Contents lists available at ScienceDirect

Journal of Automation and Intelligence

Research article

Generalized spline adaptive filtering algorithm based on q-hyperbolic function

Shiwei Yun [a,b], Sihai Guan [a,b,*], Chuanwu Zhang [a,b], Bharat Biswal [c]

[a] *College of Electronic and Information, Southwest Minzu University, Chengdu, 610041, China*
[b] *Key Laboratory of Electronic and Information Engineering, State Ethnic Affairs Commission, Chengdu, 610041, China*
[c] *Department of Biomedical Engineering, New Jersey Institute of Technology (NJIT), Newark, NJ, 07102, USA*

ARTICLE INFO

Keywords:
Nonlinear systems
Spline adaptive filtering
q-deformed hyperbolic functions

ABSTRACT

Based on the superiority of adaptive filtering algorithms designed with hyperbolic function-like objective functions, this paper proposes generalized spline adaptive filtering (SAF) algorithms designed with hyperbolic function-like objective functions. Specifically, a series of generalized new SAF algorithms are proposed by introducing the q-deformed hyperbolic function as the cost function, named SAF-qDHSI, SAF-qDHCO, SAF-qDHTA & SAF-qDHSE algorithms, respectively. Then, the proposed algorithm is theoretically demonstrated with detailed mean convergence and computational complexity analysis; secondly, the effect of different q values on the performance of the new algorithm is verified through data simulation; the new algorithm still has better performance under the interference of Gaussian noise and non-Gaussian noise even when facing the system mutation; finally, the new algorithm is verified through the measured engineering data, and the results show that the new algorithm has better convergence and robustness compared with the existing algorithm. In conclusion, the generalized algorithm based on the new cost function proposed in this paper is more effective in nonlinear system identification.

1. Introduction

With the advancement of technology, adaptive filters have gained widespread application [1]. These filters, categorized into linear and nonlinear types based on the relationship between input and output signals [2], face limitations in practical engineering scenarios where systems often exhibit nonlinear characteristics [3]. Traditional linear models fail to accurately capture these nonlinearities, significantly diminishing their effectiveness. Consequently, the study of nonlinear systems is both essential and critical.

Various methods have been developed for nonlinear system identification, including orthogonal functions [4], neural networks (NN) [5], Volterra series [6,7], TS-fuzzy models [8], NARMAX models [9], support vector machines (SVMs) [10], and function link adaptive filters (FLAF) [11]. Nonlinear adaptive filtering (NLAF) [12] is a potent tool for implementing NLAF, though it often entails high computational costs and demands sophisticated data processing capabilities from experimental equipment [13]. In contrast, spline adaptive filtering (SAF) has garnered significant attention due to its robust modeling capabilities and relatively low computational complexity [14]. Building on block-oriented nonlinear structures [15], Comminiello et al. introduced the SAF least mean square (SAF-LMS) algorithm. Further, an adaptive nonlinear Hammerstein filter based on uniform cubic spline functions was proposed in [16]. Besides, SAF-LMS, characterized by a finite impulse response (FIR) linear part and an adaptive lookup table with local low-order spline interpolation curves, can be classified into Wiener, Sandwich, and Hammerstein types based on nonlinearization order. Then, Guan et al. enhanced the stability of the SAF algorithm by proposing SAF-NLMS [17].

Nonlinear system identification remains a pivotal research area. Liu et al. [18] developed a nonlinear SAF based on the Geman–McClure estimator, eliminating large-amplitude outliers and enhancing interference immunity. Yu et al. [19] combined a novel cost function with the Nesterov accelerated gradient (MNAG), resulting in the SAF-LHC-MNAG algorithm exhibiting superior convergence properties. The q-gradient LMS spline adaptive filtering algorithm addresses the slow convergence issue by reducing the over-reliance on LMS-type algorithms [20]. In the realm of linear backscatter filtering, a nonlinear backscattering adaptive filter was developed [21]. A Wiener SAF-based generalized maximal sux criterion algorithm (GMVC) was introduced in [22], incorporating gradient descent (MSGD) and a variable step size algorithm, SAF-GMVC-VMSGD. Researchers have also proposed the optimized adaptive filtering algorithm (SPOAF) by employing dual cost functions for linear and nonlinear parts, effectively tackling colored noise environments and high steady-state issues [23]. Given the importance of anti-jamming in radar and wireless communication, the optimized adaptive filter (SPOAF-MVC) was designed [24]. A metaheuristic algorithm for Hammerstein SAF enables simultaneous updates of spline control points and linear filter weights [25]. Guan et al. introduced SAF-Adagrad and SAF-RMSprop by integrating various iterative gradient methods, proving highly effective in nonlinear system identification [26]. The frequency domain spline adaptive Fourier algorithm based on the semi-quadratic criterion, named FDSAF-HQC, excels in echo cancellation and biosignal denoising [27].

Extensive research has been conducted on the SAF-LMS algorithm, with numerous enhanced algorithms developed to address nonlinear system identification challenges through appropriate cost functions. For instance, Yang et al. [28] proposed the SAF-ARCMMSGD algorithm, utilizing an inverse tangent function insensitive to large outliers and the stochastic gradient descent method (MMSGD), enhancing the interference resistance of spline filtering algorithms. Lu et al. [29] introduced a new LMS algorithm for transient analysis based on Taylor expansion and energy conservation relationships, demonstrating superior anti-jamming capabilities compared to traditional LMS methods. Bhattacharjee et al. [30] employed a generalized maximum entropy criterion (GMCC) based robust adaptive processing algorithm with a generalized hyperbolic secant function, offering more robust system identification than traditional GMCC algorithms. Liu et al. [31] derived the lncosh algorithm using a lncosh cost function, which exhibits good robustness in impulsive noise environments, with the introduced variable λLlncosh scheme achieving a balance between convergence rate and steady-state performance. A spline adaptive transformation algorithm using a hyperbolic tangent function as the cost function, combined with a pre-triggered observation strategy, further addresses these issues [32]. Patel et al. [33] proposed a SAF using a logarithmic, hyperbolic cosine function, incorporating variable parameters to automatically ignore larger and smaller error signal values. The modified Huber function used in the MAPSAF algorithm demonstrates superior resistance to shock interference, leading to the development of the CSS-MPSAF algorithm [34]. Based on the Heaviside step function (HSF), Guan et al. introduced four new algorithms: SAF-HSF-sigmoid, SAF-HSF-erfc, SAF-HSF-atan, and SAF-HSF-tanh [35]. An enhanced SPOAF algorithm based on the generalized hyperbolic secant function was proposed in [36].

From the above exposition, it is evident that the design of SAF-LMS class algorithms based on hyperbolic functions has become a highly active area of research. Then, is it possible to propose a design framework that includes the above design of SAF-LMS class algorithms based on hyperbolic functions? This is a fascinating area of research. The article [37,38] mentioned the q-deformation hyperbolic function. When $q = 1$, the function is the traditional hyperbolic function, so this paper adopts the q-deformed hyperbolic function as a new cost function, introduces the q parameter based on the existing hyperbolic function, and combines it with the traditional SAF algorithm, and obtains the new SAF algorithm through theoretical derivation. The simulation results demonstrate that the novel algorithm exhibits superior performance in comparison to the existing five algorithms. In terms of convergence speed and steady-state characteristics, the SAF-LMS, SAF-NLMS, SAF-MCC, SAF-VSSNLMS, and SAF-ARCMMSGD algorithms were compared.

The following is a description of the structure of the paper: Section 2 provides a detailed description of the traditional SAF algorithm. Based on this description, a new SAF algorithm with a modified cost function is proposed in Section 3. Section 4 outlines the methodology employed to compare and analyze the performance of the two algorithms. Finally, Section 5 offers a concluding summary of the paper.

2. Proposed algorithm

2.1. Structure of SAF

SAF is a linear–nonlinear combinatorial filter, where the dynamic linear part is an FIR filter and the static nonlinear part is a cubic spline mechanism, including an adaptive lookup table and cubic spline interpolation. Fig. 1 shows the structural framework of the SAF algorithm. SAF is a new class of Wiener models customized for LN blocks [39] and belongs to causal shift recursive nonlinear filters (such as VAF, GFLN, and KAF). Unlike recent approaches where specific and fixed nonlinearities have to be chosen, the proposed architecture consists of a linear combiner and an adaptive lookup table (LUT). The LUT is processed by the outputs of the linear combiner and interpolated using local low-order polynomial spline curves. Both the weights of the linear filter and the interpolation points of the LUT can be adjusted by minimizing a specified cost function.

