FEATURE LMS ALGORITHMS

Paulo S. R. Diniz, Fellow, IEEE, Hamed Yazdanpanah, and Markus V. S. Lima

Universidade Federal do Rio de Janeiro
DEE-DEL/Poli & PEE/COPPE/UFRJ
P.O. Box 68504, Rio de Janeiro, RJ, 21941-972, Brazil

ABSTRACT

In recent years, there is a growing effort in the learning algorithms area to propose new strategies to detect and exploit sparsity in the model parameters. In many situations, the sparsity is hidden in the relations among these coefficients so that some suitable tools are required to reveal the potential sparsity. This work proposes a set of LMS-type algorithms, collectively called Feature LMS (F-LMS) algorithms, setting forth a hidden feature of the unknown parameters, which ultimately would improve convergence speed and steady-state mean-squared error. The key idea is to apply linear transformations, by means of the so-called feature matrices, to reveal the sparsity hidden in the coefficient vector, followed by a sparsity-promoting penalty function to exploit such sparsity. Some F-LMS algorithms for lowpass and highpass systems are also introduced by using simple feature matrices that require only trivial operations. Simulation results demonstrate that the proposed F-LMS algorithms bring about several performance improvements whenever the hidden sparsity of the parameters is exposed.

Index Terms— adaptive filtering, LMS algorithm, feature matrix, lowpass system, highpass system

1. INTRODUCTION

The adaptive filtering algorithms are employed in several applications for at least five decades. In particular, the popular LMS algorithm, first introduced in 1960 [1, 2], has been widely considered as the benchmark in the field. Elaborate studies of the LMS algorithm were presented in [3, 4]. Also, the LMS and its variants can be found to solve real problems including active noise control [5], digital equalizers [6], continuous-time filter tuning [7], system identification [8], among others.

In recent years, a number of adaptive filtering algorithms exploiting the sparsity in the model coefficients by imposing some constraints in the cost function were proposed [9–14]. This strategy allows the attraction of some coefficient values to zero enabling the detection of the nonrelevant parameters of the model. To the best of our knowledge, there is no work exploiting the sparsity arising from linear combinations of the unknown parameters in the adaptive filtering context.

In this paper, we introduce the feature LMS (F-LMS) family of algorithms inducing simple sparsity properties hidden in the parameters. The type of feature to seek determines the structure of the feature matrix $\mathbf{F}(k)$ to be applied in the constraints of the F-LMS algorithm. In fact, a plethora of featured algorithms is possible to be defined by applying smart combinations of feature matrices to the coefficient vector. In this work, some simple cases are discussed whereas many more advanced solutions will be exploited in future publications.

This work is organized as follows. Section 2 proposes the F-LMS family of algorithms. Some examples of F-LMS algorithms for systems with lowpass and highpass spectrum are introduced in Section 3. Simulation results are presented in Section 4 and the conclusions are drawn in Section 5.

Notation: Scalars are represented by lower-case letters. Vectors (matrices) are denoted by lowercase (uppercase) boldface letters. For a given iteration k, the weight vector and the input vector are denoted by $\mathbf{w}(k), \mathbf{x}(k) \in \mathbb{R}^{N+1}$, respectively, where N is the adaptive filter order. The error signal at the k-th iteration is defined as $e(k) \triangleq d(k) - \mathbf{w}^T(k)\mathbf{x}(k)$, where $d(k) \in \mathbb{R}$ is the desired signal. The l_1-norm of a vector $\mathbf{w} \in \mathbb{R}^{N+1}$ is given by $\|\mathbf{w}\|_1 = \sum_{i=0}^{N} |w_i|$.

