

Accepted Manuscript

Fixed-Point Generalized Maximum Correntropy: Convergence
Analysis and Convex Combination Algorithms

Ji Zhao, Hongbin Zhang, Gang Wang

PII: S0165-1684(18)30209-3
DOI: [10.1016/j.sigpro.2018.06.012](https://doi.org/10.1016/j.sigpro.2018.06.012)
Reference: SIGPRO 6849

To appear in: *Signal Processing*

Received date: 12 February 2018
Revised date: 8 June 2018
Accepted date: 12 June 2018

Please cite this article as: Ji Zhao, Hongbin Zhang, Gang Wang, Fixed-Point Generalized Maximum Correntropy: Convergence Analysis and Convex Combination Algorithms, *Signal Processing* (2018), doi: [10.1016/j.sigpro.2018.06.012](https://doi.org/10.1016/j.sigpro.2018.06.012)

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



Highlights

- A fixed-point algorithm is proposed to estimation the maximum of generalized correntropy (termed FP-GMC).
- A sufficient condition is obtained for the convergence of the FP-GMC algorithm.
- The sliding-window method and recursive method are applied to the FP-GMC algorithm for online signal processing. And, call these online algorithms as SW-GMC and RGMC, respectively.
- A convex combination algorithm is proposed by adaptively combine two RGMC algorithms to improve the convergence rate of RGMC algorithm. And, call this combination algorithm as AC-RGMC.
- The convergence rate of the AC-RGMC has been further increased by a simple and efficient weight control scheme. And, call this control algorithm as AC-RGMC-C.

Fixed-Point Generalized Maximum Correntropy: Convergence Analysis and Convex Combination Algorithms

Ji Zhao^a, Hongbin Zhang^{a,*}, Gang Wang^a

^a*School of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu, 611731, China*

Abstract

Compared with the MSE criterion, the generalized maximum correntropy (GMC) criterion shows a better robustness against impulsive noise. Some gradient based GMC adaptive algorithms have been derived and available for practice. But, the fixed-point algorithm on GMC has not yet been well studied in the literature. In this paper, we study a fixed-point GMC (FP-GMC) algorithm for linear regression, and derive a sufficient condition to guarantee the convergence of the FP-GMC. Also, we apply sliding-window and recursive methods to the FP-GMC to derive online algorithms for practice, these two called sliding-window GMC (SW-GMC) and recursive GMC (RGMC) algorithms, respectively. Since the solution of RGMC is not analyzable, we derive some approximations that fundamentally result in the poor **convergence** rate of the RGMC in nonstationary situations. To overcome this issue, we propose a novel robust filtering algorithm (termed adaptive convex combination of RGMC algorithms (AC-RGMC)), which relies on the convex combination of two RGMC algorithms with different memories. Moreover, by an efficient weight control method, the tracking performance of the AC-RGMC is further improved, and this new one is called AC-RGMC-C algorithm. The good performance of proposed algorithms are tested in plant identification scenarios with abrupt change under impulsive noise environment.

Keywords: Convergence, Fixed-point algorithm, Convex combination, Adaptive filter, Non-Gaussian noise, Generalized maximum correntropy (GMC) criterion

1. Introduction

Adaptive filtering algorithms (AFAs) have been successfully applied in various fields such as system identification, channel equation, acoustic echo cancelation, active noise control, and so forth [1, 2, 3]. The most existing AFAs are based on the Gaussian scenarios justified by the central limit theorem, such as the family of least mean-square (LMS) algorithms, the class of affine projection algorithms (APA), and the tribe of recursive least squares (RLS) algorithms [3, 4]. Among these algorithms, the LMS is the most widely used filtering algorithm because of its simplicity, the

*Corresponding author

Email address: zhanghb@uestc.edu.cn (Hongbin Zhang)

RLS can accelerate the convergence rate of the LMS in the presence of colored input signals, and the APA appears as intermediate complexity between the LMS and the RLS.

Note that almost all aforementioned algorithms performances may degrade dramatically under non-Gaussian distributions, such as the light-tailed (e.g., binary, uniform, etc.) and the fat-tailed (e.g., Laplace, Cauchy, mixed Gaussian, alpha-stable, etc.) distributions [5, 6, 7, 8, 9, 10]. Generally, different p -powers of the error signal, i.e., $|e|^p$, are used as cost functions to obtain robust algorithms [5, 11, 12, 13]. Specifically, when the desired signals are contaminated by the non-Gaussian interferences with light-tailed distribution, a high-order power of the error is usually more desirable to achieve a better tradeoff between the transient and steady-state performance. For example, the least mean absolute third (LMAT) algorithm with $p = 3$ [14], and the least mean fourth (LMF) algorithm with $p = 4$ [15]. On the other hand, for these heavy-tailed impulsive interferences, a lower-order power of the error with $p < 2$ is usually more robust. For example, the LMS type algorithms (e.g., the least mean p -powers algorithm (LMP), especially the sign algorithm with $p = 1$) [16], the APA type algorithms (e.g., the affine projection sign algorithm (APSA)) [17, 18], and the RLS type algorithms (e.g., the recursive least mean p -powers algorithm (RLMP)) [19, 20].

In addition, based on the information theoretical learning (ITL), the minimum error entropy (MEE) and maximum correntropy (MC) criterion have been developed as alternative robust and efficient cost criteria for non-Gaussian signal processing and machine learning [21, 22], such as the stochastic information gradient algorithms [23, 24, 25, 26, 27, 28], and the recursive adaptive algorithms [29, 30, 31]. However, the MEE possesses heavier computational complexity than that of the MC, and more relations between the MEE and MC can be found in [32, 33]. Moreover, the MC is only based on the Gaussian kernel, while this kernel is not always the best choice (i.e., other Mercer kernels may be used to define the correntropy). In recent years, a generalized maximum correntropy (GMC) criterion has been proposed and applied for robust signal processing [6, 7, 34]. However, an analytical solution of the GMC (the MC is a special case of the GMC) cannot be derived even with a linear model (e.g., a finite impulse response (FIR) filter). A practical approach to maximize the GMC criterion is to update the solution by an iterative algorithm. Usually, a simple gradient based search algorithm is applied. For instance, the GMC filtering algorithm and the convergence of it has been studied in [6]. Alternatively, the fixed-point iterative algorithm can be used to update the solution of the GMC, which is step-size free and usually converges to the solution more quickly [7, 34]. Nevertheless, for the fixed-point GMC (FP-GMC) algorithms, up to now there is still no study concerning the convergence. In this work, we study, firstly, the convergence of a FP-GMC algorithm and derive a sufficient condition to guarantee the convergence of this fixed-pointed algorithm. Also, we apply the sliding-window approach and recursive approach to the FP-GMC to derive the corresponding online algorithms for practical use, and call these online ones as the sliding-window GMC (SW-GMC) and recursive GMC (RGMC) algorithms, respectively.

Besides the robustness of the AFAs, the convergence rate is another critical role for filtering algorithms. There are numerous variable step-size strategies to improve the convergence rate of the LMS type or APA type filtering algorithms

[35, 36, 37, 38]. Moreover, the convex combination of adaptive filters is another interesting and efficient way to improve the performance of adaptive filters [1, 25, 39, 40, 41]. However, most existing combination filters are motivated by the idea of combing different filters with different step-sizes to offer complementary capabilities. By combining two RGMC algorithms with different memories, we obtain a novel adaptive filtering algorithm, termed adaptive convex combination of RGMC algorithms (AC-RGMC) to overcome the slower **convergence** rate of the RGMC when the abrupt change occurs, that issue was not properly discussed in [7]. Furthermore, we apply a simple and efficient method to further modify the **convergence** rate of the AC-RGMC algorithm, and call the new algorithm as AC-RGMC with control scheme (AC-RGMC-C).

The rest of the paper is organized as follows. In Section 2, after briefly reviewing the background of alpha-stable distribution (α -SD) which can model such types of non-Gaussian noises, and the concept of the GMC criterion, we derive a fixed-point GMC algorithm and present a sufficient condition for guaranteeing the convergence of the FP-GMC. In Section 3, we apply the sliding-window method and the recursive method to obtain the solutions of the FP-GMC algorithm for practical application, and we also explain the reason why this work uses the combination strategy to propose a novel adaptive filtering algorithm. We derive a adaptive convex combination of RGMC algorithms in Section 4. Section 5 shows the simulation results in nonstationary scenarios with impulsive noise modeled by the α -SD. Finally, conclusions are given in Section 6.

Notation: Throughout this paper, the superscript T denotes the transpose; the small or capital italic letters denote scalar variables, such as: a , b , A and B ; the small bold letters denote vectors, e.g., $\boldsymbol{\omega} \in \mathbb{R}^l$, where \mathbb{R} means the real-value set; the capital bold letters denote matrices, e.g., $\mathbf{R} \in \mathbb{R}^{l \times l}$; the notation $\|\cdot\|_p$ is an l_p -norm of a vector or an induced norm of a matrix defined by $\|\mathbf{M}\|_p = \max_{\|\mathbf{u}\|_p \neq 0} \frac{\|\mathbf{M}\mathbf{u}\|_p}{\|\mathbf{u}\|_p}$ with $p \geq 1$, $\mathbf{M} \in \mathbb{R}^{l \times l}$, and $\mathbf{u} \in \mathbb{R}^l$.

2. Fixed-point Algorithm for GMC Estimation and It's Convergence Analysis

2.1. Alpha-stable Distribution

In this work, the impulsive noise is modeled by alpha-stable distribution (α -SD) with following characteristic function [5]

$$\phi(t) = \exp\{i\delta t - \lambda|t|^\alpha[1 + i\rho \operatorname{sgn}(t)s(t, \alpha)]\} \quad (1)$$

in which

$$s(t, \alpha) = \begin{cases} \tan \frac{\alpha\pi}{2} & \text{if } \alpha \neq 1 \\ \frac{2}{\pi} \log |t| & \text{if } \alpha = 1 \end{cases} \quad (2)$$

where $i = \sqrt{-1}$; $\operatorname{sgn}(\cdot)$ is the sign function; $\alpha \in (0, 2]$ denotes the characteristic factor which measures the tail heaviness of the distribution; $\delta \in (-\infty, +\infty)$ denotes the location parameter; $\lambda > 0$ is the dispersion parameter and plays a role similar to the variance of the Gaussian distribution; and $\rho \in [-1, 1]$ means the symmetry parameter. As we can see

65 that, when $\alpha = 2$ and $\rho = 0$, the α -SD is equivalent to Gaussian distribution and the λ equals to the variance. For the sake of simplicity, we collect the parameters of the α -SD in a vector, namely, $\mathbf{p}_\alpha = [\alpha, \delta, \lambda, \rho]$.