Assuming that the system input at the moment n is $\boldsymbol{x}(n)$, the input vector of the FIR filter is then defined as $\boldsymbol{x}(n)=[x(n)\ x(n\text{-}1)\ldots x(n\text{-}M+1)]^{\mathrm{T}}$. The intermediate output signal of the FIR, $s(n)$, can be defined as:

$$s(n) = \boldsymbol{\omega}(n)^{\mathrm{T}} \cdot \boldsymbol{x}(n) \tag{1}$$

The weight vector of the FIR filter at moment n can be defined as $\boldsymbol{\omega}(n)=[\omega_0\ \omega_1\ldots\omega_{M-1}]^{\mathrm{T}}$. After that, the intermediate output signal $s(n)$ enters the cubic interpolation mechanism and is processed to obtain the final output signal $y(n)$. The Catmull–Rom spline curve (CR spline curve) has an excellent local interpolation capability, which can pass through all control points except the first and last two control points. Because it is represented as a matrix, it is very easy to calculate. We can get i and u by the following formula:

$$i = [\frac{s(n)}{\Delta x}] + \frac{N}{2} \tag{2}$$

$$u = \frac{s(n)}{\Delta x} - [\frac{s(n)}{\Delta x}] \tag{3}$$

where $[\cdot]$ denotes the lower bound operator, Δx is the equidistant interval between two points, and u is the normalized horizontal coordinate between $q_{x,i}$ and $q_{x,i+1}$. After that, by calculating the four neighboring nodes of the i interval, we can get the output:

$$y(n) = \boldsymbol{u}^{\mathrm{T}}(n) \cdot \boldsymbol{C} \cdot \boldsymbol{q}_i(n) \tag{4}$$

where $\boldsymbol{u}(n)=[u^3\ u^2\ u\ 1]^{\mathrm{T}}$, $\boldsymbol{q}_i(n)$ denotes the neighborhood node vector, which can be expressed as $\boldsymbol{q}_i(n)=[q_{y,i-1}\ q_{y,i}\ q_{y,i+1}\ q_{y,i+2}]$, and $\boldsymbol{C}$ is the spline matrix, which can be defined as:

$$\boldsymbol{C} = \frac{1}{2}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \tag{5}$$

To facilitate the derivation of the algorithm that follows, a partial derivation of the output signal $y(n)$ concerning u, we obtain:

$$\frac{\partial y(n)}{\partial \boldsymbol{u}} = \dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n) \tag{6}$$

where $\dot{\boldsymbol{u}}(n)=[3u^2\ 2u\ 1\ 0]^{\mathrm{T}}$ denotes the derivation of $\boldsymbol{u}(n)$.

The adaptive iterative strategy of the conventional SAF is obtained by minimizing the instantaneous squared error, expressed by Eq. (7):

$$J(\boldsymbol{\omega}(n), \boldsymbol{q}_i(n)) = e^2(n) \tag{7}$$

The error signal is denoted as $e(n)$=$d(n)$-$y(n)$. The partial derivatives of $J(\boldsymbol{\omega}(n),\ \boldsymbol{q}_i(n))$ with respect to $\boldsymbol{\omega}(n)$ and $\boldsymbol{q}_i(n)$, respectively, are obtained:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \mu_w \frac{e(n)\dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)}{\Delta x} \tag{8}$$

$$\boldsymbol{q}_i(n+1) = \boldsymbol{q}_i(n) + \mu_q \cdot e(n) \cdot \boldsymbol{C}^{\mathrm{T}} \cdot \boldsymbol{u}(n) \tag{9}$$

where μ_ω and μ_q represent the learning rate of the two, respectively, and this iterative method is also known as SAF-LMS [39].

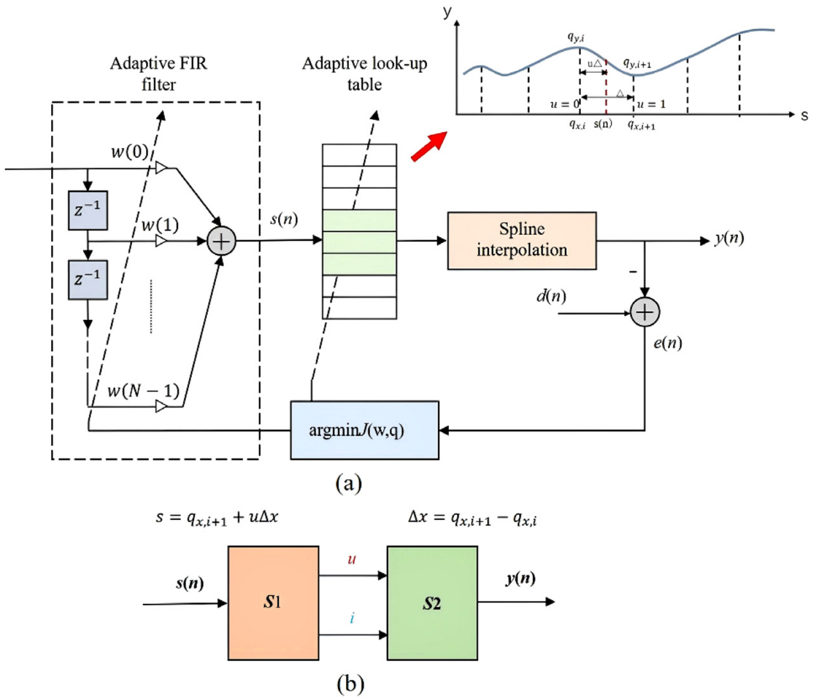


Fig. 1. The structure of SAF is described by (a) and (b).

2.2. q-deformed cost function

The new algorithm employs the q-deformed hyperbolic function as a cost function, offering a significant improvement over the traditional approach. When $q = 1$, the algorithm is equivalent to the traditional algorithm with ordinary hyperbolic function as the cost function; when $q \neq 1$, the algorithm is the improved algorithm of this paper and is accompanied by different performance for different values of q. Among them, the q-deformed hyperbolic function is introduced as follows [38], where q is the deformation parameter. The underlying principle is to apply a similar formula that depends on the positive parameter in place of the parameterless hyperbolic function. The transformed function's characteristics are conducive to facilitating the research of the filtering algorithm.

$$\begin{aligned} &\sinh_q x = \frac{e^x - qe^{-x}}{2}, \cosh_q x = \frac{e^x + qe^{-x}}{2} \\ &\tanh_q x = \frac{\sinh_q x}{\cosh_q x}, \mathrm{sech}_q x = \frac{1}{\cosh_q x}; \\ &x \in R, q \in (0, +\infty) \end{aligned} \tag{10}$$

when $q = 1$, the functions (11) are all odd, and the functions (12) are all even functions:

$$\begin{aligned} &\sinh_q(-x) = -\sinh_q x \\ &\tanh_q(-x) = -\tanh_q x; x \in R \end{aligned} \tag{11}$$

$$\begin{aligned} &\cosh_q x = \cosh_q(-x) \\ &\mathrm{sech}_q x = \mathrm{sech}_q(-x); x \in R \end{aligned} \tag{12}$$

It is worth noting that when $q \neq 1$:

$$\begin{aligned} \sinh_q(-x) &= -q \sinh_{\frac{1}{q}} x, \cosh_q(-x) = q \cosh_{\frac{1}{q}} x \\ \tanh_q(-x) &= -\tanh_{\frac{1}{q}} x, \mathrm{sech}_q(-x) = \frac{1}{q} \mathrm{sech}_{\frac{1}{q}} x; x \in R \end{aligned} \tag{13}$$

2.3. Proposed SAF-qD-HS algorithm

As shown in Fig. 2, to improve the convergence of the SAF-LMS algorithm, four cost functions are proposed, named SAF-qDHSI, SAF-qDHCO, SAF-qDHTA, and SAF-qDHSE. It can be observed that as the error term approaches infinity, the first derivative of the proposed cost function approaches 0. It can be determined that the cost function is insensitive to large disturbances in the error, indicating the algorithm's robustness.

