L_0-Norm Adaptive Volterra Filters

Hamed Yazdanpanah
Department of Computer Science
Institute of Mathematics and Statistics
University of São Paulo (USP)
São Paulo, Brazil
hamed.yazdanpanah@smt.ufrj.br

Alberto Carini
Department of Engineering and Architecture
University of Trieste
Trieste, Italy
acarini@units.it

Markus V. S. Lima
Department of Electrical Engineering
PEE/COPPE
Federal University of Rio de Janeiro (UFRJ)
Rio de Janeiro, Brazil
markus.lima@smt.ufrj.br

***Abstract*—The paper addresses adaptive algorithms for Volterra filter identification capable of exploiting the sparsity of nonlinear systems. While the l_1-norm of the coefficient vector is often employed to promote sparsity, it has been shown in the literature that superior results can be achieved using an approximation of the l_0-norm. Thus, in this paper, the Geman-McClure function is adopted to approximate the l_0-norm and to derive l_0-norm adaptive Volterra filters. It is shown through experimental results, also involving a real-world system, that the proposed adaptive filters can obtain improved performance in comparison with classical approaches and l_1-norm solutions.**

***Index Terms*—Nonlinear adaptive filter, Volterra series, sparsity, l_0-norm, Geman-McClure function**

I. Introduction

Nonlinear systems are frequently found in many different areas, such as in image and speech/audio processing [1], [2], in devices using amplifiers like loudspeakers [3], [4], in the study of biological systems [5], in wireless sensor networks [6], to name just a few.

The adaptive Volterra filter (AVF) is widely used in those areas to identify the involved nonlinear system [7]. AVFs rely on the Volterra series expansion, which can be regarded as a Taylor series expansion with memory [1]. Although AVFs truncate the series expansion at a given order P and memory length N, the number of parameters to be estimated is usually very high since it increases exponentially with the order P and geometrically with the memory N. This usually constraints the application of AVFs to small values of P [7]. In most cases, P is set to 2 or 3, which means that the AVF is capable of modeling quadratic or cubic input-output relations.

As we increase P or N, the number of parameters to be estimated increases rapidly, and the AVF convergence slows down. However, in such high-dimensional space it is very common to observe sparsity in the parameters [8]–[10]. This sparsity can be exploited in order to accelerate convergence and/or reduce steady-state mean squared error (MSE). There are several ways of taking sparsity into account in the adaptive filter. *A priori* knowledge on the nonlinear system can be exploited to limit the number of parameters to be adapted, e.g., using simplified filters [10], [11] or imposing specific filter structures [8], [9]. On the other hand, effective sparsity promoting adaptive algorithms have been developed [12]–[26]. Very often the solutions proposed in the literature are based on the l_1-norm of the coefficient vector [12]–[17]. However, it has been recently shown [18]–[26] that superior results can be achieved by adopting for the same purpose an approximation of the l_0-norm. Thus, in this paper a regularization based on the l_0-norm approximation of the parameter vector [20] is adopted in order to derive three new sparsity promoting algorithms for Volterra filters: the l_0-norm Volterra LMS (l_0-VLMS), the l_0-norm Volterra normalized LMS (l_0-VNLMS), and the l_0-norm Volterra Affine Projection (l_0-VAP) algorithms. We show through experimental results that the proposed algorithms can obtain superior performance with respect to classical algorithms and algorithms based on the l_1-norm.

This work is organized as follows. Section II reviews the fundamentals of the Volterra series and the l_0-norm approximation. Then, in Section III, three LMS-based Volterra algorithms capable of exploiting sparsity in the parameters by means of the l_0-norm approximation are proposed. These algorithms are tested in Section IV considering both synthetic and real-world data coming from a preamplifier experiment. In both cases, the performance of the proposed algorithms are superior, in comparison with the classical Volterra algorithms, which do not benefit from the system sparsity, and their zero-attractor (ZA) versions, which employ the l_1 norm to model sparsity. The conclusions are drawn in Section V.

II. Volterra series and l_0-norm

This section provides a brief review of the Volterra series expansion and the l_0-norm approximation in Subsections II-A and II-B, respectively.