2. THE FEATURE LMS ALGORITHMS

Feature LMS (F-LMS) refers to a family of LMS-type algorithms capable of exploiting the features inherent to the unknown systems to be identified. These algorithms minimize the following general objective function:

$$\xi_{\text{F-LMS}}(k) = \underbrace{\frac{1}{2}|e(k)|^2}_{\text{standard LMS term}} + \underbrace{\alpha\mathcal{P}\left(\mathbf{F}(k)\mathbf{w}(k)\right)}_{\text{feature-inducing term}}, \qquad (1)$$

where $\alpha \in \mathbb{R}_+$ stands for the weight given to the *sparsity-promoting penalty function* $\mathcal{P}$, which maps a vector to the nonnegative reals $\mathbb{R}_+$, and $\mathbf{F}(k)$ is the so-called *feature matrix* responsible for revealing the hidden sparsity, i.e., the result of applying $\mathbf{F}(k)$ to $\mathbf{w}(k)$ should be a sparse vector (in the sense that most entries of the vector $\mathbf{F}(k)\mathbf{w}(k)$ should be close or equal to zero).

The penalty function $\mathcal{P}$ can be any sparsity-promoting penalty function that is almost everywhere differentiable in order to allow for gradient-based methods. Examples of suitable functions are: (i) vector norms, especially the widely used l_1 norm [10,13]; (ii) vector norms combined with shrinking strategies [9]; (iii) a function that approximates the l_0 norm [11,15].

The feature matrix $\mathbf{F}(k)$ can vary at each iteration and it represents any linear combination that applied to $\mathbf{w}(k)$ results in a sparse vector. In practice, $\mathbf{F}(k)$ should be chosen based on some previous knowledge about the unknown system $\mathbf{w}_o$. For instance, $\mathbf{w}_o$ can represent a lowpass or highpass filter, it can have linear phase, it can be an upsampled or downsampled signal, etc. All these features can be exploited by the F-LMS algorithm in order to accelerate convergence and/or achieve lower mean-squared error (MSE).

The resulting gradient-based algorithms using the objective function given in (1) are known as F-LMS algorithms, and their recursions have the following general form:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu e(k)\mathbf{x}(k) - \mu\alpha\mathbf{p}(k), \tag{2}$$

where $\mu \in \mathbb{R}_+$ is the step size, which should be small enough to ensure convergence [4], and $\mathbf{p}(k) \in \mathbb{R}^{N+1}$ is the gradient of function $\mathcal{P}\left(\mathbf{F}(k)\mathbf{w}(k)\right)$.

3. EXAMPLES OF F-LMS ALGORITHMS

From Section 2, it is clear that the F-LMS family contains infinitely many algorithms. So, in this section we introduce some of these algorithms in order to illustrate how some specific features of the unknown system can be exploited. For the sake of clarity, we focus on simple algorithms and, therefore, we choose function $\mathcal{P}$ to be the l_1 norm and the feature matrix to be time-invariant $\mathbf{F}$ so that the cost function in (1) simplifies to

$$\xi_{\text{F-LMS}}(k) = \frac{1}{2}|e(k)|^2 + \alpha\|\mathbf{F}\mathbf{w}(k)\|_1. \tag{3}$$

As a consequence, the reader will notice that the computational complexity of the algorithms proposed in this section is only slightly superior to the complexity of the LMS algorithm, as the computation of $\mathbf{p}(k)$ required in (2) is very simple (does not involve multiplications or divisions).

3.1. The F-LMS algorithm for lowpass systems

Most systems found in practice have their energy concentrated mainly in the low frequencies. If the unknown system has lowpass narrowband spectrum, then its impulse response $\mathbf{w}_o$ is smooth, meaning that the difference between adjacent coefficients is small (probably close to zero).

