Generalized Correntropy Induced Metric based Total Least Squares for Sparse System Identification

Ji Zhao [1*], J. Andrew Zhang [2], Hongbin Zhang [3], Qiang Li [1]

[1] *School of Information Engineering, Southwest University of Science and Technology, Mianyang, Sichuan 621010, PR China*

[2] *Global Big Data Technologies Centre, University of Technology Sydney, Sydney, NSW 2007 Australia*

[3] *School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, PR China*

Abstract

The total least squares (TLS) method has been successfully applied to system identification in the errors-in-variables (EIV) model, which can efficiently describe systems where input-output pairs are contaminated by noise. In this paper, we propose a new gradient-descent TLS filtering algorithm based on the generalized correntropy induced metric (GCIM), called as GCIM-TLS, for sparse system identification. By introducing GCIM as a penalty term to the TLS problem, we can achieve improved accuracy of sparse system identification. We also characterize the convergence behaviour analytically for GCIM-TLS. To reduce computational complexity, we use the first-order Taylor series expansion and further derive a simplified version of GCIM-TLS. Simulation results verify the effectiveness of our proposed algorithms in sparse system identification.

Keywords: Total least squares, Sparse system identification, Generalized correntropy, Adaptive filtering.

1. Introduction

Sparse system identification (SSI) based on adaptive filtering (AF) has been widely studied [1, 2, 3, 4, 5, 6]. A sparse system has long impulse response with many zero or near zero tap coefficients. Examples include wireless systems with sparse multipath fading channels, acoustic channels and television transmission channels [1, 2]. In the last two decades, a number of sparse AF algorithms have been derived for SSI. The priori sparsity knowledge is very important for adaptive SSI based on, e.g., least mean square (LMS) algorithms. And, various effective zero-attracting methods have been developed to retrieve the priori sparsity knowledge implicitly. Well-known examples include the ℓ_0-norm LMS [2], zero-attracting LMS (ZA-LMS) [3], reweighted ZA-LMS (RZA-LMS) [3] and Versoria ZA-LMS [4]. In addition, the proportionate method is another efficient way to train AF algorithms for SSI [5, 6].

However, the above-mentioned sparse LMS algorithms suffer from performance degradation when the input signal is corrupted by noise. In addition, the input signal to the filter is often distorted by sampling, modeling and instrumental errors [7, 8]. To deal with this problem, the errors-in-variables (EIV) model can be applied [9, 10]. As a useful approach based on the EIV model, the total least squares (TLS) can minimize the sum of squared perpendicular distances from each point to the hyperplane defined by $d = \boldsymbol{x}^T \boldsymbol{w}_o$ in a Euclidean vector space, where $\boldsymbol{x}$ and $\boldsymbol{w}_o$ are the input and impulse response, respectively [11]. Generally, a TLS problem can be solved by two categories of methods. The first is the direct method, which is based on the computation of singular-value

*Corresponding author

Email address: zhaoji@swust.edu.cn (Ji Zhao [1])

decomposition (SVD) [12]. However, the high computational complexity of SVD limits the real-time applications of TLS. The other one is an iterative method based on the Rayleigh quotient; it has lower complexity and has been widely studied and applied in real-time signal processing [7, 13, 14].

Recently, in order to identify a sparse system using the EIV model, a reweighted ℓ_1-penalty is added to the cost function for the TLS problem, leading to the rewighted ZA-TLS (RZA-TLS) excitatory and inhibitory learning [15]. However, the reweighted ℓ_1-penalty is suboptimal for solving a sparse problem. It is known that the ℓ_0-norm defines the sparsity of a sparse system, however, it is an NP-hard combinatorial optimization problem. To solve this intractable problem, various approximators for the ℓ_0-norm have been developed, such as the correntropy induced metric (CIM) [16] and generalized CIM (GCIM) [17]. As a nonlinear metric, CIM with a proper value of the kernel width can be a good approximator for ℓ_0-norm. As an extension of CIM, GCIM with proper parameters can provide better measure for different sparsity levels [18].

In this paper, we introduce and use the GCIM as a penalty term in the TLS problem, and propose a new sparse AF algorithm, called as GCIM-TLS. The GCIM-TLS algorithm can perform efficiently in time-varying SSI under the EIV model, because the GCIM penalty term can enable the algorithm to accurately measure the sparse information of the impulse response and the TLS method can suppress the input-output noise. In this work, we further solve the following problems for combining TLS and GCIM: 1) Convergence is an important problem for AF algorithms, and it is not a simple task to analytically prove the convergence of an algorithm. As a major contribution, we provide rigorous convergence analysis for the proposed algorithms; 2) The exponential operations in GCIM is computationally complicated for some real-world applications. Therefore, we propose an exponential-free algorithm, i.e., simplified GCIM-TLS (SGCIM-TLS). Without requiring the exponential operations, SGCIM-TLS is more suitable for practical applications. Simulation results show that our proposed algorithms generally outperform the zero-attracting LMS and TLS methods for sparse identification problems.

This paper is organized as follows. In Section 2, we give a brief introduction to the EIV model and the GCIM concept. In Section 3, we introduce the proposed GCIM-TLS algorithm, conduct the convergence analysis, and present the SGCIM-TLS algorithm. In Section 4, simulation results are presented. Finally, some conclusions are presented in Section 5.

2. Preliminary Knowledge

To make the paper self-contained, we first provide a brief description on the EIV model and TLS solution, and then introduce the concept of GCIM.