2.2. Generalized Maximum correntropy criterion

The correntropy between two random variables X and Y is a local similarity measure defined by [24]

$$V(X, Y) = E[\kappa(X - Y)] = \int \kappa(x - y) dF_{x,y}(x, y) \quad (3)$$

where $E[\cdot]$ denotes the expectation operator; $\kappa(\cdot)$ stands for a Mercer kernel; and $F_{x,y}(x, y)$ is the joint distribution function of X and Y . Generally, the following Gaussian kernel is chosen as the kernel function in (3)

$$\kappa_\sigma(x - y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|x - y|^2}{2\sigma^2}\right) \quad (4)$$

where $\sigma > 0$ denotes the kernel size and $1/(\sqrt{2\pi}\sigma)$ is the normalization parameter. Practically, the joint distribution $F_{x,y}(x, y)$ is estimated by the Parzen kernel estimator based on available data $\{x_n, y_n\}_{n=1}^N$ of the variables X and Y , i.e.,

$$\hat{V}_N^\sigma(X, Y) = \frac{1}{N} \sum_{n=1}^N \kappa_\sigma(x_n - y_n) \quad (5)$$

The estimator (5) has been widely used as a new cost function for adaptive systems, because, in statistical meaning, one obtains the maximum correntropy (MC) of error yielding the maximum error probability density at the origin [21]. However, the Gaussian kernel used in (5) maybe not always the best selection. Recently, a more reasonable kernel function based on the generalized Gaussian density is defined as

$$\kappa_{s,t}(x - y) = \frac{s}{2t\Gamma(s-1)} \exp\left(-\frac{|x - y|^s}{t^s}\right) \quad (6)$$

where s and t are positive numbers and present the shape parameter and the scale parameter, respectively; $\Gamma(\cdot)$ is the gamma function; t^{-s} is the kernel parameter; and $s/(2t\Gamma(s-1))$ denotes the normalization constant. Similar to (5), a new Parzen estimator is

$$\hat{V}_N^{s,t}(X, Y) = \frac{1}{N} \sum_{n=1}^N \kappa_{s,t}(x_n - y_n) \quad (7)$$

70 which results in a new correntropy called generalized correntropy (GC). The GC has some well properties presented in [6], and based on these properties, we can use the GC of error as a cost function for adaptation (called the generalized MC (GMC)) [6, 7, 34].

2.3. Fixed-point GMC Algorithm

In this work, we consider the input vector $\mathbf{u}(n) = [u_n, u_{n-1}, \dots, u_{n-l+1}]^T$ passing through an FIR system with intrinsic weight vector $\boldsymbol{\omega}^o = [\omega_1^o, \omega_2^o, \dots, \omega_l^o]^T$, where l denotes the length of the memory. Therefore, the system output is

$$d(n) = \mathbf{u}(n)^T \boldsymbol{\omega}^o + \nu(n) \quad (8)$$

where $\nu(n)$ denotes an additive α -SD noise. In order to estimate the weight vector $\boldsymbol{\omega}^o$, following weighted cost function is optimized by an exponentially-weighted mechanism to put more emphasis on recent sample and to deemphasis on sample from the remote past

$$\begin{aligned} J(\boldsymbol{\omega}) &= \sum_{i=1}^N \beta^{N-i} \exp(-\tau |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}|^s) \\ &= \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) \end{aligned} \quad (9)$$

where $\tau = t^{-s}$ denotes a kernel parameter; $e(i) = d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}$ is the estimation error; $\beta \in (0, 1]$ is a forgetting factor; and $\boldsymbol{\omega} = [\omega_1, \omega_2, \dots, \omega_l]^T$ denotes the estimated weight vector. Note that, for simplicity, the normalization constant $s/(2t\Gamma(s^{-1}))$ in (6) has been removed in (9).

Taking the gradient of $J(\boldsymbol{\omega})$ with respect to the $\boldsymbol{\omega}$, we have

$$\begin{aligned} \frac{\partial J(\boldsymbol{\omega})}{\partial \boldsymbol{\omega}} &= \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) \tau s |e(i)|^{s-1} \text{sgn}(e(i)) \mathbf{u}(i) \\ &\stackrel{(a)}{=} \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) \tau s |e(i)|^{s-2} e(i) \mathbf{u}(i) \\ &= \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) \tau s |e(i)|^{s-2} d(i) \mathbf{u}(i) \\ &\quad - \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) \tau s |e(i)|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T \boldsymbol{\omega} \end{aligned} \quad (10)$$

where (a) uses the fact $e(i) = |e(i)| \text{sgn}(e(i))$. Setting this gradient equal to null vector, we obtain

$$\begin{aligned} \boldsymbol{\omega} &= \left[\sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) |e(i)|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T \right]^{-1} \\ &\quad \times \sum_{i=1}^N \beta^{N-i} \exp(-\tau |e(i)|^s) |e(i)|^{s-2} d(i) \mathbf{u}(i) \end{aligned} \quad (11)$$

To derive concisely, we introduce some symbols as

$$\begin{cases} \mathbf{R}_U = \sum_{i=1}^N \beta^{N-i} f(i) \mathbf{u}(i) \mathbf{u}(i)^T \\ \mathbf{z}_d = \sum_{i=1}^N \beta^{N-i} f(i) d(i) \mathbf{u}(i) \\ f(i) = \exp(-\tau |e(i)|^s) |e(i)|^{s-2} \end{cases} \quad (12)$$

where \mathbf{R}_U denotes a weighted autocorrelation matrix of the input data, and \mathbf{z}_d is a weighted cross correlation vector between the desired and the input signals. Then, we obtain the matrix form of (11) as

$$\boldsymbol{\omega} = [\mathbf{R}_U]^{-1} \mathbf{z}_d \quad (13)$$

The above solution is, in form, very similar to the RLS update rule that tracks the Wiener solution with every update. As one can see, since both the matrix \mathbf{R}_U and the vector \mathbf{z}_d are themselves functions of $\boldsymbol{\omega}$ through $f(i)$, (13) is not a closed form solution. Hence, the solution of (13) is actually a fixed-point equation, which can be represented as

$$\boldsymbol{\omega} = \mathbf{f}_p(\boldsymbol{\omega}) = [\mathbf{R}_U]^{-1} \mathbf{z}_d \quad (14)$$

In practice, we can apply the following iterative fixed-point algorithm to get the solution of (14) as

$$\boldsymbol{\omega}(n) = \mathbf{f}_p(\boldsymbol{\omega}(n-1)) \quad (15)$$

75 where $\boldsymbol{\omega}(n)$ means the estimated weight vector at instance n . And, we call this algorithm as the fixed-point GMC algorithm (FP-GMC). In this work, for tractability we assume that the matrix \mathbf{R}_U is invertible with $\lambda_{\min}[\mathbf{R}_U] > 0$, where $\lambda_{\min}[\cdot]$ means the minimum eigenvalue of a matrix. Following, we will derive a sufficient condition under which the FP-GMC algorithm surely converges to a unique fixed-point.

2.4. Convergence of the FP-GMC algorithm

In mathematics, the Banach Fixed-Point Theorem (also known as the contraction mapping theorem) is a standard method for proving the convergence of a fixed-point algorithm [42, 43]. Based on the Banach Fixed-Point Theorem, the convergence of the FP-GMC is guaranteed if $\exists A > 0$ and $0 < B < 1$ such that the initial weight vector $\|\boldsymbol{\omega}(0)\|_p \leq A$, and $\forall \boldsymbol{\omega} \in \{\boldsymbol{\omega} \in \mathbb{R}^l : \|\boldsymbol{\omega}\|_p \leq A\}$, it holds that

$$\begin{cases} \|\mathbf{f}_p(\boldsymbol{\omega})\|_p \leq A \\ \|\nabla_{\boldsymbol{\omega}} \mathbf{f}_p(\boldsymbol{\omega})\|_p \leq B \end{cases} \quad (16)$$

where $\nabla_{\boldsymbol{\omega}} \mathbf{f}_p(\boldsymbol{\omega})$ is the $l \times l$ Jacobian matrix of $\mathbf{f}_p(\boldsymbol{\omega})$ with respect to $\boldsymbol{\omega}$, denoted by

$$\nabla_{\boldsymbol{\omega}} \mathbf{f}_p(\boldsymbol{\omega}) = \left[\frac{\partial}{\partial \omega_1} f_p(\boldsymbol{\omega}), \frac{\partial}{\partial \omega_2} f_p(\boldsymbol{\omega}), \dots, \frac{\partial}{\partial \omega_l} f_p(\boldsymbol{\omega}) \right] \quad (17)$$

where

$$\begin{aligned} \frac{\partial}{\partial \omega_j} f_p(\boldsymbol{\omega}) &= \frac{\partial}{\partial \omega_j} ([\mathbf{R}_U]^{-1} \mathbf{z}_d) \\ &= -[\mathbf{R}_U]^{-1} \left(\frac{\partial}{\partial \omega_j} \mathbf{R}_U \right) [\mathbf{R}_U]^{-1} \mathbf{z}_d + [\mathbf{R}_U]^{-1} \left(\frac{\partial}{\partial \omega_j} \mathbf{z}_d \right) \\ &= -[\mathbf{R}_U]^{-1} \left(\sum_{i=1}^N \beta^{N-i} g_e(i) u_j(i) f(i) \mathbf{u}(i) \mathbf{u}(i)^T \right) \mathbf{f}_p(\boldsymbol{\omega}) \\ &\quad + [\mathbf{R}_U]^{-1} \left(\sum_{i=1}^N \beta^{N-i} g_e(i) u_j(i) f(i) d(i) \mathbf{u}(i) \right) \end{aligned} \quad (18)$$

80 where $j \in \{1, 2, \dots, l\}$ and $g_e(i) = \tau s |e(i)|^{s-2} e(i) - (s-2) \text{sgn}(e(i)) |e(i)|^{-1}$. Before deriving a sufficient condition to guarantee the convergence of the FP-GMC algorithm, following two theorems shall be proven.

