IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 14, 2019, accepted September 24, 2019, date of publication September 27, 2019, date of current version October 10, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2944187*

# Low-Complexity Feature Stochastic Gradient Algorithm for Block-Lowpass Systems

**HAMED YAZDANPANAH[1], (Member, IEEE), PAULO S. R. DINIZ[2], (Fellow, IEEE), AND MARKUS V. S. LIMA[2], (Member, IEEE)**

[1]Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, São Paulo 05508-090, Brazil
[2]DEL-DEE/Poli and the Program of Electrical Engineering, COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro 68504, Brazil

Corresponding author: Hamed Yazdanpanah (hamed.yazdanpanah@smt.ufrj.br)

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

**ABSTRACT** New approaches have been proposed to detect and exploit sparsity in adaptive systems. However, the sparsity is not always explicit among the system coefficients, thus requiring some tools to reveal it. By means of the so-called feature function, we propose the low-complexity feature stochastic gradient (LF-SG) algorithm to exploit hidden sparsity. The proposed algorithm aims at reducing the computational load of the learning process, as compared to the least-mean-square (LMS) algorithm. We focus on block-lowpass systems, but the proposed approach can easily be adapted to exploit other kinds of features of the unknown system, e.g., highpass and bandpass characteristics. Then, we analyze some properties of the LF-SG algorithm, namely its steady-state mean squared error (MSE), its bias, and the choice of the step-size parameter. Simulation results illustrate the competitive MSE performance of the LF-SG in comparison with the LMS, but the former algorithm requires much fewer multiplication operations to identify lowpass systems. For instance, to identify a measured room impulse response, the LF-SG algorithm realized less than half of the multiplication operations required by the LMS algorithm.

**INDEX TERMS** Adaptive filtering, block-lowpass systems, computational complexity, feature, LMS, sparsity.

## I. INTRODUCTION

Sparse signals and models have received great attention from researchers working in the adaptive filtering field in the last years. Different approaches have been utilized to exploit sparsity, such as the $l_1$-norm regularization [1], [2], the $l_0$-norm regularization [3]–[7], the proportionate technique [8], [9], the thresholding and shrinkage strategies [10]–[12], and the oracle algorithm [11], [12]. The interest to exploit sparsity has been increased in the last years due to several practical issues that make sparsity more likely to happen [13], [14]. For example, increasing of the sampling rate of speech/audio signals leads to long room impulse responses and echo paths, whose energies mostly concentrate in a few coefficients. Moreover, with the huge increase in computational power, researchers are using more complex models (i.e., models with more coefficients), thus frequently requiring the usage of a sparsity-promoting regularization to accelerate convergence and/or prevent overfitting [15], [16].

The associate editor coordinating the review of this manuscript and approving it for publication was Yu-Huei Cheng.

All sparsity-aware adaptive filtering algorithms mentioned in the last paragraph assume that sparsity is directly observed in the impulse response of the system to be identified. Recently, the feature LMS (F-LMS) algorithm has been proposed to exploit the hidden sparsity of systems [17]. More precisely, the F-LMS algorithm uses prior information about a given characteristic of the unknown system, herein called *feature*, to map the filter coefficients to a different domain. This feature domain of the coefficients admits sparse representation. After this transformation of variables, the so-called hidden sparsity is revealed, and the F-LMS algorithm capitalizes on it by applying some sparsity-promoting regularization. In [17], [18], features frequently found in practical systems (like lowpass and highpass frequency responses [19]–[23]) have been exploited in order to accelerate convergence or reduce steady-state mean squared error (MSE). The price to be paid is an increase in computational complexity in comparison with the LMS algorithm.

In this paper, we propose a new algorithm, called *low-complexity feature stochastic gradient* (LF-SG) algorithm that exploits the system feature to reduce the number of arithmetic operations while obtaining similar accuracy, as compared to the LMS algorithm. Due to its practical appeal, we focus on *lowpass and block-lowpass systems*, but adapting the proposed approach to other kinds of feature is straightforward. Block-lowpass systems have impulse responses comprised of several blocks of coefficients with a smooth variation. That is, each block individually has lowpass frequency response, but abrupt transitions may occur from one block to another. In this view, a lowpass system is a particular case of a block-lowpass system in which there is a single block of smoothly-varying coefficients. The LF-SG algorithm reveals the hidden sparsity within block-lowpass systems by using the first coefficient of a given block to "predict" the remaining coefficients of the same block. Then, it quantizes the residuals when they are very small, thus generating a sparse set of coefficients. Besides to proposing a new algorithm, we also analyze some of its properties concerning the step-size, its bias and steady-state MSE performance.

This paper is organized as follows. Section II presents the proposed LF-SG algorithm. Some properties related to the bias, the step-size, and the steady-state MSE of the LF-SG algorithm are analyzed in Section III. Simulation results considering both synthetic and measured unknown systems are shown in Section IV. The conclusions are drawn in Section V.

*Notation:* Scalars are represented by lowercase letters. Vectors (matrices) are denoted by lowercase (uppercase) boldface letters. The $i$th element of a given vector $\mathbf{z}$ is presented by $z_i$. At iteration $k$, the input vector and the weight vector are represented by $\mathbf{x}(k)$, $\mathbf{w}(k) \in \mathbb{R}^{N+1}$, respectively, where $N$ is the adaptive filter order. We define the usual adaptive filter output and error as $y(k) \triangleq \mathbf{w}^T(k)\mathbf{x}(k)$ and $e(k) \triangleq d(k) - y(k)$, respectively, where $d(k) \in \mathbb{R}$ is the desired signal.

## II. THE LF-SG ALGORITHM

In this section, we propose the low-complexity feature stochastic gradient (LF-SG) algorithm to exploit the block-lowpass feature of the system to be identified aiming at reducing the computational cost of calculating the output signal, in comparison with the LMS algorithm. As previously explained, we have chosen to address the block-lowpass feature due to its high practical appeal, but generalizing the approach proposed here to other features is straightforward.