2.1. EIV model and TLS solution

Consider a general linear system

$$d(i) = \boldsymbol{w}_o^T \boldsymbol{x}(i), \tag{1}$$

where $\boldsymbol{w}_o \in \mathbb{R}^{L\times 1}$ is the unknown parameter vector to be identified, L is the length of taps, $\boldsymbol{x}(i) = [x(i), x(i-1), \ldots, x(i-L+1)]^T$ is the input data at instance i, and $d(i) \in \mathbb{R}$ denotes the corresponding output. Due to the existence of various errors, we apply an EIV model to rewrite (1) as

$$\begin{cases} \tilde{\boldsymbol{x}}(i) = \boldsymbol{x}(i) + \boldsymbol{v}_{in}(i) \\ \tilde{d}(i) = d(i) + v_{out}(i), \end{cases} \tag{2}$$

where $\boldsymbol{v}_{in}(i) \in \mathbb{R}^{L\times 1}$ denotes the zero-mean input noise with auto-covariance matrix $\sigma_{in}^2\boldsymbol{I}_{L\times L}$, and $v_{out}(i) \in \mathbb{R}$ stands for the zero-mean output noise with variance σ_{out}^2. Let the augmented input vector be $\overline{\boldsymbol{x}}(i) = [\widetilde{\boldsymbol{x}}^T(i), \widetilde{d}(i)]^T \in \mathbb{R}^{(L+1)\times 1}$ and the augmented weight vector be $\overline{\boldsymbol{w}}(i) = [\boldsymbol{w}^T(i), -1]^T$, where $\boldsymbol{w}(i)$ is the estimated filter weight vector for $\boldsymbol{w}_o$. Hence, the output error is $e(i) = \overline{\boldsymbol{x}}(i)^T\overline{\boldsymbol{w}}(i) = \widetilde{\boldsymbol{x}}(i)^T\boldsymbol{w}(i) - \widetilde{d}(i)$.

In real-world signal processing, the following Rayleigh quotient is generally used as a TLS cost function [9]

$$J(i) = \frac{\overline{\boldsymbol{w}}(i)^T\boldsymbol{R}\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^2}, \tag{3}$$

where $\boldsymbol{R} = E\left\{\overline{\boldsymbol{x}}(i)\overline{\boldsymbol{x}}(i)^T\right\}$ denotes the autocorrelation matrix of $\overline{\boldsymbol{x}}(i)$, and $\|\cdot\|_2$ is the ℓ_2-norm of a vector. To solve the TLS problem, adaptive algorithms can be used to extract the eigenvector associated with the smallest eigenvalue of $\boldsymbol{R}$ [15].

2.2. Generalized correntropy induced metric

In signal processing, the generalized correntropy (GC) has been widely used as an optimization criterion, since it is an efficient non-linear and local similarity measure between two random variables [17, 19, 20]. Given two sample vectors, e.g, $\boldsymbol{a} = [a_1, a_2, \ldots, a_N]^T$ and $\boldsymbol{b} = [b_1, b_2, \ldots, b_N]^T$, the GC defines a metric between $\boldsymbol{a}$ and $\boldsymbol{b}$ as follows

$$\mathrm{GCIM}(\boldsymbol{a}, \boldsymbol{b}) = \sqrt[\alpha]{\frac{\gamma}{N}\left(1 - \sum_{k=1}^{N}\exp\left(-\lambda|a_k - b_k|^\alpha\right)\right)}, \tag{4}$$

where $\gamma = \alpha/(2\beta\Gamma(1/\alpha))$ is the normalization constant, $\Gamma(\cdot)$ is the Gamma function, $\alpha > 0$ is the shape parameter, $\beta > 0$ is the scale parameter, and $\lambda = 1/\beta^\alpha$ is the kernel factor. Literature has proven that if $\boldsymbol{b} = 0$, and $|a_k| > \delta > 0, \forall k : a_k \neq 0$ as $\lambda \to \infty$, then $\mathrm{GCIM}^\alpha(\boldsymbol{a}, \boldsymbol{0})$ can be a good approximation for the ℓ_0-norm [17]. Hence, we have

$$\|\boldsymbol{a}\|_0 \sim \mathrm{GCIM}^\alpha(\boldsymbol{a}, \boldsymbol{0}) = \frac{\gamma}{N}\left(1 - \sum_{k=1}^{N}\exp\left(-\lambda|a_k|^\alpha\right)\right). \tag{5}$$

As a good approximation to the ℓ_0-norm, the GCIM characterizes sparsity [18] and can be adopted as a penalty term in a cost function for SSI. In this paper, by combining the TLS problem with GCIM, we aim to derive a sparse AF algorithm. The GCIM with different values of λ and α provides better selectivity in terms of zero-attraction, which in turn leads to performance improvement when the system has high sparseness.

3. The Proposed GCIM-TLS Algorithm

Combining (3) and (5) together, we can obtain the following cost function

$$\bar{J}(i) = \frac{\overline{\boldsymbol{w}}^T(i)\boldsymbol{R}\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^2} + \rho\frac{\gamma}{L+1}\left(1 - \sum_{k=1}^{L+1}\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right)\right). \tag{6}$$

Taking the gradient of (6) with respect to $\overline{\boldsymbol{w}}(i)$ yields

$$\nabla\bar{J}(i) = \frac{\|\overline{\boldsymbol{w}}(i)\|_2^2\boldsymbol{R}\overline{\boldsymbol{w}}(i) - \left(\overline{\boldsymbol{w}}^T(i)\boldsymbol{R}\overline{\boldsymbol{w}}(i)\right)\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^4} + \frac{\rho\gamma\lambda\alpha}{L+1}\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i)), \tag{7}$$

where $\boldsymbol{D}(\overline{\boldsymbol{w}}(i)) = \mathrm{diag}\left[\exp\left(-\lambda|\overline{w}_1(i)|^\alpha\right)|\overline{w}_1(i)|^{\alpha-1}, \ldots, \exp\left(-\lambda|\overline{w}_{L+1}(i)|^\alpha\right)|\overline{w}_{L+1}(i)|^{\alpha-1}\right]$ is a diagonal matrix, and $\mathbf{sign}(\overline{\boldsymbol{w}}(i)) = [\mathrm{sign}(\overline{w}_1(i)), \ldots, \mathrm{sign}(\overline{w}_{L+1}(i))]^T$ denotes a column vector, with $\mathrm{sign}(\cdot)$ being the sign operation. Using the gradient descent to update the weight vector, we obtain

$$\overline{\boldsymbol{w}}(i+1) = \overline{\boldsymbol{w}}(i) - \mu\frac{\|\overline{\boldsymbol{w}}(i)\|_2^2\boldsymbol{R}\overline{\boldsymbol{w}}(i) - \left(\overline{\boldsymbol{w}}^T(i)\boldsymbol{R}\overline{\boldsymbol{w}}(i)\right)\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^4} - \eta\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i)), \tag{8}$$

where $\mu > 0$ is the step-size, and $\eta = \frac{\mu\rho\gamma\lambda\alpha}{L+1}$. Since (8) requires to compute $\boldsymbol{R}$, which, in practice, is not known, this algorithm is referred to as a theoretical GCIM-TLS algorithm. An estimate of $\boldsymbol{R}$ can be generally used instead.

