IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 24, 2019, accepted May 3, 2019, date of publication May 9, 2019, date of current version May 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916003

A Novel Hybrid Kernel Adaptive Filtering Algorithm for Nonlinear Channel Equalization

QISHUAI WU[1], YINGSONG LI[1,2], (Senior Member, IEEE), ZHENGXIONG JIANG[1], AND YOUWEN ZHANG[3]

[1]College of Information and Communication Engineering, Harbin Engineering University, Harbin 150001, China
[2]Key Laboratory of Microwave Remote Sensing, National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China
[3]Acoustic Science and Technology Laboratory, Harbin Engineering University, Harbin 150001, China

Corresponding author: Yingsong Li (liyingsong@ieee.org)

This paper was supported in part by the National Key Research and Development Program of China under Grant 2016YFE0111100, in part by the Key Research and Development Program of Heilongjiang under Grant GX17A016, in part by the Science and Technology innovative Talents Foundation of Harbin under Grant 2016RAXXJ044, in part by the Natural Science Foundation of Beijing under Grant 4182077, in part by the China Postdoctoral Science Foundation under Grant 2017M620918, in part by the Fundamental Research Funds for the Central Universities under Grant HEUCFG201829, and in part by the Opening Fund of Acoustics Science and Technology Laboratory under Grant SSKF2016001.

ABSTRACT In this paper, a novel kernel mixed error criterion (KMEC) algorithm is proposed for nonlinear system identification, which uses a combination of two different error schemes to implement a newly constructed cost function, which is realized by using a logarithmic squared error and a generalized maximum correntropy criterion (GMCC) to devise the KMEC algorithm. The proposed KMEC is derived in the context of the kernel adaptive filter and it provides good performance for identifying the nonlinear channels in different mixed noise environments in terms of the mean square error (MSE) at its steady-state and convergence performance.

INDEX TERMS Kernel adaptive filtering, mixed error criterion algorithm, generalized maximum correntropy, non-Gaussian noise environments, nonlinear adaptive filtering.

I. INTRODUCTION

Kernel adaptive filtering (KAF) is a useful adaptive filtering (AF) method within the framework of nonlinear AF, which is to implement the AF algorithms based on the kernel learning method [1]. The KAF is an online learning method in reproducing kernel Hilbert space, which has excellent effects for nonlinear signal processing. Recently, many KAF algorithms have been reported for nonlinear signal processing [1]–[4], including the kernel least mean square (KLMS) [2], kernel least mean fourth (KLMF) and the kernel recursive least squares (KRLS) [5]. Additionally, the variants of these algorithms have also been reported to improve the performance of the basic KAF algorithms [6]–[12]. Although KLMS and KRLS algorithms perform well for handling the nonlinear systems, they are not suitable for non-Gaussian environments because of the second-order statistics in the updating [13], [14].

In order to find an AF algorithm that can perform better with respect to the convergence and MSE in non-Gaussian noise environments which are always existing in the practical wireless communication environments, some AF algorithms have been proposed using sign error [15], high-order error [16]–[18], maximum correntropy criterion (MCC) [19]–[25] and mixed error criterion algorithms [17], [18], [26]–[28]. However, most of these algorithms are developed for Gaussian noise environment applications. For dealing with non-Gaussian noise environment, the least mean p-power (LMP) [29] algorithm utilizes only a single high-order error norm to implement the cost function against non-Gaussian noise, and hence, the LMP algorithm has better convergence performance than least mean square (LMS) algorithm that considers only the second-order statistic of the error signal. Another kind of nonlinear AF algorithms are realized by using the kernel theory to implement least mean algorithm and its mixed-norm [28], [30]–[32], including the kernel least mean

The associate editor coordinating the review of this manuscript and approving it for publication was Guan Gui.

mixed-norm (KLMMN) algorithm [28], [30]–[32]. In comparison with the KLMS, the KLMF and KLMMN algorithms can accelerate the convergence speed and reduce the estimation error at the steady-state. Then, the kernel method has been considered in MCC, recursive generalized mixed norm (RGMN) algorithms [32] to realize the kernel MCC (KMCC) [33] and kernel RGMN (KRGMN) for non-linear system identification under non-Gaussian noise environments [32]. From the KLMS to KMCC and KRGMN, the kernel has been promoted to the non-Gaussian applications.

In this paper, the main goal is to propose a novel KAF algorithm to deal with nonlinear channel equalization (NCE). The logarithmic squared error function and the generalized maximum correntropy criterion (GMCC) [23] scheme are used to construct a new cost function, and then, the gradient descent method is used to minimize it to realize a more robust kernel algorithm which is denoted as kernel mixed error criterion (KMEC). The proposed algorithm is derived within the framework of the KAF, and its performance is investigated over a nonlinear channel to analyze the NCE. The simulation results show that the KMEC outperforms the KLMS, KLMF, KMCC, KLMMN and KRGMN algorithms in terms of the convergence and the estimation error at the steady-state.

The rest of this paper is shown as follows: In Section II, the basic idea on kernel methods has been introduced briefly. In Section III, the mixed error algorithm based on the logarithmic squared error function and generalized maximum correntropy is presented and the proposed KMEC algorithm is derived in detail. In Section IV, the performance of the devised KMEC algorithm is verified through the simulation experiments. Finally, in section V, the conclusion is presented.

II. KERNEL METHODS

The basic idea of the KAF is to construct a framework using the kernel method which transforms the original input signal vector space $\mathbf{X}$ into a high-dimensional feature space $\mathbf{F}$ to implement the nonlinear signal processing. The nonlinear mapping in the kernel method is written as:

$$\boldsymbol{\varphi}: \mathbf{X} \rightarrow \mathbf{F}. \tag{1}$$

The kernel function $\kappa(\mathbf{x}, \mathbf{x}')$ is to implement inner product in the desired feature space for not knowing the exact nonlinear mapping. Then, the kernel method has been combined with the linear AF algorithms, where the kernel methods are realized based on Mercer's theorem.

$$\kappa(\mathbf{x}, \mathbf{x}') = \boldsymbol{\varphi}(\mathbf{x})^T \boldsymbol{\varphi}(\mathbf{x}'). \tag{2}$$

To implement the kernel method, one of a general Gaussian kernel function is used in the KAF algorithms, which is given by

$$\kappa(\mathbf{x}, \mathbf{x}') = \exp(\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma^2}), \tag{3}$$

where σ is the kernel width in the kernel function.