The key idea is to reduce the number of multiplication operations required for computing the output signal when there is a strong relation between neighboring coefficients. In block-lowpass systems, for example, the neighboring coefficients within a given block vary smoothly (see Figure 1(a) for an example). Therefore, for a block of coefficients whose amplitudes are similar up to a given small tolerance $\epsilon > 0$, we quantize these small differences to zero to obtain simpler blocks (see Figure 1(d)) that admit a sparse representation (see Figure 1(b)), thus revealing the hidden sparsity on the coefficients and also allowing a reduction in the number of multiplication operations required to compute the adaptive filter output $y(k) \triangleq \mathbf{w}^T(k)\mathbf{x}(k)$. Mathematically, if the value of $|w_{m+i}(k) - w_m(k)| < \epsilon$ for $i = 1, 2, \ldots, j$, then in the calculation of $y(k)$ instead of computing part of the inner product as follows

$$w_m(k)x_m(k) + \cdots + w_{m+j}(k)x_{m+j}(k), \tag{1}$$

we can approximate it this way

$$w_m(k)(x_m(k) + x_{m+1}(k) + \cdots + x_{m+j}(k)), \tag{2}$$

where only a single coefficient $w_m(k)$ is used. As a result, we decrease the number of multiplication operations from $j+1$ to one. Hence, the first coefficient of a given block, herein called *the reference coefficient*, is used to predict/approximate the other coefficients of the same block. A given block of coefficients ends when we find the first coefficient whose amplitude is not $\epsilon$-close to the amplitude of the reference coefficient, i.e., the absolute value of their difference exceeds the tolerance $\epsilon$.

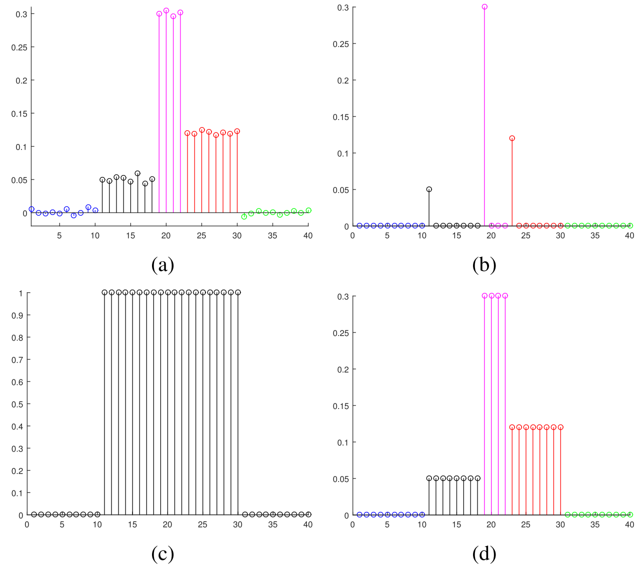



**FIGURE 1.** Toy example illustrating: (a) vector $\mathbf{w}(k)$; (b) $\mathbf{w}_s(k)$, which is the result of applying $\mathcal{F}_\epsilon$ to $\mathbf{w}(k)$; (c) auxiliary vector $\mathbf{b}(k)$; and (d) $\hat{\mathbf{w}}_s(k)$, the estimate of $\mathbf{w}(k)$ by using $\mathbf{w}_s(k)$ and $\mathbf{b}(k)$. Each color indicates a group of coefficients that belong to the same block, i.e., which are represented by a single reference coefficient in $\mathbf{w}_s(k)$.

For each block of coefficients, the LF-SG algorithm keeps only the reference coefficient and replaces the remaining ones with zeros. Furthermore, when the absolute value of a coefficient is less than $\epsilon$, the LF-SG algorithm also replaces it with zero to avoid unnecessary multiplications [24], [25]. Thus, two subsets of parameters are replaced by zero: (I) the coefficients whose absolute values are less than $\epsilon$, and (II) the consecutive coefficients whose amplitudes are $\epsilon$-close to the reference coefficient.

The above reasoning can be implemented by means of the *feature function*, $\mathcal{F}_\epsilon : \mathbb{R}^{N+1} \to \mathbb{R}^{N+1}$, applied to the weight vector of the adaptive filter. The algorithm of the feature function is described in Table 1.

As can be observed in Table 1, the feature function replaces the subsets (I) and (II) of the coefficients of $\mathbf{w}(k)$ with zero. Let us define $\mathbf{w}_s(k) \triangleq \mathcal{F}_\epsilon(\mathbf{w}(k))$. Figures 1(a) and 1(b) depict an example of the impulse responses $\mathbf{w}(k)$ and $\mathbf{w}_s(k)$ for $\epsilon = 0.02$, respectively. As can be observed, $\mathbf{w}(k)$ has

**TABLE 1. Feature function algorithm.**

```
w_s(k) ≜ F_ε(w(k))

Initialization
p = 0
Do for i = 0 to N
  if |w_i(k) − p| < ε
    w_{s_i}(k) = 0
  else if |w_i(k) − p| ≥ ε and |w_i(k)| ≥ ε
    w_{s_i}(k) = w_i(k)
    p = w_i(k)
  else if |w_i(k) − p| ≥ ε and |w_i(k)| < ε
    w_{s_i}(k) = 0
    p = 0
  end
end
```

40 nonzero coefficients, and after using the feature function, 37 of them are replaced by zeros. In summary, the feature function maps $\mathbf{w}(k)$ to another domain; if the original set of coefficients $\mathbf{w}(k)$ represents a block-lowpass system, then the application of the feature function to $\mathbf{w}(k)$ results in a sparse system $\mathbf{w}_s(k)$.

