector of the unknown system; $\mathbf{u}(n) = [u(n), u(n-1), \dots, u(n-M+1)]^T$ denotes the input regressor vector. Note that the measurement noise is assumed to consist of two components: background additive Gaussian white noise (AGWN) denoted as $v(n)$ and impulse noise

denoted as $\Omega(n)$. The AGWN noise $v(n)$ has a zero mean and variance σ_v^2 . This variance is related to the signal-to-noise ratio (SNR) as follows:

$$\text{SNR}_v = 10 \log_{10}(\sigma_y^2 / \sigma_v^2), \tag{2}$$

where σ_y^2 represents the variance of $y(n)$. In addition, impulse noise is accounted for in the system model. Two conventional models are employed in this study. The first one is the Bernoulli Gaussian (BG) model [30], defined as follows:

$$\Omega(n) = b(n) \cdot v_\Omega(n), \tag{3}$$

where, $b(n)$ takes the value of one with a probability of Pr and zero with a probability of $(1 - Pr)$. Additionally, $v_\Omega(n)$ represents a Gaussian white process characterized by a mean of zero and a variance of σ_Ω^2 . The strength of this impulse noise is SNR_Ω is used to quantify its strength as follows:

$$\text{SNR}_\Omega = 10 \log_{10}(\sigma_y^2 / \sigma_\Omega^2). \tag{4}$$

Another model utilized is the alpha-stable impulse noise model [13], which can be characterized by the parameter vector $V = (\alpha_s, \beta_s, \Gamma_s, \Delta_s)$. Here $\alpha_s \in (0, 2]$ represents the characteristic factor, $\beta_s \in [-1, 1]$ denotes the symmetry parameter, $\Gamma_s \geq 0$ stands for the dispersion parameter, and Δ_s indicates the location parameter. A reduced α_s value signifies a heightened presence of impulse noise.

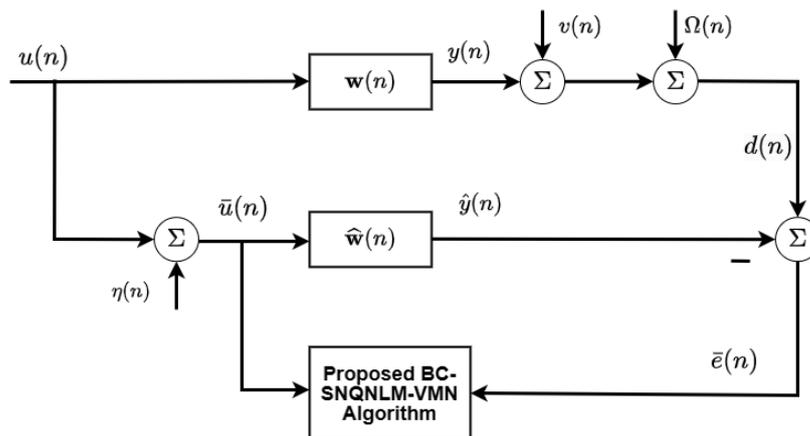


Figure 1. Model of system identification with the proposed BC-SNQNLM-VMN algorithm.

In this paper, we consider the noisy input case, i.e., the input of the adaptive filter $\bar{u}(n)$ differs from that of the unknown system. We assume an AGWN input noise $\eta(n)$ with zero-mean and variance σ_η^2 is added to the original input $u(n)$, i.e., $\bar{u}(n) = u(n) + \eta(n)$. The strength of $\eta(n)$ is determined by the SNR_η as follows:

$$\text{SNR}_\eta = 10 \log_{10}(\sigma_u^2 / \sigma_\eta^2), \tag{5}$$

where σ_u^2 denotes the variance of $u(n)$. The weights of the adaptive filter, denoted by $\hat{\mathbf{w}}(n)$, are updated iteratively through an adaptive algorithm, which computes correction terms based on $\hat{y}(n)$. These corrections rely on the error signal, expressed as:

$$\bar{e}(n) = d(n) - \hat{y}(n) = d(n) - \bar{\mathbf{u}}^T(n) \hat{\mathbf{w}}(n), \tag{6}$$

where $\bar{\mathbf{u}} = [\bar{u}(n), \bar{u}(n - 1), \dots, \bar{u}(n - M + 1)]^T$ denotes the input regressor vector linked to the adaptive filter.

3. Proposed BC-SNQLM-VMN Adaptive Filtering Algorithm

3.1. Review of SQNLMMN Algorithm [13]

The cost function of the SQNLMMN algorithm is expressed as:

$$J(\mathbf{w}(n)) = \frac{\delta}{2} J_2 \mathbf{w}(n) + \frac{1-\delta}{4} J_4 \mathbf{w}(n) + \gamma S(\mathbf{w}(n)), \tag{7}$$

where $J_2 \triangleq E[e^2(n)]$ and $J_4 \triangleq E[e^4(n)]$ are the cost functions for least mean square (LMS) and LMF algorithms, respectively; a fixed mixing parameter $0 \leq \delta \leq 1$ is used to control the mixture of the two cost functions; $S(\cdot)$ denotes the sparsity-promoting term, which is regulated by a positive parameter γ . According to [13], we have the resulting updating recursion of the sparse quasi-Newton least-mean mixed-norm (SQNLMMN) algorithm as follows:

$$\widehat{\mathbf{w}}(n+1) = \widehat{\mathbf{w}}(n) + \mu_1 \mathbf{P}(n) \mathbf{u}(n) e(n) + \mu_2 \mathbf{P}(n) \mathbf{u}(n) e^3(n) - p \mathbf{P}(n) \mathbf{g}(n), \tag{8}$$

where the step size is chosen as $\mu_1 = \delta\mu$ and $\mu_2 = (1-\delta)\mu$ that controls the convergence rate for LMS and LMF algorithms, respectively. Note that μ is a common step size; $p \mathbf{P}(n) \mathbf{g}(n)$ denotes the sparsity penalty term, and p denotes the parameter that controls zero-attraction [13]. The matrix $\mathbf{P}(n) \in \mathbf{R}^{M \times M}$ that approximates the inverse of the Hessian matrix of the cost function can be expressed as follows:

$$\mathbf{P}(n) = \mathbf{B}(n) [\mathbf{I} - \alpha\gamma(\mathbf{I} - \mathbf{H}(n-1)\mathbf{B}(n)\alpha\gamma)\mathbf{H}(n-1)\mathbf{B}(n)], \tag{9}$$

where $\mathbf{B}(n) \in \mathbf{R}^{M \times M}$ is described as follows:

$$\mathbf{B}(n) = \frac{1}{1-\alpha} \left[\mathbf{P}(n-1) \frac{\mathbf{P}^T(n-1)\mathbf{u}(n)\mathbf{u}^T(n)\mathbf{P}(n-1)}{\frac{1-\alpha}{\alpha_1 + \alpha_2 e^2(n)} + \mathbf{u}^T(n)\mathbf{P}(n-1)\mathbf{u}(n)} \right] \tag{10}$$

with $\alpha_1 = \delta\alpha$ and $\alpha_2 = 3(1-\delta)\alpha$. Note that $\mathbf{H}(n) \in \mathbf{R}^{M \times M}$ is the Hessian matrix for $S(\widehat{\mathbf{w}}(n))$. Let $S(\cdot)$ be the L_0 norm and approximate as follows:

$$S(\widehat{\mathbf{w}}(n)) \approx \sum_{i=0}^{M-1} \left(1 - e^{-\beta|\widehat{w}_i(n)|} \right), \tag{11}$$

where the parameter $\beta > 0$ is used to determine the region of zero attraction [31].