A. The Volterra series

Assume a nonlinear, time-invariant, causal, finite-memory, and continuous relationship $d(k) = f(\mathbf{x}_1(k))$, where $\mathbf{x}_1(k) \triangleq [x(k)\ x(k-1)\ \cdots\ x(k-N)]^T$, $x(k)$ and $d(k)$ are the input and the desired signals at the discrete time instant k, and N is the system memory length. This desired signal can be estimated by a truncated Volterra series expansion of order P as

$$d(k) = \sum_{p=0}^{P} W_p(\mathbf{x}_1(k)) + n(k), \tag{1}$$

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) Finance Code 001 and by the São Paulo Research Foundation (FAPESP) grants #2015/22308-2, #2019/06280-1.

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

where $n(k)$ is the measurement noise, and using the triangular form of the Volterra series [1] we define $W_p(\mathbf{x}_1(k))$ as

$$W_p(\mathbf{x}_1(k)) \triangleq \sum_{l_1=0}^{N} \dots \sum_{l_p=l_{p-1}}^{N} w_p(l_1,\dots,l_p) \prod_{i=1}^{p} x(n-l_i), \quad (2)$$

with $w_p(l_1,\dots,l_p)$, for all $l_1,\dots,l_p$, being the p-th order Volterra kernel. Assuming the constant term of the Volterra series expansion w_0 is zero, for given input and desired signals, our target is to calculate the Volterra kernels for $p=1,\dots,P$, for all $l_1,\dots,l_p$.

To allow for a compact representation, it is convenient to rewrite equations (1) and (2) using vector notation [1], [7]. Thus, by setting $W_p(\mathbf{x}_1(k)) = \mathbf{x}_p^T(k)\mathbf{w}_p$, where $\mathbf{x}_p(k)$ is the vector formed by all input sample products appearing in (2), and $\mathbf{w}_p$ is the vector containing the corresponding Volterra coefficients $w_p(l_1,\dots,l_p)$, (1) can be expressed as

$$d(k) = \mathbf{x}^T(k)\mathbf{w} + n(k), \quad (3)$$

where $\mathbf{x}(k) \triangleq [\mathbf{x}_1^T(k) \ \cdots \ \mathbf{x}_P^T(k)]^T$, $\mathbf{w} \triangleq [\mathbf{w}_1^T \ \cdots \ \mathbf{w}_P^T]^T$, and we have assumed that the constant term of the Volterra series expansion is zero.

B. The l_0-norm approximation

The l_0-norm of a vector $\mathbf{w} = [w_0 \ w_1 \ \cdots \ w_N]^T \in \mathbb{R}^{N+1}$ is defined as $\|\mathbf{w}\|_0 \triangleq \#\{i|w_i \neq 0\}$, where $\#$ stands for the cardinality of a finite set, i.e., the l_0-norm is the number of nonzero elements of a vector. Although the l_0-norm is directly related to the sparsity of vectors, its practical use is very limited because [20], [27]: (i) it leads to an NP-hard problem, (ii) most "sparse systems" found in practice are actually *compressible systems* (roughly, their energy is concentrated in a few coefficients, but every coefficient can be different from zero), and (iii) it is often an ill-conditioned problem, meaning that small perturbations on its argument may lead to very different l_0-norm results.

In the compressive sensing literature, most works circumvent the aforementioned issues by replacing the l_0-norm with the l_1-norm, which also has the advantage of being convex [12]–[17]. However, recent works indicate that superior results can be achieved by employing an approximation of the l_0-norm [18]–[26]. Such approximation should be an almost everywhere differentiable function to allow for gradient-based algorithms. Among the numerous approximations for the l_0-norm, in this paper we use the Geman-McClure function (GMF) [20], [23] which, for $\mathbf{w} \in \mathbb{R}^{N+1}$, is defined as

$$G_\beta(\mathbf{w}) \triangleq \sum_{i=0}^{N} \left(1 - \frac{1}{1+\beta|w_i|}\right), \quad (4)$$

where $\beta \in \mathbb{R}_+$ is a parameter responsible for controlling the agreement between quality of the approximation and smoothness of G_β. Fig. 1 depicts the univariate and bivariate GMFs for some values of β. The gradient of G_β is given by

$$\nabla G_\beta(\mathbf{w}) \triangleq \mathbf{g}_\beta(\mathbf{w}) \triangleq [g_\beta(w_0) \ \cdots \ g_\beta(w_N)]^T, \quad (5)$$