Our goal is to utilize $\mathbf{w}_s(k)$ in the calculation of the output signal. To do so, we must determine from which subset of coefficients of $\mathbf{w}(k)$ the zero elements of $\mathbf{w}_s(k)$ came from, i.e., subsets (I) or (II). In fact, for some $i$, $w_{s_i}(k)$ is zero if and only if $w_i(k)$ belongs to the subsets (I) or (II). If $w_i(k)$ belongs to the subset (I), then we can directly apply $w_{s_i}(k)$ to calculate the output signal. That is, we use $w_{s_i}(k)x_i(k) = 0$. However, if $w_i(k)$ belongs to the subset (II), then we must apply the last nonzero coefficient of $\mathbf{w}_s(k)$ detected before $w_{s_i}(k)$ to compute the output signal. Assume that this nonzero coefficient has index $m$, then we use $w_{s_m}(k)$ instead of $w_i(k)$ since their values are $\epsilon$-close to each other. Hence, in the calculation of the output signal, we use $w_{s_m}(k)x_i(k)$ instead of $w_{s_i}(k)x_i(k)$.

In order to determine the origin of the zero coefficients in $\mathbf{w}_s(k)$, we define the binary vector $\mathbf{b}(k) \in \{0, 1\}^{N+1}$ as

$$\mathbf{b}(k) \triangleq \mathbf{g}_\epsilon(\mathbf{w}(k)) = [g_\epsilon(w_0(k)) \ \cdots \ g_\epsilon(w_N(k))]^T, \quad (3)$$

where $g_\epsilon(\cdot) : \mathbb{R} \to \{0, 1\}$ is given by

$$g_\epsilon(w) \triangleq \begin{cases} 1, & \textit{if } |w| \geq \epsilon, \\ 0, & \textit{if } |w| < \epsilon. \end{cases} \quad (4)$$

In Figure 1(c), we can observe $\mathbf{b}(k)$ for the given example $\mathbf{w}(k)$ and $\epsilon = 0.02$. Then, for some $i$, if $w_{s_i}(k)$ and $b_i(k)$ are zero, we infer that $w_i(k)$ belongs to the subset (I). However, if $w_{s_i}(k) = 0$ and $b_i(k) = 1$, then we conclude that $w_i(k)$ belongs to the subset (II). The result of combining the pieces of information given in Figures 1(b) and 1(c) is the $\hat{\mathbf{w}}_s(k)$ depicted in Figure 1(d).

Similarly to a stochastic gradient algorithm, the objective function of the LF-SG algorithm is

$$\xi_{\text{LF-SG}}(k) = \frac{1}{2}|e_s(k)|^2, \quad (5)$$

**TABLE 2. Low-complexity feature stochastic gradient algorithm.**

```
LF-SG Algorithm

Initialization
w_s(0) = b(0) = w(0) = [0 ··· 0]^T
choose μ in the range 0 < μ ≪ 1
choose small constant ε > 0
Do for k ≥ 0
  x_aux = 0, w_aux = 0, y_s(k) = 0
  for i = 0 to N
    if w_{s_i}(k) ≠ 0
      y_s(k) = y_s(k) + (w_aux × x_aux)
      w_aux = w_{s_i}(k)
      x_aux = x_i(k)
    else
      x_aux = x_aux + (x_i(k) × b_i(k))
    end
  end
  y_s(k) = y_s(k) + (w_aux × x_aux)
  e_s(k) = d(k) − y_s(k)
  w(k + 1) = w(k) + μ e_s(k) x(k)          (recursion)
  w_s(k + 1) = F_ε(w(k + 1))               (refer to Table 1)
  b(k + 1) = g_ε(w(k + 1))
end
```

where $e_s(k) \triangleq d(k) - y_s(k)$ is the error signal of the LF-SG algorithm. The LF-SG algorithm is summarized in Table 2. It is important to observe that the LF-SG algorithm requires fewer multiplication operations to calculate its output signal $y_s(k)$ in comparison with the LMS algorithm, which computes every multiplication related to the inner product $y(k) = \mathbf{w}^T(k)\mathbf{x}(k)$. In addition, notice that the LF-SG algorithm reduces to the LMS algorithm when $\epsilon = 0$.

## III. SOME PROPERTIES OF THE LF-SG ALGORITHM

In this section, we study the convergence behavior of the coefficient vector and the steady-state performance of the LF-SG algorithm. To this end, firstly, we introduce some useful variables. Then we use the update equation of the LF-SG algorithm to continue our analysis. We assume the data model described in Definition 1.

*Definition 1: The random processes* $\{d(k), \mathbf{x}(k), n(k)\}$ *satisfy the following conditions*:

(i) *Each of them is a wide-sense stationary (WSS) process with zero mean;*
(ii) *The covariance matrix of* $\mathbf{x}(k)$ *is* $\mathbf{R} = \mathbb{E}\big[\mathbf{x}(k)\mathbf{x}^T(k)\big]$*;*
(iii) $n(k)$ *is i.i.d. with variance* $\sigma_n^2 = \mathbb{E}[n^2(k)]$*;*
(iv) $n(k_1)$ *is independent of* $\mathbf{x}(k_2)$ *for all* $k_1, k_2$*;*
(v) *The initial condition* $\mathbf{w}(0)$ *is independent of* $\{d(k), \mathbf{x}(k), n(k)\}$ *for all* $k$*;*
(vi) *There exists a vector* $\mathbf{w}_* \in \mathbb{R}^{N+1}$*, called the optimum weight vector, such that* $d(k) = \mathbf{w}_*^T\mathbf{x}(k) + n(k)$.