Theorem 1. If $0 < s \leq 2$, $|d(i)| \geq A\|\mathbf{u}(i)\|_1$ with $i \in \{1, 2, \dots, N\}$, and

$$A > \xi = \frac{\sqrt{L} \sum_i^N \beta^{N-i} (|d(i)| - A\|\mathbf{u}(i)\|_1)^{s-2} |d(i)| \|\mathbf{u}(i)\|_1}{\lambda_{\min} [\sum_i^N \beta^{N-i} |\Delta|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T]}$$

where $\Delta = |d(i)| + A\|\mathbf{u}(i)\|_1$, and $\tau \leq \tau^*$, where τ^* is the solution of equation $\varphi(\tau) = A$, where

$$\varphi(\tau) = \frac{\sqrt{l} \sum_i^N \beta^{N-i} (|d(i)| - A\|\mathbf{u}(i)\|_1)^{s-2} |d(i)| \|\mathbf{u}(i)\|_1}{\lambda_{\min} [\sum_i^N \beta^{N-i} \exp(-\tau|\Delta|^s) |\Delta|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T]} \quad (19)$$

Then $\|F(\boldsymbol{\omega})\|_1 \leq A$ for all $\boldsymbol{\omega} \in \{\boldsymbol{\omega} \in \mathbb{R}^l : \|\boldsymbol{\omega}\|_1 \leq A\}$.

Proof. Actually, the induced matrix norm is compatible with the corresponding vector l_p -norm, and we get

$$\|\mathbf{f}_p(\boldsymbol{\omega})\|_1 = \|[\mathbf{R}_U]^{-1} \mathbf{z}_d\|_1 \leq \|[\mathbf{R}_U]^{-1}\|_1 \|\mathbf{z}_d\|_1 \quad (20)$$

where $\|[\mathbf{R}_U]^{-1}\|_1$ denotes the column-sum norm of the inverse matrix $[\mathbf{R}_U]^{-1}$, which is simply the maximum absolute column sum of the matrix. Based on the matrix theory, we have following inequality

$$\|[\mathbf{R}_U]^{-1}\|_1 \leq \sqrt{l} \|[\mathbf{R}_U]^{-1}\|_2 = \sqrt{l} \lambda_{\max} [[\mathbf{R}_U]^{-1}] \quad (21)$$

where $\|[\mathbf{R}_U]^{-1}\|_2$ denotes the spectral norm of $[\mathbf{R}_U]^{-1}$, which equals the maximum eigenvalue of the matrix. Further, we have

$$\begin{aligned} \lambda_{\max} [[\mathbf{R}_U]^{-1}] &= \frac{1}{\lambda_{\min} [\mathbf{R}_U]} \\ &= \frac{1}{\lambda_{\min} [\sum_{i=1}^N \beta^{N-i} f(i) \mathbf{u}(i) \mathbf{u}(i)^T]} \\ &\stackrel{(b)}{\leq} \frac{1}{\lambda_{\min} [\sum_{i=1}^N \beta^{N-i} \exp(-\tau|\Delta|^s) |\Delta|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T]} \end{aligned} \quad (22)$$

where (b) comes from

$$\begin{cases} |e(i)| = |d(i) - \boldsymbol{\omega}^T \mathbf{u}(i)| \leq |d(i)| + A\|\mathbf{u}(i)\|_1 = \Delta \\ f(i) = \exp(-\tau|e(i)|^s) |e(i)|^{s-2} \geq \exp(-\tau|\Delta|^s) |\Delta|^{s-2} \end{cases} \quad (23)$$

Moreover, we have

$$\begin{aligned} \|\mathbf{z}_d\|_1 &= \left\| \sum_{i=1}^N \beta^{N-i} f(i) d(i) \mathbf{u}(i) \right\|_1 \\ &\stackrel{(c)}{\leq} \sum_{i=1}^N \beta^{N-i} |f(i)| |d(i)| \|\mathbf{u}(i)\|_1 \\ &\stackrel{(d)}{\leq} \sum_{i=1}^N \beta^{N-i} (|d(i)| - A\|\mathbf{u}(i)\|_1)^{s-2} |d(i)| \|\mathbf{u}(i)\|_1 \end{aligned} \quad (24)$$

where (c) follows from the convexity of the vector l_1 -norm, and (d) comes from

$$\begin{aligned} |f(i)| &= |\exp(-\tau|e(i)|^s)|e(i)|^{s-2} \\ &\leq |e(i)|^{s-2} \\ &\leq (|d(i) - A\|\mathbf{u}(i)\|_1|)^{s-2} \end{aligned} \quad (25)$$

Combining (20),(21),(22) and (24), we obtain

$$\|\mathbf{f}_p(\boldsymbol{\omega})\|_1 \leq \frac{\sqrt{l} \sum_i^N \beta^{N-i} (|d(i)| - A\|\mathbf{u}(i)\|_1)^{s-2} |d(i)| \|\mathbf{u}(i)\|_1}{\lambda_{\min}[\sum_i^N \beta^{N-i} \exp(-\tau|\Delta|^s) |\Delta|^{s-2} \mathbf{u}(i)\mathbf{u}(i)^T]} = \varphi(\tau) \quad (26)$$

Clearly, the function $\varphi(\tau)$ is a continuous and monotonically increasing function of $\tau \in (0, \infty)$, satisfying $\lim_{\tau \rightarrow \infty} \varphi(\tau) = \infty$ and $\lim_{\tau \rightarrow 0^+} \varphi(\tau) = \xi$. Therefore, if $A > \xi$, the equation $\varphi(\tau) = A$ will get a unique solution $\tau^* \in (0, \infty)$, and if $\tau < \tau^*$, we obtain $\varphi(\tau) < A$, which completes the proof. \square

Theorem 2. If $0 < s \leq 2$, $|d(i)| \geq A\|\mathbf{u}(i)\|_1$ with $i \in \{1, 2, \dots, N\}$, and

$$A > \xi = \frac{\sqrt{L} \sum_i^N \beta^{N-i} (|d(i)| - A\|\mathbf{u}(i)\|_1)^{s-2} |d(i)| \|\mathbf{u}(i)\|_1}{\lambda_{\min}[\sum_i^N \beta^{N-i} |\Delta|^{s-2} \mathbf{u}(i)\mathbf{u}(i)^T]}$$

where $\Delta = |d(i)| + A\|\mathbf{u}(i)\|_1$, and $\tau \leq \min(\tau^*, \tau^\diamond)$, where τ^* is the solution of equation $\varphi(\tau) = A$, and τ^\diamond is the solution of equation $\psi(\tau) = B$ ($0 < B < 1$), where

$$\psi(\tau) = \frac{\sqrt{l} \sum_i^N \beta^{N-i} \mathcal{D}(i) \|\mathbf{u}(i)\|_1 [A\|\mathbf{u}(i)\mathbf{u}(i)^T\|_1 + \|d(i)\mathbf{u}(i)\|_1]}{\lambda_{\min}[\sum_i^N \beta^{N-i} \exp(-\tau|\Delta|^s) |\Delta|^{s-2} \mathbf{u}(i)\mathbf{u}(i)^T]} \quad (27)$$

in which

$$\mathcal{D}(i) = \frac{\tau s (|d(i)| + A\|\mathbf{u}(i)\|_1)^s + 2 - s}{(|d(i)| - A\|\mathbf{u}(i)\|_1)^{3-s}} \quad (28)$$

Then $\|\mathbf{f}_p(\boldsymbol{\omega})\|_1 \leq A$, and $\|\nabla_{\boldsymbol{\omega}} \mathbf{f}_p(\boldsymbol{\omega})\|_1 \leq B$ for all $\boldsymbol{\omega} \in \{\boldsymbol{\omega} \in \mathbb{R}^l : \|\boldsymbol{\omega}\|_1 \leq A\}$.

Proof. According to Theorem 1, we get $\|\mathbf{f}_p(\boldsymbol{\omega})\|_1 \leq A$. To prove $\|\nabla_{\boldsymbol{\omega}} \mathbf{f}_p(\boldsymbol{\omega})\|_p \leq B$, it suffices to prove $\forall j, \|\frac{\partial}{\partial \omega_j} \mathbf{f}_p(\boldsymbol{\omega})\|_1 \leq B$. By (19), we obtain

$$\begin{aligned} \left\| \frac{\partial}{\partial \omega_j} \mathbf{f}_p(\boldsymbol{\omega}) \right\|_1 &\leq \left\| [\mathbf{R}_U]^{-1} \underbrace{\left(\sum_{i=1}^N \beta^{N-i} g_e(i) u_j(i) f(i) \mathbf{u}(i) \mathbf{u}(i)^T \right) F(\boldsymbol{\omega})}_{f_{uu}} \right\|_1 \\ &\quad + \left\| [\mathbf{R}_U]^{-1} \underbrace{\left(\sum_{i=1}^N \beta^{N-i} g_e(i) u_j(i) f(i) d(i) \mathbf{u}(i) \right)}_{f_u} \right\|_1 \end{aligned} \quad (29)$$

And we can derive

$$\begin{aligned}
f_{\mathbf{u}\mathbf{u}} &\leq \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| g_e(i) u_j(i) f(i) \mathbf{u}(i) \mathbf{u}(i)^T \right\|_1 \|f_p(\boldsymbol{\omega})\|_1 \\
&\leq A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| g_e(i) u_j(i) f(i) \mathbf{u}(i) \mathbf{u}(i)^T \right\|_1 \\
&= A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| u_j(i) f(i) \operatorname{sgn}(e(i)) [s\tau |e(i)^{s-1}| + (2-s)|e(i)|^{-1}] \mathbf{u}(i) \mathbf{u}(i)^T \right\|_1 \\
&\leq A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| u_j(i) f(i) [s\tau |e(i)^{s-1}| + (2-s)|e(i)|^{-1}] \mathbf{u}(i) \mathbf{u}(i)^T \right\|_1 \\
&= A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| u_j(i) \exp(-\tau |e(i)|^s) |e(i)|^{s-3} [s\tau |e(i)^s| + (2-s)] \mathbf{u}(i) \mathbf{u}(i)^T \right\|_1 \\
&\leq A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \left\| \exp(-\tau |e(i)|^s) |e(i)|^{s-3} [s\tau |e(i)^s| + (2-s)] \|\mathbf{u}(i)\|_1 \|\mathbf{u}(i) \mathbf{u}(i)^T\|_1 \right\|_1 \\
&\leq A \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \underbrace{\frac{s\tau(|d(i)| + A\|\mathbf{u}(i)\|_1)^s + 2-s}{(|d(i)| - A\|\mathbf{u}(i)\|_1)^{3-s}}}_{\mathcal{D}(i)} \|\mathbf{u}(i)\|_1 \|\mathbf{u}(i) \mathbf{u}(i)^T\|_1
\end{aligned} \tag{30}$$

similarly

$$f_{\mathbf{u}} \leq \|[\mathbf{R}_U]^{-1}\|_1 \sum_{i=1}^N \beta^{N-i} \mathcal{D}(i) \|\mathbf{u}(i)\|_1 \|d(i) \mathbf{u}(i)\|_1 \tag{31}$$