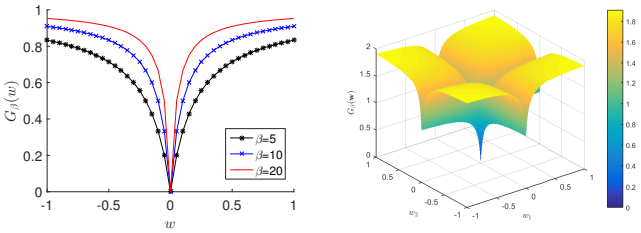


Fig. 1. (Left) univariate GMFs for $\beta = 5$, 10, and 20. (Right) bivariate GMF for $\beta = 20$. Observe that G_β gets closer to the l_0-norm as $\beta \to \infty$.

where the entries of vector $\mathbf{g}_\beta(\mathbf{w})$ are

$$g_\beta(w_i) \triangleq \frac{\partial G_\beta(\mathbf{w})}{\partial w_i} = \frac{\beta \mathrm{sgn}(w_i)}{(1+\beta|w_i|)^2}, \quad (6)$$

where $\mathrm{sgn}(\cdot)$ denotes the sign function.

III. l_0-NORM LMS-BASED VOLTERRA FILTERS

One of the main issues involving Volterra filters is the large number of parameters to be estimated since both $\mathbf{w}$ and $\mathbf{x}(k)$ in (3) usually belong to a very high dimensional space. In fact, the Volterra filter tackles the problem of modeling nonlinear relations by accounting for every possible nonlinear combination of input data. Due to the excessive amount of parameters and to the resulting bad conditioning of the autocorrelation matrix, Volterra filters usually converge slowly. However, many times the nonlinear relation between $d(k)$ and $\mathbf{x}(k)$ depends only on a few entries of $\mathbf{x}(k)$, meaning that $\mathbf{w}$ is sparse. So, in this section, we propose some LMS-based adaptive Volterra filters capable of exploiting systems sparsity in order to achieve faster convergence and/or lower steady-state MSE. More specifically, we derive the l_0-norm Volterra LMS (l_0-VLMS), the l_0-norm Volterra normalized LMS (l_0-VNLMS), and the l_0-norm Volterra Affine Projection (l_0-VAP) algorithms in Subsections III-A, III-B, and III-C, respectively.

A. The l_0-VLMS algorithm

The adaptive filter that approximates the desired signal $d(k)$ utilizing a truncated Volterra series of order P has output

$$y(n) = \mathbf{x}^T(k)\mathbf{w}(k), \quad (7)$$

where $\mathbf{x}^T(k)$ is the same input data vector of (3) and $\mathbf{w}(k)$ is the vector collecting the corresponding coefficients of the adaptive filter. The error signal $e(k)$ of the nonlinear adaptive filter is $e(k) = d(k) - y(k)$.

We can now define the objective function of the l_0-VLMS algorithm as follows

$$\zeta_{l_0\text{-VLMS}} = \frac{1}{2}|e(k)|^2 + \alpha\|\mathbf{w}(k)\|_0, \quad (8)$$

where $\alpha \in \mathbb{R}_+$ is the l_0-norm penalty weight. If we replace the l_0-norm with its approximation, we get

$$\zeta_{l_0\text{-VLMS}} = \frac{1}{2}|e(k)|^2 + \alpha G_\beta(\mathbf{w}(k)). \quad (9)$$

Using the steepest-descent method, the recursion rule of the l_0-VLMS algorithm can be given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}(k) - \mu\alpha \mathbf{g}_\beta(\mathbf{w}(k)), \quad (10)$$

where μ is the step-size parameter. However, to improve convergence speed, it is often convenient to adopt different step-size parameters for the terms of different order, which generally have different average power. Therefore, the update equation of the l_0-VLMS algorithm is characterized by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{M} e(k)\mathbf{x}(k) - \mu_1\alpha \mathbf{g}_\beta(\mathbf{w}(k)), \quad (11)$$

where $\mathbf{M} = \mathrm{diag}([\mu_1, \cdots, \mu_1, \mu_2, \cdots, \mu_2, \cdots, \mu_P, \cdots, \mu_P])$.