III. THE PROPOSED KMEC ALGORITHM

Since the LMS cannot combat the impulse noise, the performance of the KLMS will be degraded for identifying the system in impulse noise environments. Thus, we first propose a novel MCE algorithm that minimizes a new cost function combined by logarithmic squared error and the correntropy function. Then, the kernel learning principle is incorporated into the MCE algorithm to realize the KMCE algorithm. Herein, the MCE and the KMCE will be discussed in detail.

A. THE MIXED ERROR CRITERION(MEC) ALGORITHM

In the framework of the system identification, an input signal is defined as $\mathbf{x}(i) = [x(i), x(i-1), \ldots, x(i-L+1)]^T$ with a length of L, and the desired signal $d(i)$ is given by

$$d(i) = \mathbf{h}^T(i)\mathbf{x}(i) + v(i), \tag{4}$$

where $\mathbf{h}(i) = [h_0(i), h_1(i), \ldots, h_{L-1}(i)]^T$ is the unknown system vector needed to be identified, $v(i)$ is the measurement noise. Then, the error signal $e(i)$ is expressed as

$$e(i) = d(i) - \hat{\mathbf{h}}^T(i)\mathbf{x}(i), \tag{5}$$

where $\hat{\mathbf{h}}(i)$ is an estimation of $\mathbf{h}(i)$. On the basis of the LMS and MCC in the context of the AF, a mixed error criterion (MEC) algorithm is created using the logarithmic squared error and a generalized maximum correntropy criterion (GMCC) schemes to improve the LMS and MCC, whose cost function is given by

$$J_{\mathrm{MEC}}(i) = J_{\mathrm{A}}(i) - J_{\mathrm{B}}(i), \tag{6}$$

where $J_{\mathrm{A}}(i) = \frac{\omega}{2}\log\left[1 + \frac{|e(i)|^2}{2}\right]$ and $J_{\mathrm{B}}(i) = \frac{1-\omega}{\alpha} \cdot \frac{\alpha}{2\beta\Gamma(1/\alpha)}\exp\left(-\left|\frac{e(i)}{\beta}\right|^{\alpha}\right)$. $\Gamma(x) = \int_0^{+\infty} t^{x-1}e^{-t}dt$ represents the gamma function, and α is a shaping parameter, β is the scale (bandwidth) parameter, and $0 < \omega < 1$ is a mixed factor. Then, the gradient of the cost function $J_{\mathrm{MEC}}(i)$ is

$$\begin{aligned}
&\frac{\partial J_{\mathrm{MEC}}(i)}{\partial \hat{\mathbf{h}}(i)} \\
&= -\left(\frac{\omega}{2+|e(i)|^2} e(i)\mathbf{x}(i)\,\mathrm{sgn}(e(i))\right. \\
&\quad \left. + \frac{(1-\omega)\alpha}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \exp\left(-\left|\frac{e(i)}{\beta}\right|^{\alpha}\right) e(i)^{\alpha-1}\mathbf{x}(i)\,\mathrm{sgn}(e(i))\right) \\
&= -\left(\frac{\omega}{2+|e(i)|^2} + \frac{(1-\omega)\alpha}{2\beta^{\alpha+1}\Gamma(1/\alpha)} e(i)^{\alpha-2}\right. \\
&\quad \left. \times \exp\left(-\left|\frac{e(i)}{\beta}\right|^{\alpha}\right)\right) e(i)\mathbf{x}(i)\,\mathrm{sgn}(e(i)) \\
&= -\left(\frac{\omega}{2+|e(i)|^2} + \frac{(1-\omega)\alpha}{2\beta^{\alpha+1}\Gamma(1/\alpha)}\right. \\
&\quad \left. \times \exp\left(-\left|\frac{e(i)}{\beta}\right|^{\alpha}\right) e(i)^{\alpha-2}\right) e(i)\mathbf{x}(i),
\end{aligned} \tag{7}$$

where $\mathrm{sgn}(e(i))$ denotes the sign function [32]. According to the steepest descent theory [1], the updating equation of MEC algorithm is

$$\begin{aligned}\hat{\mathbf{h}}(i+1) &= \hat{\mathbf{h}}(i) - \mu \frac{\partial J_{\text{MEC}}(i)}{\partial \hat{\mathbf{h}}(i)} \\ &= \hat{\mathbf{h}}(i) + \mu \left(\frac{\omega}{2+|e(i)|^2} + \frac{(1-\omega)}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \right. \\ &\quad \left. \times \exp\left(-\left| \frac{e(i)}{\beta} \right|^{\alpha} \right) e(i)^{\alpha-2} \right) e(i)\,\mathbf{x}(i), \quad (8)\end{aligned}$$

where μ is the adaption factor.

B. KERNEL MEC (KMEC) ALGORITHM

Similar to the other KAF algorithms, the KMEC algorithm is proposed and derived based on the kernel method and MEC algorithm. An exponentially-weighted cost function is introduced to implement the KMEC algorithm, which is expressed as

$$\begin{aligned}&J(\mathbf{\Omega}) \\ &= \min_{\mathbf{\Omega}} \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{\omega}{2} \log[1 + \frac{\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2}{2}] \\ &\quad - \frac{(1-\omega)\alpha}{2\alpha\beta\Gamma(1/\alpha)} \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha})\} + \frac{1}{2}\gamma^i \lambda \|\mathbf{\Omega}\|^2, \quad (9)\end{aligned}$$

where γ is the forgetting factor to gradually strengthen the weights, ω is the mixed factor, λ is the regularization factor. The second term is a norm penalty operation, which is to guarantee the existence of the inverse of the autocorrelation matrix especially during the initial update stages. Based on the gradient descent method, we can get

$$\begin{aligned}&\frac{\partial J(\mathbf{\Omega})}{\partial \mathbf{\Omega}} \\ &= -\boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2}\} \\ &\quad \times (d(j) - \boldsymbol{\varphi}(j)^T \mathbf{\Omega}) + \gamma^i \lambda \mathbf{\Omega}, \\ &= -\boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2}\} \\ &\quad \times d(j) + \boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2} \\ &\quad + \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha})\} \boldsymbol{\varphi}(j)^T \mathbf{\Omega} + \gamma^i \lambda \mathbf{\Omega} \\ &= -\boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2}\} \\ &\quad \times d(j) + \left(\boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2} \right. \\ &\quad + \frac{(1-\omega)\alpha\left|d(j) - \Omega^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \left. \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha})\} \boldsymbol{\varphi}(j)^T + \gamma^i \lambda \right) \mathbf{\Omega}. \quad (10)\end{aligned}$$