Then, combining (21), (22), (29), (30) and (31), we get

$$\left\| \frac{\partial}{\partial \omega_j} f_p(\boldsymbol{\omega}) \right\|_1 \leq \frac{\sqrt{l} \sum_i^N \beta^{N-i} \mathcal{D}(i) \|\mathbf{u}(i)\|_1 [A\|\mathbf{u}(i) \mathbf{u}(i)^T\|_1 + \|d(i) \mathbf{u}(i)\|_1]}{\lambda_{\min}[\sum_i^N \beta^{N-i} \exp(-\tau |\Delta|^s) |\Delta|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T]} = \psi(\tau) \tag{32}$$

Obviously, $\psi(\tau)$ is also a continuous and monotonically increasing function of $\tau \in (0, \infty)$, satisfying $\lim_{\tau \rightarrow \infty} \psi(\tau) = \infty$ and $\lim_{\tau \rightarrow 0^+} \psi(\tau) = 0$. Therefore, given $0 < B < 1$, the equation $\psi(\tau) = B$ gets a unique solution $\tau^\diamond \in (0, \infty)$, and if $\tau < \tau^\diamond$, we obtain $\psi(\tau) < B$, which completes the proof. \square

Remark 1. By Theorem 2 and Banach Fixed-Point Theorem, given an initial weight vector satisfying $\|\boldsymbol{\omega}(0)\|_1 \leq A$, the FP-GMC algorithm (15) will surely converge to a unique fixed-point in the range $\boldsymbol{\omega} \in \{\boldsymbol{\omega} \in \mathbb{R}^l : \|\boldsymbol{\omega}\|_1 \leq A\}$ provided that kernel parameter τ is smaller than a certain value. Moreover, the value of B ($0 < B < 1$) guarantees the convergence rate. It is worth noting that the derived sufficient condition will be a little loose, due to the zooming out in the proof process. In addition, when the value of s is set as 2, the results of Theorem 2 are equivalent to that of Theorem 2 in [44].

3. The Sliding-window and Recursive GMC Algorithms

As one can see that the FP-GMC algorithm (15) is not suitable for online situations. In this part, we apply the sliding-window method and recursive method to the FP-GMC algorithm for practical applications.

3.1. Sliding-window GMC Algorithm

Similar to the RLMP [19], when a new sample is acquired, one can use the sliding-window method and set some stopping criteria to obtain the solution of (13). Hence, we rewrite (12) and (13) as follows

$$\boldsymbol{\omega}_{sw}(n) = [\mathbf{R}_U^{sw}(n)]^{-1} \mathbf{z}_d^{sw}(n) \quad (33)$$

with

$$\begin{cases} \mathbf{R}_U^{sw}(n) = \sum_{i=n-W+1}^n \beta^{n-i} f_e^{sw}(n, i) \mathbf{u}(i) \mathbf{u}(i)^T \\ \mathbf{z}_d^{sw}(n) = \sum_{i=n-W+1}^n \beta^{n-i} f_e^{sw}(n, i) d(i) \mathbf{u}(i) \\ f_e^{sw}(n, i) = \exp(-\tau |e_{sw}(n, i)|^s) |e_{sw}(n, i)|^{s-2} \end{cases} \quad (34)$$

where W is the length of the sliding-window and $e_{sw}(n, i) = d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{sw}(n)$. And, we can use an iterative scheme to solve for $\boldsymbol{\omega}_{sw}(n)$ at each instant. Such a scheme is called *sliding-window generalized maximum correntropy* algorithm (SW-GMC) with filter parameters vector $\mathbf{p}_{sw} = [s, \tau, \beta, W, K, \epsilon_1]$, where K denotes the number of subcycles and ϵ_1 is a filtering accuracy. The SW-GMC is summarized in Algorithm 1. Although, the SW-GMC can be used to obtain the solution of (33), there are two drawbacks of the sliding-algorithm: 1) this algorithm is not a truly online algorithm, because it buffers previous samples (decided by W) within a window; 2) the computational complexity of this algorithm is not known a priori, and it varies at each instance n relying on the stopping criteria (the number of subcycles K and the filtering accuracy ϵ_1).

3.2. Recursive GMC Algorithm

Alternatively, a online algorithm can be derived by some properly operations [7, 29]. Investigating the structure of the \mathbf{R}_U^{rg} and the \mathbf{z}_d^{rg} in (12), we obtain a approximately recursive formula to update them when a new arrives as

Algorithm 1: The Sliding-window GMC Algorithm**Initialization:**

$s > 0, \tau > 0, 0 < \beta \leq 1, 1 \leq W, 1 \leq K, 0 < \epsilon_1 \ll 1$ and $\boldsymbol{\omega}_{sw}(0) = \mathbf{0}$.

Computation:

while $\{\mathbf{u}(n), d(n)\}_{n \geq 1}$ available do

1 : $\boldsymbol{\omega}_{sw}^0(n) = \boldsymbol{\omega}_{sw}(n-1),$

2 : for $m = 1 : K$

3 : $[\mathbf{R}_U^{sw}]^m(n) = \sum_{i=n-W+1}^n \beta^{n-i} f_e^{sw}(n, i) \mathbf{u}(i) \mathbf{u}(i)^T,$

4 : $[\mathbf{z}_d^{sw}]^m(n) = \sum_{i=n-W+1}^n \beta^{n-i} f_e^{sw}(n, i) d(i) \mathbf{u}(i),$

5 : $\boldsymbol{\omega}_{sw}^m(n) = [[\mathbf{R}_U^{sw}]^m(n)]^{-1} [\mathbf{z}_d^{sw}]^m(n),$

6 : if $\|\boldsymbol{\omega}_{sw}^m(n) - \boldsymbol{\omega}_{sw}^{m-1}(n)\| / \|\boldsymbol{\omega}_{sw}^m(n)\| < \epsilon_1$, then end for

7 : else $m = m + 1$, go to step 3,

end while

follows

$$\begin{aligned}
\mathbf{R}_U^{rg}(n) &= \sum_{i=1}^n \beta^{n-i} \exp(-\tau |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)|^s) |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T \\
&= \underbrace{\sum_{i=1}^{n-1} \beta^{n-i} \exp(-\tau |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)|^s) |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T}_{(e)} \\
&\quad + \exp(-\tau |d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rg}(n)|^s) |d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rg}(n)|^{s-2} \mathbf{u}(n) \mathbf{u}(n)^T \\
&\approx \underbrace{\beta \sum_{i=1}^{n-1} \beta^{(n-1)-i} \exp(-\tau |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n-1)|^s) |d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n-1)|^{s-2} \mathbf{u}(i) \mathbf{u}(i)^T}_{(f)} \\
&\quad + \exp(-\tau |d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rg}(n)|^s) |d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rg}(n)|^{s-2} \mathbf{u}(n) \mathbf{u}(n)^T \\
&= \beta \mathbf{R}_U^{rg}(n-1) + f_e^{rg}(n, n) \mathbf{u}(n) \mathbf{u}(n)^T
\end{aligned} \tag{35}$$

where $f_e^{rg}(n, i) = \exp(-\tau |e_{rg}(n, i)|^s) |e_{rg}(n, i)|^{s-2}$ with $e_{rg}(n, i) = d(i) - \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)$. The main difference of (e) and (f) in (35) is that the $\boldsymbol{\omega}_{rg}(n)$ in (e) is replaced by $\boldsymbol{\omega}_{rg}(n-1)$ in (f). In this way, the $\mathbf{z}_d^{rg}(n)$ can also be approximated as

$$\mathbf{z}_d^{rg}(n) \approx \beta \mathbf{z}_d^{rg}(n-1) + f_e^{rg}(n, n) d(n) \mathbf{u}(n) \tag{36}$$

By applying the *matrix inversion lemma* (5.4) [3], the inverse of $\mathbf{R}_U^{rg}(n)$ denoted as $\mathbf{P}_U^{rg}(n)$ can be represented as

$$\begin{aligned}\mathbf{P}_U^{rg}(n) &= [\mathbf{R}_U^{rg}(n)]^{-1} \\ &= [\beta \mathbf{R}_U^{rg}(n-1) + f_e^{rg}(n, n) \mathbf{u}(n) \mathbf{u}(n)^T]^{-1} \\ &= \beta^{-1} \mathbf{P}_U^{rg}(n-1) - \beta^{-1} \mathbf{g}(n) \mathbf{u}(n)^T \mathbf{P}_U^{rg}(n-1)\end{aligned}\quad (37)$$

where the gain vector $\mathbf{g}(n)$ is defined as

$$\mathbf{g}(n) = \frac{f_e^{rg}(n, n) \mathbf{P}_U^{rg}(n-1) \mathbf{u}(n)}{\beta + f_e^{rg}(n, n) \mathbf{u}(n)^T \mathbf{P}_U^{rg}(n-1) \mathbf{u}(n)} \quad (38)$$

Substituting the (36) and (37) into $\boldsymbol{\omega}_{rg}(n) = [\mathbf{R}_U^{rg}(n)]^{-1} \mathbf{z}_d^{rg}(n)$, and after some manipulations, we readily obtain

$$\boldsymbol{\omega}_{rg}(n) = \boldsymbol{\omega}_{rg}(n-1) + \mathbf{g}(n) e_{rg}(n, n-1) \quad (39)$$

where $e_{rg}(n, n-1) = d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rg}(n-1)$ denotes the *a priori error*, and the update equation (39) is called *recursive generalized maximum correntropy* (RGMC) algorithm with filter parameters vector $\mathbf{p}_{rg} = [s, \tau, \beta]$

Remark 2. From (35) and (36), one can observe that $f_e^{rg}(n, i) = \exp(-\tau |e_{rg}(n, i)|^s) |e_{rg}(n, i)|^{s-2}$ is approximated by $\tilde{f}_e^{rg}(n, i) = \exp(-\tau |e_{rg}(n-1, i)|^s) |e_{rg}(n-1, i)|^{s-2}$ for $i \in [1, n-1]$, which means as $n \rightarrow +\infty$, the $\boldsymbol{\omega}_{rg}(n-1) \rightarrow \boldsymbol{\omega}_{rg}(n)$ resulting in $\mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n-1) \rightarrow \mathbf{u}(i)^T \boldsymbol{\omega}_{rg}(n)$ [7]. Unfortunately, the approximations used in (35) and (36) may dramatically damage filter performance when the impulse response of FIR changes abruptly. Because, in the abrupt case, the difference $\tilde{f}_e^{rg}(n, i) = f_e^{rg}(n, i) - \tilde{f}_e^{rg}(n, i)$ can be obvious for many instances, and thus the errors incurred by the approximations of $\mathbf{R}_U^{rg}(n)$ and $\mathbf{z}_d^{rg}(n)$ can be very significant. [7] and [29] did not pay enough attention to this problem.