B. The l_0-VNLMS algorithm

The cost function of the l_0-VNLMS algorithm, using the l_0-norm approximation, is defined as

$$\begin{aligned} &\text{minimize } \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 + \alpha G_\beta(\mathbf{w}(k+1)), \\ &\text{subject to } d(k) = \mathbf{x}^T(k)\mathbf{w}(k+1). \end{aligned} \quad (12)$$

By using the method of Lagrange multipliers, we have

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\lambda}{2}\mathbf{x}(k) - \frac{\alpha}{2}\mathbf{g}_\beta(\mathbf{w}(k+1)), \quad (13)$$

where λ is the Lagrange multiplier. Then, by utilizing the constraint in (12), we can find λ and substitute it in the equation above. Thus, we attain

$$\begin{aligned} \mathbf{w}(k+1) = \mathbf{w}(k) &+ \frac{\mu\mathbf{x}(k)e(k)}{\mathbf{x}^T(k)\mathbf{x}(k)+\delta} \\ &+ \frac{\mu\alpha}{2}\Big[\frac{\mathbf{x}(k)\mathbf{x}^T(k)}{\mathbf{x}^T(k)\mathbf{x}(k)+\delta} - \mathbf{I}\Big]\mathbf{g}_\beta(\mathbf{w}(k+1)), \end{aligned} \quad (14)$$

where $\mathbf{I}$ is the identity matrix with dimension equal to the AVF number of coefficients, and $\delta > 0$ is a small constant to avoid division by zero. If we replace $\mathbf{g}_\beta(\mathbf{w}(k+1))$ with $\mathbf{g}_\beta(\mathbf{w}(k))$ in order to form the recursion and use different step-sizes for the terms with different orders, then the update equation of the l_0-VNLMS algorithm can be characterized as

$$\begin{aligned} \mathbf{w}(k+1) = \mathbf{w}(k) &+ \frac{\mathbf{M}\mathbf{x}(k)e(k)}{\mathbf{x}^T(k)\mathbf{x}(k)+\delta} \\ &+ \frac{\mu_1\alpha}{2}\Big[\frac{\mathbf{x}(k)\mathbf{x}^T(k)}{\mathbf{x}^T(k)\mathbf{x}(k)+\delta} - \mathbf{I}\Big]\mathbf{g}_\beta(\mathbf{w}(k)), \end{aligned} \quad (15)$$

where $\mathbf{M}$ is the diagonal matrix containing the step-size parameters and μ_1 is its first entry.

C. The l_0-VAP algorithm

When the input signal is correlated, a potential approach to increase convergence rate is data-reusing [7], [28]. To employ the data-reusing technique in AVF algorithms, let us use the last $L+1$ values of $\mathbf{x}(k)$ and $d(k)$ and introduce the input matrix $\mathbf{X}(k)$ and the desired signal vector $\mathbf{d}(k)$ as

$$\mathbf{X}(k) = [\mathbf{x}(k)\ \mathbf{x}(k-1)\ \cdots\ \mathbf{x}(k-L)], \quad (16)$$

$$\mathbf{d}(k) = [d(k)\ d(k-1)\ \cdots\ d(k-L)]^T, \quad (17)$$

and define the error signal vector $\mathbf{e}(k) \triangleq \mathbf{d}(k) - \mathbf{X}^T(k)\mathbf{w}(k)$.

By utilizing the l_0-norm approximation, we can now introduce the objective function of the l_0-VAP algorithm as

$$\begin{aligned} &\text{minimize } \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 + \alpha G_\beta(\mathbf{w}(k+1)), \\ &\text{subject to } \mathbf{d}(k) = \mathbf{X}^T(k)\mathbf{w}(k+1). \end{aligned} \quad (18)$$

Similarly to the l_0-VNLMS algorithm, we may employ the method of Lagrange multipliers to solve the problem. After doing so, we obtain the following recursion rule

$$\begin{aligned} \mathbf{w}(k+1) = \mathbf{w}(k) &+ \mu\mathbf{X}(k)\mathbf{S}(k)\mathbf{e}(k) \\ &+ \frac{\mu\alpha}{2}\Big[\mathbf{X}(k)\mathbf{S}(k)\mathbf{X}^T(k) - \mathbf{I}\Big]\mathbf{g}_\beta(\mathbf{w}(k+1)), \end{aligned} \quad (19)$$