Then, introduce

$$\mathbf{d}(i) = \left[d(1), \quad d(2), \quad \ldots \quad , d(i) \right]^T, \quad (11)$$

$$\mathbf{\Phi}(i) = \left[\boldsymbol{\varphi}(1), \quad \boldsymbol{\varphi}(2), \quad \ldots \quad , \boldsymbol{\varphi}(i) \right], \quad (12)$$

$$\begin{aligned}&\mathbf{\Psi}(i) \\ &= \text{diag}[\gamma^{i-1} \{ \frac{(1-\omega)\alpha\left|d(1) - \mathbf{\Omega}^T \boldsymbol{\varphi}(1)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(1) - \Omega^T \boldsymbol{\varphi}(1)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(1) - \mathbf{\Omega}^T \boldsymbol{\varphi}(1)\right|^2}\}, \\ &\quad \gamma^{i-2} \{ \frac{(1-\omega)\alpha\left|d(2) - \mathbf{\Omega}^T \boldsymbol{\varphi}(2)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(2) - \mathbf{\Omega}^T \boldsymbol{\varphi}(2)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(2) - \mathbf{\Omega}^T \boldsymbol{\varphi}(2)\right|^2}\}, \\ &\quad \ldots, \{ \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2} + \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha}) \\ &\quad \times \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \}], \quad (13)\end{aligned}$$

and let the gradient $\frac{\partial J(\mathbf{\Omega})}{\partial \mathbf{\Omega}}$ be zero, and consider $\mathbf{\Omega}$

$$\begin{aligned}\mathbf{\Omega} &= (\boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2}\} \\ &\quad \times, \boldsymbol{\varphi}(j)^T + \gamma^i \lambda)^{-1} \cdot \boldsymbol{\varphi}(j) \sum_{j=1}^{i} \gamma^{i-j} \{ \frac{\omega}{2+\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^2} \\ &\quad + \frac{(1-\omega)\alpha\left|d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\quad \times \exp(-\left|\frac{d(j) - \mathbf{\Omega}^T \boldsymbol{\varphi}(j)}{\beta}\right|^{\alpha})\} d(j), \quad (14)\end{aligned}$$

we will get

$$\Omega(i) = \left(\Phi(i)\, \Psi(i)\, \Phi(i)^T + \gamma^i \lambda \mathbf{I} \right)^{-1} \Phi(i)\, \Psi(i)\, \mathbf{d}(i). \quad (15)$$

Using the Matrix inverse lemma

$$(\mathbf{A}+\mathbf{BCD})^{-1}=\mathbf{A}^{-1}-\mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1}, \tag{16}$$

and considering

$$\gamma^i\lambda\mathbf{I}\rightarrow\mathbf{A},\quad \Phi(i)\rightarrow\mathbf{B},\ \Psi(i)\rightarrow\mathbf{C},\ \Phi(i)^T\rightarrow\mathbf{D}, \tag{17}$$

equation (15) is changed to be

$$\left(\Phi(i)\Psi(i)\Phi(i)^T+\gamma^i\lambda\mathbf{I}\right)^{-1}\Phi(i)\Psi(i) = \Phi(i)\left(\Phi(i)^T\Phi(i)+\gamma^i\lambda\Psi(i)^{-1}\right)^{-1}. \tag{18}$$

$\Omega(i)$ can also be expressed as

$$\Omega(i)=\Phi(i)\left(\Phi(i)^T\Phi(i)+\gamma^i\lambda\Psi(i)^{-1}\right)^{-1}\mathbf{d}(i), \tag{19}$$

Then, the weight vector $\Omega(i)$ can be explicitly replaced by using a linear combination of the input data

$$\Omega(i)=\Phi(i)\,\mathbf{a}(i), \tag{20}$$

where $\mathbf{a}(i)$ is given by

$$\mathbf{a}(i)=\left(\Phi(i)^T\Phi(i)+\gamma^i\lambda\Psi(i)^{-1}\right)^{-1}\mathbf{d}(i). \tag{21}$$

We define $\mathbf{Q}(i)$ as

$$\mathbf{Q}(i)=\left(\Phi(i)^T\Phi(i)+\gamma^i\lambda\Psi(i)^{-1}\right)^{-1}, \tag{22}$$

where $\Phi(i)=\{\Phi(i-1),\ \boldsymbol{\varphi}(i)\}$, and then, $\mathbf{Q}(i)$ can be expressed as

$$\mathbf{Q}(i) = \begin{bmatrix} \Phi(i-1)^T\Phi(i-1) + \gamma^i\lambda\Psi(i-1)^{-1} & \Phi(i-1)^T\boldsymbol{\varphi}(i) \\ \boldsymbol{\varphi}(i)^T\Phi(i-1) & \begin{aligned}&\{\frac{\omega}{2+\left|d(i)-\Omega^T\boldsymbol{\varphi}(i)\right|^2} + \frac{\left|d(i)-\Omega^T\boldsymbol{\varphi}(i)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \\ &\times\exp(-\left|\frac{d(i)-\Omega^T\boldsymbol{\varphi}(i)}{\beta}\right|^{\alpha}) \\ &\times(1-\omega)\alpha\}^{-1} \\ &\times\gamma^i\lambda+\boldsymbol{\varphi}(i)^T\boldsymbol{\varphi}(i)\end{aligned} \end{bmatrix}^{-1}. \tag{23}$$

We can define $\delta(i)$ as

$$\delta(i)=\{\frac{(1-\omega)\alpha\left|d(i)-\Omega^T\boldsymbol{\varphi}(i)\right|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \times\exp(-\left|\frac{d(i)-\Omega^T\boldsymbol{\varphi}(i)}{\beta}\right|^{\alpha}) + \frac{\omega}{2+\left|d(i)-\Omega^T\boldsymbol{\varphi}(i)\right|^2}\}^{-1}, \tag{24}$$

Then, $\mathbf{Q}(i)$ can be expressed as

$$\mathbf{Q}(i)=\begin{bmatrix} \Phi(i-1)^T\Phi(i-1)+\gamma^i\lambda\Psi(i-1)^{-1} & \Phi(i-1)^T\boldsymbol{\varphi}(i) \\ \boldsymbol{\varphi}(i)^T\Phi(i-1) & \boldsymbol{\varphi}(i)^T\boldsymbol{\varphi}(i)+\gamma^i\lambda\delta(i) \end{bmatrix}^{-1}. \tag{25}$$