To explain this influence, we plot the averaging time evolution of the cost function (9) in Fig. 1 (a), when using the sliding-window and recursive solutions for $\mathbf{R}_U(n)$ and $\mathbf{z}_d(n)$, respectively, with an abrupt change in the $\boldsymbol{\omega}^o$ to $-\boldsymbol{\omega}^o$ at $n = 2000$ under the α -SD noise with $\mathbf{p}_\alpha = [1.45, 0, 0.5, 0]$, and the filters with $\mathbf{p}_{sw} = [1.4, 0.001, 0.99, 1000, 50, 0.001]$ and $\mathbf{p}_{rg} = [1.4, 0.001, 0.99]$, respectively. As we can observe that, in comparison with the SW-GMC algorithm, the RGMC algorithm induces a non-negligible error after the change. Furthermore, Fig. 1 (b) shows the evolution of the difference $\tilde{f}_e^{rg}(n, i)$ for sample $i = 1900$, and it also illustrates that $\tilde{f}_e^{rg}(n, i)$ does fluctuate obviously for many instances after the change, which results in the error induced by the approximations in (35) and (36). Actually, the negative effect of this error can be alleviated by the small forgetting factor β , since $f_e^{rg}(n, i)$ is multiplied by β^{n-i} in (35). And, this is the reason why we consider the combination schemes to modify overall filtering performance of the RGMC algorithm.

4. Adaptive Convex Combination of RGMC algorithms

In this section, we propose to convexly combine two RGMC algorithms with different forgetting factors, a large one β_1 and a small one β_2 , i.e., $0 < \beta_2 < \beta_1 \leq 1$. Based on the conventional idea of convex combination [1, 25, 39, 40, 41],

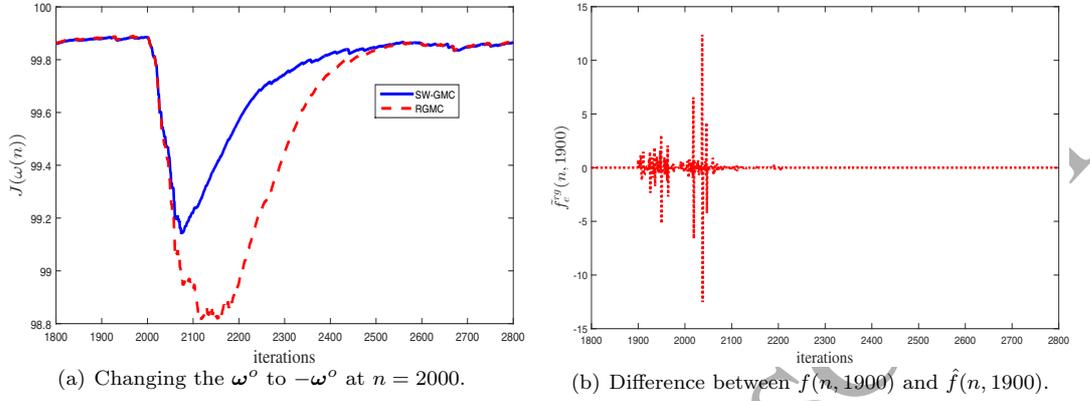


Figure 1: Influence of approximations for $\mathbf{R}_U^{rg}(n)$ and $\mathbf{z}_d^{rg}(n)$. (a) Evolution of $J(\omega(n))$ averaged over 100 Monte Carlo simulations when using the sliding-window and recursive solutions, respectively. (b) Averaging value of the difference $\bar{f}_e^{rg}(n, i) = f_e^{rg}(n, i) - \hat{f}_e^{rg}(n, i)$ for $i = 1900$ under the RGMC algorithm.

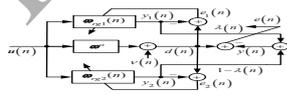


Figure 2: Adaptive convex combination of two RGMC algorithms. Each component is updated using its own adaptation rules, while the mixing parameter $\lambda(n)$ is chosen to maximize the GMC of the overall filter's error.

the two component filters with different forgetting factors are independently updated using their own adaption rules as shown in Fig. 2. To be more specific, the overall output of the proposed algorithm is obtained by combining the output of the component RGMC algorithms as

$$y(n) = \lambda(n)y_1(n) + (1 - \lambda(n))y_2(n) \quad (40)$$

where $\lambda(n)$ is the mixing parameter; $y_1(n) = \mathbf{u}(n)^T \boldsymbol{\omega}_{rg1}(n)$ is the component filter with large β_1 , this filter provides accurate identification in steady-state, but suffers the abrupt change problem regarding the lack of adaption for $f_e^{rg}(n, i)$, thus poor convergence; $y_2(n) = \mathbf{u}(n)^T \boldsymbol{\omega}_{rg2}(n)$ is another component filter with small β_2 , this one provides better behavior in fast varying situations, and also serve as a control mechanism for the first component with β_1 , since the influence of errors affecting $f_e^{rg}(n, i)$ is significantly reduced, and approximations (35) and (36) are more acceptable. Based on (40), the overall filter output error is that

$$\begin{aligned} e(n) &= d(n) - y(n) \\ &= d(n) - (\lambda(n)y_1(n) + (1 - \lambda(n))y_2(n)) \\ &= \lambda(n)(d(n) - y_1(n)) + (1 - \lambda(n))(d(n) - y_2(n)) \\ &= \lambda(n)e_1(n) + (1 - \lambda(n))e_2(n) \end{aligned} \quad (41)$$

where $e_i(n) = d(n) - \mathbf{u}(n)^T \boldsymbol{\omega}_{rgi}(n)$, $i \in \{1, 2\}$, denote the component errors.

Following relevant literature, the mixing parameter $\lambda(n)$ is defined as the output of a sigmoidal function

$$\lambda(n) = \frac{1}{1 + \exp(-\chi(n))} \quad (42)$$

which results in $\lambda(n) \in [0, 1]$. The $\chi(n)$ is an adaption parameter related to the estimation error $e(n)$, and it usually applies a stochastic gradient method to update [1, 25, 39, 40, 41]. On the downside, updating the $\chi(n)$, in conventional combination scheme, by minimizing the square error is not robust to α -SD noises. Hence, we maximize the generalized correntropy with a normalized gradient ascend scheme to modify the adaptive rule [41], i.e.,

$$\begin{aligned} \chi(n+1) &= \chi(n) + \frac{\mu_\chi}{\tau L_p(n)} \frac{\partial \exp(-\tau |e(n)|^s)}{\partial \chi(n)} \\ &= \chi(n) + \frac{\mu_\chi}{L_p(n)} \exp(-\tau |e(n)|^s) |e(n)|^{s-2} e(n) \\ &\quad \times \lambda(n)(1 - \lambda(n))(y_1(n) - y_2(n)) \end{aligned} \quad (43)$$

where $\mu_\chi > 0$ denotes a combination step-size, and $L_p(n)$ is a low-pass filtered estimation of $|y_1(n) - y_2(n)|^s$ [11], namely

$$L_p(n) = \theta L_p(n-1) + (1 - \theta) |y_1(n) - y_2(n)|^s \quad (44)$$

where $\theta \in (0, 1)$ denotes a selection parameter. Furthermore, when the value of $\lambda(n)$ is too close to 1 or 0, the adaptation of $\chi(n)$ will be very slow or stop. Thus, we constrain the range of $\chi(n) \in [-4, +4]$ to avoid this issue.

So far we have derived the adaptive convex combination of two RGMC algorithms, and call it as AC-RGMC algorithm. This new algorithm is able to effectively assemble the smaller steady-state error of the β_1 -RGMC and the faster convergence rate of the β_2 -RGMC. In addition, we can use the weight transfer idea to further improve the convergence rate of the AC-RGMC in some instants [25]. Specifically, the faster component β_2 -RGMC uses its $\mathbf{P}_{U_2}^{rg}(n)$ to reset $\mathbf{P}_{U_1}^{rg}(n)$ of the slower one β_1 -RGMC whenever $\lambda(n) < \epsilon_2$, where ϵ_2 is a relatively small positive constant. It is worth noting that, when the ϵ_2 is very small, the component with β_2 achieves better filtering accuracy than that of the filter with β_1 . This situation usually occurs following abrupt changes in the intrinsic weight vector $\boldsymbol{\omega}^o$, because the large forgetting factor of β_1 -RGMC leads this filter keep using very outdated values of $f_e^{rg}(n, i)$ for a long instances. Resetting $\mathbf{P}_{U_1}^{rg}(n)$ guarantees β_1 -RGMC immediately discarding these outdated terms, thus improving its convergence rate. And, in this work, we call the AC-RGMC with control scheme as AC-RGMC-C. Moreover, we summarize the proposed algorithms in Algorithm 2¹.