The derivation of the gradient for this penalty term is as follows:

$$\mathbf{g}(n) = [t(\widehat{w}_0(n)), t(\widehat{w}_1(n)), \dots, t(\widehat{w}_{M-1}(n))]^T, \tag{12}$$

where

$$t(\widehat{w}_i(n)) = \beta \operatorname{sgn}(\widehat{w}_i(n)) e^{-\beta|\widehat{w}_i(n)|}, \quad \forall \quad 0 \leq i \leq (M-1), \tag{13}$$

and the operator $\operatorname{sgn}(\cdot)$ denotes the sign function. In order to streamline Equation (13), we utilize the first-order Taylor approximation of the exponential function in the following manner:

$$e^{-\beta|u|} \approx \begin{cases} 1 - \beta|u|, & |u| \leq \beta^{-1} \\ 0, & \text{otherwise} \end{cases}. \tag{14}$$

Therefore, we can approximate Equation (13) as follows:

$$t(\hat{w}_i(n)) = \begin{cases} -\beta(1 + \beta\hat{w}_i(n)), & -\beta^{-1} \leq \hat{w}_i(n) < 0 \\ \beta(1 - \beta\hat{w}_i(n)), & 0 < \hat{w}_i(n) \leq \beta^{-1} \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

The Hessian of Equation (11) can be derived as:

$$\mathbf{H}(n) = \text{diag}(t'(\hat{w}_0(n)), t'(\hat{w}_1(n)), \dots, t'(\hat{w}_{M-1}(n))) \quad (16)$$

with

$$t'(\hat{w}_i(n)) = \begin{cases} -\beta^2, & |\hat{w}_i(n)| \leq \beta^{-1} \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

3.2. Normalized SQNLMMN

Inspired by the design of normalized LMS and normalized LMF [32], we propose a normalized version of SQNNLMMN algorithm by modifying Equation (8) that considers the noisy inputs $\bar{\mathbf{u}}(n)$ as follows:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{\mu_1 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}(n)}{\|\bar{\mathbf{u}}(n)\|^2} + \frac{\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n)}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n))} - p \bar{\mathbf{P}}(n) \mathbf{g}(n) \quad (18)$$

where $\bar{\mathbf{u}}(n) = \mathbf{u}(n) + \boldsymbol{\eta}(n)$ denotes the noisy input regressor vector and $\boldsymbol{\eta}(n) = [\eta(n), \eta(n-1), \dots, \eta(n-M+1)]^T$ represents the input noise vector. The noisy error signal $\bar{e}(n)$ is calculated as follows:

$$\bar{e}(n) = d(n) - \bar{\mathbf{u}}^T(n) \hat{\mathbf{w}}(n) = e(n) - \boldsymbol{\eta}^T(n) \hat{\mathbf{w}}(n). \quad (19)$$

Note that the matrix $\bar{\mathbf{P}}(n)$ is a contaminated version of $\mathbf{P}(n)$ (see Equation (9)) defined as follows:

$$\bar{\mathbf{P}}(n) = \bar{\mathbf{B}}(n) - \bar{\mathbf{B}}(n) \alpha \gamma (\mathbf{I} - \mathbf{H}(n-1) \bar{\mathbf{B}}(n) \alpha \gamma) \mathbf{H}(n-1) \bar{\mathbf{B}}(n), \quad (20)$$

where $\gamma = p/\mu > 0$ governs the impact of the penalty term; $0 < \alpha \leq 0.1$ denotes a forgetting factor [13]; the matrix $\bar{\mathbf{B}}(n)$ is a contaminated version of $\mathbf{B}(n)$ (see Equation (10)) defined as follows:

$$\bar{\mathbf{B}}(n) = \frac{1}{1-\alpha} \left[\mathbf{P}(n-1) \frac{\bar{\mathbf{P}}^T(n-1) \bar{\mathbf{u}}(n) \bar{\mathbf{u}}^T(n) \bar{\mathbf{P}}(n-1)}{\frac{1-\alpha}{\alpha_1 + \alpha_2 \bar{e}^2(n)} + \bar{\mathbf{u}}^T(n) \bar{\mathbf{P}}(n-1) \bar{\mathbf{u}}(n)} \right]. \quad (21)$$

Note that the difference between $\bar{e}(n)$ and $e(n)$, i.e., $\boldsymbol{\eta}^T(n) \hat{\mathbf{w}}(n)$, results in the biases during the weight updating process.

3.3. Bias Compensation Design

To compensate for the bias of the normalized SQNLMMN algorithm, we introduce a bias compensation vector $\mathbf{b}(n) \in \mathbf{R}^{M \times 1}$ into the weight-updating recursion and rewrite Equation (18) as follows:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \Delta \hat{\mathbf{w}}(n) + \mathbf{b}(n), \quad (22)$$

with

$$\Delta \widehat{\mathbf{w}}(n) = \frac{\mu_1 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}(n)}{\|\bar{\mathbf{u}}(n)\|^2} + \frac{\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n)}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n))} - p \bar{\mathbf{P}}(n) \mathbf{g}(n). \quad (23)$$

We further define the weight estimation error vector as follows:

$$\tilde{\mathbf{w}}(n) = \widehat{\mathbf{w}}(n) - \mathbf{w}(n). \quad (24)$$

By combining Equations (22) and (24), we then have the following recursion

$$\tilde{\mathbf{w}}(n+1) = \tilde{\mathbf{w}}(n) + \Delta \widehat{\mathbf{w}}(n) + \mathbf{b}(n). \quad (25)$$

It has been reported that the sparsity terms in Equation (23), i.e., $p \bar{\mathbf{P}}(n) \mathbf{g}(n)$, should be ignored when deriving the bias compensation term $\mathbf{b}(n)$; otherwise the derived vector $\mathbf{b}(n)$ will compensate for the bias caused by both the input noise and this term [28]. Hence, the recursion for weight updating can be formulated as follows:

$$\tilde{\mathbf{w}}(n+1) = \tilde{\mathbf{w}}(n) + \frac{\mu_1 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} + \frac{\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n)}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n))} + \mathbf{b}(n). \quad (26)$$

Given the noisy input vector $\bar{\mathbf{u}}(n)$, we then derive $\mathbf{b}(n)$ based on the unbiased criterion as follows:

$$E[\tilde{\mathbf{w}}(n+1) | \bar{\mathbf{u}}(n)] = \mathbf{0} \quad \text{whenever} \quad E[\tilde{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] = \mathbf{0}. \quad (27)$$

Furthermore, to simplify the analysis, two commonly used assumptions have been employed [33] as follows:

Assumption 1. The input noise $\eta(n)$ and background noise are zero-mean AGWN noises and the ratio $\rho = \sigma_v^2 / \sigma_\eta^2$ is a prior knowledge.

Assumption 2. The signals $\eta(n)$, $v(n)$, $\Omega(n)$, $u(n)$, and $\tilde{\mathbf{w}}(n)$ are statistically independent.

By taking expectation on both sides of Equation (26) for the given $\bar{\mathbf{u}}(n)$ and assuming $E[\tilde{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] = \mathbf{0}$, we have

$$\begin{aligned} E[\tilde{\mathbf{w}}(n+1) | \bar{\mathbf{u}}(n)] &= E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right] \\ &+ E \left[\frac{\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n)}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n))} \middle| \bar{\mathbf{u}}(n) \right] + E[\mathbf{b}(n) | \bar{\mathbf{u}}(n)]. \end{aligned} \quad (28)$$

By replacing $\bar{\mathbf{u}}(n)$ and $\bar{e}(n)$ in the first term of the right-hand side (RHS) of Equation (28), we have

$$\begin{aligned} E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right] &= E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \mathbf{u}(n) e(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right] \\ &- E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \boldsymbol{\eta}(n) \boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right], \end{aligned} \quad (29)$$

where

$$\begin{aligned} E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \mathbf{u}(n) e(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right] &= \frac{\mu_1 E[\bar{\mathbf{P}}(n) | \bar{\mathbf{u}}(n)]}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} E \left[\mathbf{u}(n) \left(-\mathbf{u}^T(n) \tilde{\mathbf{w}}(n) + v(n) \right) \middle| \bar{\mathbf{u}}(n) \right] \\ &= \frac{\mu_1 E[\bar{\mathbf{P}}(n) | \bar{\mathbf{u}}(n)]}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \left\{ -E \left[\mathbf{u}(n) \mathbf{u}^T(n) | \bar{\mathbf{u}}(n) \right] E[\tilde{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] + E[\mathbf{u}(n) v(n) | \bar{\mathbf{u}}(n)] \right\} = \mathbf{0} \end{aligned} \quad (30)$$

and

$$\begin{aligned}
 E \left[\frac{\mu_1 \bar{\mathbf{P}}(n) \boldsymbol{\eta}(n) \boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n)}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} \middle| \bar{\mathbf{u}}(n) \right] &= \frac{\mu_1 E[\bar{\mathbf{P}}(n) | \bar{\mathbf{u}}(n)]}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} E[\boldsymbol{\eta}(n) \boldsymbol{\eta}^T(n)] E[\widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] \\
 &= \frac{\mu_1 E[\bar{\mathbf{P}}(n) | \bar{\mathbf{u}}(n)] \sigma_\eta^2}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} E[\widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)].
 \end{aligned} \tag{31}$$