The adaptive filtering algorithm can take advantage of this feature present in the unknown system by selecting the feature matrix properly. Indeed, by selecting $\mathbf{F}$ as $\mathbf{F}_l$, where $\mathbf{F}_l$ is a $N \times N+1$ matrix defined as

$$\mathbf{F}_l = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \\ 0 & 0 & \cdots & 1 & -1 \end{bmatrix} \tag{4}$$

and $\|\mathbf{F}_l\mathbf{w}(k)\|_1 = \sum_{i=0}^{N-1}|w_i(k) - w_{i+1}(k)|$, the optimization problem in (3) can be interpreted as: we seek for $\mathbf{w}(k)$ that minimizes both the squared error (LMS term) and the distances between adjacent coefficients of $\mathbf{w}(k)$. In other words, the F-LMS algorithm for lowpass systems acts like the LMS algorithm, but enforcing $\mathbf{w}(k)$ to be a lowpass system. It is worth mentioning that if $\mathbf{w}_o$ is indeed a lowpass system, then matrix $\mathbf{F}_l$ yields a sparse vector $\mathbf{F}_l\mathbf{w}(k)$.[1]

Thus, the F-LMS algorithm for lowpass systems is defined by the recursion given in (2), but replacing vector $\mathbf{p}(k)$ with $\mathbf{p}_l(k)$ defined as

$$\begin{cases} p_{l,i}(k) = \text{sgn}(w_0(k) - w_1(k)), & \text{if } i = 0 \\ p_{l,i}(k) = -\text{sgn}(w_{i-1}(k) - w_i(k)) & \\ \qquad\quad +\text{sgn}(w_i(k) - w_{i+1}(k)), & \text{if } i = 1, \cdots, N-1 \\ p_{l,i}(k) = -\text{sgn}(w_{N-1}(k) - w_N(k)), & \text{if } i = N \end{cases} \tag{5}$$

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

As previously explained, the F-LMS algorithm above tries to reduce the distances between consecutive coefficients of $\mathbf{w}(k)$, i.e., matrix $\mathbf{F}_l$ can be understood as the process of windowing $\mathbf{w}(k)$ with a window of length 2 (i.e., two coefficients are considered at a time). We can increase the window length, in order to make a smoothing considering more coefficients simultaneously, by nesting linear combinations as follows:

$$\mathbf{F}_l^{M-\text{nested}} = \prod_{m=1}^{M} \mathbf{F}_l^{(m)}\mathbf{F}_l, \tag{6}$$

where $\mathbf{F}_l^{(m)}$ has the same structure given in (4), but losing m rows and m columns in relation to the dimensions of $\mathbf{F}_l$.

In addition to the previous examples, suppose that the unknown system is the result of upsampling a lowpass system by a factor of L. In this case, we should use matrix $\mathbf{F}_l^*$, whose rows have $L-1$ zeros between the ± 1 entries, in (3). For $L = 2$, we have the following matrix

$$\mathbf{F}_l^* = \begin{bmatrix} 1 & 0 & -1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & -1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & 0 & \cdots & 1 & 0 & -1 \end{bmatrix}, \tag{7}$$

and $\|\mathbf{F}_l^*\mathbf{w}(k)\|_1 = \sum_{i=0}^{N-2}|w_i(k) - w_{i+2}(k)|$.

[1] A matrix similar to the $\mathbf{F}_l$ in (4) is already known by the statisticians working on a field called *trend filtering* [16].

Next the F-LMS algorithm using such $\mathbf{F}_l^*$ has the update rule given in (2), but replacing $\mathbf{p}(k)$ with $\mathbf{p}_l^*(k)$ defined as

$$\begin{cases} p_{l,i}^*(k) = \text{sgn}(w_i(k) - w_{i+2}(k)), & \text{if } i = 0, 1 \\ p_{l,i}^*(k) = -\text{sgn}(w_{i-2}(k) - w_i(k)) & \\ \qquad\qquad +\text{sgn}(w_i(k) - w_{i+2}(k)), & \text{if } i = 2, \cdots, N-2 \\ p_{l,i}^*(k) = -\text{sgn}(w_{i-2}(k) - w_i(k)), & \text{if } i = N-1, N \end{cases} \tag{8}$$