5. Simulation Results

In this section, we assess the filtering performances of the proposed AC-RGMC and AC-RGMC-C algorithms in a plant identification. The desired output $d(n)$ is modeled by $d(n) = \mathbf{u}(n)^T \boldsymbol{\omega}^o + \nu(n)$, where $\boldsymbol{\omega}^o$ is randomly generated with length $l = 32$ and is the impulse response of the system to identify, and the $\nu(n)$ is an additive impulsive noise modeled by α -SD. The input signal is obtained by filtering a white, zero-mean Gaussian signal with power as 1 through a second-order system $H(z) = (1 + 0.6z^{-1})/(1 + z^{-1} + 0.21z^{-2})$. We test the performances of the AC-RGMC and AC-RGMC-C algorithms in scenario that changes $\boldsymbol{\omega}^o$ to $-\boldsymbol{\omega}^o$ at $n = 2000$. The convergence performance is measured by the normalized mean-square deviation (NMSD)

$$NMSD = 10 \log_{10} \frac{\|\boldsymbol{\omega}^o - \boldsymbol{\omega}(n)\|_2^2}{\|\boldsymbol{\omega}^o\|_2^2} \quad (45)$$

The NMSD learning curves are obtained in dB scale and averaged over 100 Monte Carlo simulations.

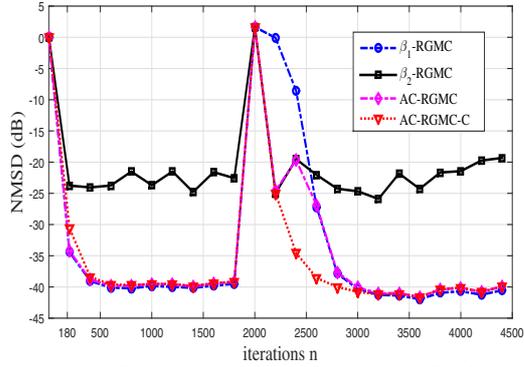
In the first experiment, we demonstrate the efficiency of the AC-RGMC and AC-RGMC-C algorithms in comparison with the component algorithms under the α -SD noise with $\mathbf{p}_\alpha = [1.5, 0, 0.1, 0]$. For β_1 -RGMC, the parameter vector is $\mathbf{p}_{rg1} = [1.4, 0.001, 0.99]$; for β_2 -RGMC, it has the same setting as β_1 -RGMC except $\beta_2 = 0.9$; for AC-RGMC and AC-RGMC-C, the parameters are: $\mu_\chi = 2$, $\theta = 0.9$ and $\epsilon_2 = 0.05$. Fig. 3 (a) plots the NMSD curves for the component algorithms and for the combination-based algorithms, and Fig. 3 (b) shows the evolution of mixing parameters for the combination-based algorithms. From Fig. 3 (a) we can see that: 1) in comparison with other algorithms, the

¹Proposed AC-RGMC algorithm : step 1 to step 14 and 16; Proposed AC-RGMC-C algorithm : step 1 to step 16.

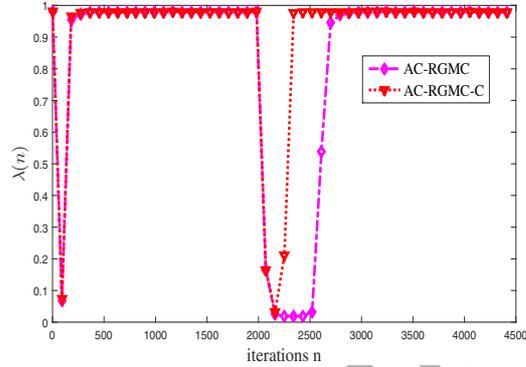
150 filtering accuracy of β_2 -RGMC is the worst and fluctuates obviously; 2) in steady-state, both the AC-RGMC and AC-RGMC-C can achieve almost the same filtering accuracy as that of β_1 -RGMC, since the mixing parameters $\lambda_{acr}(n)$ (for AC-RGMC) and $\lambda_{acrc}(n)$ (for AC-RGMC-C) are very close to 1 as plotted in Fig. 3 (b) resulting in AC-RGMC and AC-RGMC-C ignoring the effect of β_2 -RGMC; 3) all RGMCs almost present the similar initial convergence rate when $n < 180$, however, after $n = 2000$, the β_1 -RGMC slows convergence rate down due to the lack of adaptation of the $f_e^{Tg}(n, i)$, and the AC-RGMC can capture the contribution of faster convergence of β_2 -RGMC realizing faster convergence rate than that of β_1 -RGMC; 4) the application of the simple control mechanism allows AC-RGMC-C to accelerate the convergence rate of AC-RGMC after the abrupt change as shown in Fig. 3 (a), and the convergence improvement can be explained that, when the AC-RGMC-C and AC-RGMC-C both capture the fast initial convergence rate of the β_2 -RGMC, the AC-RGMC-C rapidly switches to the β_1 -RGMC while the AC-RGMC gets bogged down in the β_2 -RGMC more instances as shown in Fig. 3 (b). Furthermore, Fig. 3 (c) and (d) show the influences of different values of the shape parameter $s \in \{1, 1.5, 2, 3\}$ on the proposed algorithms. From Fig. 3 (c), we can find that: 1) the β_1 -RGMC with small values of s (such as $s = 1$ or $s = 1.5$) realize better filtering accuracies, however, when the abrupt change occurs, these small values of s slow down the convergence rate of corresponding β_1 -RGMC algorithms; 2) the β_2 -RGMC with large values of s (such as $s = 1.5, s = 2$ or $s = 3$) yield much fluctuation in terms of filtering accuracies, and the different values of s have less influences on the convergence rate of RGMC with small forgetting factor. In addition, Fig. 3 (d) reveals that: 1) the AC-RGMC and AC-RGMC-C with large values of s (such as $s = 2$ or $s = 3$) result in much fluctuation in terms of filtering accuracies; 2) the smaller s is the better filtering performances of the AC-RGMC and AC-RGMC-C are; 3) combining the filtering results of Fig. 3 (a) and (d), one can see that the AC-RGMC-C can achieve best filtering performances.

170 In the second experiment, we investigate the effects of the parameter θ on the AC-RGMC and AC-RGMC-C. Based on the results of the first experiment, we choose the same parameters for these two algorithms except that $\theta \in \{0.1, 0.25, 0.4, 0.55, 0.7, 0.9, 0.99\}$. As shown in Fig. 4, one can observe that: 1) when the θ is very close to 1 (such as $\theta = 0.99$), for the AC-RGMC and AC-RGMC-C, after the abrupt change, the convergence rates are slower than other combination-based algorithms, especially for AC-RGMC-C with $\theta = 0.99$ the initial convergence rate slows down compared with other AC-RGMC-C algorithms; 2) when the $\theta \leq 0.9$, all combination-based algorithms achieve also the same filtering performances in terms of convergence rate and filtering accuracy, in other words, the smaller θ the less influence on AC-RGMC and AC-RGMC-C. Hence, we can conclude that, when the θ is small, this θ is not critical for the good performance of the proposed algorithms.

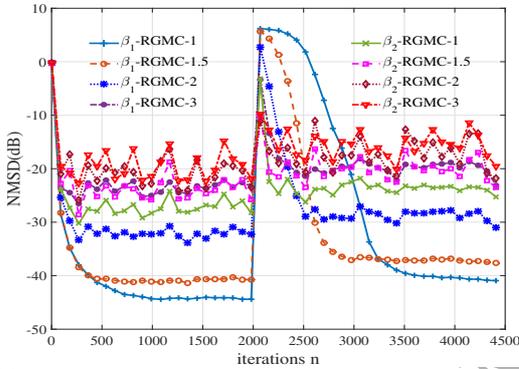
180 In the third experiment, we study the influence of the parameter ϵ_2 on the AC-RGMC-C algorithm. According to the conclusion of the second trial, we fix the $\theta = 0.9$ and consider different values of $\epsilon_2 \in \{0.01, 0.02, 0.03, 0.04, 0.05\}$. From Fig. 5, we can see that: 1) different values of ϵ_2 only affects the convergence rates of the AC-RGMC-C algorithms after the abrupt change; 2) in this experiment, the relatively small ones may slow down the convergence rate, actually,



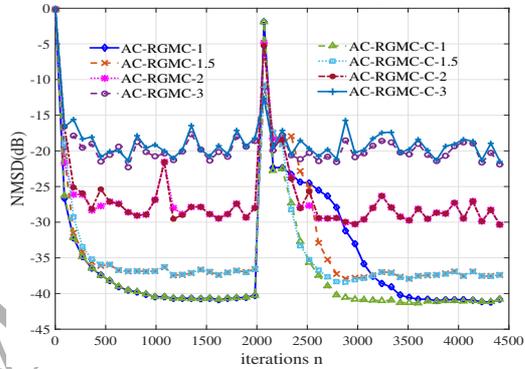
(a) The NMSD learning curves of various RGMC algorithms (where $\beta_1 = 0.99$ and $\beta_2 = 0.9$).



(b) The evolution of mixing parameters $\lambda_{aep}(n)$ (for AC-RGMC) and $\lambda_{aerc}(n)$ (for AC-RGMC-C).

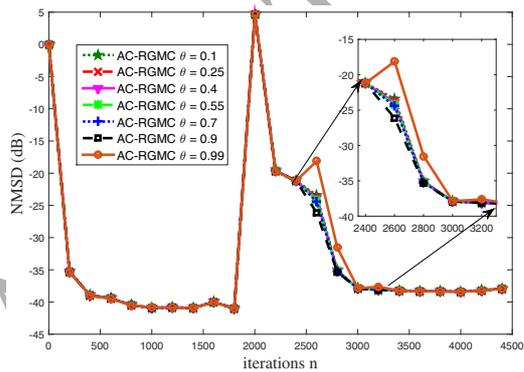


(c) The NMSD learning curves of β_1 -RGMC and β_2 -RGMC algorithms with different values of s .

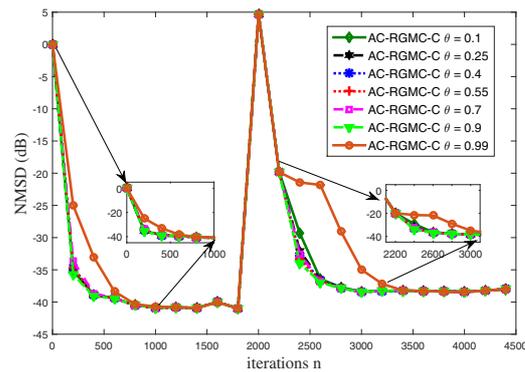


(d) The NMSD learning curves of AC-RGMC and AC-RGMC-C algorithms with different values of s .

Figure 3: The filtering performances comparison between the component algorithms and combination algorithms; The influence of shape parameter s on the proposed algorithms.