The values of parameters in the algorithm, in particularly λ and α, have an important impact on the performance of the algorithm, as the scale of the GCIM-norm is mainly controlled by these values. From the theoretical point of view, when $\lambda \to 0$, GCIM behaves like various norms (from ℓ_α-norm to ℓ_0-norm) of the data in different regions; when $\lambda \to \infty$, GCIM can approximate the ℓ_0-norm irrespective of the values of α. However, from the experimental point of view, when λ is large enough, a large α value generally can not improve the filtering accuracy of GCIM-based sparse adaptive algorithms. Hence, in practice, a relatively large value, such as 1000, is set for λ, and a moderate value, such as 1.5 or 2, is set for α [18].

3.1. Convergence analysis

The stochastic approximation theory shows that, under some conditions, the asymptotic limit of the above difference equations (8) can be analysed by using the following continuous time differential equations [9, 15]

$$\frac{\mathrm{d}\overline{\boldsymbol{w}}(t)}{\mathrm{d}t} = -\frac{\|\overline{\boldsymbol{w}}(t)\|_2^2 \boldsymbol{R}\overline{\boldsymbol{w}}(t) - \left(\overline{\boldsymbol{w}}^T(t)\boldsymbol{R}\overline{\boldsymbol{w}}(t)\right)\overline{\boldsymbol{w}}(t)}{\|\overline{\boldsymbol{w}}(t)\|_2^4} - \eta \boldsymbol{D}(\overline{\boldsymbol{w}}(t))\mathbf{sign}(\overline{\boldsymbol{w}}(t)). \tag{9}$$

Let the eigenvalue decomposition of $\boldsymbol{R}$ be $\boldsymbol{C}\boldsymbol{\Lambda}\boldsymbol{C}^T$, where $\boldsymbol{C} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{L+1}] \in \mathbb{R}^{(L+1)\times(L+1)}$ denotes a unitary matrix, $\boldsymbol{\Lambda} = \mathrm{diag}\,[\lambda_1, \ldots, \lambda_{L+1}]$ is a diagonal matrix with $L+1$ ordered eigenvalues, and λ_{L+1} is the smallest one. Since $\overline{\boldsymbol{w}}(t)$ can be represented as a linear combination of the elements of $\boldsymbol{C}$ [9, 15], we obtain

$$\overline{\boldsymbol{w}}(t) = \sum_{k=1}^{L+1} f_k(t)\boldsymbol{c}_k, \quad \boldsymbol{R} = \sum_{k=1}^{L+1} \lambda_k \boldsymbol{c}_k \boldsymbol{c}_k^T, \tag{10}$$

where $f_k(t)$ is a scalar function. Therefore, (9) can be rewritten as

$$\frac{\mathrm{d}\overline{\boldsymbol{w}}(t)}{\mathrm{d}t} = \sum_{k=1}^{L+1} \frac{\mathrm{d}f_k(t)}{\mathrm{d}t}\boldsymbol{c}_k = \frac{-A\sum_{k=1}^{L+1}\lambda_k f_k(t)\boldsymbol{c}_k + \left(\overline{\boldsymbol{w}}^T(t)\boldsymbol{R}\overline{\boldsymbol{w}}(t)\right)\sum_{k=1}^{L+1} f_k(t)\boldsymbol{c}_k}{A^2} - \eta \boldsymbol{D}(\overline{\boldsymbol{w}}(t))\mathbf{sign}(\overline{\boldsymbol{w}}(t)), \tag{11}$$

where $A = \|\overline{\boldsymbol{w}}(t)\|_2^2$. Let $\boldsymbol{\delta} = -\eta\boldsymbol{D}(\overline{\boldsymbol{w}}(t))$, some parameters can be set to constrain the bound of $\boldsymbol{\delta}\mathbf{sign}(\overline{\boldsymbol{w}}(t))$ in $[-\delta\boldsymbol{I}, \delta\boldsymbol{I}]$, where $\delta = \max\left\{x = \left|-\eta\exp\left(-\lambda\left|\overline{w}_k(t)\right|^\alpha\right)\left|\,\overline{w}_k(t)\right|^{\alpha-1}\right|, k \in [1, L+1]\right\}$. Therefore, we can assume that $\boldsymbol{\delta}\mathbf{sign}(\overline{\boldsymbol{w}}(t))$ is distributed evenly from $\boldsymbol{c}_1$ to $\boldsymbol{c}_{L+1}$, and also assume that the contribution intensity of $\boldsymbol{c}_k$ belongs to $[-\delta', \delta']$ with $\delta > \delta' > 0$. Based on these assumptions, we have

$$\frac{\mathrm{d}f_k(t)}{\mathrm{d}t} = \frac{\left(\overline{\boldsymbol{w}}^T(t)\boldsymbol{R}\overline{\boldsymbol{w}}(t) - \lambda_k A\right) f_k(t)}{A^2} + \epsilon, \tag{12}$$

where ϵ denotes a random variable uniformly distributed over $[-\delta', \delta']$. Since $\overline{\boldsymbol{w}}(t) = \sum_{k=1}^{L+1} f_k(t)\boldsymbol{c}_k$, we obtain $f_k(t) = \overline{\boldsymbol{w}}^T(t)\boldsymbol{c}_k$. In addition, if $\overline{\boldsymbol{w}}^T(0)\boldsymbol{c}_{L+1} \neq 0$, we have $\forall t : f_{L+1}(t) \neq 0$. Hence, we can denote $h_k(t) = \frac{f_k(t)}{f_{L+1}(t)}$ for $k \in [1, L]$. And based on (12), we can obtain the following differential equation