Meanwhile, according to Eq. (10), we design four new cost functions:

$$J(\boldsymbol{\omega}(n), \boldsymbol{q}_i(n)) = \frac{\exp\left[e^2(n)\right] - q\exp\left[-e^2(n)\right]}{2} \tag{14}$$

$$J(\boldsymbol{\omega}(n), \boldsymbol{q}_i(n)) = \frac{\exp\left[e^2(n)\right] + q\exp\left[-e^2(n)\right]}{2} \tag{15}$$

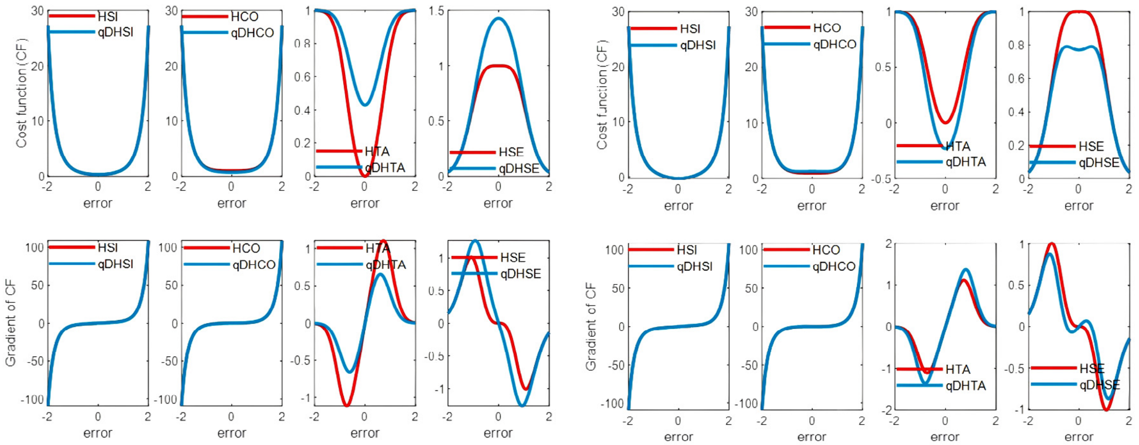
$$J(\boldsymbol{\omega}(n), \boldsymbol{q}_i(n)) = \frac{\exp\left[e^2(n)\right] - q\exp\left[-e^2(n)\right]}{\exp\left[e^2(n)\right] + q\exp\left[-e^2(n)\right]} \tag{16}$$

$$J(\boldsymbol{\omega}(n), \boldsymbol{q}_i(n)) = \frac{2}{\exp\left[e^2(n)\right] + q\exp\left[-e^2(n)\right]} \tag{17}$$

The proposed algorithms are summarized as shown in Table 1. It is noteworthy that SAF-qDHSI, SAF-qDHCO, SAF-qDHTA, and SAF-qDHSE represent the SAF algorithms designed based on the four hyperbolic functions of sinh, cosh, tanh, and sech as the objective functions when $q = 1$, respectively.

3. Performances of the proposed algorithm

This section details the convergence performance and computational complexity of proving the above four algorithms (SAF-qDHSI, SAF-qDHCO, SAF-qDHTA, SAF-qDHSE). The convergence of the four algorithms proposed above in identifying Wiener-type nonlinear systems will be analyzed next. The cost functions (14), (15), (16), and (17) depend on two coupled variables $\boldsymbol{\omega}(n)$, $\boldsymbol{q}_i(n)$ such that $\lim_{n\to\infty} E[\boldsymbol{\omega}(n)] = \boldsymbol{\omega}^0$ and $\lim_{n\to\infty} E[\boldsymbol{q}(n)] = \boldsymbol{q}^0$, where $\boldsymbol{\omega}_0$ and $\boldsymbol{q}_0$ are the weight vectors of the system. We determine whether the algorithm converges or not, as well as determine the value of the learning rate by the condition $|\, e(n+1)| < |e(n)|$. The adaptive process of the algorithm proposed in this paper can be realized in two stages, $\boldsymbol{\omega}(n)$ and $\boldsymbol{q}_i(n)$, respectively, so the analysis of convergence is divided into two parts.



(a)

(b)

Fig. 2. Cost function and gradient variation: (a) the q values are set to 0.4; (b) the q values are set to 1.6.

Table 1
Algorithm summary.

```
Initialize: $\omega(n)$, $\boldsymbol{q}_i(n)$, $\mu_\omega$, $\mu_q$
for $n$=1,2...
do
    $s(n)=\boldsymbol{\omega}^{\mathrm{T}}(n)\boldsymbol{x}(n)$
    $\boldsymbol{q}i(n)=[q_{y,i-1}\ q_{y,i}\ q_{y,i+1}\ q_{y,i+2}]$
    $\boldsymbol{u}(n)=[u^3\ u^2\ u\ 1]^{\mathrm{T}}$
    $y(n)=\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}q_i(n)$
    $e(n)=d(n)$-$y(n)$
    SAF-qDHSI:
        $\boldsymbol{g}_{\omega(n)}=-e(n)\left\{\exp\left[e^2(n)\right]+q\exp\left[-e^2(n)\right]\right\}\dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)$   (18)
        $\boldsymbol{g}_{\boldsymbol{q}i(n)}=-e(n)\left\{\exp\left[e^2(n)\right]+q\exp\left[-e^2(n)\right]\right\}\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)$   (19)
    SAF-qDHCO:
        $\boldsymbol{g}_{\omega(n)}=-e(n)\left\{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]\right\}\dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)$   (20)
        $\boldsymbol{g}_{\boldsymbol{q}i(n)}=-e(n)\left\{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]\right\}\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)$   (21)
    SAF-qDHTA:
        $\boldsymbol{g}_{\omega(n)}=-2e(n)\left\{1-\left[\frac{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]}{\exp\left[e^2(n)\right]+q\exp\left[-e^2(n)\right]}\right]^2\right\}\dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)$   (22)
        $\boldsymbol{g}_{\omega(n)}=-2e(n)\left\{1-\left[\frac{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]}{\exp\left[e^2(n)\right]+q\exp\left[-e^2(n)\right]}\right]^2\right\}\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)$   (23)
    SAF-qDHSE:
        $\boldsymbol{g}_{\omega(n)}=4e(n)\frac{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]}{\left\{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]\right\}^2}\dot{\boldsymbol{u}}(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)$   (24)
        $\boldsymbol{g}_{q_i}(n)=4e(n)\frac{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]}{\left\{\exp\left[e^2(n)\right]-q\exp\left[-e^2(n)\right]\right\}^2}\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)$   (25)
end
```

3.1. SAF-qDHSI

Firstly, the Taylor series expansion of the error signal $e(n+1)$ is obtained from the following equation:

$$e(n+1)=e(n)+\frac{\partial e(n)}{\partial\boldsymbol{\omega}^{\mathrm{T}}(n)}\Delta\boldsymbol{\omega}(n)+h.o.t \tag{26}$$

where $h.o.t$ is the higher-order term of the Taylor series expansion, $\frac{\partial e(n)}{\partial\boldsymbol{\omega}^{\mathrm{T}}(n)}=-\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)$ and we get $\Delta\boldsymbol{\omega}(n)$=$\boldsymbol{\omega}(n+1)$−$\boldsymbol{\omega}(n)$ by Eq. (8), thus:

$$\Delta\omega(n)=-\mu_\omega g_\omega(n) \tag{27}$$

Substituting Eq. (27) into (26) yields:

$$\begin{aligned}
&e(n+1)\\
&=e(n)+\mu_\omega\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)g_{\boldsymbol{\omega}}(n)\\
&=e(n)-\mu_{\boldsymbol{\omega}}\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\boldsymbol{x}(n)e(n)\left\{e^{\left[e^2(n)\right]}+qe^{\left[-e^2(n)\right]}\right\}\boldsymbol{u}^{\mathrm{T}}(n)\cdot\boldsymbol{C}\cdot\boldsymbol{q}_i(n)\\
&=e(n)\left\{1-\mu_{\boldsymbol{\omega}}\left[\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\right]^2\|\boldsymbol{x}(n)\|^2\left[e^{\left[e^2(n)\right]}+qe^{\left[-e^2(n)\right]}\right]\right\}
\end{aligned} \tag{28}$$

To ensure the convergence performance of the algorithm, we specify that the paradigm of the error cannot be larger than the paradigm of the right-hand side of the equation:

$$|e(n+1)| \leq |e(n)| \left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \boldsymbol{x}(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{[e^2(n)]} + q e^{[-e^2(n)]} \right] \right\} \right| \tag{29}$$

To ensure that the above equation holds, the following relationship can be assumed:

$$\left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{[e^2(n)]} + q e^{[-e^2(n)]} \right] \right\} \right| \leq 1 \tag{30}$$

We can obtain a range of values for the learning rate $\mu_{\boldsymbol{\omega}}$:

$$0 < \mu_\omega \leq \frac{\operatorname{sech}_q \left[e^2(n) \right]}{\left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q} i(n) \right]^2 \|\boldsymbol{x}(n)\|^2} \tag{31}$$

when $q = 1$, at this point:

$$0 < \mu_\omega \leq \frac{2}{\left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q} i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{e^2(n)} + e^{-e^2(n)} \right]} \tag{32}$$