From the derivation, we can easily get

$$\mathbf{Q}(i)^{-1}=\begin{bmatrix} \mathbf{Q}(i-1)^{-1} & \mathbf{b}(i) \\ \mathbf{b}(i)^T & \boldsymbol{\varphi}(i)^T\boldsymbol{\varphi}(i)+\gamma^i\lambda\delta(i) \end{bmatrix}, \tag{26}$$

where $\mathbf{b}(i)=\Phi(i-1)^T\boldsymbol{\varphi}(i)$, and the block matrix inversion operation is shown as

$$\begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}^{-1} = \begin{bmatrix} (\mathbf{A}-\mathbf{BD}^{-1}\mathbf{C})^{-1} & -\mathbf{A}^{-1}\mathbf{B}(\mathbf{D}-\mathbf{CA}^{-1}\mathbf{B})^{-1} \\ -\mathbf{D}^{-1}\mathbf{C}(\mathbf{A}-\mathbf{BD}^{-1}\mathbf{C})^{-1} & (\mathbf{D}-\mathbf{CA}^{-1}\mathbf{B})^{-1} \end{bmatrix}. \tag{27}$$

Till now, the equation (25) can be obtained by inverting the block matrix

$$\mathbf{Q}(i)=\varepsilon(i)^{-1}\begin{bmatrix} \mathbf{Q}(i-1)\varepsilon(i)+\mathbf{f}(i)\mathbf{f}(i)^T & -\mathbf{f}(i) \\ -\mathbf{f}(i)^T & 1 \end{bmatrix}, \tag{28}$$

where $\mathbf{f}(i)=\mathbf{Q}(i-1)\mathbf{b}(i)$, and $\varepsilon(i)=\gamma^i\lambda\delta(i)+\boldsymbol{\varphi}(i)^T\boldsymbol{\varphi}(i)-\mathbf{f}(i)^T\mathbf{b}(i)$. Thus, $\mathbf{a}(i)$ is obtained

$$\begin{aligned}\mathbf{a}(i)&=\mathbf{Q}(i)\mathbf{d}(i)\\ &=\begin{bmatrix} \mathbf{Q}(i-1)+\mathbf{f}(i)\mathbf{f}(i)^T\varepsilon(i)^{-1} & -\mathbf{f}(i)\varepsilon(i)^{-1} \\ -\mathbf{f}(i)^T\varepsilon(i)^{-1} & \varepsilon(i)^{-1} \end{bmatrix} \times\begin{bmatrix}\mathbf{d}(i-1)\\ d(i)\end{bmatrix},\\ &=\begin{bmatrix}\mathbf{a}(i-1)-\mathbf{f}(i)\varepsilon(i)^{-1}e(i)\\ \varepsilon(i)^{-1}e(i)\end{bmatrix},\end{aligned} \tag{29}$$

where $e(i)=d(i)-\mathbf{b}(i)^T\mathbf{a}(i-1)$. The KMEC algorithm is developed and summarized in table 1 with a description in pseudo code.

IV. SIMULATION RESULT

In this section, simulation experiments will be setup to verify the performance of the KMEC algorithm over a NCE under different noise environments. In this paper, a nonlinear channel model consisting of a linear filter and a memoryless nonlinear model, and Gaussian kernel function in equation (3) is used in the simulation to model the NCE channel whose structure is shown in Fig. 1 and the conditions of this model are constructed as follows: a binary signal $\{s(1), s(2), \ldots, s(L)\}$ is fed into the nonlinear channel. The input signal is pass through a linear system with a transform function of $H(z)=1-0.5z^{-1}$ to get a memoryless nonlinear filter $x(i)$. At the receiving end of the channel, the signal is contaminated by additive noise $n(i)$,

TABLE 1. The KMEC algorithm.

Initialization:

$$\mathbf{Q}(1) = (\gamma\lambda + \kappa(\mathbf{x}(1), \mathbf{x}(1)))^{-1},$$
$$\mathbf{a}(1) = \mathbf{Q}(1)\, d(1), \gamma, \lambda, \omega, \alpha, \beta$$

Computation

Iterate for $i > 1$:

$$\mathbf{b}(i) = [\kappa(\mathbf{x}(1), \mathbf{x}(1)), \ldots, \kappa(\mathbf{x}(i), \mathbf{x}(i-1))]^T$$
$$\mathbf{f}(i) = \mathbf{Q}(i-1)\,\mathbf{b}(i)$$
$$\delta(i) = \{\frac{\omega}{2 + |d(i) - \mathbf{\Omega}^T \boldsymbol{\varphi}(i)|^2} + \frac{(1-\omega)\alpha |d(i) - \mathbf{\Omega}^T \boldsymbol{\varphi}(i)|^{\alpha-2}}{2\beta^{\alpha+1}\Gamma(1/\alpha)} \cdot \exp(-\left|\frac{d(i) - \mathbf{\Omega}^T \boldsymbol{\varphi}(i)}{\beta}\right|^{\alpha})\}^{-1},$$
$$\varepsilon(i) = \gamma^i \lambda \delta(i) + \boldsymbol{\varphi}(i)^T \boldsymbol{\varphi}(i) - \mathbf{f}(i)^T \mathbf{b}(i)$$
$$\mathbf{Q}(i) = \varepsilon(i)^{-1} \begin{bmatrix} \mathbf{Q}(i-1)\varepsilon(i) + \mathbf{f}(i)\mathbf{f}(i)^T & -\mathbf{f}(i) \\ -\mathbf{f}(i)^T & 1 \end{bmatrix}$$
$$e(i) = d(i) - \mathbf{b}(i)^T \mathbf{a}(i-1) = d(i) - \mathbf{\Omega}^T \boldsymbol{\varphi}(i)$$
$$\mathbf{a}(i) = \begin{bmatrix} \mathbf{a}(i-1) - \mathbf{f}(i)\varepsilon(i)^{-1} e(i) \\ \varepsilon(i)^{-1} e(i) \end{bmatrix}$$

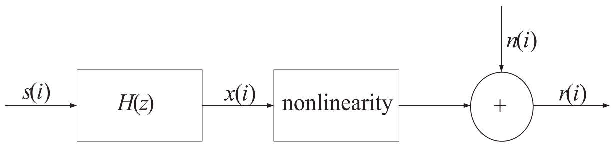


FIGURE 1. An example of nonlinear channel.

TABLE 2. Parameters for algorithms.