Note that as the condition $\|\bar{\mathbf{u}}(n)\|^2 \gg \bar{e}^2(n)$ and the deviation of $\|\bar{\mathbf{u}}(n)\|^2$ being small hold, the the second term of the RHS of Equation (28) can be approximated as follows [33]:

$$E \left[\frac{\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n)}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n))} \middle| \bar{\mathbf{u}}(n) \right] \approx \frac{E[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n) | \bar{\mathbf{u}}(n)]}{E[\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n)) | \bar{\mathbf{u}}(n)]}. \tag{32}$$

Thus, we can rewrite the nominator of Equation (32) as follows:

$$\begin{aligned}
 E[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) \bar{e}^3(n) | \bar{\mathbf{u}}(n)] &= E \left[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) (e(n) - \boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^3 \middle| \bar{\mathbf{u}}(n) \right] \\
 &= E[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) e^3(n) | \bar{\mathbf{u}}(n)] \\
 &\quad + E \left[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) 3e(n) (\boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^2 \middle| \bar{\mathbf{u}}(n) \right] \\
 &\quad - E \left[\mu_2 \bar{\mathbf{P}}(n) \bar{\mathbf{u}}(n) (\boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^3 \middle| \bar{\mathbf{u}}(n) \right] \\
 &= \sigma_\eta^4 E[\mu_2 \bar{\mathbf{P}}(n) \widehat{\mathbf{w}}(n) \widehat{\mathbf{w}}^T(n) \widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)].
 \end{aligned} \tag{33}$$

Furthermore, we can rewrite the denominator of the RHS of Equation (32) as follows:

$$\begin{aligned}
 &E[\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \bar{e}^2(n)) | \bar{\mathbf{u}}(n)] \\
 &= E \left[\|\bar{\mathbf{u}}(n)\|^2 \left(\|\bar{\mathbf{u}}(n)\|^2 + e^2(n) - 2e(n) \boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n) + (\boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^2 \right) \middle| \bar{\mathbf{u}}(n) \right] \\
 &= \|\bar{\mathbf{u}}(n)\|^2 \left(\|\bar{\mathbf{u}}(n)\|^2 + E[e^2(n) | \bar{\mathbf{u}}(n)] + E \left[(\boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^2 \middle| \bar{\mathbf{u}}(n) \right] \right),
 \end{aligned} \tag{34}$$

where

$$E[e^2(n) | \bar{\mathbf{u}}(n)] = E \left[(\mathbf{u}^T(n) \widetilde{\mathbf{w}}(n))^2 - 2\mathbf{u}^T(n) \widetilde{\mathbf{w}}(n) v(n) + v^2(n) \middle| \bar{\mathbf{u}}(n) \right] = \sigma_v^2 \tag{35}$$

and

$$E \left[(\boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n))^2 \middle| \bar{\mathbf{u}}(n) \right] = E[\widehat{\mathbf{w}}^T(n) \boldsymbol{\eta}(n) \boldsymbol{\eta}^T(n) \widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] = \sigma_\eta^2 E[\widehat{\mathbf{w}}^T(n) \widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)]. \tag{36}$$

Combining the results Equations (29) to (36) and substituting them into Equation (28), we obtain the following results:

$$\begin{aligned}
 E[\mathbf{b}(n) | \bar{\mathbf{u}}(n)] &= \frac{\mu_1 E[\bar{\mathbf{P}}(n) | \bar{\mathbf{u}}(n)] \sigma_\eta^2}{\bar{\mathbf{u}}^T(n) \bar{\mathbf{u}}(n)} E[\widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)] \\
 &\quad + \mu_2 \frac{\sigma_\eta^4 E[\bar{\mathbf{P}}(n) \widehat{\mathbf{w}}(n) \widehat{\mathbf{w}}^T(n) \widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)]}{\|\bar{\mathbf{u}}(n)\|^2 (\|\bar{\mathbf{u}}(n)\|^2 + \sigma_\eta^2 E[\widehat{\mathbf{w}}^T(n) \widehat{\mathbf{w}}(n) | \bar{\mathbf{u}}(n)])}.
 \end{aligned} \tag{37}$$

By using the stochastic approximation [34], we derive the bias-compensation vector as follows:

$$\mathbf{b}(n) = \mu_1 \frac{E[\bar{\mathbf{P}}(n)|\bar{\mathbf{u}}(n)]\sigma_{\eta}^2\hat{\mathbf{w}}(n)}{\bar{\mathbf{u}}^T(n)\bar{\mathbf{u}}(n)} + \mu_2 \frac{E[\bar{\mathbf{P}}(n)|\bar{\mathbf{u}}(n)]\sigma_{\eta}^4\hat{\mathbf{w}}(n)\hat{\mathbf{w}}^T(n)\hat{\mathbf{w}}(n)}{\|\bar{\mathbf{u}}(n)\|^2(\|\bar{\mathbf{u}}(n)\|^2 + \sigma_v^2 + \sigma_{\eta}^2\hat{\mathbf{w}}^T(n)\hat{\mathbf{w}}(n))} \quad (38)$$

with

$$\begin{aligned} E[\bar{\mathbf{P}}(n)|\bar{\mathbf{u}}(n)] &:= \mathbf{P}_{bias}(n) \\ &= \mathbf{B}_{bias}(n)[\mathbf{I} - \alpha\gamma(\mathbf{I} - \mathbf{H}(n-1)\mathbf{B}_{bias}(n)\alpha\gamma)\mathbf{H}(n-1)\mathbf{B}_{bias}(n)], \end{aligned} \quad (39)$$

where

$$\begin{aligned} \mathbf{B}_{bias}(n) &:= E[\bar{\mathbf{B}}(n)|\bar{\mathbf{u}}(n)] \\ &= \frac{1}{1-\alpha} \left[\mathbf{P}_{bias}(n-1) \frac{\mathbf{P}_{bias}^T(n-1)\bar{\mathbf{u}}(n)\bar{\mathbf{u}}^T(n)\mathbf{P}_{bias}(n-1)}{\frac{1-\alpha}{\alpha_1+\alpha_2(\sigma_v^2+\sigma_{\eta}^2\hat{\mathbf{w}}^T(n)\hat{\mathbf{w}}(n))} + \bar{\mathbf{u}}^T(n)\mathbf{P}_{bias}(n-1)\bar{\mathbf{u}}(n)} \right]. \end{aligned} \quad (40)$$

3.4. Variable Mixing Parameter Design

For the conventional SQNLMMN algorithm, it was suggested to use a fixed mixing parameter $\delta = 0.8$ to achieve the best performance in terms of the convergence rate. However, a small mixing parameter, say $\delta = 0.2$, could slowly achieve a lower misadjustment in the steady state than that with a large δ . This inspires us to use a variable mixing parameter to attain a fast convergence rate and small misadjustment simultaneously.

Figure 2 depicts the block diagram of the variable mixing parameter scheme design. Two adaptive filters are combined as follows:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}_1(n+1)\lambda_C(n+1) + \hat{\mathbf{w}}_2(n+1)[1 - \lambda_C(n+1)], \quad (41)$$

where $\hat{\mathbf{w}}_1(n)$ and $\hat{\mathbf{w}}_2(n)$ are the fast and slow filters, respectively, i.e., $\delta_1 > \delta_2$; $\lambda(n+1)$ is the smoothed combination factor. Referring to Equation (18), the adaptation recursion for each filter can be expressed as follows:

$$\hat{\mathbf{w}}_i(n+1) = \hat{\mathbf{w}}_i(n) + \Delta\hat{\mathbf{w}}_i(n) + \kappa\mathbf{b}_i(n), \quad (42)$$

with

$$\Delta\hat{\mathbf{w}}_i(n) = \frac{\mu_{1,i}\bar{\mathbf{P}}_i(n)\bar{\mathbf{u}}(n)\bar{\mathbf{e}}_i(n)}{\|\bar{\mathbf{u}}(n)\|^2} + \frac{\mu_{2,i}\bar{\mathbf{P}}_i(n)\bar{\mathbf{u}}(n)\bar{\mathbf{e}}_i^3(n)}{\|\bar{\mathbf{u}}(n)\|^2(\|\bar{\mathbf{u}}(n)\|^2 + \bar{\mathbf{e}}_i^2(n))} - p\bar{\mathbf{P}}_i(n)\mathbf{g}_i(n), \quad (43)$$