$$\begin{aligned}
\frac{\mathrm{d}h_k(t)}{\mathrm{d}t} &= \frac{f_{L+1}(t)\frac{\mathrm{d}f_k(t)}{\mathrm{d}t} - f_k(t)\frac{\mathrm{d}f_{L+1}(t)}{\mathrm{d}t}}{f_{L+1}^2(t)} \\
&= \frac{f_{L+1}(t)\left(\frac{B-\lambda_k A}{A^2}f_k(t) + \epsilon\right) - f_k(t)\left(\frac{B-\lambda_{L+1}A}{A^2}f_{L+1}(t) + \epsilon\right)}{f_{L+1}^2(t)} \\
&= \frac{f_{L+1}(t)f_k(t)(\lambda_{L+1} - \lambda_k)A^{-1} + \epsilon(f_{L+1}(t) - f_k(t))}{f_{L+1}^2(t)} \\
&= \left(\frac{\lambda_{L+1} - \lambda_k}{A} - \frac{\epsilon}{f_{L+1}(t)}\right)h_k(t) + \frac{\epsilon}{f_{L+1}(t)},
\end{aligned} \tag{13}$$

where $B = \overline{\boldsymbol{w}}(t)^T \boldsymbol{R}\overline{\boldsymbol{w}}(t)$. The solution to (13) in $[0, \infty]$ is

$$h_k(t) = \frac{\epsilon}{f_{L+1}(t)} \exp\left((\lambda_{L+1} - \lambda_k)\int_0^\infty A^{-1}\mathrm{d}\tau\right) \exp\left(-\frac{\epsilon}{f_{L+1}(t)}t\right), \tag{14}$$

If λ_{L+1} is the smallest and the single one, and $|\frac{\epsilon}{f_{L+1}(t)}| \ll 1$, then $\lim_{t\to\infty} h_k(t) \to 0$ for $k \in [1, L]$. Such results lead to $\lim_{t\to\infty} f_k(t) \to 0$ for $k \in [1, L]$. Therefore, under the condition of enough training, we can conclude that $\overline{\boldsymbol{w}}(t)$ will approach the eigenvector $\boldsymbol{c}_{L+1}$ corresponding to the smallest eigenvalue, namely

$$\lim_{t\to\infty} \overline{\boldsymbol{w}}(t) = \lim_{t\to\infty} \sum_{k=1}^{L+1} f_k(t)\boldsymbol{c}_k = \lim_{t\to\infty} f_{L+1}(t)\boldsymbol{c}_{L+1}. \tag{15}$$

3.2. GCIM-TLS and its simplified version

In practice, an instantaneous approximation method can be used to replace (8), i.e., replacing $\boldsymbol{R}$ by $\overline{\boldsymbol{x}}(i)\overline{\boldsymbol{x}}^T(i)$ and yielding

$$\begin{aligned}\overline{\boldsymbol{w}}(i+1) &= \overline{\boldsymbol{w}}(i) - \mu\frac{\|\overline{\boldsymbol{w}}(i)\|_2^2\overline{\boldsymbol{x}}(i)\overline{\boldsymbol{x}}^T(i)\overline{\boldsymbol{w}}(i) - \left(\overline{\boldsymbol{w}}^T(i)\overline{\boldsymbol{x}}(i)\overline{\boldsymbol{x}}^T(i)\overline{\boldsymbol{w}}(i)\right)\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^4} - \eta\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i)) \\ &= \overline{\boldsymbol{w}}(i) - \mu\frac{e(i)\|\overline{\boldsymbol{w}}(i)\|_2^2\overline{\boldsymbol{x}}(i) - e^2(i)\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^4} - \eta\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i)),\end{aligned} \tag{16}$$

where $e(i) = \overline{\boldsymbol{x}}(i)^T\overline{\boldsymbol{w}}(i)$ is the prediction error. And, (16) is the update equation for the GCIM-TLS algorithm. The computational complexity of GCIM-TLS is slightly larger than that of RZA-TLS, because the term $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))$ additionally requires to calculate $\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right)|\overline{w}_k(i)|^{\alpha-1}$ for $k \in [1, L+1]$. To reduce the complexity, we can apply the first-order Taylor series expansion to the exponential operations as follows

$$\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right) \approx \begin{cases} \underbrace{1 - \lambda|\overline{w}_k(i)|^\alpha}_{\theta_k(i)}, & \underbrace{|\overline{w}_k(i)| < \lambda^{-\frac{1}{\alpha}}}_{\phi_k(i)} \\ 0, & \text{otherwise.}\end{cases} \tag{17}$$

Therefore, we call GCIM-TLS with the following update equation as simplified GCIM-TLS, named as SGCIM-TLS

$$\overline{\boldsymbol{w}}(i+1) = \overline{\boldsymbol{w}}(i) - \mu\frac{e(i)\|\overline{\boldsymbol{w}}(i)\|_2^2\bar{\boldsymbol{x}}(i) - e^2(i)\overline{\boldsymbol{w}}(i)}{\|\overline{\boldsymbol{w}}(i)\|_2^4} - \eta\boldsymbol{D}_s(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i)), \tag{18}$$

where $\boldsymbol{D}_s(\overline{\boldsymbol{w}}(i)) = \mathrm{diag}\left[\mathbb{I}(\phi_1(i))\theta_1|\overline{w}_1(i)|^{\alpha-1}, \ldots, \mathbb{I}(\phi_{L+1}(i))\theta_{L+1}(i)|\overline{w}_{L+1}(i)|^{\alpha-1}\right]$ with $\mathbb{I}(\cdot)$ standing for an indicator function. In addition, observing the third term $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i))$ in (16) or $\boldsymbol{D}_s(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i))$ in (18), we can find that such terms are similar to the derivative of (3) regularized with ℓ_α-norm of $\boldsymbol{w}(i)$, i.e., $\|\boldsymbol{w}(i)\|^\alpha$. Thus the third term in (18) becomes $\boldsymbol{D}_\alpha(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i))$ with $\boldsymbol{D}_\alpha(\overline{\boldsymbol{w}}(i)) = \mathrm{diag}\left[\alpha|\overline{w}_1(i)|^{\alpha-1}, \ldots, \alpha|\overline{w}_{L+1}(i)|^{\alpha-1}\right]$. Generally, the α of ℓ_α-norm is not lower than 1, however, neither $\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right)$ in $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))$ nor $\mathbb{I}(\phi_k(i))\theta_k$ in $\boldsymbol{D}_s(\overline{\boldsymbol{w}}(i))$ is greater than 1. Such a difference results in small weight adjustment in our proposed GCIM-TLS algorithms when the weight is close to the optimum.