Algorithm	α	β	ω	μ	λ	γ
KLMS	-	-	0	0.125	-	-
KLMF	-	-	0	0.01	-	-
KLMMN	-	-	0.25	0.025	-	-
KMCC	-	1	0	0.0125	-	-
KRGMN	1	-	0.25	-	0.45	1
KRMEC	1	1	0.25	-	0.45	1
KRMEC	1	0.45	0.25	-	0.45	1
KRMEC	2	0.45	0.25	-	0.45	1

and the observed value is $\{r(1), r(2), \ldots, r(L)\}$. Considering it as a simple regression problem, the sample is $\{([r(i), r(i+1), \ldots, r(i+l)], s(i-D))\}$, where l is the time embedded length and D is the equilibrium lag time. In all the experiments, $l = 3$ and $D = 2$ are selected. The nonlinear channel model is defined based on its input and output, where the input is $x(i) = s(i) + 0.5s(i-1)$, and the output is $r(i) = x(i) - 0.9x(i)^2 + n(i)$, where $n(i)$ is the noise mixed by $n_1(i)$ and $n_2(i)$ [34]. The proposed KMEC algorithm is investigated using Monte Carlo simulation, whose performance is compared with the KLMS, KLMF, KLMMN, KRGMN and KMCC. In the simulation experiments, the parameters for the mentioned algorithms are listed in table 2, and $\sigma = 1$ is used in all the experiments.

A. PERFORMANCE OF THE KMEC ALGORITHM UNDER DIFFERENT MIXED NOISES

Three simulation experiments are constructed under different mixed noises to observe the convergence of the

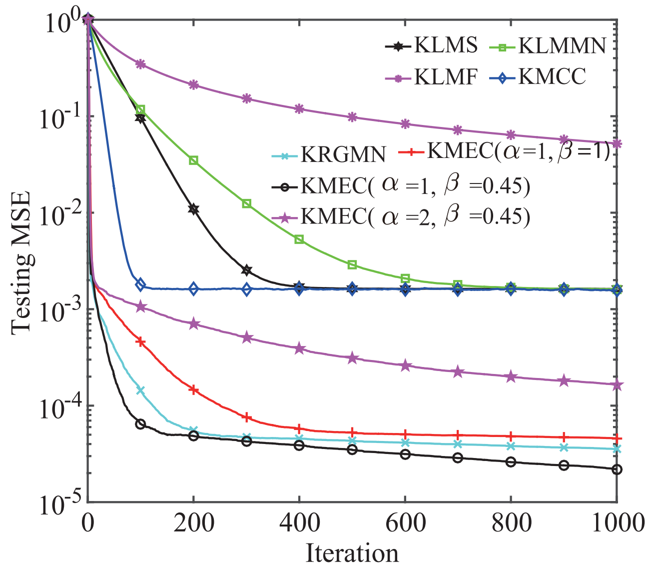


FIGURE 2. Convergence performance of the KMEC under the Bernoulli-Gaussian noise.

KMEC algorithm. The used mixed noises are presented as follows to discuss of the robust of the proposed KMEC algorithm for giving a resistant of impulse noises. 1) A Bernoulli distributed noise $n_1(i)$ with power of 0.45 and a Gaussian distributed noise $n_2(i)$ with power of 0.08 is used in experiment 1.

2) A Bernoulli distributed noise $n_1(i)$ with power of 0.45 and a Laplace distributed noise $n_2(i)$ with power of 0.45 is considered in experiment 2.

3) A Bernoulli distributed noise $n_1(i)$ with power of 0.45 and a uniformly distributed noise $n_2(i)$ with power of 1 is used in experiment 3.

In these simulations, the kernel bandwidth parameter is set to be 1, and the total mixed noise power is set to be 0.1. 1000 iterations are considered to train the proposed KMEC algorithm, and 100 independent tests are run to get a point. The convergence of the KMEC in these three mixed noises are presented in Figs. 2, 3, and 4 compared with KLMS, KLMF, KLMMN, KMCC, KRGMN algorithms. It is found that the KMEC provides the fastest convergence speed for different noises. Additionally, the MSE performance of the proposed KMEC is also superior to other mentioned KAF algorithms, which is because that the KMEC algorithm employs the logarithmic squared error and GMCC to combat against the non-Gaussian measurement noise. Then, the effects of single non-Gaussian distribution noise on the performance of the KMCE is discussed and shown in Fig.5, where a Laplace distribution noise is used with power of 0.45 and $\beta = 0.55$. We can see that the KMEC algorithm still achieves the fastest convergence and lowest MSE. Thus, we can get a conclusion that the proposed KMEC is robust for non-Gaussian noises and provide the best performance with respect to the MSE and convergence speed.

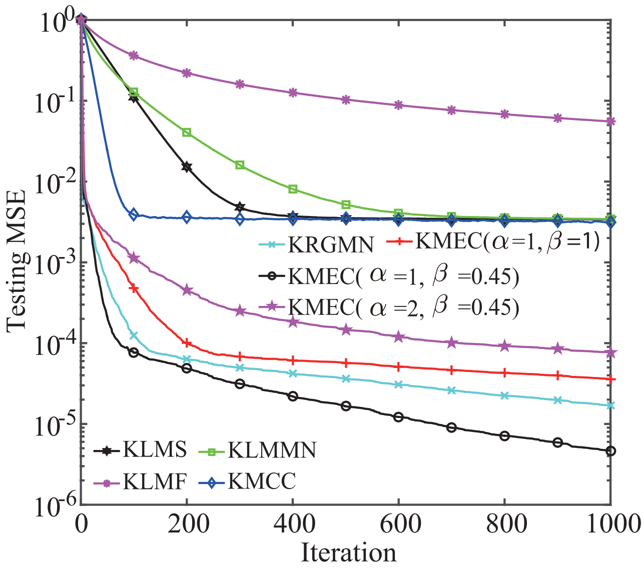


FIGURE 3. **Convergence performance of the KMEC under the bernoulli-laplace noise.**

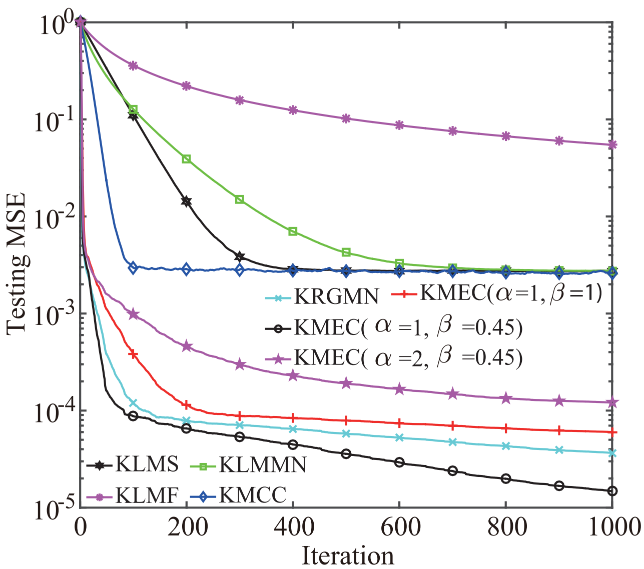


FIGURE 4. **Convergence performance of the KMEC under the bernoulli-uniform noise.**