3.2. The F-LMS algorithm for highpass systems

If the unknown system $\mathbf{w}_o$ has a highpass narrowband spectrum, then adjacent coefficients tend to have similar absolute values, but with opposite signs. Therefore, the sum of two consecutive coefficients is close to zero and we can exploit this feature in the learning process by minimizing the sum of adjacent coefficients of $\mathbf{w}(k)$. This can be accomplished by selecting $\mathbf{F}$ as $\mathbf{F}_h$, where $\mathbf{F}_h$ is a $N \times N+1$ feature matrix defined as

$$\mathbf{F}_h = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}, \tag{9}$$

such that $\|\mathbf{F}_h\mathbf{w}(k)\|_1 = \sum_{i=0}^{N-1} |w_i(k) + w_{i+1}(k)|$.

The F-LMS algorithm for highpass systems is characterized by the recursion given in (2), but replacing $\mathbf{p}(k)$ with $\mathbf{p}_h(k)$, which is defined as

$$\begin{cases} p_{h,i}(k) = \text{sgn}(w_0(k) + w_1(k)), & \text{if } i = 0 \\ p_{h,i}(k) = \text{sgn}(w_{i-1}(k) + w_i(k)) & \\ \qquad\qquad +\text{sgn}(w_i(k) + w_{i+1}(k)), & \text{if } i = 1, \cdots, N-1 \\ p_{h,i}(k) = \text{sgn}(w_{N-1}(k) + w_N(k)), & \text{if } i = N \end{cases} \tag{10}$$

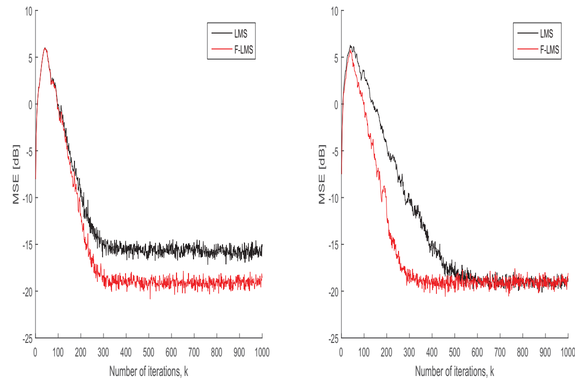
Similar to the lowpass case, let us consider that the unknown system is the result of interpolating a highpass system by a factor $L = 2$. The set of interpolated highpass systems leads to a notch filter with zeros at $z = \pm \jmath$. In this case, we can utilize $\mathbf{F}_h^*$ in the objective function (3), where $\mathbf{F}_h^*$ is described by

$$\mathbf{F}_h^* = \begin{bmatrix} 1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & 0 & \cdots & 1 & 0 & 1 \end{bmatrix}, \tag{11}$$

and $\|\mathbf{F}_h^*\mathbf{w}(k)\|_1 = \sum_{i=0}^{N-2} |w_i(k) + w_{i+2}(k)|$.

Using $\mathbf{F}_h^*$, the F-LMS recursion in (2) should substitute $\mathbf{p}(k)$ by $\mathbf{p}_h^*(k)$ defined as

$$\begin{cases} p_{h,i}^*(k) = \text{sgn}(w_i(k) + w_{i+2}(k)), & \text{if } i = 0, 1 \\ p_{h,i}^*(k) = \text{sgn}(w_{i-2}(k) + w_i(k)) & \\ \qquad\qquad +\text{sgn}(w_i(k) + w_{i+2}(k)), & \text{if } i = 2, \cdots, N-2 \\ p_{h,i}^*(k) = \text{sgn}(w_{i-2}(k) + w_i(k)), & \text{if } i = N-1, N \end{cases} \tag{12}$$



(a) (b)

Fig. 1. MSE learning curves of the LMS and F-LMS algorithms considering $\mathbf{w}_{o,l}$: (a) both algorithms with the same step size: $\mu = 0.03$; (b) LMS and F-LMS with step sizes equal to 0.01 and 0.03, respectively.