Using the same approach, we can obtain a range of values for μ_q:

$$e(n+1) = e(n) + \frac{\partial e(n)}{\partial \boldsymbol{q}_i(n)} \Delta \boldsymbol{q}_i(n) + h.o.t \tag{33}$$

where $h.o.t$ is the higher-order term of the Taylor series expansion, $\frac{\partial e(n)}{\partial \boldsymbol{q}_i(n)} = -\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)$, $\Delta\boldsymbol{q}_i\ (n)=\boldsymbol{q}_i(n+1)-\boldsymbol{q}_i(n)$ thus:

$$\Delta \boldsymbol{q}_i(n) = \mu_{\boldsymbol{q}} g_{\boldsymbol{q} i(n)} \tag{34}$$

Substituting Eq. (34) into Eq. (33), we get:

$$\begin{aligned} & e(n+1) = e(n) + \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) g_{\boldsymbol{q} i(n)} \\ & = e(n) - \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) e(n) \left\{ e^{[e^2(n)]} + q e^{[-e^2(n)]} \right\} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \\ & = e(n) \left\{ 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} + q e^{[-e^2(n)]} \right] \right\} \end{aligned} \tag{35}$$

Again, we stipulate that the paradigm of the error is smaller than the paradigm on the right-hand side of the equation:

$$|e(n+1)| \leq |e(n)| \left| \left\{ 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} + q e^{[-e^2(n)]} \right] \right\} \right| \tag{36}$$

If the following relation holds:

$$\left| 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} + q e^{[-e^2(n)]} \right] \right| \leq 1 \tag{37}$$

We can then obtain bounds on μ_q:

$$0 < \mu_{\boldsymbol{q}} \leq \frac{\operatorname{sech}_q e^2(n)}{\left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2} \tag{38}$$

when $q = 1$, at this point:

$$0 < \mu_{\boldsymbol{q}} \leq \frac{2}{\left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{e^2(n)} + e^{-e^2(n)} \right]} \tag{39}$$

3.2. SAF-qDHCO

Analyzing the process as above, we get:

$$\begin{aligned} & e\,(n+1) \\ & = e\,(n) + \mu_w \boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n)\, \boldsymbol{x}\,(n)\, g_w\,(n) \\ & = e\,(n) \left\{ 1 - \mu_w \left[\boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n) \right]^2 \|\boldsymbol{x}\,(n)\|^{\mathbf{2}} \left[e^{e^2(n)} - q e^{-e^2(n)} \right] \right\} \end{aligned} \tag{40}$$

To ensure the convergence performance of the algorithm, we specify that the paradigm of the error cannot be larger than the paradigm of the right-hand side of the equation:

$$|e(n+1)| \leq |e(n)| \left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{[e^2(n)]} - q e^{[-e^2(n)]} \right] \right\} \right| \tag{41}$$

To ensure that the above equation holds, the following relationship can be assumed:

$$\left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{[e^2(n)]} - q e^{[-e^2(n)]} \right] \right\} \right| \leq 1 \tag{42}$$

We can obtain a range of values for the learning rate $\mu_{\boldsymbol{\omega}}$:

$$0 < \mu_\omega \leq \frac{1}{\left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \sinh_q e^2(n)} \tag{43}$$

when $q = 1$, at this point:

$$0 < \mu_\omega \leq \frac{\left[e^{e^2(n)} + e^{-e^2(n)} \right]}{2 \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2} \tag{44}$$

Using the same approach, we can obtain a range of values for μ_q:

$$\begin{aligned} & e(n+1) = e(n) + \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) g_{\boldsymbol{q} i(n)} \\ & = e(n) - \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) e(n) \left\{ e^{[e^2(n)]} - q e^{[-e^2(n)]} \right\} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \\ & = e(n) \left\{ 1 - \mathrm{\mu_q} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} - q e^{[-e^2(n)]} \right] \right\} \end{aligned} \tag{45}$$

Again, we stipulate that the paradigm of the error is smaller than the paradigm on the right-hand side of the equation:

$$|e(n+1)| \leq |e(n)| \left| \left\{ 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} - q e^{[-e^2(n)]} \right] \right\} \right| \tag{46}$$

If the following relation holds:

$$\left| 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[e^{[e^2(n)]} - q e^{[-e^2(n)]} \right] \right| \leq 1 \tag{47}$$

$$0 < \mu_{\boldsymbol{q}} \leq \frac{1}{\left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \sinh_q e^2(n)} \tag{48}$$

$$0 < \mu_{\boldsymbol{q}} \leq \frac{\left[e^{e^2(n)} + e^{-e^2(n)} \right]}{2 \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2} \tag{49}$$

3.3. SAF-qDHTA

Analyzing the process as above, we get:

$$\begin{aligned} & e\,(n+1) \\ & = e\,(n) + \mu_w \boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n)\, \boldsymbol{x}\,(n)\, g_w\,(n) \\ & = e\,(n) \left\{ 1 - \mu_w \left[\boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n) \right]^2 \|\boldsymbol{x}\,(n)\|^{\mathbf{2}} \left[1 - \tanh_{\boldsymbol{q}} e^2(n) \right]^{\mathbf{2}} \right\} \end{aligned} \tag{50}$$

To ensure the convergence performance of the algorithm, we specify that the paradigm of the error cannot be larger than the paradigm of the right-hand side of the equation:

$$|e(n+1)| \leq |e(n)| \left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \right\} \left[1 - \tanh_q e^2(n) \right]^2 \right| \tag{51}$$

To ensure that the above equation holds, the following relationship can be assumed:

$$\left| \left\{ 1 - \mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[1 - \tanh_q e^2(n) \right]^2 \right\} \right| \leq 1 \tag{52}$$

We can obtain a range of values for the learning rate $\mu_{\boldsymbol{\omega}}$:

$$0 < \mu_\omega \leq \frac{1}{\left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \|\boldsymbol{x}(n)\|^2 \left[1 - \tanh_q e^2(n) \right]} \tag{53}$$

when $q = 1$, at this point:

$$0 < \mu_\omega \leq \frac{2 e^{e^2(n)}}{\left[\boldsymbol{u}^{\mathrm{T}}(n) \boldsymbol{C} \boldsymbol{q}_i(n) \right]^2 \left[e^{e^2(n)} + e^{-e^2(n)} \right]} \tag{54}$$

Using the same approach, we can obtain a range of values for μ_q:

$$\begin{aligned} & e(n+1) = e(n) + \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) g_{q i(n)} \\ & = e(n) - \mu_{\boldsymbol{q}} \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) e(n) \left[1 - \tanh_q e^2(n) \right]^2 \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \\ & = e(n) \left\{ 1 - \mu_{\boldsymbol{q}} \left\| \boldsymbol{C}^{\mathrm{T}}(n) \boldsymbol{u}(n) \right\|^2 \left[1 - \tanh_q e^2(n) \right]^2 \right\} \end{aligned} \tag{55}$$

Again, we stipulate that the paradigm of the error is smaller than the paradigm on the right-hand side of the equation:

$$|e(n+1)| \le |e(n)| \left| \left\{ 1 - \mu_q \left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \left[1 - \tanh_q e^2(n)\right]^2 \right\} \right| \tag{56}$$

If the following relation holds:

$$\left| 1 - \mu_q \left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \left[1 - \tanh_q e^2(n)\right]^2 \right| \le 1 \tag{57}$$

We can then obtain bounds on μ_q:

$$0 < \mu_q \le \frac{1}{\left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \left[1 - \tanh_q e^2(n)\right]^2} \tag{58}$$

when $q = 1$, at this point:

$$0 < \mu_q \le \frac{2e^{e^2(n)}}{\left[\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)\right]^2 \left[e^{e^2(n)} + e^{-e^2(n)}\right]} \tag{59}$$

3.4. SAF-qDHSE

Analyzing the process as above, we get:

$$\begin{aligned} &e\,(n+1) \\ &= e\,(n) + \mu_w \boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n)\,\boldsymbol{x}\,(n)\,g_w\,(n) \\ &= e\,(n) \left\{ 1 + 4\mu_w \left[\boldsymbol{u}\,(n)^{\mathrm{T}} \boldsymbol{C} \boldsymbol{q}_i\,(n)\right]^2 \|\boldsymbol{x}\,(n)\|^{\mathbf{2}} \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^{\mathbf{2}}} \right\} \end{aligned} \tag{60}$$

To ensure the convergence performance of the algorithm, we specify that the paradigm of the error cannot be larger than the paradigm of the right-hand side of the equation:

$$|e(n+1)| \le |e(n)| \left| \left\{ 1 + 4\mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\right]^2 \|\boldsymbol{x}(n)\|^2 \right\} \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \right| \tag{61}$$

To ensure that the above equation holds, the following relationship can be assumed:

$$\left| \left\{ 1 + 4\mu_\omega \left[\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\right]^2 \|\boldsymbol{x}(n)\|^2 \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \right\} \right| \le 1 \tag{62}$$