Remark: In practice, the step-size μ in both GCIM-TLS algorithms should be small enough to avoid instability and divergence. In addition, because the prediction error is $e(i) = \overline{\boldsymbol{x}}(i)^T\overline{\boldsymbol{w}}(i) = \widetilde{\boldsymbol{x}}(i)^T\boldsymbol{w}(i) - \widetilde{d}(i)$, after updating (16) and (18), we need to set $\overline{\boldsymbol{w}}_{L+1}(i) = -1$. Furthermore, we can set different values for α in GCIM, which can lead to various ℓ_0-norm approximations. For instances, if $\alpha = 1$, GCIM becomes the ℓ_0-norm used in [2]; and if $\alpha = 2$, GCIM becomes the CIM used in [6, 16]. Hence, we expect that GCIM with different α values can result in different performance improvement for SSI. As will be shown in the simulation results, in comparison with GCIM-TLS, our proposed SGCIM-TLS algorithm can achieve very similar filtering performance in terms of convergence rate and filtering accuracy, while it does not require the exponential operations. Hence, SGCIM-TLS is more suitable for hardware equipments, such as digital signal processor and field-programmable gate array.

4. Simulation Results

In this section, some simulation results about synthetic models and real impulse responses for SSI are presented to verify the efficiency of the filtering accuracy and convergence rate of proposed GCIM-TLS algorithms. Comparisons are made with several related algorithms, i.e., RZA-LMS, ℓ_0-LMS, RZA-TLS and GD-TLS [7]. The filtering performance of all the algorithms is measured by the error of the weight vector (EWV) defined as

$$\text{EWV} = 20\log_{10}\left(\|\boldsymbol{w}(i) - \boldsymbol{w}_o\|_2\right). \tag{19}$$

In addition, the following definition for the sparsity level of system is adopted [21]

$$\zeta = \frac{L}{L-\sqrt{L}}\left(1 - \frac{\|\boldsymbol{w}_o\|_1}{\sqrt{L}\|\boldsymbol{w}_o\|_2}\right) \in [0,1]. \tag{20}$$

The definition shows that a sparser weight vector leads to a larger $\zeta \to 1$, while a more dispersive weight vector results in a smaller $\zeta \to 0$. In the following experiments, unless stated otherwise, the length of $\boldsymbol{w}_o$ is $L = 64$, $\boldsymbol{w}_o$ is randomly generated with $\zeta = 0.95$, and $\|\boldsymbol{w}_o\|_2 = 1$. The input signal $x(i)$ follows Gaussian distribution with zero mean and unit-variance, and the noise variances are $\sigma_{in}^2 = \sigma_{out}^2 = 0.1$. Thus the signal noise ratio is 10 dB. All simulation results are averaged over 100 independent runs.

4.1. The influence of λ and α on our proposed algorithms

First, we study the influence of λ and α on the performance of two GCIM-TLS algorithms. We set the values of λ and α over $[50, 1000]$ and $[1, 8]$, respectively. Other parameters are set as $\mu = 0.01$ and $\rho = 0.0001$. In this part, the length of each trial is 3000, and we use the steady-state EWV (SS-EWV) to compare filtering accuracy of GCIM-TLS and SGCIM-TLS. Here, SS-EWV is obtained by averaging over the last 100 EWV results of each trial. Figure 1 plots the corresponding SS-EWV results. From this figure, we observe that: 1) Irrespective of the λ values, when $\alpha \geq 2.5$, increasing α value does not have a notable effect on improving the filtering accuracy of the two GCIM-TLS algorithms. This is because, when $|\overline{w}_k(i)| < 1$ and $\alpha - 1$ is relatively large, the elements, i.e., $\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right)|\overline{w}_k(i)|^{\alpha-1}$ in $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))$, are very small, and then the gain vector $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i))$ has very small entries. Although we can slightly adjust the weight vector, this will not notably improve the filtering accuracy of our proposed algorithms. However, in this situation, SGCIM-TLS realizes almost the same misalignment as GCIM-TLS; 2) When α is close to 1, a larger λ leads to worse filtering accuracy for these two algorithms. This is because, when $\alpha \to 1^+$, $|\overline{w}_k(i)| < 1$ and λ becomes large, the entry $\exp\left(-\lambda|\overline{w}_k(i)|^\alpha\right)|\overline{w}_k(i)|^{\alpha-1}$ gets close to 1, and then $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))$ approaches to an identity matrix. In this case, $\boldsymbol{D}(\overline{\boldsymbol{w}}(i))\mathbf{sign}(\overline{\boldsymbol{w}}(i))$ will make too much adjustment of the weight vector, and therefore worsen the filtering accuracy; 3) When $\alpha \in (1, 2.5)$, there exits a valley in the SS-EWV surface, which means that moderate α values lead to a better filtering accuracy of the proposed algorithms; 4) Although the difference of SS-EWV between GCIM-TLS and SGCIM-TLS is in the range of $(-10, 2)$ dB, Figure 1 (b) shows that, in comparison with GCIM-TLS, SGCIM-TLS has a lower computational complexity but can achieve comparable steady-state misalignments in most cases except for a neighbourhood of $\alpha = 1.5$.