4. SIMULATIONS

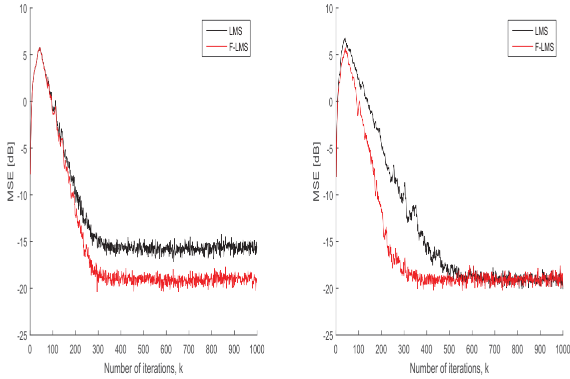
In this section, we apply the LMS and the F-LMS algorithms to identify some unknown lowpass and highpass systems. The order of all the unknown systems is 39, i.e., they have 40 coefficients. The first example considers extremely lowpass and highpass systems defined as $\mathbf{w}_{o,l} = [0.4, \cdots, 0.4]^T$ and $\mathbf{w}_{o,h} = [0.4, -0.4, 0.4, \cdots, -0.4]^T$, respectively. The second example uses the interpolated models $\mathbf{w}'_{o,l} = [0.4, 0, 0.4, \cdots, 0, 0.4, 0]^T$ and $\mathbf{w}'_{o,h} = [0.4, 0, -0.4, 0, 0.4, \cdots, 0]^T$. The third example uses block-sparse lowpass and block-sparse highpass models, $\mathbf{w}''_{o,l}$ and $\mathbf{w}''_{o,h}$, whose entries are defined in (13) and (14), respectively.

$$w''_{o,l_i} = \begin{cases} 0, & \text{if } 0 \leq i \leq 9 \\ 0.05(i-9), & \text{if } 10 \leq i \leq 14 \\ 0.3, & \text{if } 15 \leq i \leq 24 \\ 0.3 - 0.05(i-24), & \text{if } 25 \leq i \leq 29 \\ 0, & \text{if } 30 \leq i \leq 39 \end{cases}, \tag{13}$$

$$w''_{o,h_i} = (-1)^{i+1} w''_{o,l_i}. \tag{14}$$

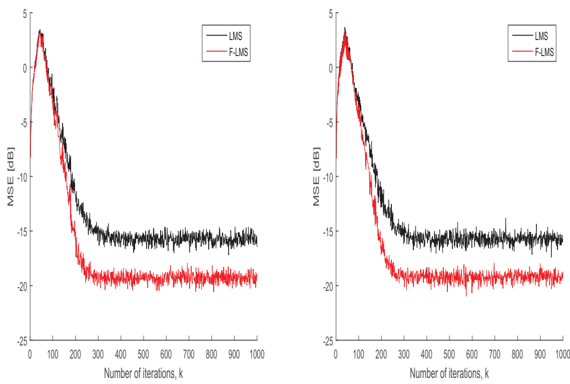
The input signal is a zero-mean white Gaussian noise with unit variance. The signal-to-noise ratio (SNR) is chosen as 20 dB. For all algorithms, the initial vector is $\mathbf{w}(0) = [0, \cdots, 0]^T$ and $\alpha = 0.05$. The values of the step size μ are informed later for each simulated scenario. The MSE learning curves of the LMS and F-LMS algorithms depicted in Figures 1 to 4 are computed by averaging the outcomes of 200 independent trials.