We can obtain a range of values for the learning rate $\mu_{\boldsymbol{\omega}}$:

$$0 < \mu_\omega \le \frac{\left[\cosh_q e^2(n)\right]^2}{2\left[\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\right]^2 \|\boldsymbol{x}(n)\|^2 \sinh_q e^2(n)} \tag{63}$$

when $q = 1$, at this point:

$$0 < \mu_\omega \le \frac{\left[e^{e^2(n)} + e^{-e^2(n)}\right]^2}{4\left[\boldsymbol{u}^{\mathrm{T}}(n)\boldsymbol{C}\boldsymbol{q}_i(n)\right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{e^2(n)} - e^{-e^2(n)}\right]} \tag{64}$$

Using the same approach, we can obtain a range of values for μ_q:

$$\begin{aligned} e(n+1) &= e(n) + \mu_q \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) g_{qi(n)} \\ &= e(n) + \mu_q \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) e(n) \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \\ &= e(n) \left\{ 1 + 4\mu_q \left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \right\} \end{aligned} \tag{65}$$

Again, we stipulate that the paradigm of the error is smaller than the paradigm on the right-hand side of the equation:

$$|e(n+1)| \le |e(n)| \left| \left\{ 1 + 4\mu_q \left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \right\} \right| \tag{66}$$

If the following relation holds:

$$\left| 1 + 4\mu_q \left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \frac{\sinh_q e^2(n)}{\left[\cosh_q e^2(n)\right]^2} \right| \le 1 \tag{67}$$

We can then obtain bounds on μ_q:

$$0 < \mu_q \le \frac{\left[\cosh_q e^2(n)\right]^2}{2\left\| \boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n) \right\|^2 \sinh_q e^2(n)} \tag{68}$$

when $q = 1$, at this point:

$$0 < \mu_q \le \frac{\left[e^{e^2(n)} + e^{-e^2(n)}\right]^2}{4\left[\boldsymbol{C}^{\mathrm{T}}(n)\boldsymbol{u}(n)\right]^2 \|\boldsymbol{x}(n)\|^2 \left[e^{e^2(n)} - e^{-e^2(n)}\right]} \tag{69}$$

3.5. Computational complexity

The computational complexity of the adaptive filtering algorithm is defined as the number of iterative arithmetic operations, such as addition, multiplication, division, and so forth, performed on the weight vector at each iteration. However, the multiplication operation is considerably more complex than addition and thus occupies a greater proportion of the computational complexity of the filtering algorithm. This is also true of exponential functions. During each iteration, the $\boldsymbol{u}\,(n)^{\mathrm{T}}\boldsymbol{C}\boldsymbol{q}_i\,(n)$ and $\boldsymbol{C}^{\mathrm{T}}\boldsymbol{u}\,(n)$ need to be multiplied $4K_M$ times using the past operations repeatedly, where $4K_M$ is a constant whose size depends on the spline structure in practice. The SAF-qDHSI algorithm and the SAF-qDHCO algorithm exhibit low computational complexity, whereas the SAF-qDHTA algorithm and the SAF-qDHSE algorithm demonstrate high computational complexity. However, the distinction between these two groups is not substantial in comparison to the complexity of the other algorithms. For details, please refer to Table 2.

Table 2
Computational complexity of algorithms.

Algorithms	Multiplication	exp (·)
SAF-LHC	$4K_M$ +14	1
SAF-ARC-EHC	$4K_M$ +24	14
SAF-ARC-MVC	$4K_M$ +34	–
SAF-GMCC	$4K_M$ +16	2
SAF-HSF-Tanh	$4K_M$ +12	16
SAF-qDHSI	$4K_M$ +10	4
SAF-qDHCO	$4K_M$ +10	4
SAF-qDHTA	$4K_M$ +34	16
SAF-qDHSE	$4K_M$ +26	12

4. Experimental and results

This section presents a three-part division of the experiment to provide a detailed description of the performance of the proposed algorithm. The initial phase of the experiment serves to ascertain the impact of varying values of parameter q on the convergence rate of the four novel algorithms. In the second part, the new algorithm is compared with SAF-LHC [33], SAF-ARC-EHC [22], SAF-ARC-MVC [24], SAF-GMCC [40], SAF-HSV-TANH [35], and other algorithms in the application of nonlinear system identification in different noise environments. In the third part of the experiment, the measured engineering data further proves the superiority of the proposed algorithm.

To validate the effectiveness of the proposed algorithm, several numerical simulations are conducted to compare the convergence rate of the proposed algorithm with the steady state error in different noise environments. Where M is the length of the weight vector $\boldsymbol{\omega}$, the coefficient vector is initialized to zero. The input signal $\boldsymbol{x}_{\mathrm{uc}}(n)$ is a Gaussian white noise with zero mean and $\sigma_{\mathrm{x}}^2 = 1$. The correlated input signal $y(n)$ is calculated by using $y(n) = a \times y(n-1) + \boldsymbol{x}_{\mathrm{uc}}(n-1)$. We compare the performance of these SAF algorithms by the steady-state mean square error, in all cases, the system vector is initialized to $\mathbf{0}$.

The number of independent Monte Carlo numbers is 10. In the first two sections, the step size is set to μ_{ω}=μ_q=0.005 and in the measured data section, the step size is set to μ_{ω}=μ_q=0.001.

4.1. Effect of q-value

The number of independent Monte Carlo simulations is 10, with 2000 iterations conducted at each stage. The step size μ_{ω}=μ_q=0.005 was employed. The optimal value is determined by observing the convergence performance of the proposed new algorithm after modifying the value of q. The effect on the improved algorithm when q takes different values, is illustrated in Fig. 3. The values of q are set as [0.2, 0.4, 0.8, 1.0, 1.6, 2, 2.4, 3]. Where (a), (b), and (c) describe the changes in the convergence of the algorithm SAF-qDHSI, the algorithm SAF-qDHCO and the algorithm SAF-qDHSE, respectively, produced by a change in the q value, algorithm SAF-qDHTA is not shown because of its performance divergence. When q is set to 1, the improved algorithm's cost function is equivalent to the traditional hyperbolic function. The results demonstrate that modifying the value of q enhances the performance of the algorithms, representing a significant contribution to this work. Mathematical derivation allows us to identify the optimal value of q; however, the derivation process is inherently tedious and, therefore, not described in detail here. The results in this paper are the optimal values obtained by simulating the experiments and adjusting the parameters [41].

4.2. Different types of measurement noise effects

This section is divided into two parts for verification through simulation. The first of these, designated **Case 1**, establishes a white noise environment, while the second, designated **Case 2**, establishes a colored noise environment. Concurrently, when independent simulation experiments are conducted in each environment, the input signal is divided into two groups: one with the same input signal and another with a different input signal. The independent Monte Carlo number is 10, and the value of the algorithm q is set to 0.4.

Case 1: White noise

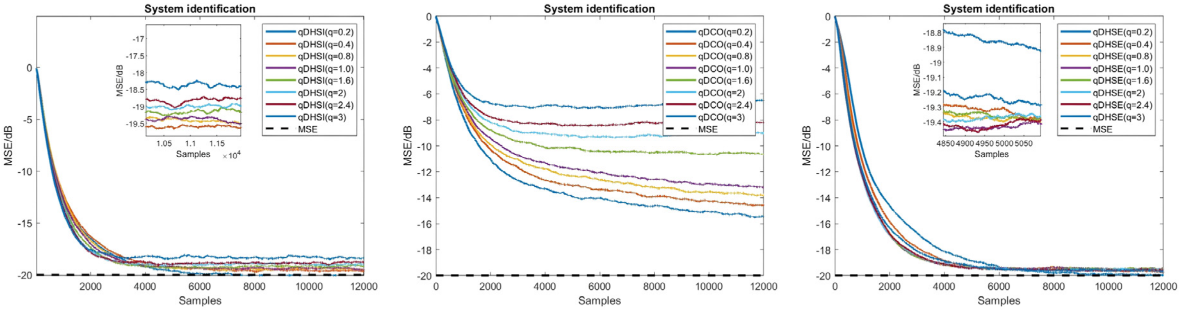
Case 1 verifies the performance of the proposed algorithm in a white noise environment where the step size μ_{ω}=μ_q=0.005, the correlation coefficients were a = 0.9 and a = 0.1, respectively, and the SNR=20 dB. Fig. 4 depicts the performance comparison of different algorithms for the same input signal in a white noise environment, where no systematic mutation is introduced on the left side of Fig. 4 and systematic mutation is introduced on the right side. The results show that the performance of algorithm SAF-qDHSI, algorithm SAF-qDHSE, and algorithm SAF-qDHCO is excellent with the same input signal, in which the convergence speed of algorithm SAF-qDHSI and algorithm SAF-qDHSE is only second to that of algorithm SAF-LHC in the early stage, but with the increase of iteration number, they can catch up with algorithm SAF-LHC very quickly and have better steady-state performance. SAF-LHC, and has better steady-state performance; after adding the system mutation, the algorithm SAF-qDHSI, algorithm SAF-qDHSE, and algorithm SAF-qDHCO can quickly adapt to and return to normal, showing the excellent robustness of the new algorithm. Fig. 5 depicts the performance comparison of different algorithms with different input signals in a white noise environment, with no systematic mutations on the left side of Fig. 5 and systematic mutations introduced on the right side. Fig. 5 exhibits similar results to Fig. 4, also showing the excellent convergence rate, steady state, and robustness of the new algorithm. Among them, Algorithm SAF-qDHSI and Algorithm SAF-qDHSE are the best.