Second, to better compare the filtering behaviours of both GCIM-TLS methods, we set α from $\{1.1, 1.5, 2, 3, 4\}$, λ from $\{100, 1000\}$, and other parameters are kept the same as before. Figure 2 plots the corresponding EWV curves. This figure shows that: 1) In comparison with SGCIM-TLS, GCIM-TLS with some α values can achieve better filtering accuracy; 2) When $\alpha = 1.1$, the filtering performance of these two algorithms decreases with the

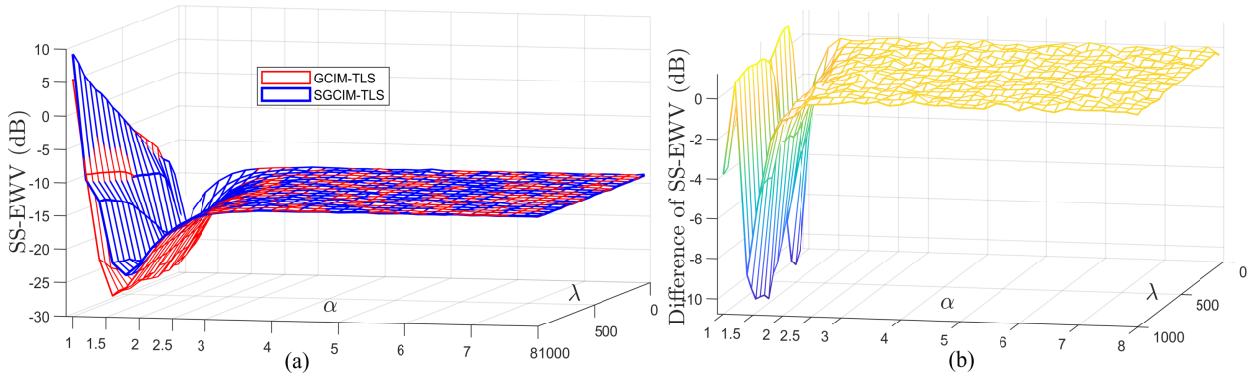

Figure 1: The SS-EWV results for both GCIM-TLS algorithms with $L = 64$ and $\zeta = 0.95$. (a) Respective SS-EWV results; (b) Difference of SS-EWV: GCIM-TLS - SGCIM-TLS.

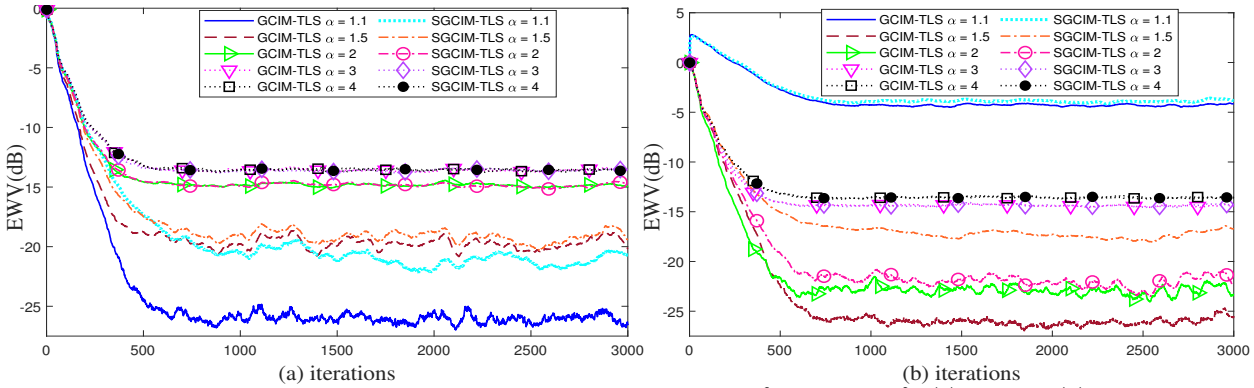

Figure 2: The EWV curves of GCIM-TLS and SGCIM-TLS with $\alpha \in \{1.1, 1.5, 2, 3, 4\}$. (a) $\lambda = 100$; (b) $\lambda = 1000$.

increase of λ values, which is in accordance with the observation from Figure 1; 3) As revealed in Figure 1, when $\alpha \in \{3, 4\}$, GCIM-TLS and SGCIM-TLS achieve almost the same filtering results, but it is hard for them to make further performance improvement; 4) When $(\alpha, \lambda) = (1.5, 100)$ and $(\alpha, \lambda) = (2, 1000)$, SGCIM-TLS demonstrates similar filtering behaviours with GCIM-TLS, while the former does not require the exponential operations.

4.2. Performance comparisons under time-varying system

Third, we test the filtering performance in terms of accuracy and convergence rate, by comparing our proposed algorithms with RZA-LMS, ℓ_0-LMS, RZA-TLS and GD-TLS. The time-varying system (with varying weights) is modelled by changing $\boldsymbol{\omega}_o$ to $-\boldsymbol{\omega}_o$ at a middle time. For all algorithms the step-size $\mu = 0.01$, and other parameters are set, shown in Figures 3 and 4, to realize similar initial convergence behaviours. Figure 3 plots the EWV curves of these algorithms for different sparsity levels, i.e., $\zeta = 0.95$ and $\zeta = 0.65$. From this figure we have the following observations: 1) As expected, when the sparsity level is very high, namely, $\zeta = 0.95$, the filtering accuracy of GD-TLS is the worst among the six algorithms. Since GD-TLS does not use the sparsity knowledge of the sparse system, it is not suitable for the SSI; 2) When $\zeta = 0.95$, the TLS based sparse methods, including RZA-TLS and our proposed algorithms, can realize better filtering accuracy than ℓ_0-LMS and RZA-LMS under EIV systems. Although GCIM-TLS has a lower convergence rate after system change, it is superior to others in terms of steady-state misalignment; 3) When $\zeta = 0.65$, GD-TLS achieves better filtering accuracy than that of RZA-LMS, ℓ_0-LMS and RZA-TLS. Such results reveal that RZA-TLS is not robust to the variation of sparseness of a system. However, in this situation, at the cost of reduced convergence rate, SGCIM-TLS achieves the best filtering accuracy among the other methods.