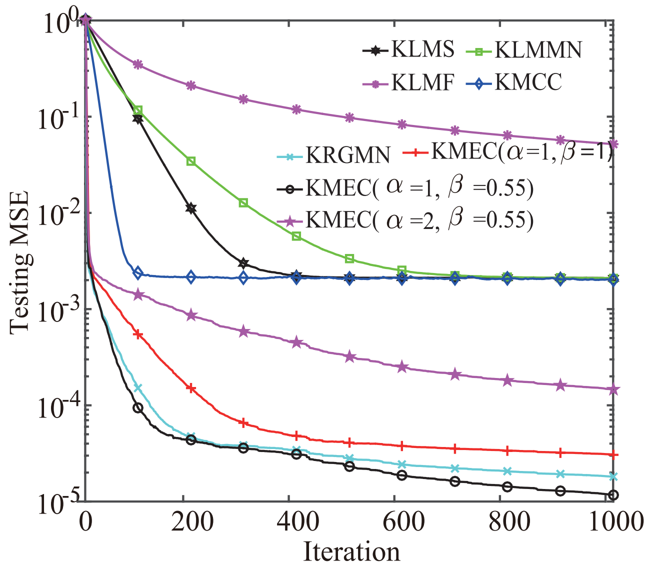


FIGURE 5. **Convergence performance of the KMEC under the laplace noise.**

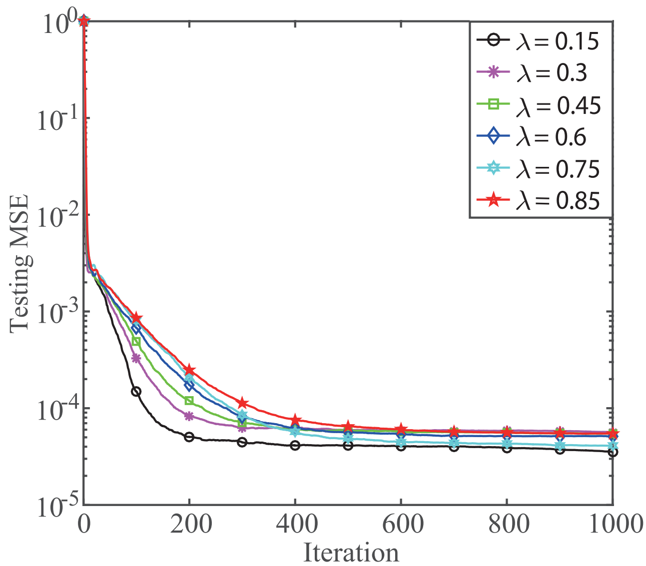


FIGURE 6. **Convergence performance of the KMEC with various λ.**

B. PARAMETER ANALYSIS OF THE KMEC ALGORITHM

Herein, the effects of the parameters of the proposed KMEC algorithm are analyzed and the simulation parameters are same as those in table 2. When we tune one parameter, other parameters are fixed. Firstly, the effect of different regularization factors of the KMEC is investigated in terms of the convergence, where the regularization factor λ is set to be (0.15, 0.3, 0.45, 0.6, 0.75, and 0.9). The other parameters for the mentioned algorithms are $\omega = 0.25$, $\gamma = 1$, $\alpha = 1$ and $\beta = 0.45$ for getting nearly the same MSE level. The experiment is implemented in a Gaussian and uniformly distributed measurement noise with a power of 0.1. The simulation results are presented in Fig.6. It is observed from Fig.6 that the KMEC has the best performance when $\lambda = 0.15$, which means that the forgetting factor λ has an important effect on the estimation performance of the KMEC algorithm.

Next, the characteristics of the KMCC, KRGMN and KMEC algorithms with different weights are verified, where the mixed factor ω is set to be (0.15, 0.35, 0.55, 0.75, 0.95). In this experiment, the Gaussian and uniform distribution noise is used, and the MSE is obtained from the last 100 iterations which is assumed to be worked in steady-state. The simulation results are presented in Fig.7. With an increase of ω, the MSE of the KMEC is increased. However, the KMEC algorithm is still better than the KMCC and the KRGMN when ω ranges from 0.1 to 1. Thus, the proposed KMEC

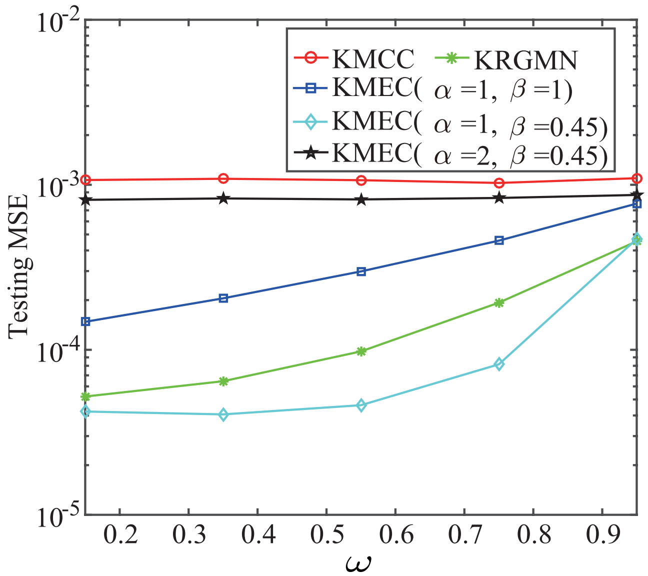


FIGURE 7. Performance comparison of the KMEC with different weight ω.

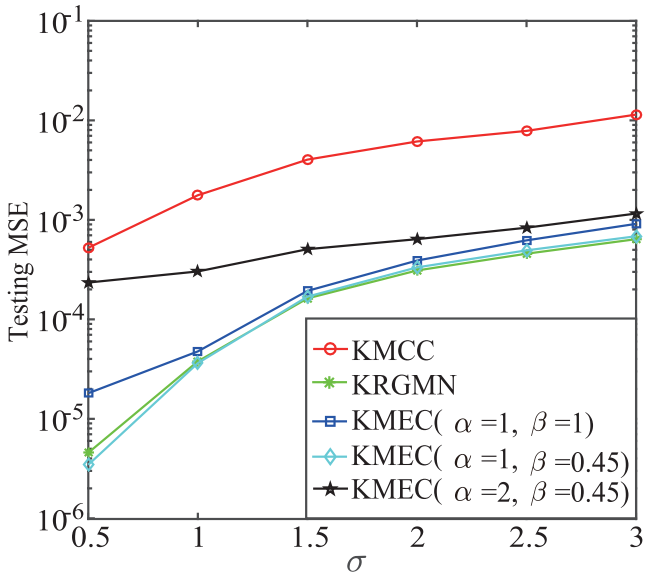


FIGURE 8. Performance comparison of the KMEC with different values of σ.

can exhibit better tracking ability under non-Gaussian noise environments.