(a) The different AC-RGMC algorithms.



(b) The different AC-RGMC-C algorithms.

Figure 4: The influences of the parameter θ on the combination-based algorithms.

when the ϵ_2 is too small (e.g., smaller than 0.01) or relatively large (e.g., larger than 0.04), the filtering performances of the AC-RGMC-C will be invariant.

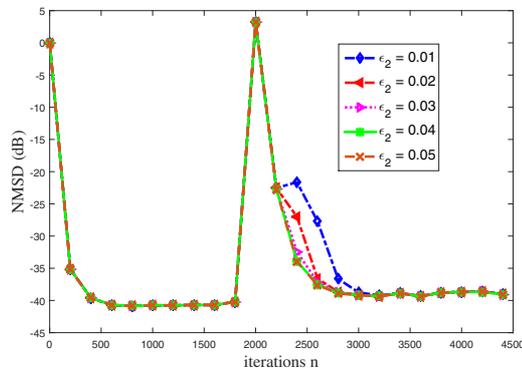


Figure 5: The influence of the parameter ϵ_2 on the AC-RGMC-C algorithm with $\theta = 0.9$.

185 Finally, under different alpha stable noises, we compare the performance of the AC-RGMC-C algorithm (with the same parameters as the first experiment except ϵ_2) against other state-of-the-art methods, e.g., 1) the combination of the affine projection algorithm and the affine projection sign algorithm (C-APA-SA) with parameters vector $\mathbf{p}_{ca} = [\mu_1, \mu_2, P, \alpha_{ca}, \gamma_{ca}, \beta_{ca}]$, where $\mu_i, i \in \{1, 2\}$, is the step-size, P denotes the projection orders, α_{ca} is the degree of weighting parameter, γ_{ca} stands for the ratio threshold, and β_{ca} means the scale factor [1]; 2) the combination of maximum correntropy (CMC) criterion algorithm with parameters vector $\mathbf{p}_{cm} = [\mu_1, \mu_2, P, \sigma, \gamma_{cm}, \beta_{cm}]$, where σ denotes the kernel-size of entropy, γ_{ca} stands for the smoothing factor, and β_{cm} is similar as the β_{ca} [25]; 3) the SW-GMC presented in Algorithm 1. In this experiment, the combination-based algorithms have the same combination step-size as 2 and the initial value as 4, and all parameters vectors are listed in Table 1. Fig. 6 plots the NMSD curves of all adaptive filtering algorithms, from this figure, we can observe that: 1) the correntropy-based algorithms can achieve the better filtering accuracy than that of C-APA-SA; 2) as we expect, the convergence rate of the CMC can be slowed down by the correlated input signal; 3) compared with other algorithms, the proposed algorithm can realize the best filtering performances in terms of convergence rate and filtering accuracy; 4) the NMSD of the C-APA-SA dramatically fluctuates under vary fat-tailed noise (e.g., α -SD with $\alpha = 1.5$) as shown in Fig. 6 (a); 5) the SW-GMC algorithms can achieve fast initial convergence rate, and the SW-GMC with longer window (e.g., $W = 300$) realizes better filtering accuracy than that of SW-GMC with shorter window (e.g., $W = 50$). Moreover, Table 1 also summarizes the steady-state NMSD averaged from the last 100 iterations (where S1 and S2 stand for steady-state results before and after the abrupt change, respectively), and the average cost time for one simulation, where the cost time is measured on a PC configured with a 3.4-GHz with 8-GB of RAM, running Matlab R2015a on Windows7. As one can see that: 1) the gradient based algorithm, CMC, is the fastest approach; 2) the computational requirements for the SW-GMC algorithm depend upon the number of a samples window, and thus the SW-GMC is not practical for

195

200

205

real-time applications; 3) the AC-RGMC-C algorithm, in comparison with other algorithms, can achieve best filtering accuracy with an affordable computation time.

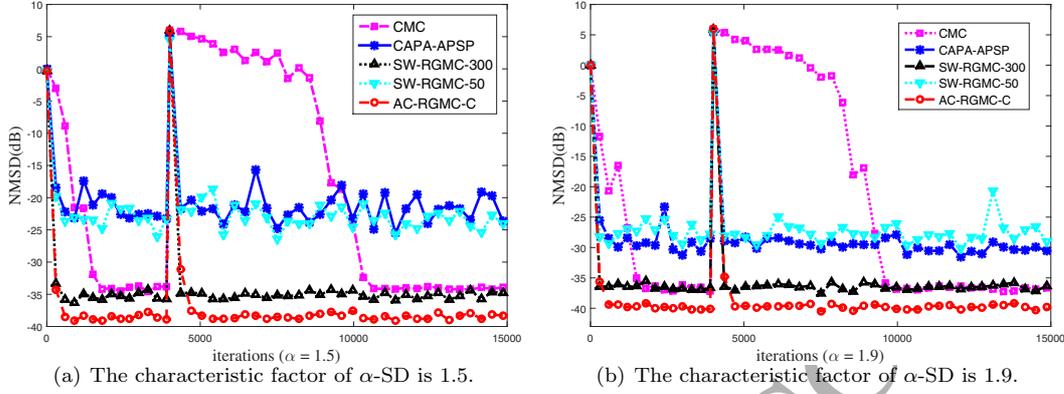


Figure 6: The NMSD curves of different algorithms in system changes at $n = 4000$ with different α -SD noises.

6. Conclusions

The generalized maximum correntropy (GMC) criterion is very useful for robust signal processing. In this paper, we investigated a fixed-point algorithm for GMC estimation, and obtained a sufficient condition to guarantee the convergence of the FP-GMC algorithm. Results of the proposed theorem show that the FP-GMC will surely converge to a unique fixed point if the kernel parameter τ is small than a certain value. For a specific application, we didn't solve the problem of how to set the best kernel parameter, but the theoretic results of this paper may provide a possible range for selecting a kernel parameter for the FP-GMC algorithm.

In addition, based on the proposed FP-GMC algorithm, we applied the sliding-window method and recursive method to the FP-GMC for adaptive filtering. However, since the SW-GMC is not a truly online algorithm and the computational complexity of SW-GMC is uncertain, this sliding algorithm is not proper for practical environments. As we mentioned in **Remark 2**, the approximations in (35) and (36) lead to the RGMC algorithm suffering, when abrupt change occurs, slower convergence rate demonstrated in Fig. 1. This drawback of the RGMC is the reason why we propose the convex combination RGMC (AC-RGMC) algorithm relying on two RGMC algorithms with big and small forgetting factors. Furthermore, the AC-RGMC with control scheme (AC-RGMC-C) is proposed to increase the convergence rate of the AC-RGMC. We have studied the influences of free parameters on AC-RGMC and AC-RGMC-C, and obtained that: 1) when the θ is small, the value of θ is not critical for the AC-RGMC and AC-RGMC-C; 2) when the ϵ_2 gets too small value or relative large value, the AC-RGMC-C can remain its filtering performances. Moreover, the comparison experiment shows that the AC-RGMC-C outperforms the CMC, CAPA-APSP, and SW-GMC algorithms

for the abrupt change problem. Finally, the computational complexity is affordable for the AC-RGMC-C algorithm, since it realizes better filtering accuracy than that of CMC, and thus the proposed algorithm is suitable for real-time applications.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant nos. 61374117, 61004048, 61174137, and 61104064), in part by the National Science Foundation of Jiang Su Province (Grant no. BK2010493), in part by the grant from the Science and Technology Department of Sichuan Province (Grant no. 2014GZ0156), in part by the National Natural Science Foundation of China under Grants 61371182.

References

References

- [1] S. H. Kim, J. J. Jeong, G. Koo, S. W. Kim, Robust convex combination of affine projection-type algorithms using an impulsive noise indicator, *Signal Processing* 129 (2016) 33–37.
- [2] S. Wang, Y. Zheng, S. Duan, L. Wang, C. K. Tse, A class of improved least sum of exponentials algorithms, *Signal Processing* 128 (2016) 340–349.
- [3] A. H. Sayed, *Adaptive filters*, John Wiley & Sons, Inc. Hoboken, New Jersey, 2008.
- [4] P. S. Diniz, *Adaptive filtering: algorithms and practical implementation*, Springer US, 2013.
- [5] C. L. Nikias, M. Shao, *Signal processing with alpha-stable distributions and applications*, Wiley-Interscience, New York, NY, USA, 1995.
- [6] B. Chen, L. Xing, H. Zhao, N. Zheng, J. C. Príncipe, Generalized correntropy for robust adaptive filtering, *IEEE Transactions on Signal Processing* 64 (13) (2016) 3376–3387.
- [7] W. Ma, J. Duan, B. Chen, G. Gui, W. Man, Recursive generalized maximum correntropy criterion algorithm with sparse penalty constraints for system identification, *Asian Journal of Control* 19 (3) (2017) 1164–1172.
- [8] S. Peng, B. Chen, L. Sun, W. Ser, Z. Lin, Constrained maximum correntropy adaptive filtering, *Signal Processing* 140 (2017) 116–126.
- [9] G. Wang, R. Xue, C. Zhou, J. Gong, Complex-Valued adaptive networks based on entropy estimation, *Signal Processing* 149 (2018) 124–134.