Case 2: Colored noise

Theoretically, noise other than white noise is called colored noise. In this subsection, we set up three different noise environments, i.e., Rayleigh distribution, uniform distribution, and Poisson distribution. The step size μ_{ω}=μ_q=0.005, the correlation coefficients are a = 0.9 and a = 0.1, and the SNR=20 dB, respectively. Fig. 6 illustrates the comparison of different algorithms with the same input signal under Rayleigh distributions without systematic mutation (left) and with systematic mutation (right), and Fig. 7 illustrates the comparison of different algorithms with the same input signals under Rayleigh distributions without systematic mutation (left) and with systematic mutation (right). (left) and with systematic mutations (right) for different input signals. Fig. 8 depicts the comparison of different algorithms for the same input signals without systematic mutation (left) and with systematic mutation (right) in a uniform distribution; Fig. 9 depicts the comparison of different algorithms for different input signals without systematic mutation (upper panel) and with systematic mutation (lower panel) in a uniform distribution. Fig. 10 depicts the comparison of different algorithms for the same input signal in the case of Poisson distribution without systematic mutation (upper panel) and with systematic mutation (lower panel); Fig. 11 depicts the comparison of different algorithms for different input signals in the case of Poisson distribution without systematic mutation (upper panel) and with systematic mutation (lower panel). The superiority of the SAF-qDHSI algorithm versus the SAF-qDHSE algorithm is evident in all algorithm comparisons. In a comprehensive comparison, the SAF-qDHSI and SAF-qDHSE algorithms outperform the other algorithms, and the SAF-qDHCO algorithm performs similarly to the remaining algorithms.

Case 3: Real empirical datasets from DaISy

To evaluate the efficacy of our proposed novel algorithm, we have employed two empirical datasets (No. 96-004, data from the ball and beam practicum at ESAT-SISTA; No. 96-009, Data from a flexible robot arm), which were sourced from (https://homes.esat.kuleuven.be/~smc/daisy/). The experimental results are shown in Fig. 12 and Fig. 13, where we set μ_{ω}=μ_q=0.005, and the value of q for the new algorithm is uniformly set to 0.4 for 200 iterations per run. The results presented in Fig. 12 demonstrate that the two most effective algorithms



(a) (b) (c)

Fig. 3. Effect of different q values on the learning rate of the algorithm (a=0.1, SNR=20 dB).

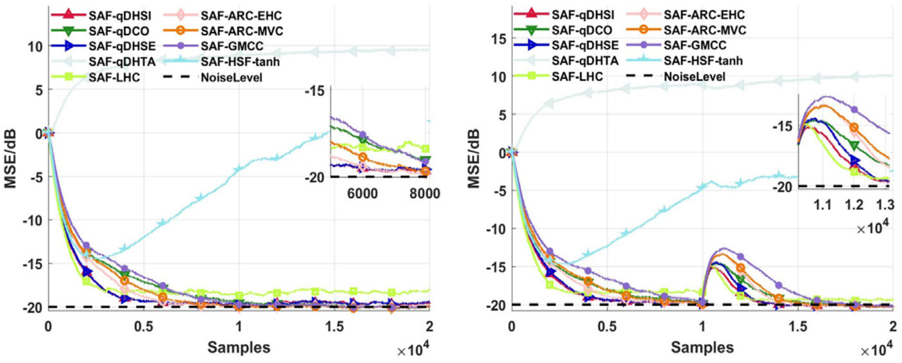


Fig. 4. Comparison of different algorithms for the same input signal in white noise environment (left) without systematic mutation (right) with systematic mutation (a=0.9, SNR=20 dB).

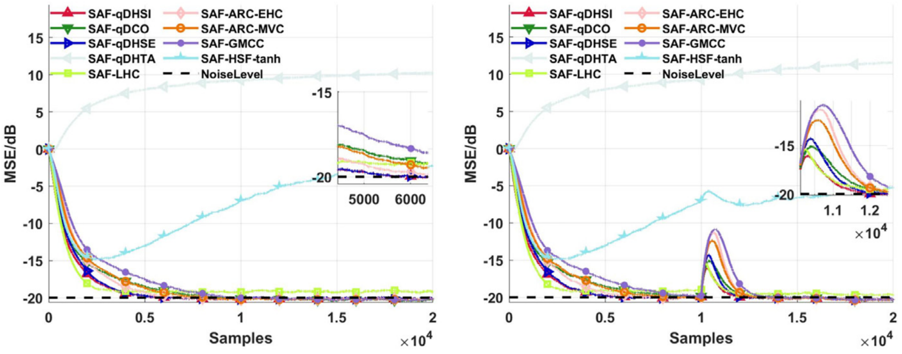


Fig. 5. Comparison of different algorithms for the different input signals in white noise environment (left) without systematic mutation (right) with systematic mutation (a=0.1, SNR=20 dB).

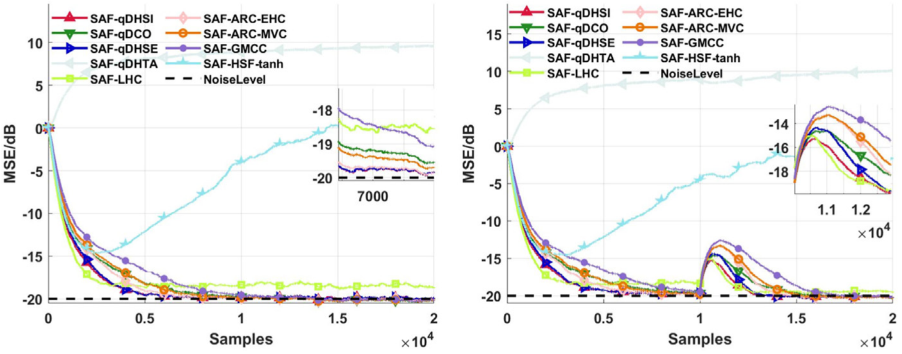


Fig. 6. Comparison of algorithms for the same input signal in a Rayleigh distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.9, SNR=20 dB).

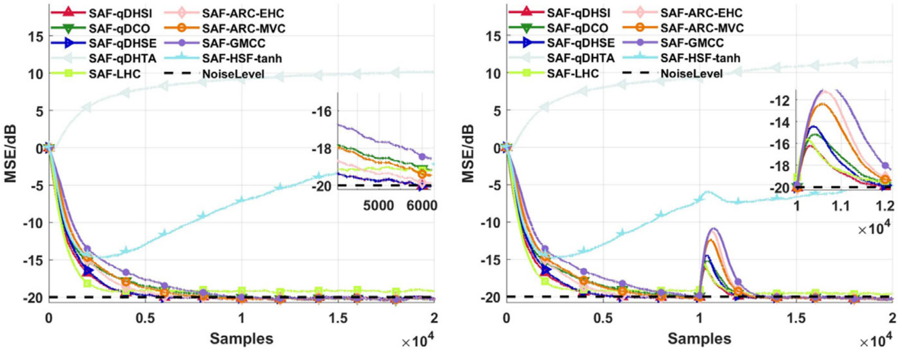


Fig. 7. Comparison of algorithms for different input signals in a Rayleigh distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.1, SNR=20 dB).

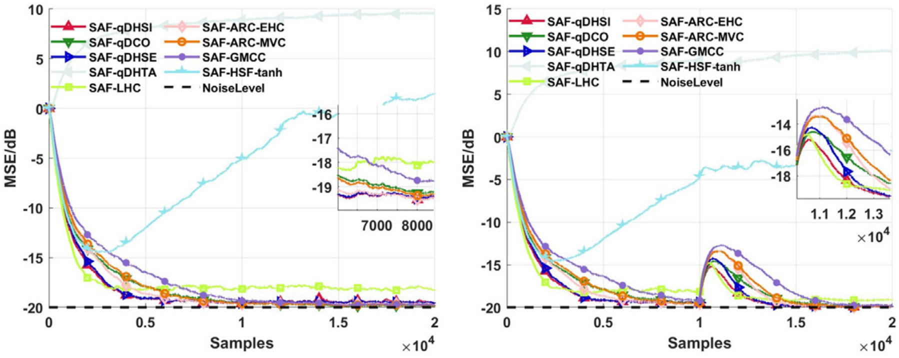


Fig. 8. Comparison of algorithms for the same input signal in a uniformly distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.9, SNR=20 dB).