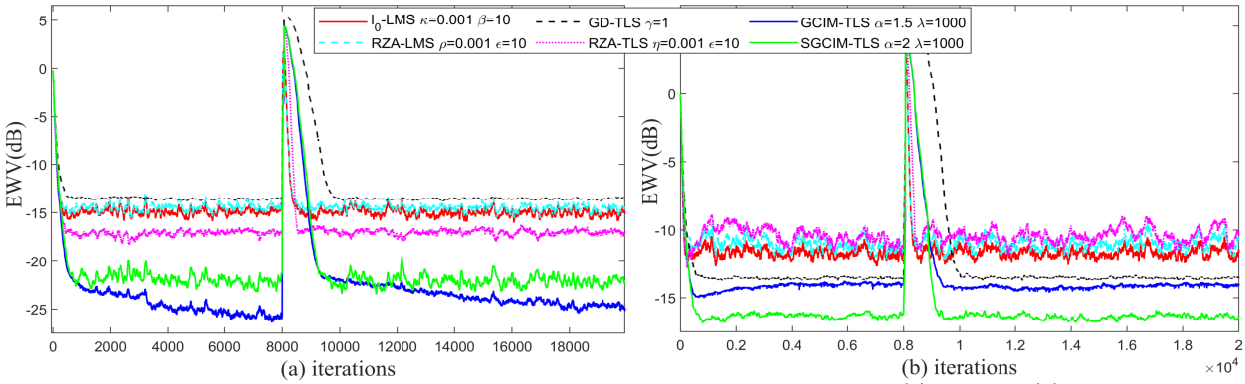

Figure 3: The EWV curves of these algorithms for weight time-varying systems. (a) $\xi = 0.95$; (b) $\xi = 0.65$.

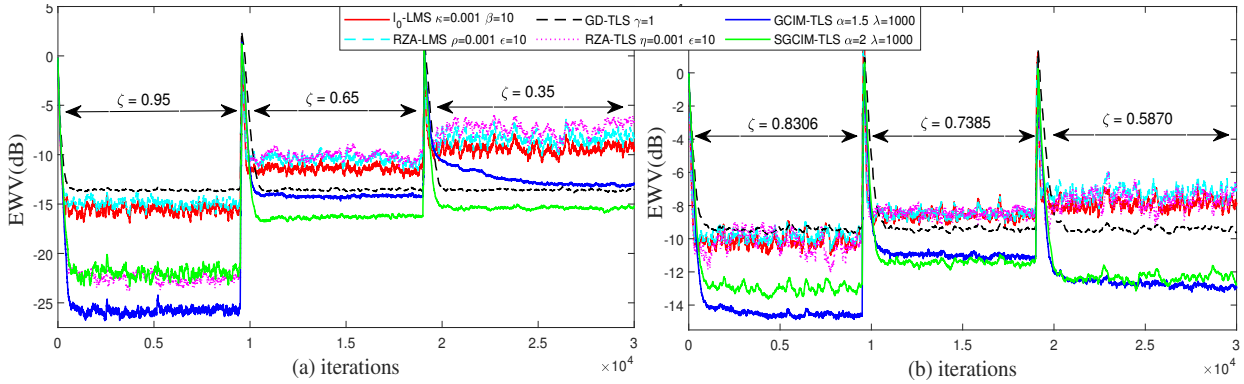

Figure 4: The EWV curves of these algorithms for sparsity time-varying system. (a) Synthetic model; (b) Real impulse responses.

Finally, we consider another time-vary system (with varying sparsity), which is modelled by changing ζ from 0.95 to 0.65 and then to 0.35. Figure 4 (a) plots the corresponding EWV curves, and reveals that: 1) In comparison with other algorithms, the filtering performance of GD-TLS is almost consistent in all cases; 2) Except for GD-TLS, for the other algorithms, the filtering accuracy degrades when ζ decreases. Nevertheless, with $\zeta = 0.65$ and $\zeta = 0.35$, SGCIM-TLS can achieve the smallest misalignment results[1]. Moreover, we also consider a practical problem of identifying real impulse responses, i.e., the impulse response of level measurement device (LMD), the impulse response of echo path model 4 (M4) and the impulse response of echo path model 2 (M2), with $\zeta = 0.8306$, $\zeta = 0.7385$ and $\zeta = 0.5870$, respectively [22]. Here, we unify the length of these three impulse responses to be $L = 128$, and set their ℓ_2-norms to be 1. All algorithms have the same parameter setting as those in Figure 4 (a). The EWV curves are plotted in Figure 4 (b). The figure shows that: 1) Our proposed algorithms achieve the best filtering accuracy in all cases; and 2) With the decrease of ζ, SGCIM-TLS can realize comparable filtering performance as GCIM-TLS. Table 1 summarizes the corresponding SS-EWV results for performance comparisons. In the table, the smallest SS-EWV values are highlighted in bold for different situations. Based on the above observations and results, we can conclude that our proposed algorithms with different α and λ values can achieve various and better filtering performance, compared to existing algorithms. However, how to set an optimal (α, λ) pair remains as an open research problem.

[1]Actually, as shown in Figure 2, compared with SGCIM-TLS, GCIM-TLS with the same parameter setting can achieve similar or even better filtering performance.

Table 1: The SS-EWV results of theses algorithms for time-varying systems.