where $\mathbf{g}_i(n)$ can be calculated by Equation (12) for $\hat{\mathbf{w}}_i(n)$, and

$$\bar{\mathbf{P}}_i(n) = \bar{\mathbf{B}}_i(n)\{\mathbf{I} - \alpha\gamma[\mathbf{I} - \mathbf{H}_i(n-1)\bar{\mathbf{B}}_i(n)\alpha\gamma]\mathbf{H}_i(n-1)\bar{\mathbf{B}}_i(n)\}, \quad (44)$$

where $\mathbf{H}_i(n-1)$ can be calculated by Equation (16) for $\hat{\mathbf{w}}_i(n-1)$ and the scaling factor $0 \ll \kappa < 1$ in Equation (42) is used to mitigate the interaction between $\Delta\hat{\mathbf{w}}_i(n)$ and $\mathbf{b}_i(n)$; $\mu_{1,i} = \mu\delta_i$ and $\mu_{2,i} = \mu(1 - \delta_i)$; the matrix $\bar{\mathbf{B}}(n)$ is a contaminated version of $\mathbf{B}(n)$ (see Equation (10)) defined as follows:

$$\bar{\mathbf{B}}_i(n) = \frac{1}{1-\alpha} \left[\bar{\mathbf{P}}_i(n-1) \frac{\bar{\mathbf{P}}_i^T(n-1)\bar{\mathbf{u}}(n)\bar{\mathbf{u}}^T(n)\bar{\mathbf{P}}_i(n-1)}{\frac{1-\alpha}{\alpha_{1,i}+\alpha_{2,i}\bar{\mathbf{e}}_i^2(n)} + \bar{\mathbf{u}}^T(n)\bar{\mathbf{P}}_i(n-1)\bar{\mathbf{u}}(n)} \right], \quad (45)$$

where $\alpha_{1,i} = \delta_i\alpha$ and $\alpha_{2,i} = 3(1 - \delta_i)\alpha$. The bias compensation vector $\mathbf{b}_i(n)$ associated with $\hat{\mathbf{w}}_i(n)$ can be expressed as follows:

$$\mathbf{b}_i(n) = \mu_{1,i} \frac{E[\bar{\mathbf{P}}_i(n)|\bar{\mathbf{u}}(n)]\sigma_{\eta}^2\hat{\mathbf{w}}_i(n)}{\bar{\mathbf{u}}^T(n)\bar{\mathbf{u}}(n)} + \mu_{2,i} \frac{E[\bar{\mathbf{P}}_i(n)|\bar{\mathbf{u}}(n)]\sigma_{\eta}^4\hat{\mathbf{w}}_i(n)\hat{\mathbf{w}}_i^T(n)\hat{\mathbf{w}}_i(n)}{\|\bar{\mathbf{u}}(n)\|^2(\|\bar{\mathbf{u}}(n)\|^2 + \sigma_v^2 + \sigma_{\eta}^2\hat{\mathbf{w}}_i^T(n)\hat{\mathbf{w}}_i(n))} \quad (46)$$

with

$$E[\bar{\mathbf{P}}_i(n)|\bar{\mathbf{u}}(n)] := \mathbf{P}_{bias,i}(n) = \mathbf{B}_{bias,i}(n) - \mathbf{B}_{bias,i}(n)\alpha\gamma(\mathbf{I} - \mathbf{H}_i(n-1)\mathbf{B}_{bias,i}(n)\alpha\gamma)\mathbf{H}_i(n-1)\mathbf{B}_{bias,i}(n), \quad (47)$$

where

$$\mathbf{B}_{bias,i}(n) = \frac{1}{1-\alpha} \left[\mathbf{P}_{bias,i}(n-1) \frac{\mathbf{P}_{bias,i}^T(n-1)\bar{\mathbf{u}}(n)\bar{\mathbf{u}}^T(n)\mathbf{P}_{bias,i}(n-1)}{\frac{1-\alpha}{\alpha_{1,i}+\alpha_{2,i}}(\sigma_v^2 + \sigma_{\eta}^2\hat{\mathbf{w}}_i^T(n)\hat{\mathbf{w}}_i(n)) + \bar{\mathbf{u}}^T(n)\mathbf{P}_{bias,i}(n-1)\bar{\mathbf{u}}(n)} \right]. \quad (48)$$

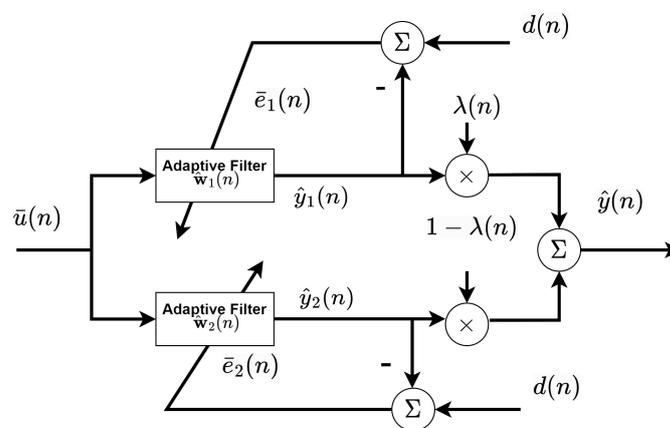


Figure 2. The design of variable mixing parameter scheme.

The smoothed combination equation for λ_C is given by

$$\lambda_C(n+1) = \frac{1}{C} \sum_{k=n-C+2}^{n+1} \lambda(k), \quad (49)$$

where C is the length used to smooth the combination factor λ , which can be calculated as follows [35]:

$$\lambda(n+1) = \text{sgm}[a(n+1)] = \frac{1}{1 + e^{-a(n+1)}} \quad (50)$$

with

$$a(n+1) = a(n) + \mu_a \text{sgn}(\bar{e}(n)) [\hat{y}_1(n) - \hat{y}_2(n)] \lambda(n) [1 - \lambda(n)], \quad (51)$$

where μ_a is the step size for adjusting the recursion of $a(n)$; $\text{sgm}(\cdot)$ and $\text{sgn}(\cdot)$ denote the sigmoid and sign function, respectively. Note that we confine $|a(n+1)| \leq a^+$, and we check if the condition holds every N_0 iterations. We force $\lambda(n+1) = 0$ when $a(n+1) \leq -a^+$ and set $\lambda(n+1) = 1$ when $a(n+1) \geq a^+$ [36].

3.5. Robustness Consideration

To obtain the impact of impulse noise $\Omega(n)$ on the convergence of the adaptive filter $\hat{\mathbf{w}}_i(n)$ in the proposed BC-SNQNLM-VMN algorithm, we propose applying the modified Huber function $\psi(\cdot)$ on $\bar{e}_i(n)$ as follows [37]:

$$\psi(\bar{e}_i(n)) = \begin{cases} \bar{e}_i(n), & \text{for } -\xi_i \leq \bar{e}_i(n) \leq \xi_i \\ 0, & \text{otherwise} \end{cases}, \quad (52)$$

where ζ_i is a threshold as follows:

$$\zeta_i = k_\zeta \hat{\sigma}_{\bar{e}_i(n)}, \tag{53}$$

with $k_\zeta = 2.576$.

$$\hat{\sigma}_{\bar{e}_i(n)} = \lambda_\sigma \hat{\sigma}_{\bar{e}_i(n-1)} + c_1(1 - \lambda_\sigma) + \text{med}(A_{\bar{e}_i}(n)), \tag{54}$$

where $\text{med}(\cdot)$ denotes the median operation; $A_{\bar{e}_i}(n)$ is an observation vector for $\bar{e}_i^2(n)$ with length N_w defined as follows:

$$A_{\bar{e}_i}(n) = [\bar{e}_i^2, \dots, \bar{e}_i^2(n - N_w + 1)]. \tag{55}$$

Note that we choose $c_1 = 2.13$, $N_w = 9$, and $\lambda_\sigma = 0.99$ in the computer simulations.

Furthermore, the estimation of σ_η^2 is required to calculate the bias compensation vector $\mathbf{b}_i(n)$. Concerning robustness, it has been reported that σ_η^2 can be estimated as follows [22,24]:

$$\hat{\sigma}_{\eta,i}^2 = \begin{cases} \bar{e}_i^2(n) / (\|\mathbf{w}_i(n)\|_2^2 + \rho), & \text{if } \bar{e}_i^2 \leq 2\|\mathbf{w}_i(n)\|_2^2 \|\bar{\mathbf{u}}(n)\|_2^2 \\ \hat{\sigma}_{\eta,i}^2, & \text{otherwise} \end{cases} \tag{56}$$

where ρ , which is the ratio of σ_σ^2 and σ_η^2 , is assumed to be available as in [38].