where $\mathbf{S}(k) = \Big(\mathbf{X}^T(k)\mathbf{X}(k) + \delta\mathbf{I}_{L+1}\Big)^{-1}$, $\delta\mathbf{I}_{L+1}$ is utilized to avoid numerical problems, $\delta > 0$ is a small constant, and $\mathbf{I}_{L+1}$ is the identity matrix of dimension $L+1$. Finally, we replace $\mathbf{g}_\beta(\mathbf{w}(k+1))$ with $\mathbf{g}_\beta(\mathbf{w}(k))$ to obtain the recursion and assume different step-sizes for the terms with different order employing the diagonal matrix $\mathbf{M}$. The update rule of the l_0-VAP algorithm results in the following

$$\begin{aligned} \mathbf{w}(k+1) = \mathbf{w}(k) &+ \mathbf{M}\mathbf{X}(k)\mathbf{S}(k)\mathbf{e}(k) \\ &+ \frac{\mu_1\alpha}{2}\Big[\mathbf{X}(k)\mathbf{S}(k)\mathbf{X}^T(k) - \mathbf{I}\Big]\mathbf{g}_\beta(\mathbf{w}(k)). \end{aligned} \quad (20)$$

IV. Experimental results

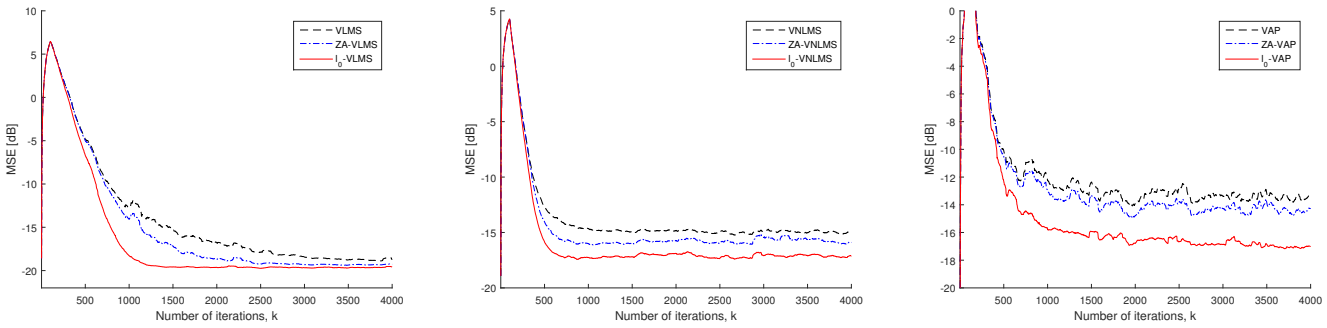
In this section, considering a system identification problem for synthetic and real-world data, we compare the performance of the proposed AVFs, l_0-VLMS, l_0-VNLMS, l_0-VAP, with the classical Volterra algorithms, VLMS, VNLMS, VAP, given by (11), (15), and (20), respectively, for $\alpha = 0$, and with ZA-VLMS [4], ZA-VNLMS [4], and ZA-VAP algorithms. The ZA-VAP algorithm is derived similarly to the l_0-VAP algorithm but the l_0-norm approximation is replaced by the l_1-norm, thus its update equation can be obained by substituting $\mathbf{g}_\beta(\mathbf{w}(k))$ with $\mathrm{sgn}(\mathbf{w}(k))$ in Equation (20). In Subsection IV-A, the algorithms are utilized with synthetic data. However, in Subsection IV-B they are used in a real-world problem. For both scenarios, the parameter β in the l_0-norm approximation is set to 20, and the learning curves have been smoothed by a box filter of length 100.

A. Simulation experiment

For the synthetic example, the nonlinear Volterra channel to be identified is given by

$$\begin{aligned} d(k) = &-0.76x(k) + 0.5x^2(k) + 2x(k)x(k-2) \\ &- 0.5x^2(k-3) + n(k), \end{aligned} \quad (21)$$

where $n(k)$ is a zero-mean white Gaussian noise with variance 0.01. The nonlinear AVF is assumed to be of order and memory length 3; i.e., $N = P = 3$. The input signal is a zero-mean white Gaussian noise with unit variance except for the VAP, the ZA-VAP, and the l_0-VAP algorithms, where the input signal is a first-order autoregressive process generated by $x(k) = 0.95x(k-1) + m(k)$ and $m(k)$ has the zero-mean Gaussian distribution with unit variance. For all algorithms, the regularization parameter δ is 10^{-9}. Moreover, all algorithms



(a) (b) (c)

Fig. 2. The MSE learning curves of: (a) the VLMS, the ZA-VLMS, and the l_0-VLMS algorithms; (b) the VNLMS, the ZA-VNLMS, and the l_0-VNLMS algorithms; (c) the VAP, the ZA-VAP, and the l_0-VAP algorithms, for the correlated input signal and $L = 1$.

are initialized with the null vector, and the learning curves have been attained by averaging the outcomes of 500 independent trials.