### A. RELATING $\mathbf{w}_s(k)$ WITH $\hat{\mathbf{w}}_s(k)$

Table 2 presents an efficient implementation of the LF-SG algorithm that avoids computing several multiplications by using $\mathbf{w}_s(k)$. This strategy can be regarded as a sparse representation of the coefficient vector $\mathbf{w}(k)$. However, the *if* *statement* and the several auxiliary variables used in Table 2 hinder the mathematical analysis of the algorithm using $\mathbf{w}_s(k)$ directly. To overcome such issue, here we define $\hat{\mathbf{w}}_s(k)$ which can be regarded as an estimate of $\mathbf{w}(k)$ obtained via $\mathbf{w}_s(k)$ (by repeating the first entry of each block for the other entries of the corresponding block, as illustrated in Figure 1(d) for $\epsilon = 0.02$). In this way, the filter output can be rewritten as

$$y_s(k) \triangleq \hat{\mathbf{w}}_s^T(k)\mathbf{x}(k). \quad (6)$$

Besides, we can relate $\mathbf{w}(k)$ with $\hat{\mathbf{w}}_s(k)$ as follows

$$\mathbf{w}_\epsilon(k) \triangleq \mathbf{w}(k) - \hat{\mathbf{w}}_s(k), \quad (7)$$

where each entry of $\mathbf{w}_\epsilon(k)$ belongs to the interval $[-\epsilon, \epsilon]$.

### B. STEP-SIZE

Let $\widetilde{\mathbf{w}}_s(k) \triangleq \hat{\mathbf{w}}_s(k) - \mathbf{w}_*$ be the discrepancy between $\hat{\mathbf{w}}_s(k)$ and the optimum weight vector $\mathbf{w}_*$. Thus, the error signal can be written as

$$\begin{aligned} e_s(k) = d(k) - y_s(k) &= \mathbf{w}_*^T\mathbf{x}(k) + n(k) - \hat{\mathbf{w}}_s^T(k)\mathbf{x}(k) \\ &= n(k) - \widetilde{\mathbf{w}}_s^T(k)\mathbf{x}(k), \end{aligned} \quad (8)$$

in which the second equality follows from Definition 1 and (6).

Now, consider the recursion given in Table 2. After using (7), subtracting $\mathbf{w}_*$ from both of its sides, and using (8), this recursion can be rewritten as

$$\begin{aligned} \widetilde{\mathbf{w}}_s(k+1) = \left(\mathbf{I} - \mu\mathbf{x}(k)\mathbf{x}^T(k)\right)\widetilde{\mathbf{w}}_s(k) + \mu\mathbf{x}(k)n(k) \\ + \left(\mathbf{w}_\epsilon(k) - \mathbf{w}_\epsilon(k+1)\right). \end{aligned} \quad (9)$$

By taking the expected value of the previous equation we obtain

$$\mathbb{E}[\widetilde{\mathbf{w}}_s(k+1)] = (\mathbf{I}-\mu\mathbf{R})\mathbb{E}[\widetilde{\mathbf{w}}_s(k)] + \mathbb{E}[\mathbf{w}_\epsilon(k) - \mathbf{w}_\epsilon(k+1)], \quad (10)$$

where we used $\mathbb{E}[\mathbf{x}(k)n(k)] = 0$, which follows from Definition 1, and assumed $\mathbf{x}(k)$ to be independent of $\widetilde{\mathbf{w}}_s(k)$.[1] After resolving this recursion the following relation is achieved

$$\begin{aligned} &\mathbb{E}[\widetilde{\mathbf{w}}_s(k+1)] \\ &= (\mathbf{I} - \mu\mathbf{R})^{k+1}\mathbb{E}[\widetilde{\mathbf{w}}_s(0)] \\ &\quad + \sum_{i=0}^{k}(\mathbf{I} - \mu\mathbf{R})^i\mathbb{E}[\mathbf{w}_\epsilon(k-i) - \mathbf{w}_\epsilon(k-i+1)] \\ &= \underbrace{\mathbf{Q}(\mathbf{I} - \mu\mathbf{\Lambda})^{k+1}\mathbf{Q}^T\mathbb{E}[\widetilde{\mathbf{w}}_s(0)]}_{1st\ term} \\ &\quad \underbrace{+\sum_{i=0}^{k}\mathbf{Q}(\mathbf{I}-\mu\mathbf{\Lambda})^i\mathbf{Q}^T\mathbb{E}[\mathbf{w}_\epsilon(k-i)-\mathbf{w}_\epsilon(k-i+1)]}_{2nd\ term}, \end{aligned} \quad (11)$$

where $\mathbf{R} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T$ (spectral decomposition), $\mathbf{Q}$ is a matrix containing the orthonormal eigenvectors of $\mathbf{R}$, whereas $\mathbf{\Lambda}$ is a diagonal matrix containing the associated eigenvalues on the main diagonal. Clearly, by setting the step-size as $0 < \mu < 2/\lambda_{max}$, with $\lambda_{max}$ being the largest eigenvalue, the 1st term converges to zero. However, for $\epsilon > 0$ there is no choice of $\mu$ that ensures the convergence of the 2nd term to zero, meaning that the LF-SG algorithm provides a biased estimate of $\mathbf{w}_*$ in general. Nevertheless, such bias can be negligible since $0 < \mu < 2/\lambda_{max}$ implies that $(\mathbf{I} - \mu\mathbf{\Lambda})^i \overset{i\to\infty}{\longrightarrow} 0$, meaning that the expectation terms are attenuated more as $i$ increases. Besides, the LF-SG algorithm should employ small $\epsilon$ so that each expectation within the 2nd term is close to zero. In this view, the LF-SG algorithm trades off bias for complexity reduction by means of $\epsilon$.

[1] This is the so-called *independence theory* which provides quite accurate analysis results [26].

### C. MSE

Let us define the MSE of the LF-SG algorithm as follows

$$\zeta_{\text{LF-SG}} \triangleq \lim_{k\to\infty}\mathbb{E}[e_s^2(k)]. \quad (12)$$

In Table 2, we can see that the update equation of the LF-SG algorithm is similar to that of the LMS algorithm. Indeed, by Equation (7), $\hat{\mathbf{w}}_s(k)$ is obtained by translating the weight vector of the LMS algorithm by $\mathbf{w}_\epsilon(k)$. Therefore, we can follow the same mathematical steps in computing the MSE of the LMS algorithm as in [26] to derive the MSE of the LF-SG algorithm.