Then, different kernel bandwidth, namely $\sigma = \{0.5, 1, 1.5, 2, 2.5, 3\}$ are used to further evaluate the performance of the KMEC algorithm. All the parameter settings are same with the latest experiment, and the performance is given in Fig.8 for various σ. It is found that the performance of the proposed KMEC algorithm is deteriorated when σ increases from 0.5 to 3. The KMEC has the lowest MSE for $\sigma = 0.5$ because of the GMCC scheme. Also, we can see that the performance of the KMEC is better than KRGMN within the range of [0.5, 1].

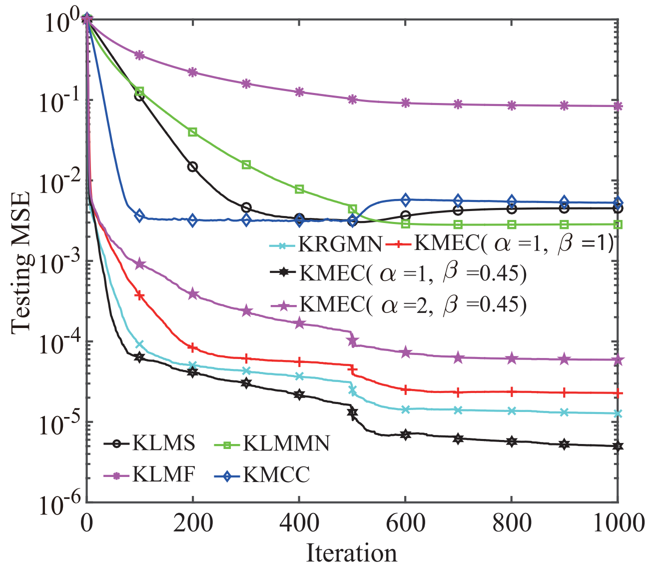


FIGURE 9. Ensemble learning curves of different algorithms with an abrupt change at the 500th iteration.

C. TRACKING OF THE PROPOSED KMEC ALGORITHM

In this experiment, the channel is changed after 500 iterations to observe the tracking performance of the KMEC algorithm and the measurement noise is Laplace and Uniform distributed noise. The channel is changed to be $r(i) = -x(i)+0.9x^2(i-1)+n(i)$ for the second 500 iterations. The performance in this case is shown in Fig.9. We can see that the performance of the KMEC algorithm is still better than other algorithms even though the learning lines are changed at 500th iteration.

V. CONCLUSIONS

In this paper, a novel kernel mixed error criterion (KMEC) algorithm has been proposed for nonlinear channel equalization under non-Gaussian noise environments. The KMEC algorithm is realized based on the kernel method in the combination of the logarithmic squared error and GMCC scheme. The simulation results verified that the KMEC algorithm achieves best performance in terms of the convergence speed and the MSE in comparison with the previously proposed kernel algorithms.

REFERENCES

[1] W. Liu, J. C. Príncipe, and S. Haykin, *Kernel Adaptive Filtering: A Comprehensive Introduction*. Hoboken, NJ, USA: Wiley, 2010.

[2] W. Liu, P. P. Pokharel, and J. C. Príncipe, "The kernel least-mean-square algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 543–554, Feb. 2008.

[3] Y. Yu, H. Zhao, R. C. de Lamare, and L. Lu, "Sparsity-aware subband adaptive algorithms with adjustable penalties," *Digit. Signal Process.*, vol. 84, pp. 93–106, Jan. 2019.

[4] L. Lu, H. Zhao, and B. Chen, "Time series prediction using kernel adaptive filter with least mean absolute third loss function," *Nonlinear Dyn.*, vol. 90, no. 2, pp. 999–1013, 2017.

[5] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2275–2285, Aug. 2004.

[6] B. Chen, S. Zhao, P. Zhu, and J. C. Príncipe, "Quantized kernel least mean square algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 1, pp. 22–32, Jan. 2012.

[7] B. Chen, S. Zhao, P. Zhu, and J. C. Príncipe, "Quantized kernel recursive least squares algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 9, pp. 1484–1491, Sep. 2013.

[8] W. Liu, I. Park, and J. C. Príncipe, "An information theoretic approach of designing sparse kernel adaptive filters," *IEEE Trans. Neural Netw.*, vol. 20, no. 12, pp. 1950–1961, Dec. 2009.

[9] W. Liu, I. Park, Y. Wang, and J. C. Príncipe, "Extended kernel recursive least squares algorithm," *IEEE Trans. Signal Process.*, vol. 57, no. 10, pp. 3801–3814, Oct. 2009.

[10] S. Wang, Y. Zheng, and C. Ling, "Regularized kernel least mean square algorithm with multiple-delay feedback," *IEEE Signal Process. Lett.*, vol. 23, no. 1, pp. 98–101, Jan. 2016.

[11] J. Zhao, X. Liao, S. Wang, and C. K. Tse, "Kernel least mean square with single feedback," *IEEE Signal Process. Lett.*, vol. 22, no. 7, pp. 953–957, Jul. 2015.

[12] B. Chen, J. Liang, N. Zheng, and J. C. Príncipe, "Kernel least mean square with adaptive kernel size," *Neurocomputing*, vol. 191, pp. 95–106, May 2016.

[13] L. Shi, Y. Lin, and X. Xie, "Combination of affine projection sign algorithms for robust adaptive filtering in non-Gaussian impulsive interference," *Electron. Lett.*, vol. 50, no. 6, pp. 466–467, Mar. 2014.

[14] K. Pelekanakis and M. Chitre, "Adaptive sparse channel estimation under symmetric α-stable noise," *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3183–3195, Jun. 2014.

[15] N. A. M. Verhoeckx and T. A. C. M. Claasen, "Some considerations on the design of adaptive digital filters equipped with the sign algorithm," *IEEE Trans. Commun.*, vol. 32, no. 3, pp. 258–266, Mar. 1984.

[16] E. Walach and B. Widrow, "The least mean fourth (LMF) adaptive algorithm and its family," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 2, pp. 275–283, Mar. 1984.