Fig. 2(a) illustrates the MSE learning curves of the VLMS, the ZA-VLMS, and the l_0-VLMS algorithms. The step-size parameters μ_1, μ_2, and μ_3 are selected to be 0.04, 0.004, and 0.0004, respectively. These parameters have been chosen using a trial and error procedure to obtain a similar convergence rate at the early iterations in all tested algorithms. Both the l_0-norm and the l_1-norm penalty weights in the l_0-VLMS and the ZA-VLMS algorithms are set to 0.005. As can be seen, the l_0-VLMS algorithm has the lowest MSE and the fastest convergence rate, followed by the ZA-VLMS and the VLMS algorithms. The better performances have been obtained thanks to the efficient system sparsity exploitation through the l_0-norm.

Fig. 2(b) shows the MSE learning curves of the VNLMS, the ZA-VNLMS, and the l_0-VNLMS algorithms. The step-size parameters μ_1, μ_2, and μ_3 are chosen as 0.5, 0.5, and 0.05, respectively. The l_0-norm and the l_1-norm penalty weights in the l_0-VNLMS and the ZA-VNLMS algorithms are 0.001 and 0.03, respectively. We can observe that the l_0-VNLMS algorithm has the lowest MSE and the highest convergence speed, followed by the ZA-VNLMS and the VNLMS algorithms. Furthermore, as can be seen, the normalized AVFs in Fig. 2(b) have higher convergence rate than those in Fig. 2(a); however, the AVFs in Fig. 2(b) cannot attain the same MSE as those in Fig. 2(a).

Fig. 2(c) depicts the MSE learning curves of the VAP, the ZA-VAP, and the l_0-VAP algorithms when the input signal is the correlated signal. The data-reuse parameter L is equal to one. The step-size parameters μ_1, μ_2, and μ_3 are set to be 0.1, 0.1, and 0.05, respectively. Both the l_0-norm and the l_1-norm penalty weights in the l_0-VAP and the ZA-VAP algorithms are equal to 0.001. We can see that the l_0-VAP algorithm has the lowest MSE, followed by the ZA-VAP and the VAP algorithms.

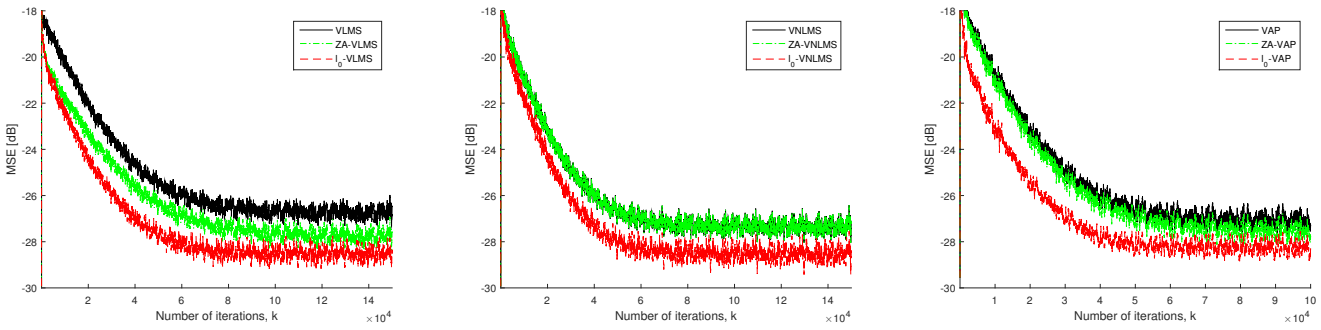
B. Real-world experiment

In this subsection, we have utilized the algorithms tested in the previous subsection in a real-world experiment. Indeed, we want to identify the coefficients of a preamplifier using a white input with the probability density function $p(x) = \frac{1}{\pi\sqrt{1-x^2}}$, where $x \in [-1, 1]$. For the settings adopted in this experiment, on a sinusoidal input at 200 Hz, the preamplifier presents a second and third order harmonic distortion of 5.6% and 20.2%, respectively. The reader can see [29] for more details about the settings in this experiment. The AVFs have memory 15 and order 3. In order to simplify the step-size choice, the samples of vector $\mathbf{x}(n)$ in (7) were normalized by the square-root of their average power and the same step-size was used for all coefficients. The learning curves are the average of outcomes of 50 independent runs.