Also, note that the relationship between the error signals of the LMS and the LF-SG algorithms is given by

$$\begin{aligned} e_s(k) = d(k) - y_s(k) &= d(k) - \hat{\mathbf{w}}_s^T(k)\mathbf{x}(k) \\ &= d(k) - (\mathbf{w}(k) - \mathbf{w}_\epsilon(k))^T\mathbf{x}(k) \\ &= e(k) + \mathbf{w}_\epsilon^T(k)\mathbf{x}(k), \end{aligned} \quad (13)$$

where $e(k)$ is the error signal of the LMS algorithm. Therefore, after some mathematical manipulations as in [26] pages 591-598, we obtain the MSE of the LF-SG algorithm as

$$\zeta_{\text{LF-SG}} = \frac{\mu(\sigma_n^2 + \epsilon^2\sigma_x^2)tr(\mathbf{R})}{2 - \mu tr(\mathbf{R})} + \epsilon^2\sigma_x^2 + \sigma_n^2, \quad (14)$$

where $\sigma_x^2$ and $\sigma_n^2$ are the variances of the input signal and the additive noise signal, respectively, and $tr(\mathbf{R})$ stands for the trace of matrix $\mathbf{R}$.

### D. SELECTION OF $\epsilon$

In this subsection, we propose some approaches to select an appropriate $\epsilon$, since it is a crucial parameter for the LF-SG algorithm.

The best value of $\epsilon$ can be adopted by using some *a priori* knowledge about the system to be identified; however, when such *a priori* knowledge is not available, we may utilize one of the following strategies.

(i) We can run the LMS algorithm until it converges. After the convergence of the LMS algorithm, by clustering the coefficients in lowpass blocks, the suitable value for $\epsilon$ can be chosen.

(ii) Note that we infer the LF-SG algorithm reaches the steady-state if its steady-state mean squared deviation (MSD) equals to the MSD of the LMS algorithm, and the MSD of the LMS algorithm is given by [27]

$$\mathrm{MSD_{LMS}} = \frac{1}{2}\mu N\sigma_n^2. \tag{15}$$

Hence, we can adopt an appropriate $\epsilon$ as follows

$$0 \leq \epsilon < \sqrt{\frac{\mathrm{MSD_{LMS}}}{N}} = \sqrt{\frac{1}{2}\mu\sigma_n^2}, \tag{16}$$

where $\sqrt{\frac{1}{2}\mu\sigma_n^2}$ expresses the average value of the steady-state estimate deviation between each coefficient of the adaptive filter and the optimal one.

## IV. SIMULATIONS

In this section, we evaluate the performance of the LMS and the LF-SG algorithms in some system identification scenarios. In all examples, the input signal is zero-mean white Gaussian with unit variance, the initial coefficient vector is adopted as $\mathbf{w}(0) = [0 \ \cdots \ 0]^T$, and the signal-to-noise ratio (SNR) is 20 dB, i.e., $\sigma_n^2 = 0.01$. The MSE learning curves are computed by averaging the outcomes of 200 independent runs.

### A. SCENARIO 1: SYNTHETIC IMPULSE RESPONSES

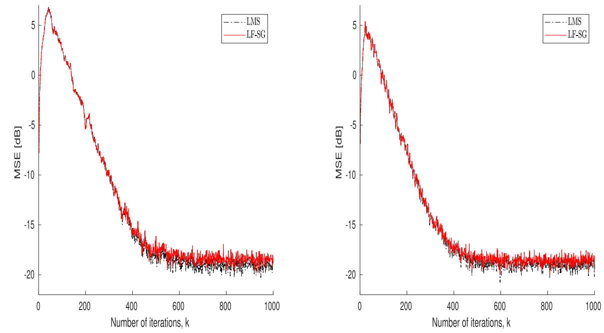
In this scenario, we apply the LMS and the LF-SG algorithms to identify two unknown systems of order 39. The first unknown system is the narrowband lowpass system $\mathbf{w}_*^l = [0.4 \ 0.4 \ \cdots \ 0.4]^T$. The second unknown model is a block-lowpass model, $\mathbf{w}_*^b$, whose coefficients are listed in Table 3. For both algorithms, the step-size is $\mu = 0.01$. Moreover, using the *a priori* information, we use $\epsilon = 0.02$.

**TABLE 3.** The coefficients of $\mathbf{w}_*^b$.

| $i$ | $w^s_{*_i}$ | $i$ | $w^s_{*_i}$ | $i$ | $w^s_{*_i}$ | $i$ | $w^s_{*_i}$ | $i$ | $w^s_{*_i}$ |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0.4 | 8 | 0.401 | 16 | -0.344 | 24 | 0.125 | 32 | 0 |
| 1 | 0.402 | 9 | 0.404 | 17 | -0.351 | 25 | 0.122 | 33 | 0 |
| 2 | 0.404 | 10 | -0.35 | 18 | 0.5 | 26 | 0.117 | 34 | 0 |
| 3 | 0.401 | 11 | -0.348 | 19 | 0.505 | 27 | 0.121 | 35 | 0 |
| 4 | 0.399 | 12 | -0.354 | 20 | 0.496 | 28 | 0.119 | 36 | 0 |
| 5 | 0.398 | 13 | -0.353 | 21 | 0.502 | 29 | 0.123 | 37 | 0 |
| 6 | 0.397 | 14 | -0.347 | 22 | 0.12 | 30 | 0 | 38 | 0 |
| 7 | 0.4 | 15 | -0.359 | 23 | 0.119 | 31 | 0 | 39 | 0 |

Figure 2(a) illustrates the MSE learning curves of the LMS and the LF-SG algorithms considering the identification of the narrowband lowpass system $\mathbf{w}_*^l$. As can be verified, both algorithms exhibit similar learning curves, but the LMS algorithm requires much more multiplication operations. For instance, during the steady-state, the LMS algorithm performed 40 multiplications per iteration to compute its output signal $y(k)$, whereas the LF-SG algorithm required just one multiplication to compute its corresponding output $y_s(k)$.