3.6. Computational Cost Analysis

Table 1 lists the major computational cost for the proposed BC-SNQNLM-VMN algorithm in terms of the required number of adders (Adds) and multipliers (Muls). Moreover, Table 2 compares the dominating computational costs with other comparative works. Note that we focus on the dominating terms of the total number of required adders and multipliers for simplicity consideration. Due to the high computational costs caused by the original SQNLMMN algorithm, the proposed method incurs much higher computational costs than comparative works.

Table 1. Computational analysis for the proposed method.

No.	Operations	Adds	Muls
1	$\hat{y}_i(n) = \hat{\mathbf{w}}_i^T(n) \bar{\mathbf{u}}_i(n)$	$M - 1$	M
2	$\bar{e}_i(n) = d(n) - \hat{y}_i(n)$	1	-
3	$\bar{\mathbf{B}}_i(n)$ in Equation (45)	$M^3 + M^2 - M + 2$	$M^3 + 4M^2 + M + 7$
4	$\bar{\mathbf{P}}_i(n)$ in Equation (44)	$2M^3$	$2M^3 + 4M^2$
5	$\Delta \hat{\mathbf{w}}_i(n)$ in Equation (43)	$3M^2$	$3M^2 + 4M + 7$
6	$\mathbf{P}_{bias,i}(n)$ in Equation (47)	$2M^3 - M^2$	$2M^3 + 4M^2 + 1$
7	$\mathbf{B}_{bias,i}(n)$ in Equation (48)	$M^3 + 2M^2 + 3$	$M^3 + 4M^2 + 2M + 4$
8	$\mathbf{b}_i(n)$ in Equation (46)	$M^2 + M$	$M^2 + 4M + 6$
9	$\hat{\mathbf{w}}_i(n + 1)$ in Equation (42)	$2M$	M
10	$\hat{\mathbf{w}}(n + 1)$ in Equation (41)	$M + 1$	$2M$

Table 2. Dominating computational costs comparisons.

Methods	Adds	Muls
BC-NLMS [22]	$2M$	$4M$
BC-NLMF [24]	$2M$	$5M$
BC-LMMN [26]	$3M$	$5M$
BC-CIM-LGMN [27]	$3M$	M^2
BC-CIM-PNLMS [23]	$3M^2$	$3M^2$
Proposed	$6M^3$	$6M^3$

4. Simulation Results

4.1. Setup

Computer simulations evaluated the effectiveness of the proposed algorithm. The unknown sparse system \mathbf{w} comprises 32 taps ($M = 32$), which has $K = 8$ nonzero taps, i.e., the sparsity is 0.75. We randomly choose the positions of the nonzero taps among M taps, and their values follow a standard Gaussian distribution. A standard AGWN models the input signal. The signal-to-noise ratio (SNR) for the input signal $\text{SNR}_\eta = 10$ dB (see Equation (5)) and the SNR of the observed signal $\text{SNR}_v = 30$ dB (see Equation (2)). For the BG impulse noise model, we choose the SNR of the additive impulse noise $\text{SNR}_\Omega = -30$ dB (see Equation (4)). Moreover, we designate the occurrence probability of BG impulse as $Pr = 10^{-3}$ for weak BG noise and $Pr = 6 \times 10^{-2}$ for strong BG noise. For the alpha-stable impulse noise model, we define $V = (1.8, 0, 0.1, 0)$ for weak alpha stable noise and $V = (1.5, 0, 0.1, 0)$ for strong alpha stable noise [11]. Other main parameters are setting as follows: $\gamma = 2 \times 10^{-6}$, $\mu = 0.5$, $p = 10^{-6}$, $\kappa = 0.8$, $\beta = 5$, $a^+ = 8$, $\mu_a = 5$, $N_0 = 2$, and $\alpha = 0.01$.

The fast filter $\hat{\mathbf{w}}_1(n)$ with $\delta_1 = 0.8$ and the slow filter $\hat{\mathbf{w}}_2(n)$ with $\delta_2 = 0.2$ were used to combine the filter $\hat{\mathbf{w}}(n)$ (see Equation (41)). We used a vector with length C to store the C consecutive values of instantaneous $\lambda(k)$ (see Equation (49)). The initial value of each element in this vector was 1. This makes the value of λ_C lean to 1 during the adjustment phase, i.e., the combined filter behaves like the fast filter. The performance metric is the normalized mean-square deviation (NMSD), which can be calculated as follows:

$$\text{NMSD}(n) = 10 \log_{10} \left(\frac{\|\mathbf{w} - \hat{\mathbf{w}}(n)\|^2}{\|\mathbf{w}\|^2} \right). \quad (57)$$

We employed MATLAB[®] R2022a installed on a Windows 10 64-bit operating system to conduct simulations on a computer equipped with an Intel[®] Core[™] i7-13700K CPU from Santa Clara, CA, United States, and 128GB DDR4 RAM. We plotted the NMSD learning curves and the evolution of mixing parameters in the simulation results by averaging over 100 independent Monte Carlo trials. The comparative works were BC-NLMS [22], BC-NLMF [24], BC-LMMN [26], BC-CIM-LGMN [27], and BC-CIM-PNLMS [23] algorithms.

4.2. Results

4.2.1. Baseline: No Impulse Noise

We compare the NMSD learning curves concerning various bias-compensated adaptive filtering algorithms in the absence of impulse noise. As shown in Figure 3, our proposed method can achieve the lowest NMSD during the steady state, outperforming the comparative works by 2.5–3.5 dB. Note that $C = 1$ (see Equation (49)) implies no smoothing was applied. The results confirmed that combining two filters with a smoothed factor (see Equation (41)) exhibits a more smooth convergence behavior in the transient stage ($0.5 \times 10^4 \leq n \leq 0.9 \times 10^4$) during the simulation. Note that: (1) the step sizes are chosen so that the convergence rate is at the same level for all algorithms to have a fair comparison; (2) the NMSD loss is about 12 dB without bias compensation.

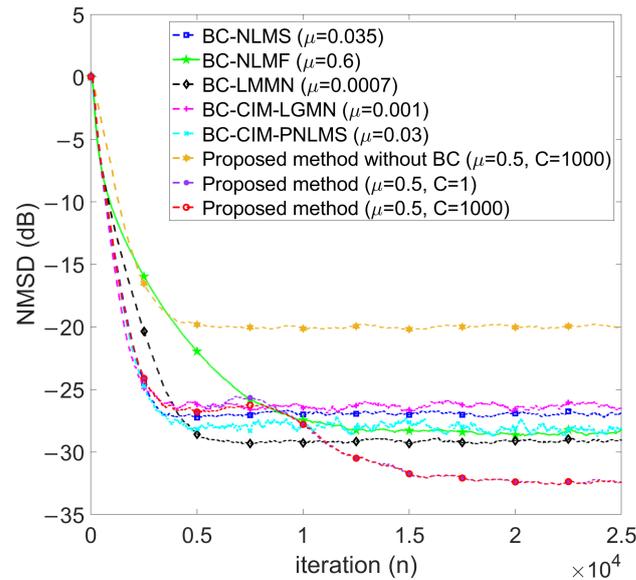


Figure 3. Comparisons of NMSD learning curve without impulse noise.

4.2.2. Evaluation of Variable Mixing Parameter Method

As shown in Figure 4, we evaluate the effectiveness of the proposed variable mixing parameter scheme. In Figure 4a,b, the additive impulse noise corresponds to the weak and strong BG impulse noise, respectively. The results exhibit that our proposed combining mixing parameter δ scheme worked well in both types of additive impulse noise. Referring to Figure 4a, we can observe the variation of the mixing parameter in the weak BG impulse noise case as follows:

1. First, λ_C keeps at 1 in the early stage ($n \in [0, 0.5 \times 10^4]$), which implies the proposed method behaves like $\delta = 0.8$.
2. Then, λ_C decreases gradually during the transient stage ($n \in (0.5 \times 10^4, 1.25 \times 10^4]$), which implies the proposed method behaves changing from $\delta = 0.8$ to $\delta = 0.2$.
3. Finally, λ_C keeps around 0 in the steady-state stage ($n \in (1.25 \times 10^4, 2.5 \times 10^4]$).