Fig. 3(a) illustrates the learning curves of the VLMS, the ZA-VLMS, and the l_0-VLMS algorithms. The step-size parameter is 0.0013. It has been obtained with a trial and error procedure and is the value that obtained the minimum MSE (apart from a dB fraction) with the fastest convergence speed in the VLMS algorithm. Moreover, the l_0-norm and the l_1-norm penalty weights are set to be 0.0001 and 0.0005, respectively. As can be seen, the l_0-VLMS algorithm has the lowest MSE and the highest convergence rate, followed by the ZA-VLMS and the VLMS algorithms.

Fig. 3(b) presents the learning curves of the VNLMS, the ZA-VNLMS, and the l_0-VNLMS algorithms. The step-size parameter is 0.9 and is the value that obtained the minimum MSE (apart from a dB fraction) with the fastest convergence speed in the VNLMS algorithm. Furthermore, the l_0-norm and the l_1-norm penalty weights are adopted as 10^{-7} and 0.0005, respectively, and the regularization parameter δ is chosen as 10^{-9}. We can observe that the l_0-VNLMS algorithm has the best MSE and convergence speed as compared to the VNLMS and the ZA-VNLMS algorithms. Also, note that the learning curves of the VNLMS and the ZA-VNLMS algorithms are very similar, and they are overlaid. It is worth to mention that the learning curves of the AVFs in Fig. 3(b) are a little faster than those in Fig. 3(a).

Fig. 3(c) shows the learning curves of the VAP, the ZA-VAP, and the l_0-VAP algorithms when $L = 1$. The step-size parameter for each coefficient is the same used for Fig. 3(b), and the regularization parameter is 10^{-9}. Both the l_0-norm and the l_1-norm penalty weights are selected as 5×10^{-7}. As can be seen, the l_0-VAP algorithm has the highest convergence



(a) (b) (c)

Fig. 3. The MSE learning curves of: (a) the VLMS, the ZA-VLMS, and the l_0-VLMS algorithms; (b) the VNLMS, the ZA-VNLMS, and the l_0-VNLMS algorithms; (c) the VAP, the ZA-VAP, and the l_0-VAP algorithms, for $L = 1$.

rate and the lowest MSE in comparison with the VAP and the ZA-VAP algorithms. Furthermore, as compared to the VAP algorithm, the superiority of the ZA-VAP algorithm is not significant.

V. Conclusions

In this paper, three novel sparsity promoting l_0-norm AVF algorithms have been proposed. The novel algorithms are based on the Geman-McClure approximation of the l_0-norm. Through experimental results, it has been shown that the proposed AVFs can obtain improved performance in terms of convergence speed and steady-state MSE with respect to both the classical AVFs, and the zero attracting AVFs based on l_1-norm.

References

[1] V. Mathews and G. Sicuranza, *Polynomial signal processing*, Wiley, New York, USA, 2000.

[2] N.V. George and G. Panda, "Advances in active noise control: A survey, with emphasis on recent nonlinear techniques," *Signal Processing*, vol. 93, no. 2, pp. 363–377, Feb. 2013.

[3] M. Zeller and W. Kellermann, "Fast and robust adaptation of DFT-domain Volterra filters in diagonal coordinates using iterated coefficient updates," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1589–1604, Mar. 2010.

[4] C. Shi and Y. Kajikawa, "Identification of the parametric array loudspeaker with a Volterra filter using the sparse NLMS algorithm," in *Proc. of ICASSP 2015*, Brisbane,Australia, Apr. 2015, pp. 3372–3376.

[5] T.W. Berger, D. Song, R.H.M. Chan, and V.Z. Marmarelis, "The neurobiological basis of cognition: Identification by multi-input, multioutput nonlinear dynamic modeling," *Proceedings of the IEEE*, vol. 98, no. 3, pp. 356–374, Mar. 2010.