The MSE learning curves of the LMS and the LF-SG algorithms, when they are employed to identify the block-lowpass system $\mathbf{w}_*^b$, are presented in Figure 2(b). Once again, the two curves are very similar, indicating that the LF-SG algorithm performed as good as the LMS algorithm, but requiring much fewer multiplications. Indeed, in the computation of their corresponding output signals during the steady-state,




(a) (b)

**FIGURE 2.** MSE learning curves of the LMS and the LF-SG algorithms considering the unknown systems in Scenario 1: (a) the narrowband lowpass system $\mathbf{w}_*^l$; (b) the block-lowpass system $\mathbf{w}_*^b$.

the LMS algorithm required 40 multiplication operations per iteration, whereas the LF-SG algorithm realized only four multiplications.

Therefore, in this scenario, we verified the competitive MSE performance of the LF-SG algorithm, in comparison with the LMS algorithm, but the proposed algorithm requires fewer multiplications to identify block-lowpass systems, as illustrated in Table 4.

**TABLE 4.** Number of required multiplication operations to compute the output signals in the identification of systems $\mathbf{w}_*^l$, $\mathbf{w}_*^b$, and RIR.

| Systems | LMS | LF-SG | Percentage of reduction |
|---|---|---|---|
| $\mathbf{w}_*^l$ | 40 | 1 | 97.5% |
| $\mathbf{w}_*^b$ | 40 | 4 | 90% |
| RIR | 16000 | 7800 | 51.25% |

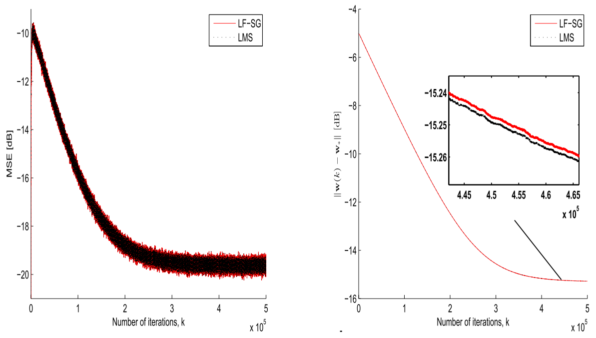
### B. SCENARIO 2: MEASURED ROOM IMPULSE RESPONSE

In this scenario, we apply the LMS and the LF-SG algorithms to identify a measured unknown system corresponding to the room impulse response (RIR) provided in [28]. Such RIR, sampled initially at 96 kHz, is downsampled to 8 kHz, thus reducing the number of coefficients to 16, 000. The step-size is $\mu = 10^{-5}$ for both algorithms, whereas $\epsilon = 5 \times 10^{-5} < \sqrt{\frac{1}{2}\mu\sigma_n^2}$ for the LF-SG algorithm.

Figures 3(a) and 3(b) depict the MSE learning curves and the values of $\mathbb{E}[\|\mathbf{w}(k) - \mathbf{w}_*\|]$, respectively, for both algorithms, considering that $\mathbf{w}_*$ is the RIR to be identified. Moreover, these curves have been smoothed using a box filter of 100 samples length. In both figures, the curves corresponding to the LMS and the LF-SG algorithms are overlaid, thus illustrating that the LF-SG algorithm is capable of achieving similar accuracy but requiring fewer multiplications than the LMS algorithm. During the steady-state, for example, the LMS algorithm realized 16, 000 multiplications to compute its output $y(k)$, whereas the LF-SG algorithm required about 7, 800 multiplications to compute $y_s(k)$.

### C. SCENARIO 3: VERIFYING THEORETICAL MSE

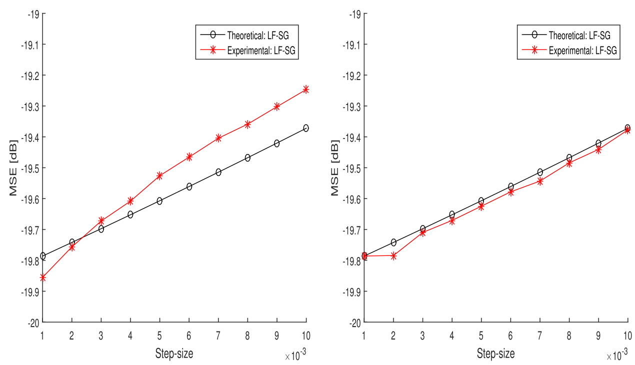
In this scenario, we test the accuracy of the theoretical steady-state MSE expression for the LF-SG algorithm given




(a) (b)

**FIGURE 3.** Results for the LMS and the LF-SG algorithms applied to the identification of the measured RIR given in scenario 2: (a) MSE learning curves; (b) $\mathbb{E}[\|\mathbf{w}(k) - \mathbf{w}_*\|]$.