Note that because the initial values for the vector with length C used to calculate λ_C are all set to 1, we observed a slight decrease followed by an increase in λ_C at the beginning of the adaptation process. Similar results were observed in the strong BG impulse noise case (see Figure 4b).

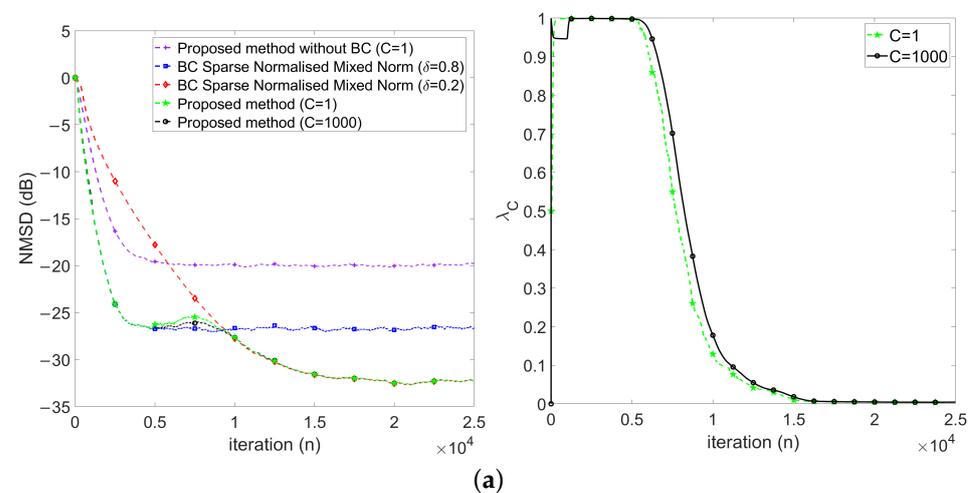


Figure 4. Cont.

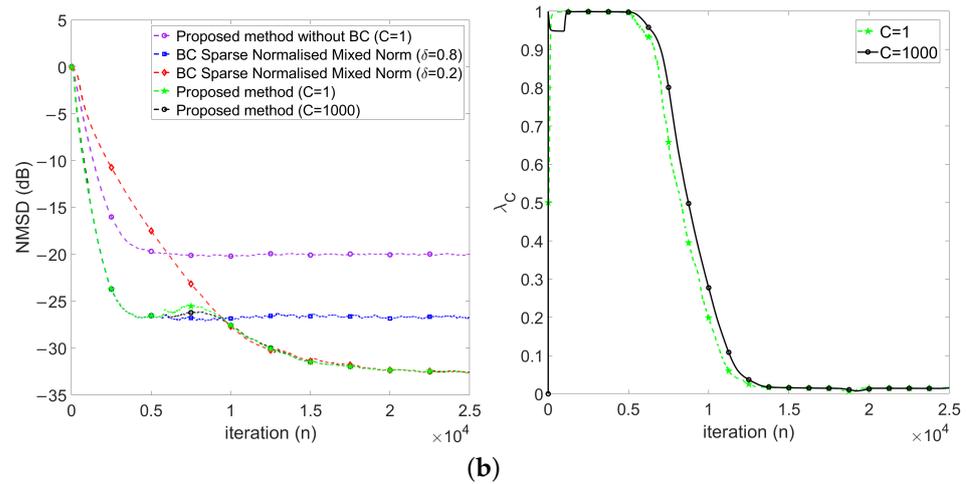


Figure 4. NMSD learning curves (left) and evolution of λ_C (right) for BG impulse noise mode: (a) weak ($Pr = 0.001$) and (b) strong ($Pr = 0.06$).

In Figure 5, we evaluated the impact of the additive alpha-stable impulse noise in both weak (Figure 5a) and strong (Figure 5b) cases. In this scenario, we observe that the strong alpha stable impulse noise makes the NMSD learning curves exhibit more fluctuations in the steady state. In addition, we observe that the evolution of λ shows more fluctuations at the beginning of the transient stage and quickly reaches its steady state. This implies the proposed method behaves by changing from $\delta = 0.8$ to $\delta = 0.2$ earlier than in the case of weak impulse noise. Compared with the baseline, the smoothed combination method exhibits fewer fluctuations, especially in the strong impulse noise cases. Thus, we choose $C = 10^3$ without an explicit statement in the following simulation. The results have shown the impact of the noisy input on the resulting NMSD. Without bias compensation, the NMSD loss in the steady status is about 12 dB, the same as the baseline. Therefore, we can confirm the robustness of our proposed method.

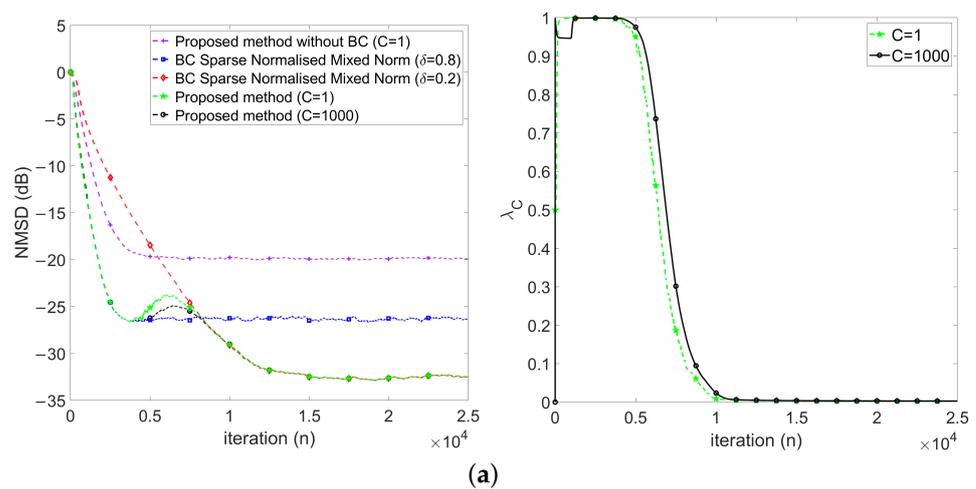


Figure 5. Cont.

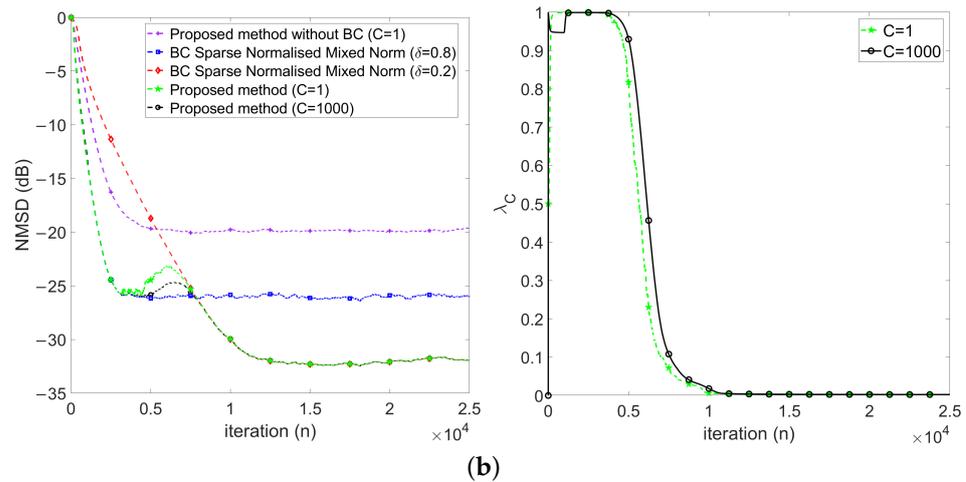


Figure 5. NMSD learning curves (left) and evolution of λ_C (right) for alpha stable impulse noise mode: (a) weak ($\alpha_s = 1.8$) and (b) strong ($\alpha_s = 1.5$).

4.2.3. Performance Comparisons in the Presence of Impulse Noise

As shown in Figure 6, we compare our proposed method with other comparative works for the BG impulse noise case. In the weak BG impulse noise case (see Figure 6a), our proposed method achieves the lowest NMSD and improves by 3 dB to 15 dB compared to comparative works. Note that the BC-CIM-PNLMS did not consider the impact of impulse noise, which resulted in the worst NMSD performance. However, BC-CIM-LGMN, BC-LMMN, and BC-NLMF algorithms diverged in the strong BG impulse noise case (see Figure 6b). In this case, only the BC-NLMS and our proposed method still function well, and our method improves the NMSD by 3 dB in the steady state.