- [10] G. Wang, R. Xue, J. Zhao, Switching criterion for sub-and super-Gaussian additive noise in adaptive filtering, *Signal Processing* 150 (2018) 166–170.
- [11] O. Arikan, M. Belge, A. E. Cetin, E. Erzin, Adaptive filtering approaches for non-gaussian stable processes, in: *International Conference on Acoustics, Speech, and Signal Processing*, Vol. 2, 1995, pp. 1400–1403.
- [12] B. Weng, K. E. Barner, Nonlinear System Identification in Impulsive Environments, *IEEE Transactions on Signal Processing* 53 (7) (2005) 2588–2594.
- [13] F. Wen, Diffusion least-mean p-power algorithms for distributed estimation in alpha-stable noise environments, *Electronics Letters* 49 (21) (2013) 1355–1356.
- [14] H. Zhao, Y. Yu, S. Gao, X. Zeng, Z. He, A new normalized LMAT algorithm and its performance analysis, *Signal Processing* 105 (2014) 399–409.
- [15] E. Walach, B. Widrow, The least mean fourth (lmf) adaptive algorithm and its family, *IEEE Transactions on Information Theory* 30 (2) (1984) 275–283.
- [16] S. C. Pei, C. C. Tseng, Least mean p-power error criterion for adaptive FIR filter, *IEEE Journal on selected areas in communications* 12 (9) (1994) 1540–1547.
- [17] T. Shao, Y. R. Zheng, J. Benesty, An affine projection sign algorithm robust against impulsive interferences, *IEEE Signal Processing Letters* 17 (4) (2010) 327–330.
- [18] J. H. Kim, J. H. Chang, S. W. Nam, Affine projection sign algorithm with l1 minimization-based variable step-size, *Signal Processing* 105 (2014) 376–380.
- [19] M. Belge, E. L. Miller, A sliding window RLS-like adaptive algorithm for filtering alpha-stable noise, *IEEE Signal Processing Letters* 7 (4) (2000) 86–89.
- [20] W. Ma, J. Duan, W. Man, H. Zhao, B. Chen, Robust kernel adaptive filters based on mean p-power error for noisy chaotic time series prediction, *Engineering Applications of Artificial Intelligence* 58 (2017) 101–110.
- [21] J. Príncipe, *Information theoretic learning: Renyi’s entropy and kernel perspectives*, Springer New York Dordrecht Heidelberg London, 2010.
- [22] R. He, W. S. Zheng, B. G. Hu, Maximum correntropy criterion for robust face recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (8) (2011) 1561–1576.
- [23] S. Han, S. Rao, D. Erdogmus, K. H. Jeong, J. Príncipe, A minimum-error entropy criterion with self-adjusting step-size (MEE-SAS), *Signal Processing* 87 (11) (2007) 2733–2745.

- [24] W. Liu, P. P. Pokharel, J. C. Príncipe, Correntropy: properties and applications in non-Gaussian signal processing, *IEEE Transactions on Signal Processing* 55 (11) (2007) 5286–5298.
- [25] L. Shi, Y. Lin, Convex combination of adaptive filters under the maximum correntropy criterion in impulsive interference, *IEEE Signal Processing Letters* 21 (11) (2014) 1385–1388.
- [26] S. Zhao, B. Chen, J. C. Príncipe, Kernel adaptive filtering with maximum correntropy criterion, in: *Proceedings of the International Joint Conference on Neural Networks*, 2011, pp. 2012–2017.
- [27] B. Chen, J. C. Príncipe, Maximum correntropy estimation is a smoothed MAP estimation, *IEEE Signal Processing Letters* 19 (8) (2012) 491–494.
- [28] B. Chen, L. Xing, J. Liang, N. Zheng, J. C. Príncipe, Steady-state mean-square error analysis for adaptive filtering under the maximum correntropy criterion, *IEEE Signal Processing Letters* 21 (7) (2014) 880–884.
- [29] A. Singh, J. C. Príncipe, A closed form recursive solution for maximum correntropy training, in: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing – Proceedings*, 2010, pp. 2070–2073.
- [30] Z. Wu, J. Shi, X. Zhang, W. Ma, B. Chen, Kernel recursive maximum correntropy, *Signal Processing* 117 (2015) 11–16.
- [31] X. Zhang, K. Li, Z. Wu, Y. Fu, H. Zhao, B. Chen, Convex regularized recursive maximum correntropy algorithm, *Signal Processing* 129 (2016) 12–16.
- [32] W. Liu, P. P. Pokharel, J. C. Principe, Error entropy, correntropy and m-estimation, in: *Machine Learning for Signal Processing*, 2006. *Proceedings of the 2006 IEEE Signal Processing Society Workshop on*, 2006, pp. 179–184.
- [33] A. R. Heravi, G. A. Hodtani, A new information theoretic relation between minimum error entropy and maximum correntropy, *IEEE Signal Processing Letters* PP (99) (2018) 1–1.
- [34] J. Zhao, H. Zhang, Kernel recursive generalized maximum correntropy, *IEEE Signal Processing Letters* PP (99) (2017) 1–1.
- [35] M. A. Iqbal, S. L. Grant, Novel variable step size nlms algorithms for echo cancellation, in: *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 241–244.
- [36] J. chang Liu, X. Yu, H. ru Li, A nonparametric variable step-size nlms algorithm for transversal filters, *Applied Mathematics and Computation* 217 (17) (2011) 7365 – 7371.
- [37] J. Shin, J. Yoo, P. Park, Variable step-size sign subband adaptive filter, *IEEE Signal Processing Letters* 20 (2) (2013) 173–176.

- [38] S. Zhang, J. Zhang, Modified variable step-size affine projection sign algorithm, *Electronics Letters* 49 (20) (2013) 1264–1265.
- 310 [39] J. Arenas-García, A. R. Figueiras-Vidal, A. H. Sayed, Mean-square performance of a convex combination of two adaptive filters, *IEEE Transactions on Signal Processing* 54 (3) (2006) 1078–1090.
- [40] W. Gao, C. Richard, J. C. M. Bermudez, J. Huang, Convex combinations of kernel adaptive filters (2014) 1–5.
- [41] L. A. Azpicueta-Ruiz, A. R. Figueiras-Vidal, J. Arenas-Garcia, A normalized adaptation scheme for the convex combination of two adaptive filters, in: *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 3301–3304.
- 315 [42] R. P. Agarwal, M. Meehan, D. O'Regan, *Fixed Point Theory and Applications*, Cambridge University Press: Cambridge, UK, 2001.
- [43] A. R. Heravi, G. A. Hodtani, A new robust fixed-point algorithm and its convergence analysis, *Journal of Fixed Point Theory and Applications* 19 (4) (2017) 3191–3215.
- 320 [44] B. Chen, J. Wang, H. Zhao, N. Zheng, J. C. Principe, Convergence of a fixed-point algorithm under maximum correntropy criterion, *IEEE Signal Processing Letters* 22 (10) (2015) 1723–1727.

Algorithm 2: AC-RGMC and AC-RGMC-C algorithms**Initialization:**

For components: $s > 0$, $\tau > 0$, $0 < \beta_2 < \beta_1 \leq 1$, $\boldsymbol{\omega}_{rgi}(0) = \mathbf{0}^T$ and $\mathbf{P}_{U_i}^{rg}(0) = \mathbf{I}$, where $i \in \{1, 2\}$ and \mathbf{I} is the unit matrix with compatible dimension.

For adaptive combination: $\mu_\chi > 0$, $\chi(1) = 4$, $L_p(0) = 0$, $0 < \theta < 1$, $0 < \epsilon_2 \ll 1$.

Computation:

while $\{\mathbf{u}(n), d(n)\}_{n \geq 1}$ available do

- 1 : $\lambda(n) = (1 + \exp(-\chi(n)))^{-1}$,
- 2 : $y_i(n) = \mathbf{u}(n)^T \boldsymbol{\omega}_{rgi}(n-1)$, $i \in \{1, 2\}$,
- 3 : $y(n) = \lambda(n)y_1(n) + (1 - \lambda(n))y_2(n)$,
- 4 : $e(n) = d(n) - y(n)$,
- 5 : $e_i(n) = d(n) - y_i(n)$,
- 6 : $f_{e_i}^{rg}(n, n) = \exp(-\tau|e_i(n)|^s)|e_i(n)|^{s-2}$,
- 7 : $\diamond_i(n) = \beta_i + f_{e_i}^{rg}(n, n)\mathbf{u}(n)^T \mathbf{P}_{U_i}^{rg}(n-1)\mathbf{u}(n)$,
- 8 : $\mathbf{g}_i(n) = \diamond_i(n)^{-1} f_{e_i}^{rg}(n, n) \mathbf{P}_{U_i}^{rg}(n-1)\mathbf{u}(n)$,
- 9 : $\mathbf{P}_{U_i}^{rg}(n) = \beta_i^{-1}(\mathbf{P}_{U_i}^{rg}(n-1) - \mathbf{g}_i(n)\mathbf{u}(n)^T \mathbf{P}_{U_i}^{rg}(n-1))$,
- 10 : $\boldsymbol{\omega}_{rgi}(n) = \boldsymbol{\omega}_{rgi}(n-1) + \mathbf{g}_i(n)e_i(n)$,
- 11 : $L_p(n) = \theta L_p(n-1) + (1 - \theta)|y_1(n) - y_2(n)|^s$,
- 12 : $f_\chi^e(n) = L_p(n)^{-1} \exp(-\tau|e(n)|^s)|e(n)|^{s-2} e(n) \lambda(n) (1 - \lambda(n))$,
- 13 : $\chi(n+1) = \chi(n) + \mu_\chi f_\chi^e(n) (y_1(n) - y_2(n))$,
- 14 : if $|\chi(n+1)| \geq 4$, then $\chi(n+1) = \text{sgn}(\chi(n+1)) \times 4$, end if,
- 15 : if $\lambda(n) < \epsilon_2$, then $\mathbf{P}_{U_1}^{rg}(n) = \mathbf{P}_{U_2}^{rg}(n)$, end if, (control scheme)
- 16 : $\boldsymbol{\omega}(n) = \lambda(n)\boldsymbol{\omega}_1(n) + (1 - \lambda(n))\boldsymbol{\omega}_2(n)$,

end while

Table 1: Parameters and Simulation results with system changes at $n = 4000$ under different α -SD noises.

α -SD noises with $\mathbf{p}_\alpha = [\alpha, 0, 0.1, 0]$		$\alpha = 1.5$			$\alpha = 1.9$		
Algorithms	Parameters	NMSD(dB)		Time(sed)	NMSD(dB)		Time(sed)
		S1	S2		S1	S2	
CMC	$\mathbf{p}_{cm} =$ [0.05, 0.01, 2, 2, 0.8]	-33.90	-33.84	1.47	-37.04	-36.84	1.48
C-APA-SA	$\mathbf{p}_{ca} =$ [0.1, 0.001, 10, 0.9, 2, 5]	-21.61	-19.85	3.25	-29.11	-30.86	3.26
SW-GMC-50	$\mathbf{p}_{sw} =$	-23.58	-21.55	27.15	-40.09	-39.95	23.05
SW-GMC-300	[1.4, 0.001, 0.99, 50 or 300, 10, 0.001]	-35.45	-34.59	42.73	-27.38	-28.05	38.8
AC-RGMC-C	$\epsilon_2 = 0.005$	-38.78	-38.08	3.36	-40.09	-39.95	3.40