[6] R.A. do Prado, F.d.R. Henriques, and D.B. Haddad, "Sparsity-aware distributed adaptive filtering algorithms for nonlinear system identification," in *Proc. of IJCNN 2018*, Rio de Janeiro, Brazil, July 2018.

[7] P.S.R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Springer, New York, USA, 4th edition, 2013.

[8] C. Hofmann, C. Huemmer, M. Guenther, and W. Kellermann, "Significance-aware filtering for nonlinear acoustic echo cancellation," *EURASIP Journal on Advances in Signal Processing*, vol. 2016, no. 1, pp. 113, Nov. 2016.

[9] E.L.O. Batista, O.J. Tobias, and R. Seara, "A sparse-interpolated scheme for implementing adaptive Volterra filters," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2022–2035, Apr. 2010.

[10] A. Fermo, A. Carini, and G.L. Sicuranza, "Low complexity nonlinear adaptive filters for acoustic echo cancellation," *European Transactions on Telecommunications*, vol. 14, no. 2, pp. 161–169, Mar. 2003.

[11] H. Yazdanpanah, P.S.R. Diniz, and M.V.S. Lima, "A simple set-membership affine projection algorithm for sparse system modeling," in *Proc. of EUSIPCO 2016*, Budapest, Hungary, Aug. 2016, pp. 1798–1802.

[12] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, New York, USA, 2010.

[13] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Springer, New York, USA, 2013.

[14] E.J. Candes, M.B. Wakin, and S.P. Boyd, "Enhancing sparsity by reweighted l_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 877–905, Dec. 2008.

[15] R. Meng, R.C. de Lamare, and V.H. Nascimento, "Sparsity-aware affine projection adaptive algorithms for system identification," in *Sensor Signal Processing for Defence*, London, U.K., Sept. 2011, pp. 1–5.

[16] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online sparse system identification and signal reconstruction using projections onto weighted l_1 balls," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 936–952, Mar. 2011.

[17] R.L. Das and M. Chakraborty, "A zero attracting proportionate normalized least mean square algorithm," in *Proc. of APSIPA 2012*, Dec. 2012, pp. 1–4.

[18] M.V.S. Lima, W.A. Martins, and P.S.R. Diniz, "Affine projection algorithms for sparse system identification," in *Proc. of ICASSP 2013*, Vancouver, Canada, May 2013, pp. 5666–5670.

[19] M.V.S. Lima, I. Sobron, W.A. Martins, and P.S.R. Diniz, "Stability and MSE analyses of affine projection algorithms for sparse system identification," in *Proc. of ICASSP 2014*, Florence, Italy, May 2014, pp. 6399–6403.

[20] M.V.S. Lima, T.N. Ferreira, W.A. Martins, and P.S.R. Diniz, "Sparsity-aware data-selective adaptive filters," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4557–4572, Sept. 2014.

[21] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l^0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, Jan. 2009.

[22] Y. Gu, J. Jin, and S. Mei, "l_0 norm constraint LMS algorithm for sparse system identification," *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 774–777, Sept. 2009.

[23] J. Trzasko and A. Manduca, "Highly undersampled magnetic resonance image reconstruction via homotopic l_0-minimization," *IEEE Transactions on Medical Imaging*, vol. 28, no. 1, pp. 106–121, Jan. 2009.

[24] H. Yazdanpanah and P.S.R. Diniz, "Recursive least-squares algorithms for sparse system modeling," in *Proc. of ICASSP 2017*, New Orleans, LA, USA, Mar. 2017, pp. 3879–3883.

[25] H. Yazdanpanah, J.A. Apolinário Jr., P.S.R. Diniz, and M.V.S. Lima, "l_0-norm feature LMS algorithms," in *Proc. of GlobalSIP 2018*, Anaheim, California, USA, Nov. 2018, pp. 311–315.

[26] Z. Shen, T. Huang, and K. Zhou, "l_0-norm constraint normalized logarithmic subband adaptive filter algorithm," *Signal, Image and Video Processing*, vol. 12, no. 5, pp. 861–868, 2018.

[27] Y. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge University Press, Cambridge, UK, 2012.

[28] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electronics and Communications in Japan*, vol. 67-A, no. 5, pp. 19–27, 1984.

[29] A. Carini and G.L. Sicuranza, "A study about Chebyshev nonlinear filters," *Signal Processing*, vol. 122, pp. 24–32, May 2016.