Algorithms	Steady-State Error of Weight Vector (dB)									
	Synthetic Model							Real Impulse Responses		
	Varying Weight				Varying Sparsity (Fig.4 (a))			Varying Sparsity (Fig.4 (b))		
	$\zeta = 0.95$ (Fig.3 (a))		$\zeta = 0.65$ (Fig.3 (b))		$\zeta = 0.95$	$\zeta = 0.65$	$\zeta = 0.35$	LDM	M4	M2
	$\boldsymbol{\omega_o}$	$-\boldsymbol{\omega_o}$	$\boldsymbol{\omega_o}$	$-\boldsymbol{\omega_o}$				$\zeta = 0.8306$	$\zeta = 0.7385$	$\zeta = 0.5870$
ℓ_0-LMS	-14.73	-15.05	-11.85	-11.77	-15.55	-11.59	-9.11	-10.01	-8.54	-7.93
RZA-LMS	-14.27	-14.58	-11.28	-11.06	-14.97	-10.74	-8.02	-9.85	-8.42	-7.23
GD-TLS	-13.55	-13.49	-13.55	-13.50	-13.52	-13.54	-13.55	-9.46	-9.38	-9.44
RZA-TLS	-17.00	-17.00	-11.16	-10.58	-22.41	-10.18	-6.54	-10.18	-8.51	-7.38
GCIM-TLS	**-25.91**	**-24.66**	-13.97	-14.06	**-25.72**	-14.19	-13.06	**-14.63**	-11.13	**-12.90**
SGCIM-TLS	-22.17	-21.86	**-16.29**	**-16.41**	-21.72	**-16.25**	**-15.32**	-13.01	**-11.43**	-12.35

5. Conclusions

For sparse system identification in the EIV situation, a sparse adaptive filtering algorithm is proposed by injecting the GCIM sparsity penalty term into the TLS cost function. The GCIM is a good approximation to the ℓ_0-norm and can efficiently extract the sparsity information of a system. Exploiting this property, the proposed GCIM-TLS outperforms many existing algorithms in terms of filtering accuracy. In addition, using the first-order Taylor series expansion, we obtain hardware-implementation-friendly SGCIM-TLS that avoids the exponential operation, while achieving comparable filtering performance with GCIM-TLS. The convergence property is also characterized analytically and verified by simulation results.

6. Acknowledgment

This work is supported in part by the Southwest University of Science and Technology Doctor Fund (Grant no. 20zx7119), in part by the National Natural Science Foundation of China (Grant No. 61971100), and in part by Sichuan Science and Technology Program (No. 2019JDTD0019)

[1] W. Ma, B. Chen, H. Qu, J. Zhao, Sparse least mean p-power algorithms for channel estimation in the presence of impulsive noise, Signal, Image and Video Processing 10 (3) (2016) 503–510.

[2] Y. Gu, J. Jin, S. Mei, l_0 norm constraint LMS algorithm for sparse system identification, IEEE Signal Processing Letters 16 (9) (2009) 774–777.

[3] Y. Chen, Y. Gu, A. O. Hero, Sparse LMS for system identification, in: 2009 International Conference on Acoustics, Speech, and Signal processing, 2009, pp. 3125–3128.

[4] S. S. Bhattacharjee, D. Ray, N. V. George, Adaptive modified versoria zero attraction least mean square algorithms, IEEE Transactions on Circuits and Systems II: Express Briefs 67 (12) (2020) 3602–3606.

[5] D. L. Duttweiler, Proportionate normalized least-mean-squares adaptation in echo cancelers, IEEE Transactions on Speech and Audio Processing 8 (5) (2000) 508–518.

[6] J. Zhao, H. Zhang, G. Wang, X. Liao, Modified memory-improved proportionate affine projection sign algorithm based on correntropy induced metric for sparse system identification, Electronics Letters 54 (10) (2018) 630–632.

[7] R. Arablouei, S. Werner, K. Dogancay, Analysis of the gradient-descent total least-squares adaptive filtering algorithm, IEEE Transactions on Signal Processing 62 (5) (2014) 1256–1264.

[8] P. Shen, C. Li, Minimum total error entropy method for parameter estimation, IEEE Transactions on Signal Processing 63 (15) (2015) 4079–4090.

[9] X. Kong, C. Han, R. Wei, Modified gradient algorithm for total least square filtering, Neurocomputing 70 (1) (2006) 568–576.

[10] I. Markovsky, S. Van Huffel, Overview of total least-squares methods, Signal Processing 87 (10) (2007) 2283–2302.

[11] C. E. Davila, An efficient recursive total least squares algorithm for fir adaptive filtering, IEEE Transactions on Signal Processing 42 (2) (1994) 268–280.

[12] D. Feng, X. Zhang, D. Chang, W. X. Zheng, A fast recursive total least squares algorithm for adaptive fir filtering, IEEE Transactions on Signal Processing 53 (3) (2004) 957–965.

[13] B. E. Dunne, G. A. Williamson, Analysis of gradient algorithms for tls-based adaptive iir filters, IEEE Transactions on Signal Processing 52 (12) (2004) 3345–3356.

[14] F. Wang, Y. He, S. Wang, B. Chen, Maximum total correntropy adaptive filtering against heavy-tailed noises, Signal Processing 141 (2017) 84–95.

[15] J. S. Lim, H. Pang, Reweighted l1 regularized tls linear neuron for the sparse system identification, Neurocomputing 173 (2016) 1972–1975.

[16] S. Seth, J. C. Principe, Compressed signal reconstruction using the correntropy induced metric, in: Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on, 2008.

[17] B. Chen, L. Xing, H. Zhao, N. Zheng, J. C. Principe, Generalized correntropy for robust adaptive filtering, IEEE Transactions on Signal Processing 64 (13) (2016) 3376–3387.

[18] G. Li, H. Zhang, J. Zhao, Generalized correntropy induced metric memory-improved proportionate affine projection sign algorithm and its combination, IEEE Transactions on Circuits and Systems II-express Briefs 67 (10) (2019) 2239–2243.

[19] J. Zhao, H. Zhang, G. Wang, Fixed-point generalized maximum correntropy: Convergence analysis and convex combination algorithms, Signal Processing 154 (2019) 64–73.

[20] K. Kumar, N. V. George, A generalized maximum correntropy criterion based robust sparse adaptive room equalization, Applied Acoustics 158 (2020) 107036.

[21] Z. Yang, Y. R. Zheng, S. L. Grant, Proportionate affine projection sign algorithms for network echo cancellation, IEEE Transactions on Audio Speech and Language Processing 19 (8) (2011) 2273–2284.

[22] T. S. Sector, Digital network echo cancellers, Series G: Transmission Systems and Meaid, Digital Systems and Netwroks, Recommendation G.168, International Telecommunication Union (ITU-T) (2015).