[17] Y. Li, Y. Wang, and T. Jiang, "Sparse least mean mixed-norm adaptive filtering algorithms for sparse channel estimation applications," *Int. J. Commun. Syst.*, vol. 30, no. 8, p. e3181, May 2017.

[18] Y. Li, Y. Wang, and T. Jiang, "Norm-adaption penalized least mean square/fourth algorithm for sparse channel estimation," *Signal Process.*, vol. 128, pp. 243–251, Nov. 2016.

[19] A. Singh and J. C. Príncipe, "Using correntropy as a cost function in linear adaptive filters," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Atlanta, GA, USA, Jun. 2009, pp. 2950–2955.

[20] S. Zhao, B. Chen, and J. C. Príncipe, "Kernel adaptive filtering with maximum correntropy criterion," in *Proc. Int. Joint Conf. Neural Netw.*, San Jose, CA, USA, Aug. 2011, pp. 2012–2017.

[21] Y. Li, Z. Jiang, W. Shi, X. Han, and B. Chen, "Blocked maximum correntropy criterion algorithm for cluster-sparse system identifications," *IEEE Trans. Circuits Syst., II, Exp. Briefs*, to be published. doi: 10.1109/TCSII.2019.2891654.

[22] R. He, W.-S. Zheng, and B.-G. Hu, "Maximum correntropy criterion for robust face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1561–1576, Aug. 2011.

[23] B. Chen, L. Xing, H. Zhao, N. Zheng, and J. C. Príncipe, "Generalized correntropy for robust adaptive filtering," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3376–3387, Jul. 2016.

[24] W. Ma, J. Duan, W. Man, J. Liang, and B. Chen, "General mixed-norm-based diffusion adaptive filtering algorithm for distributed estimation over network," *IEEE Access*, vol. 5, pp. 1090–1102, Jan. 2017.

[25] S. Yu *et al.*, "Generalized kernel normalized mixed-norm algorithm: Analysis and simulations," in *Proc. Int. Conf. Neural Inf. Process.*, Istanbul, Turkey, Nov. 2015, pp. 61–70.

[26] O. Tanrikulu and J. A. Chambers, "Convergence and steady-state properties of the least-mean mixed-norm (LMMN) adaptive algorithm," *IEE Proc-Vis. Image Signal Process.*, vol. 143, no. 3, pp. 137–142, Jun. 1996.

[27] J. Chambers and A. Avlonitis, "A robust mixed-norm adaptive filter algorithm," *IEEE Signal Process. Lett.*, vol. 4, no. 2, pp. 46–48, Feb. 1997.

[28] J. A. Chambers, O. Tanrikulu, and A. G. Constantinides, "Least mean mixed-norm adaptive filtering," *Electron. Lett.*, vol. 30, no. 19, pp. 1574–1575, Sep. 1994.

[29] S.-C. Pei and C.-C. Tseng, "Least mean p-power error criterion for adaptive FIR filter," *IEEE J. Sel. Areas Commun.*, vol. 12, no. 9, pp. 1540–1547, Dec. 1994.

[30] Q. Y. Miao and C. G. Li, "Kernel least-mean mixed-norm algorithm," in *Proc. Int. Conf. Autom. Control Artif. Intell. (ACAI)*, Xiamen, China, Mar. 2012, pp. 1285–1288.

[31] X. Luo, J. Deng, J. Liu, A. Li, W. Wang, and W. Zhao, "A novel entropy optimized kernel least-mean mixed-norm algorithm," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Vancouver, BC, Canada, Jul. 2016, pp. 1716–1722,

[32] W. Ma, X. Qiu, J. Duan, Y. Li, and B. Chen, "Kernel recursive generalized mixed norm algorithm," *J. Franklin Inst.*, vol. 355, no. 4, pp. 1596–1613, Mar. 2018.

[33] S. Zhao, B. Chen, and J. C. Príncipe, "Kernel adaptive filtering with maximum correntropy criterion," in *Proc. Int. Joint Conf. Neural Netw.*, Aug. 2011, pp. 2012–2017.

[34] V. Z. Filipovic, "Consistency of the robust recursive hammerstein model identification algorithm," *J. Franklin Inst.*, vol. 352, no. 5, pp. 1932–1945, May 2015.

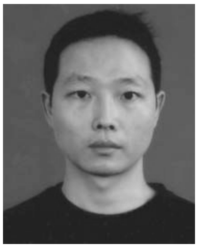
QISHUAI WU was born in Taonan, Jilin, China, in 1995. He received the B.S. degree in information engineering from Jilin University, in 2018. He is currently pursuing the M.S. degree in information and communication engineering with Harbin Engineering University. His research interests include the signal processing, and kernel adaptive filtering algorithms and applications.

YINGSONG LI (S'09–M'14–SM'19) received the B.S. degree in electrical and information engineering and the M.S. degree in electromagnetic field and microwave technology from Harbin Engineering University, in 2006 and 2011, respectively, and the Ph.D. degree from both Kochi University of Technology (KUT), Japan, and Harbin Engineering University, China, in 2014.

He was a Visiting Scholar with the University of California, Davis, from 2016 to 2017, and with University of York, U.K., in 2018. He has been a Full Professor with Harbin Engineering University, since 2014. He is also a Visiting Professor of Far Eastern Federal University (FEFU) and KUT. His recent research interests include remote sensing, underwater communications, signal processing, radar, SAR imaging, compressed sensing, and antennas.

Dr. Li is a Senior Member of the Chinese Institute of Electronics (CIE). He is an Associate Editor of the IEEE Access and *Applied Computational Electromagnetics Society Journal*, and is also an Area Editor of *AEü-International Journal of Electronics and Communications*. He serves as a reviewer for more than 20 journals.

ZHENGXIONG JIANG received the B.S. degree in electrical and information engineering from Harbin Engineering University, Harbin, China, in 2016, where he is currently pursuing the M.E. degree in information and communication engineering. His research interests mainly focus on the signal processing, and kernel adaptive filtering algorithms and applications.



YOUWEN ZHANG received the B.S. degree in computer application technology from the Wuhan University of Economics and Business, Wuhan, Hubei, China, in 1997, and the M.Sc. degree in signal and information processing and the Ph.D. degree in underwater acoustical engineering from the Harbin Engineering University, Harbin, Heilongjiang, China, in 2004 and 2005, respectively.

He was an Assistant Professor with the College of Underwater Acoustic Engineering, Harbin Engineering University, Harbin, from 2005 to 2010, where he has been an Associate Professor, since 2010. His research interests include underwater acoustic communications and networking, and the array signal processing in underwater acoustic environments.