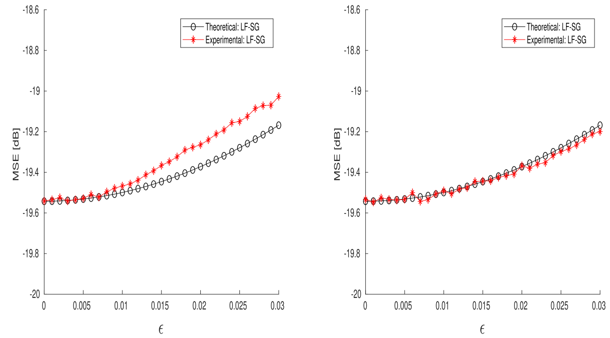
in (14) versus the variations of the step-size $\mu$ and the parameter $\epsilon$. To compare the steady-state MSE performance versus the variations of $\mu$, we apply the LF-SG algorithm, using the fixed $\epsilon = 0.02$ and step-size values ranging from 0.001 to 0.01, to identify two unknown systems of order 19. Also, to verify the steady-state MSE performance versus the variations of $\epsilon$, the LF-SG algorithm has been utilized with the fixed step-size $\mu = 0.01$, while $\epsilon$ ranging from 0 to 0.03. The first unknown system is a narrowband lowpass system given by $[w^l_{*0} \ \cdots \ w^l_{*19}]^T$. The second unknown system is a block-lowpass system given by $[0 \ \cdots \ 0 \ w^b_{*10} \ \cdots \ w^b_{*13} \ w^b_{*18} \ \cdots \ w^b_{*25} \ 0 \ \cdots \ 0]^T$. The experimental MSE results are obtained by implementing the LF-SG algorithm for $2 \times 10^4$ iterations and averaging the squared error $e_s^2(k)$ over 100 independent runs in order to produce the ensemble-average curve. Then we compute the time average over the last $5,000$ iterations (all of these iterations corresponding to the steady-state) and use it as the experimental MSE.




(a) (b)

**FIGURE 4.** MSE versus step-size for the LF-SG algorithm considering scenario 3: (a) the narrowband lowpass system of order 19; (b) the block-lowpass system of order 19.

Figures 4(a) and 4(b) depict the theoretical and the experimental steady-state MSE values achieved by the LF-SG algorithm, for different values of the step-size $\mu$, considering the identification of the narrowband lowpass system and the block-lowpass system, respectively.




(a) (b)

**FIGURE 5.** MSE versus $\epsilon$ for the LF-SG algorithm considering Scenario 3: (a) the narrowband lowpass system of order 19; (b) the block-lowpass system of order 19.

Also, Figures 5(a) and 5(b) illustrate the theoretical and the experimental steady-state MSE values of the LF-SG algorithm, for different values of $\epsilon$, considering the identification of the narrowband lowpass and the block-lowpass systems, respectively. In these figures, we can observe that the theoretical curves match quite well the experimental curves, especially in Figures 4(b) and 5(b). Observe that for the narrowband lowpass system in Figures 4(a) and 5(a), all coefficients are estimated using a single reference coefficient, whereas, for the block-lowpass system, there are four reference coefficients (or equivalently, four nonzero blocks). This means that for the case of the block-lowpass system we have less "quantization effect", thus justifying the better agreement between the theoretical and the experimental curves.

## V. CONCLUSION

This work proposed a strategy to benefit computationally from the hidden sparsity inherent to many practical systems by means of the feature function. In particular, we proposed the LF-SG algorithm for block-lowpass systems, but the algorithm can easily be adapted to exploit other kinds of feature, such as highpass and multi-bandpass systems. In comparison with the LMS algorithm, the LF-SG algorithm can significantly decrease the number of multiplication operations required to compute the adaptive filter output without impairing its accuracy. Furthermore, we analyzed some properties of the proposed algorithm, like its convergence in the mean and steady-state MSE. Finally, the potential benefits of the LF-SG algorithm in reducing computational requirements were demonstrated considering the identification of both synthetic and measured unknown systems.

## REFERENCES


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

[2] Z. F. Li, D. Li, and J. Q. Zhang, "A new penalized recursive least squares method with a variable regularization factor for adaptive sparse filtering," *IEEE Access*, vol. 6, pp. 31828–31839, 2018.

[3] Y. Gu, J. Jin, and S. Mei, "$\ell_0$ norm constraint LMS algorithm for sparse system identification," *IEEE Signal Process. Lett.*, vol. 16, no. 9, pp. 774–777, Sep. 2009.

[4] M. V. S. Lima, W. A. Martins, and P. S. R. Diniz, "Affine projection algorithms for sparse system identification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Vancouver, BC, Canada, May 2013, pp. 5666–5670.

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

[6] 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. IEEE Int. Conf. Acoust., Speech Signal Process.*, Florence, Italy, May 2014, pp. 6399–6403.

[7] C. Ye, G. Gui, L. Xu, and T. Ohtsuki, "Recovery of block-structured sparse signal using block-sparse adaptive algorithms via dynamic grouping," *IEEE Access*, vol. 6, pp. 56069–56083, 2018.

[8] Y. Li, Z. Jiang, Z. Jin, X. Han, and J. Yin, "Cluster-sparse proportionate NLMS algorithm with the hybrid norm constraint," *IEEE Access*, vol. 6, pp. 47794–47803, 2018.

[9] M. R. Petraglia and D. B. Haddad, "New adaptive algorithms for identification of sparse impulse responses—Analysis and comparisons," in *Proc. 7th Int. Symp. Wireless Commun. Syst.*, York, U.K., Sep. 2010, pp. 384–388.

[10] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, Nov. 2009.

[11] T. G. Miller, S. Xu, R. C. de Lamare, and H. V. Poor, "Distributed spectrum estimation based on alternating mixed discrete-continuous adaptation," *IEEE Signal Process. Lett.*, vol. 23, no. 4, pp. 551–555, Apr. 2016.

[12] R. C. de Lamare and R. Sampaio-Neto, "Sparsity-aware adaptive algorithms based on alternating optimization and shrinkage," *IEEE Signal Process. Lett.*, vol. 21, no. 2, pp. 225–229, Feb. 2014.

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

[14] Y. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge, U.K.: Cambridge Univ. Press, 2012.

[15] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY, USA: Springer, 2016.

[16] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006.

[17] P. S. R. Diniz, H. Yazdanpanah, and M. V. S. Lima, "Feature LMS algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Calgary, AB, Canada, Apr. 2018, pp. 4144–4148.

[18] H. Yazdanpanah, J. A. Apolinário, P. S. R. Diniz, and M. V. S. Lima, "$\ell_0$-norm feature LMS algorithms," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Anaheim, CA, USA, Nov. 2018, pp. 311–315.