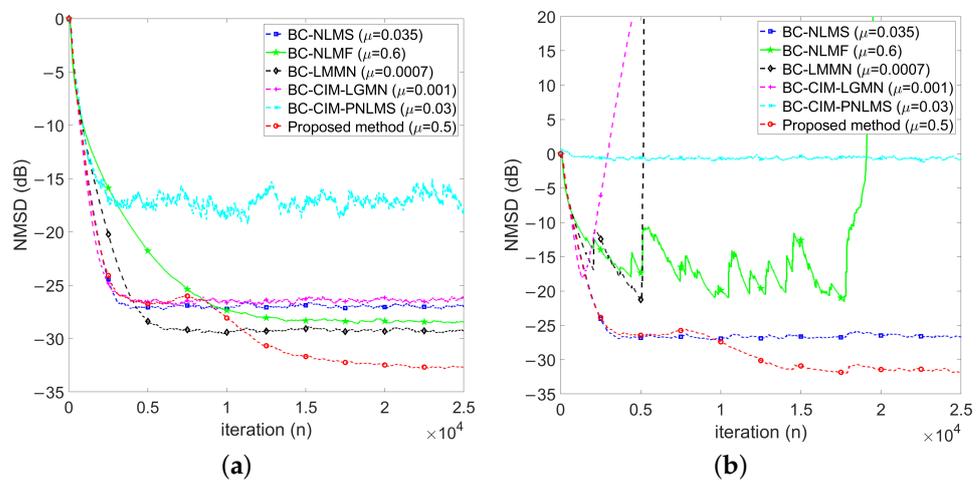


Figure 6. NMSD learning curve under BG impulse noise: (a) weak ($Pr = 0.001$) and (b) strong ($Pr = 0.06$).

As shown in Figure 7, we compare our proposed method with other comparative works for the alpha stable impulse noise case. In the weak alpha stable impulse noise case (see Figure 7a), our proposed method achieves the lowest NMSD and improves by 4.5 dB to 7 dB compared to comparative works. Note that the BC-CIM-PNLMS did not consider the impact of impulse noise, exhibiting some NMSD learning curve spikes. In addition, the comparative works exhibit poor performance in the strong alpha stable impulse noise case (see Figure 7b). In this case, the BC-CIM-PNLMS exhibits stronger spikes than that in the weak alpha stable impulse noise case. However, our proposed method shows the lowest NMSD loss (about 0.9 dB) compared to other comparative works in the steady state.

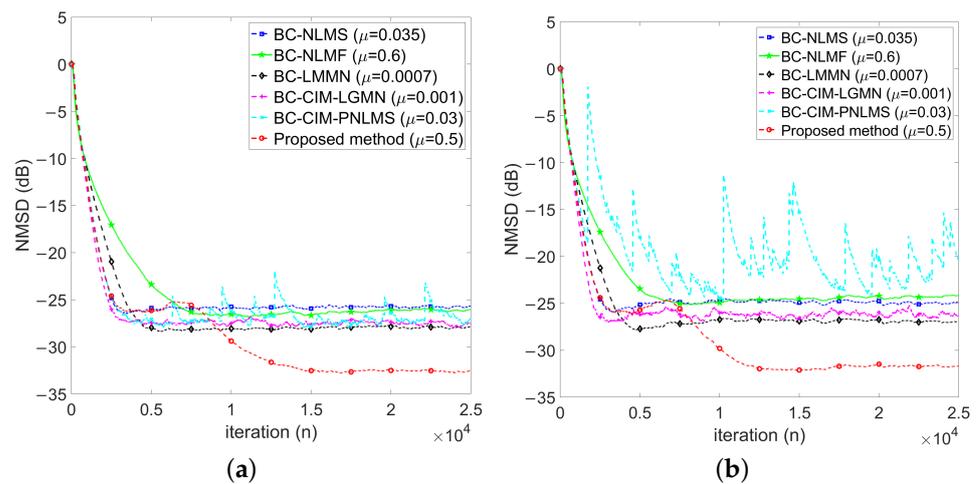


Figure 7. NMSD learning curve of AGWN input under alpha stable impulse noise: (a) weak ($\alpha_s = 1.8$) and (b) strong ($\alpha_s = 1.5$).

5. Conclusions and Future Prospects

The noisy input signals result in a significant NMSD loss even in the absence of an impulse noise scenario. In this paper, we have presented a robust bias compensation method for the SNQNL algorithm. Furthermore, we have proposed a variable mixing-norm method to attain a high convergence rate and low misadjustment during adaptation. Simulation results have confirmed that our proposed BC-SNQNL-VMN algorithm outperforms the comparative works by 3 to 15 dB for BG impulse noise and 4.5 to 7 dB for alpha stable impulse noise in terms of NMSD, respectively. Additionally, we illuminate potential pathways for overcoming remaining challenges and broadening the applicability of our methodologies:

1. The interaction between weight-vector correction term $\Delta\hat{\mathbf{w}}_i(n)$ and bias compensation term $\kappa\mathbf{b}_i(n)$ (see Equation (42)): We have employed a constant scaling factor, κ , to mitigate the interaction between the weight-vector correction term, $\Delta\hat{\mathbf{w}}_i(n)$, and the bias compensation term, $\kappa\mathbf{b}_i(n)$. Nonetheless, devising a dynamic algorithm for adapting κ would enhance the robustness of bias compensation methods to varying input noise over time or in scenarios involving time-varying unknown systems.
2. Extension to general mixed-norm algorithms: While our study focused on L_2 and L_4 norms, the methodology can be extended to encompass bias compensation in adaptive filtering algorithms utilizing a mix of L_p and L_q norms, where p and q are positive parameters.
3. Bias compensation for non-linear adaptive filtering systems: Addressing bias compensation becomes notably intricate in non-linear adaptive filtering systems. A future avenue of research involves developing techniques to estimate biases induced by noisy inputs in such non-linear contexts.