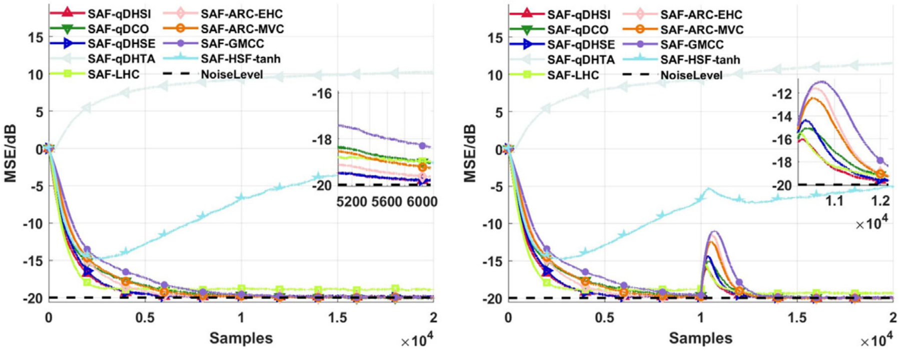


Fig. 9. Comparison of algorithms for different input signals in a uniformly distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.1, SNR=20 dB).

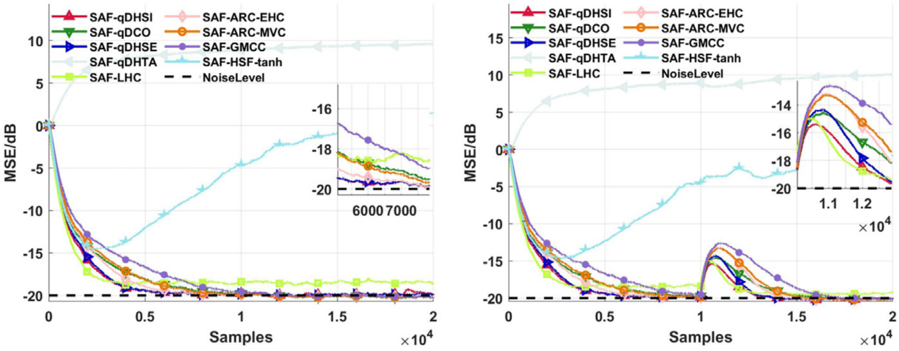


Fig. 10. Comparison of algorithms for the same input signal in a Poisson-distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.9, SNR=20 dB).

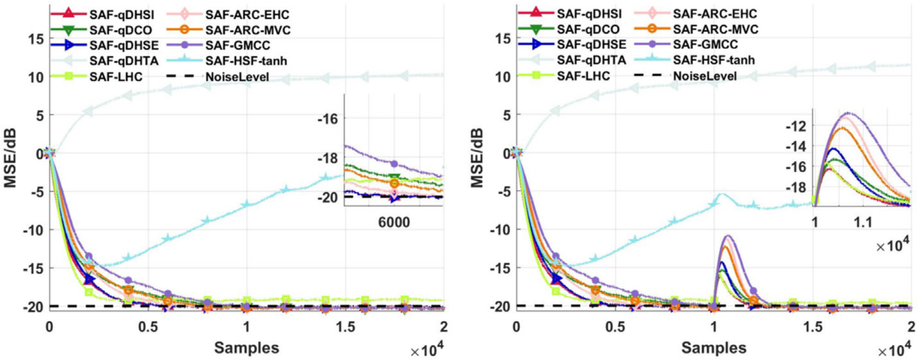
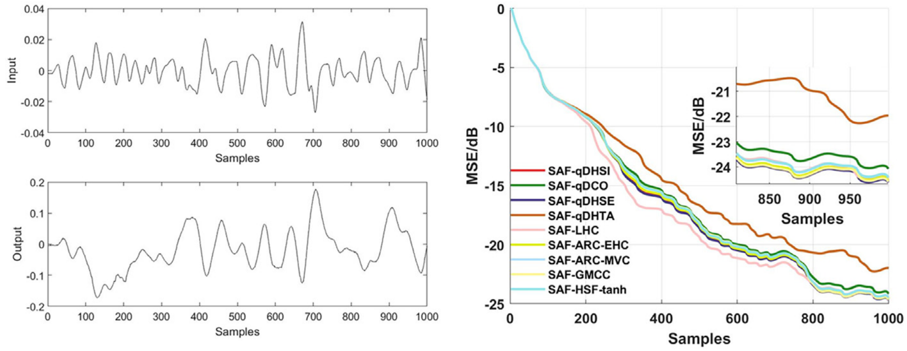
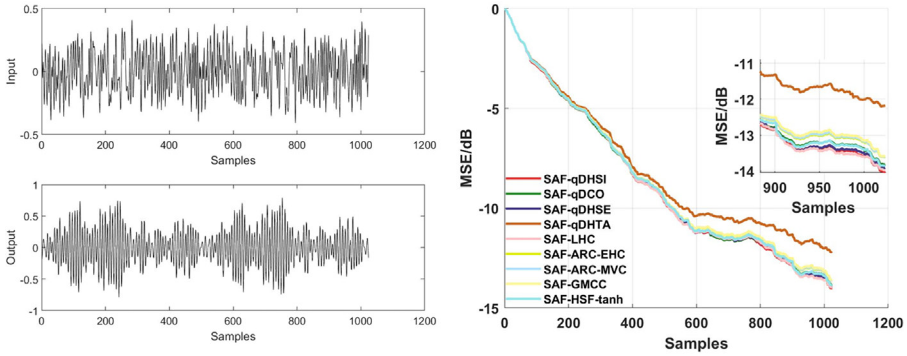


Fig. 11. Comparison of algorithms for the different input signals in a Poisson-distributed noise environment (left) without systematic mutation (right) with systematic mutation (a=0.1, SNR=20 dB).



(a) (b)

Fig. 12. Measured dataset: (a) Data of the ball and beam practicum at ESAT-SISTA with input and output variables; (b) Learning rate of the practical dataset (No. 96-004).



(a) (b)

Fig. 13. Measured dataset: (a) input and output variables of Data from a flexible robot arm; (b) Learning rate of the practical dataset (No. 96-009).

are SAF-qDHSI and SAF-qDHSE. Similarly, the results in Fig. 13 indicate that these two algorithms continue to exhibit superior performance.

5. Conclusion

This paper proposes a series of generalized SAF algorithms based on the q-deformed hyperbolic function, which are designated as SAF-qDHSI, SAF-qDHCO, SAF-qDHTA, and SAF-qDHSE, respectively. Furthermore, the range of learning rates for the proposed algorithms is derived, accompanied by a computational complexity analysis. The experimental results show that: (1) the algorithm proposed in this paper exhibits excellent performance in terms of the effectiveness of nonlinear system identification; (2) The algorithm can be adjusted by changing its parameters to adjust the algorithm's performance; (3) Compared with the existing algorithms, the algorithm proposed in this paper has better performance under the interference of Gaussian noise and non-Gaussian noise and better robustness in the face of abrupt system changes; (4) The algorithm in this paper still has better performance in real engineering data.

CRediT authorship contribution statement

Shiwei Yun: Writing – original draft, Methodology, Formal analysis, Data curation, Conceptualization. **Sihai Guan:** Writing – review & editing, Supervision, Software, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Chuanwu Zhang:** Writing – review & editing, Visualization, Supervision, Funding acquisition. **Bharat Biswal:** Writing – review & editing, Visualization, Project administration, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

No generative AI or AI assistive techniques are used in the writing process.

Declaration of competing interest

There is no conflict of interest in submitting this manuscript, and all the authors have agreed to publish it.

Acknowledgments

This work is financially supported by the National Natural Science Foundation of China (8225041038), the Sichuan Science and Technology Program (23NSFSC2916), and the Fundamental Research Funds for the Central Universities, Southwest Minzu University (ZYN2024077).

Data availability

Data will be made available on request.

References

[1] B. Farhang-Boroujeny, Adaptive Filters: Theory and Applications, second ed., John Wiley & Sons, 2013.

[2] J.W. Yoo, J.W. Shin, P.G. Park, An improved NLMS algorithm in sparse systems against noisy input signals, IEEE Trans. Circuits Syst. II 62 (2014) 271–275.

[3] Q. Zhang, J. Zhou, H. Wang, et al., Output feedback stabilization for a class of multi-variable bilinear stochastic systems with stochastic coupling attenuation, IEEE Trans. Automat. Control 62 (2016) 2936–2942.