[19] J. F. Kaiser and W. A. Reed, "Data smoothing using low-pass digital filters," *Rev. Sci. Instrum.*, vol. 48, no. 11, pp. 1447–1457, 1977.

[20] V. Szortyka, K. Raczkowski, M. Kuijk, and P. Wambacq, "A wideband beamforming lowpass filter for 60 GHz phased-array receivers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, no. 9, pp. 2324–2333, Sep. 2015.

[21] Z. Dai, S. He, J. Pang, C. Huang, J. Peng, Z. Yang, and M. Li, "Low-pass Network Synthesis Using 'Feldtkeller Correction Approach,'" *IEEE Access*, vol. 7, pp. 27970–27982, 2019.

[22] W. Yuan, X. Liu, H. Lu, W. Wu, and N. Yuan, "Flexible design method for microstrip bandstop, highpass, and bandpass filters using similar defected ground structures," *IEEE Access*, vol. 7, pp. 98453–98461, 2019.

[23] M. Á. G. de Anda and I. M. Dector, "A second-order lowpass parameter-varying filter based on the interconnection of first-order stages," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 8, pp. 1840–1853, Aug. 2011.

[24] H. Yazdanpanah, P. S. R. Diniz, and M. V. S. Lima, "A simple set-membership affine projection algorithm for sparse system modeling," in *Proc. 24th Eur. Signal Process. Conf.*, Budapest, Hungary, Aug. 2016, pp. 1798–1802.

[25] H. Yazdanpanah and P. S. R. Diniz, "Recursive least-squares algorithms for sparse system modeling," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, New Orleans, LA, USA, Mar. 2017, pp. 3879–3883.

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

[27] S. Haykin, *Adaptive Filter Theory*, 4th ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

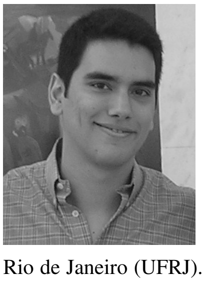
[28] R. Stewart and M. Sandler, "Database of omnidirectional and B-format room impulse responses," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Dallas, TX, USA, Mar. 2010, pp. 165–168.


**HAMED YAZDANPANAH** (S'16–M'19) was born in Mashhad, Iran, in 1986. He received the Applied Mathematics degree from Khayyam University, in 2009, the M.Sc. degree in pure mathematics focused on probability from the Instituto de Matemática Pura e Aplicada (IMPA), in 2014, and the D.Sc. degree in electrical engineering from the Federal University of Rio de Janeiro (UFRJ), in 2018. From 2018 to 2019, he was a Postdoctoral Researcher with the Military Institute of Engineering (IME), Rio de Janeiro, Brazil. He is currently a Postdoctoral Researcher with the Department of Computer Science, University of São Paulo, São Paulo, Brazil. His research interests include machine learning, adaptive filtering, and deep learning.

**PAULO S. R. DINIZ** (M'81–SM'92–F'00) was born in Niterói, Brazil. He received the Electronics Engineering degree *(cum laude)* from the Federal University of Rio de Janeiro (UFRJ), in 1978, the M.Sc. degree from COPPE/UFRJ, in 1981, and the Ph.D. degree in electrical engineering from Concordia University, Montreal, QC, Canada, in 1984.

He has published refereed articles in journals and conference papers in some of his research areas. He wrote textbooks *Adaptive Filtering: Algorithms and Practical Implementation*, (New York, NY, USA: Fourth Edition, Springer, 2013), *Digital Signal Processing: System Analysis and Design*, (Second Edition, Cambridge University Press, Cambridge, U.K., 2010) (with E. A. B. D. Silva and S. L. Netto), and the monograph *Block Transceivers: OFDM and Beyond*, (New York, NY, USA: Morgan & Claypool, 2012) (W. A. Martins, and M. V. S. Lima). His teaching and research interests include analog and digital signal processing, adaptive signal processing, digital communications, wireless communications, multirate systems, stochastic processes, and electronic circuits.

Dr. Diniz is a member of the National Academy of Engineering (ANE) and the Brazilian Academy of Science (ABC). He is a Fellow of the EURASIP 2016. He was a Distinguished Lecturer of the IEEE Circuits and Systems Society, from 2000 to 2001. In 2004, he has served as a Distinguished Lecturer of the IEEE Signal Processing Society. He has received the 2014 Charles Desoer Technical Achievement Award from the IEEE Circuits and Systems Society. He also holds some best paper awards from conferences and the IEEE journals. He has served as an Associate Editor for the following journals, including the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING, from 1996 to 1999, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, from 1999 to 2002, and the *Circuits, Systems, and Signal Processing* journal, from 1998 to 2002.



**MARKUS V. S. LIMA** (S'08–M'14) was born in Rio de Janeiro, Brazil, in 1984. He received the degree in electronics engineering from the Federal University of Rio de Janeiro (UFRJ), in 2008, and the M.Sc. and D.Sc. degrees in electrical engineering from COPPE/UFRJ, in 2009 and 2013, respectively. He is currently an Associate Professor of the program of electrical engineering (PEE/COPPE) and the Department of Electrical Engineering (DEE/Poli), Universidade Federal do Rio de Janeiro (UFRJ). He also collaborates with the Institute of Electrical and Electronics Engineers (IEEE) in a volunteer basis, having integrated the Technical Program Committee of the International Symposium on Industrial Electronics (ISIE 2015), the board of the IEEE Section in Rio de Janeiro as a Secretary, from 2016 to 2017. He has been the Chair of the IEEE Signal Processing Chapter in Rio de Janeiro, since 2018. His research interests include adaptive filtering and machine learning, digital signal processing, sound source localization and enhancement through microphone arrays, and wireless communications systems, in particular the physical layer techniques employed in 4G and 5G systems. He is a member of the Brazilian Telecommunications Society (SBrT).