Author Contributions: Conceptualization, Y.-R.C., H.-E.H. and G.Q.; methodology, Y.-R.C. and H.-E.H.; software, Y.-R.C. and H.-E.H.; validation, Y.-R.C. and G.Q.; formal analysis, Y.-R.C. and H.-E.H.; investigation, H.-E.H.; resources, Y.-R.C. and H.-E.H.; data curation, Y.-R.C. and H.-E.H.; supervision, Y.-R.C.; writing—original draft preparation, Y.-R.C. and H.-E.H.; writing—review and editing, Y.-R.C., H.-E.H. and G.Q.; project administration, Y.-R.C.; funding acquisition, Y.-R.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Science and Technology Council, Taiwan (NSTC) under Grant 112-2221-E-197-022.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We would like to thank all the reviewers for their constructive comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Diniz, P.S.R. *Adaptive Filtering Algorithms and Practical Implementation*, 5th ed.; Springer: New York, NY, USA, 2020.
2. Tan, G.; Yan, S.; Yang, B. Impulsive Noise Suppression and Concatenated Code for OFDM Underwater Acoustic Communications. In Proceedings of the 2022 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), Xi'an, China, 25–27 October 2022; pp. 1–6. [\[CrossRef\]](#)
3. Wang, J.; Li, J.; Yan, S.; Shi, W.; Yang, X.; Guo, Y.; Gulliver, T.A. A Novel Underwater Acoustic Signal Denoising Algorithm for Gaussian/Non-Gaussian Impulsive Noise. *IEEE Trans. Veh. Technol.* **2021**, *70*, 429–445. [\[CrossRef\]](#)
4. Diamant, R. Robust Interference Cancellation of Chirp and CW Signals for Underwater Acoustics Applications. *IEEE Access* **2018**, *6*, 4405–4415. [\[CrossRef\]](#)
5. Ge, F.X.; Zhang, Y.; Li, Z.; Zhang, R. Adaptive Bubble Pulse Cancellation From Underwater Explosive Charge Acoustic Signals. *IEEE J. Oceanic Eng.* **2011**, *36*, 447–453. [\[CrossRef\]](#)
6. Fieve, S.; Portala, P.; Bertel, L. A new VLF/LF atmospheric noise model. *Radio Sci.* **2007**, *42*, 1–14. [\[CrossRef\]](#)
7. Rehman, I.u.; Raza, H.; Razzaq, N.; Frnda, J.; Zaidi, T.; Abbasi, W.; Anwar, M.S. A Computationally Efficient Distributed Framework for a State Space Adaptive Filter for the Removal of PLI from Cardiac Signals. *Mathematics* **2023**, *11*, 350. [\[CrossRef\]](#)
8. Peng, L.; Zang, G.; Gao, Y.; Sha, N.; Xi, C. LMS Adaptive Interference Cancellation in Physical Layer Security Communication System Based on Artificial Interference. In Proceedings of the 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Xi'an, China, 25–27 May 2018; pp. 1178–1182. [\[CrossRef\]](#)
9. Chen, Y.E.; Chien, Y.R.; Tsao, H.W. Chirp-Like Jamming Mitigation for GPS Receivers Using Wavelet-Packet-Transform-Assisted Adaptive Filters. In Proceedings of the 2016 International Computer Symposium (ICS), Chiayi, Taiwan, 15–17 December 2016; pp. 458–461. [\[CrossRef\]](#)
10. Yu, H.C.; Chien, Y.R.; Tsao, H.W. A study of impulsive noise immunity for wavelet-OFDM-based power line communications. In Proceedings of the 2016 International Conference On Communication Problem-Solving (ICCP), Taipei, Taiwan, 7–9 September 2016; pp. 1–2. [\[CrossRef\]](#)
11. Chien, Y.R.; Wu, S.T.; Tsao, H.W.; Diniz, P.S.R. Correntropy-Based Data Selective Adaptive Filtering. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2024**, *71*, 754–766. [\[CrossRef\]](#)
12. Chien, Y.R.; Xu, S.S.D.; Ho, D.Y. Combined boosted variable step-size affine projection sign algorithm for environments with impulsive noise. *Digit. Signal Process.* **2023**, *140*, 104110. [\[CrossRef\]](#)
13. Maleki, N.; Azghani, M. Sparse Mixed Norm Adaptive Filtering Technique. *Circuits Syst. Signal Process.* **2020**, *39*, 5758–5775. [\[CrossRef\]](#)
14. Shao, M.; Nikias, C. Signal processing with fractional lower order moments: Stable processes and their applications. *Proc. IEEE* **1993**, *81*, 986–1010. [\[CrossRef\]](#)
15. Huang, F.; Song, F.; Zhang, S.; So, H.C.; Yang, J. Robust Bias-Compensated LMS Algorithm: Design, Performance Analysis and Applications. *IEEE Trans. Veh. Technol.* **2023**, *72*, 13214–13228. [\[CrossRef\]](#)
16. Danaee, A.; Lamare, R.C.d.; Nascimento, V.H. Distributed Quantization-Aware RLS Learning With Bias Compensation and Coarsely Quantized Signals. *IEEE Trans. Signal Process.* **2022**, *70*, 3441–3455. [\[CrossRef\]](#)
17. Walach, E.; Widrow, B. The least mean fourth (LMF) adaptive algorithm and its family. *IEEE Trans. Inform. Theory* **1984**, *30*, 275–283. [\[CrossRef\]](#)
18. Balasundar, C.; Sundarabalan, C.K.; Santhanam, S.N.; Sharma, J.; Guerrero, J.M. Mixed Step Size Normalized Least Mean Fourth Algorithm of DSTATCOM Integrated Electric Vehicle Charging Station. *IEEE Trans. Ind. Inform.* **2023**, *19*, 7583–7591. [\[CrossRef\]](#)
19. Papoulis, E.; Stathaki, T. A normalized robust mixed-norm adaptive algorithm for system identification. *IEEE Signal Process. Lett.* **2004**, *11*, 56–59. [\[CrossRef\]](#)
20. Zerguine, A. A variable-parameter normalized mixed-norm (VPNMN) adaptive algorithm. *Eurasip J. Adv. Signal Process.* **2012**, *2012*, 55. [\[CrossRef\]](#)
21. Pimenta, R.M.S.; Petraglia, M.R.; Haddad, D.B. Stability analysis of the bias compensated LMS algorithm. *Digital Signal Process.* **2024**, *147*, 104395. [\[CrossRef\]](#)
22. Jung, S.M.; Park, P. Normalised least-mean-square algorithm for adaptive filtering of impulsive measurement noises and noisy inputs. *Electron. Lett.* **2013**, *49*, 1270–1272. [\[CrossRef\]](#)
23. Jin, Z.; Guo, L.; Li, Y. The Bias-Compensated Proportionate NLMS Algorithm With Sparse Penalty Constraint. *IEEE Access* **2020**, *8*, 4954–4962. [\[CrossRef\]](#)
24. Lee, M.; Park, T.; Park, P. Bias-Compensated Normalized Least Mean Fourth Algorithm for Adaptive Filtering of Impulsive Measurement Noises and Noisy Inputs. In Proceedings of the 2019 12th Asian Control Conference (ASCC), Kitakyushu, Japan, 9–12 June 2019; pp. 220–223.
25. Vasundhara. Robust filtering employing bias compensated M-estimate affine-projection-like algorithm. *Electron. Lett.* **2020**, *56*, 241–243. [\[CrossRef\]](#)
26. Lee, M.; Park, I.S.; Park, C.e.; Lee, H.; Park, P. Bias Compensated Least Mean Mixed-norm Adaptive Filtering Algorithm Robust to Impulsive Noises. In Proceedings of the 2020 20th International Conference on Control, Automation and Systems (ICCAS), Busan, Republic of Korea, 13–16 October 2020; pp. 652–657. [\[CrossRef\]](#)

27. Ma, W.; Zheng, D.; Zhang, Z. Bias-compensated generalized mixed norm algorithm via CIM constraints for sparse system identification with noisy input. In Proceedings of the 2017 36th Chinese Control Conference (CCC), Dalian, China, 26–28 July 2017; pp. 5094–5099. [[CrossRef](#)]
28. Yoo, J.; Shin, J.; Park, P. An Improved NLMS Algorithm in Sparse Systems Against Noisy Input Signals. *IEEE Trans. Circuits Syst. Express Briefs* **2015**, *62*, 271–275. [[CrossRef](#)]
29. Liu, C.; Zhao, H. Efficient DOA Estimation Method Using Bias-Compensated Adaptive Filtering. *IEEE Trans. Veh. Technol.* **2020**, *69*, 13087–13097. [[CrossRef](#)]
30. Kim, S.R.; Efron, A. Adaptive robust impulse noise filtering. *IEEE Trans. Signal Process.* **1995**, *43*, 1855–1866. [[CrossRef](#)]
31. Gu, Y.; Jin, J.; Mei, S. l_0 Norm Constraint LMS Algorithm for Sparse System Identification. *IEEE Signal Process. Lett.* **2009**, *16*, 774–777. [[CrossRef](#)]
32. Eweda, E. Global Stabilization of the Least Mean Fourth Algorithm. *IEEE Trans. Signal Process.* **2012**, *60*, 1473–1477. [[CrossRef](#)]
33. Zheng, Z.; Liu, Z.; Zhao, H. Bias-Compensated Normalized Least-Mean Fourth Algorithm for Noisy Input. *Circuits Syst. Signal Process.* **2017**, *36*, 3864–3873. [[CrossRef](#)]
34. Zhao, H.; Zheng, Z. Bias-compensated affine-projection-like algorithms with noisy input. *Electron. Lett.* **2016**, *52*, 712–714. [[CrossRef](#)]
35. Arenas-Garcia, J.; Figueiras-Vidal, A.; Sayed, A. Mean-square performance of a convex combination of two adaptive filters. *IEEE Trans. Signal Process.* **2006**, *54*, 1078–1090. [[CrossRef](#)]
36. Lu, L.; Zhao, H.; Li, K.; Chen, B. A Novel Normalized Sign Algorithm for System Identification Under Impulsive Noise Interference. *Circuits Syst. Signal Process.* **2015**, *35*, 3244–3265. [[CrossRef](#)]
37. Zhou, Y.; Chan, S.C.; Ho, K.L. New Sequential Partial-Update Least Mean M-Estimate Algorithms for Robust Adaptive System Identification in Impulsive Noise. *IEEE Trans. Ind. Electron.* **2011**, *58*, 4455–4470. [[CrossRef](#)]
38. Jung, S.M.; Park, P. Stabilization of a Bias-Compensated Normalized Least-Mean-Square Algorithm for Noisy Inputs. *IEEE Trans. Signal Process.* **2017**, *65*, 2949–2961. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.