[4] O. Nelles, Orthonormal basis functions for nonlinear system identification with local linear model trees (LOLIMOT), IFAC Proc. Vol. 30 (1997) 639–644.

[5] Simon Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall PTR, Upper Saddle River, NJ, United States, 1998.
[6] C. Lesiak, A. Krener, The existence and uniqueness of Volterra series for nonlinear systems, IEEE Trans. Automat. Control 23 (1978) 1090–1095.
[7] T. Ogunfunmi, Adaptive Nonlinear System Identification: The Volterra and Wiener Model Approaches, Springer Science & Business Media, 2007.
[8] Y. Cao, P.M. Frank, Analysis and synthesis of nonlinear time-delay systems via fuzzy control approach, IEEE Trans. Fuzzy Syst. 8 (2000) 200–211.
[9] A. Rahrooh, S. Shepard, Identification of nonlinear systems using NARMAX model, Nonlinear Anal. Theory 71 (2009) e1198–e1202.
[10] D.J. Sebald, J.A. Bucklew, Support vector machine techniques for nonlinear equalization, IEEE Trans. Signal Process. 48 (2000) 3217–3226.
[11] D. Comminiello, M. Scarpiniti, L.A. Azpicueta-Ruiz, et al., Functional link adaptive filters for nonlinear acoustic echo cancellation, IEEE Trans. Audio Speech 21 (2013) 1502–1512.
[12] J.Q. Gong, B. Yao, Neural network adaptive robust control of nonlinear systems in semi-strict feedback form, Automatica 37 (2001) 1149–1160.
[13] N. Farhat, Optoelectronic neural networks and learning machines, IEEE Circuits Devices Mag. 5 (1989) 32–41.
[14] M. Scarpiniti, D. Comminiello, R. Parisi, et al., Nonlinear system identification using IIR spline adaptive filters, Signal Process. 108 (2015) 30–35.
[15] J. de Jesus Rubio, W. Yu, Stability analysis of nonlinear system identification via delayed neural networks, IEEE Trans. Circuits Syst. II 54 (2007) 161–165.
[16] M. Scarpiniti, D. Comminiello, R. Parisi, et al., Hammerstein uniform cubic spline adaptive filters: Learning and convergence properties, Signal Process. 100 (2014) 112–123.
[17] S. Guan, Z. Li, Normalised spline adaptive filtering algorithm for nonlinear system identification, Neural Process. Lett. 46 (2017) 595–607.
[18] Q. Liu, Y. He, Robust Geman–McClure based nonlinear spline adaptive filter against impulsive noise, IEEE Access 8 (2020) 22571–22580.
[19] T. Yu, W. Li, Y. Yu, et al., Robust spline adaptive filtering based on accelerated gradient learning: Design and performance analysis, Signal Process. 183 (2021) 107965.
[20] Y. Gao, H. Zhao, Y. Zhu, et al., The q-gradient LMS spline adaptive filtering algorithm and its variable step-size variant, Inf. Sci. 658 (2024) 119983.
[21] G. Xu, B. Zha, H. Yuan, et al., Underwater four-quadrant dual-beam circumferential scanning laser fuze using nonlinear adaptive backscatter filter based on pausable SAF-LMS algorithm, Def. Technol. 37 (2024) 1–13.
[22] Wenyan Guo, Yongfeng Zhi, Nonlinear spline adaptive filtering against non-gaussian noise, Circuits Syst. Signal Process. 41 (1) (2022) 579–596.
[23] S.S. Bhattacharjee, V. Patel, N.V. George, Nonlinear spline adaptive filters based on a low rank approximation, Signal Process. 201 (2022) 108726.
[24] W. Guo, Y. Zhi, Nonlinear spline versoria prioritization optimization adaptive filter for alpha-stable clutter, IEEE Trans. Aerosp. Electron. Syst. 59 (2022) 734–744.
[25] L. Janjanam, S.K. Saha, R. Kar, Optimal design of Hammerstein cubic spline filter for nonlinear system modeling based on snake optimizer, IEEE Trans. Ind. Electron. 70 (2022) 8457–8467.
[26] S. Guan, B. Biswal, Spline adaptive filtering algorithm based on different iterative gradients: Performance analysis and comparison, J. Autom. Intell. 2 (2023) 1–13.
[27] W. Li, M. Xu, J. Tang, et al., Robust frequency domain spline adaptive filtering based on the half-quadratic criterion: Performance analysis and applications, IEEE Trans. Instrum. Meas. (2023).
[28] L. Yang, J. Liu, R. Yan, et al., Spline adaptive filter with arctangent-momentum strategy for nonlinear system identification, Signal Process. 164 (2019) 99–109.
[29] L. Lu, L. Chen, Z. Zheng, et al., Behavior of the LMS algorithm with hyperbolic secant cost, J. Franklin Inst. 357 (2020) 1943–1960.
[30] S.S. Bhattacharjee, K. Kumar, N.V. George, Nearest kronecker product decomposition based generalized maximum correntropy and generalized hyperbolic secant robust adaptive filters, IEEE Signal Process. Lett. 27 (2020) 1525–1529.
[31] C. Liu, M. Jiang, Robust adaptive filter with lncosh cost, Signal Process. 168 (2020) 107348.
[32] X. Wang, A. Wang, D. Wang, et al., An improved spline adaptive filter for nonlinear system identification under impulsive noise environment, Energy Rep. 8 (2022) 832–840.
[33] V. Patel, S.S. Bhattacharjee, N.V. George, A family of logarithmic hyperbolic cosine spline nonlinear adaptive filters, Appl. Acoust. 178 (2021) 107973.
[34] T. Yu, W. Li, R.C. de Lamare, et al., M-estimate affine projection spline adaptive filtering algorithm: Analysis and implementation, Digit. Signal Process. 123 (2022) 103452.
[35] S. Guan, Q. Cheng, Y. Zhao, et al., Spline adaptive filtering algorithm based on Heaviside step function, Signal Image Video Process. 16 (2022) 1333–1343.
[36] W. Guo, Y. Zhi, Nonlinear spline prioritization optimization generalized hyperbolic secant adaptive filtering against alpha-stable noise, Nonlinear Dynam. 111 (2023) 14351–14363.
[37] G.A. Anastassiou, q-deformed hyperbolic tangent based Banach space valued ordinary and fractional neural network approximations, Rev. Real Acad. Cienc. Exactas Fis. Nat. Ser. A- Mat. 117 (2023) 83.
[38] S.A. El-Shehawy, E.A.B. Abdel-Salam, The q-deformed hyperbolic secant family, Int. J. Appl. Math. Stat. 29 (2012) 51–62.
[39] M. Scarpiniti, D. Comminiello, R. Parisi, et al., Nonlinear spline adaptive filtering, Signal Process. 93 (2013) 772–783.
[40] Y. Gao, H. Zhao, Y. Zhu, et al., Spline adaptive filtering algorithm-based generalized maximum correntropy and its application to nonlinear active noise control, Circuits Syst. Signal Process. 42 (2023) 6636–6659.
[41] Alexandru-George Rusu, Silviu Ciochină, Constantin Paleologu, On the step-size optimization of the LMS algorithm, in: 2019 42nd International Conference on Telecommunications and SIGNAL PROCESS, TSP, IEEE, 2019.

Shiwei Yun Currently, he is pursuing his B.S. degree in the School of Electronic Information, Southwest Minzu University, Chengdu, China. His current research interest is adaptive signal processing.

Sihai Guan received a Ph.D. degree (2017) from Xidian University. He is currently working at the College of Electronic and Information, Southwest Minzu University, and the Key Laboratory of Electronic and Information Engineering, State Ethnic Affairs Commission. His research interests included adaptive signal processing and fMRI analysis.

Chuanwu Zhang received a Ph.D. degree (2003) from the University of Electronic Science and Technology of China. He is currently a Professor at the College of Electronic and Information at Southwest Minzu University and dean of the Key Laboratory of Electronic and Information Engineering, State Ethnic Affairs Commission.

Bharat Biswal is Distinguished Professor and Chair in the Department of Biomedical Engineering at the New Jersey Institute of Technology, Newark, NJ, USA. Dr. Bharat Biswal was a fellow of the American Institute for Medical and Biological Engineering (AIMBE). His impact is reflected in the fact that he is an ISI highly cited researcher in neuroscience and behavior. He has authored some of the most highly cited papers in clinical imaging since 2014. He has authored and co-authored 250 articles appearing in peer-reviewed journals. His work has appeared in high-profile journals in psychology, neuroscience, and engineering, including Nature, Neuron, and Proceedings of the National Academy of Science, Nature neuroscience, Trends in cognitive sciences, Nature Communications, and IEEE transactions on medical imaging.