Fig. 1 depicts the MSE learning curves of the LMS and F-LMS algorithms considering the lowpass system $\mathbf{w}_{o,l}$. In Fig. 1(a), both algorithms use the same step size $\mu = 0.03$ so that they exhibit similar convergence speeds. In this figure, we can observe that the F-LMS algorithm achieved a steady-state MSE which is more than 3 dB lower than the MSE re-



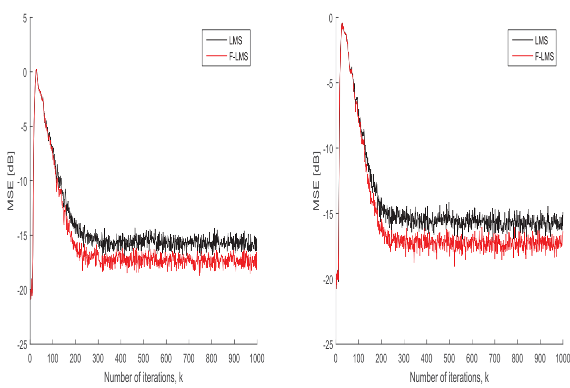
(a) (b)

Fig. 2. MSE learning curves of the LMS and F-LMS algorithms considering $\mathbf{w}_{o,h}$: (a) both algorithms with the same step size: $\mu = 0.03$; (b) LMS and F-LMS with step sizes equal to 0.01 and 0.03, respectively.



(a) (b)

Fig. 3. MSE learning curves of the LMS and F-LMS algorithms, both with step size $\mu = 0.03$, considering the unknown systems: (a) $\mathbf{w}'_{o,l}$ and (b) $\mathbf{w}'_{o,h}$.



(a) (b)

Fig. 4. MSE learning curves of the LMS and F-LMS algorithms, both with step size $\mu = 0.03$, considering the unknown systems: (a) $\mathbf{w}''_{o,l}$ and (b) $\mathbf{w}''_{o,h}$.

sults of the LMS algorithm. In Fig. 1(b), the steady-state MSE of the algorithms are fixed in order to compare their convergence speeds. Thus, we set the step sizes of the LMS and F-LMS algorithms as 0.01 and 0.03, respectively. We can observe, in this figure, that the F-LMS algorithm converged much faster than the LMS algorithm.

In Fig. 2 we present results equivalent to the ones presented in Fig. 1, but considering the highpass system $\mathbf{w}_{o,h}$. Once again, when the step sizes of both algorithms are the same ($\mu = 0.03$), refer to Fig. 2(a), the F-LMS algorithm achieved lower steady-state MSE; whereas the F-LMS algorithm (with $\mu = 0.03$) converged much faster than the LMS algorithm (with $\mu = 0.01$) when their steady-state MSEs are fixed, as illustrated in Fig. 2(b).

Figs. 3(a) and 3(b) depict the MSE learning curves of the LMS and F-LMS algorithms, both using $\mu = 0.03$, considering the interpolated systems $\mathbf{w}'_{o,l}$ and $\mathbf{w}'_{o,h}$, respectively. Notice, in both figures, that the F-LMS algorithm achieved lower steady-state MSE, thus outperforming the LMS algorithm.

Figs. 4(a) and 4(b) depict the MSE learning curves of the LMS and F-LMS algorithms, both using $\mu = 0.03$, considering the block-sparse systems $\mathbf{w}''_{o,l}$ and $\mathbf{w}''_{o,h}$, respectively. In both cases, the F-LMS algorithm achieved lower steady-state MSE, thus outperforming the LMS algorithm.

5. CONCLUSIONS

In this paper, we proposed a family of algorithms called Feature LMS (F-LMS). The F-LMS algorithms are capable of exploiting specific features of the unknown system to be identified in order to accelerate convergence speed and/or reduce steady-state MSE, obtaining a more accurate estimate. The main idea is to apply a sparsity-promoting function to a linear combination of the parameters, in which this linear combination should reveal the sparsity hidden in the parameters, i.e., the linear combination exploits the specific structure/feature in order to generate a sparse vector. Some examples of F-LMS algorithms having low computational complexity and exploiting the lowpass and highpass characteristics of unknown systems were introduced. Simulation results confirmed the superior performance of the F-LMS algorithm in comparison with the LMS algorithm.

In future works, we intend to investigate other choices for the sparsity-promoting penalty function and the feature matrix, analyze the stability and MSE of the F-LMS algorithm, and demonstrate how the F-LMS approach can be used to reduce the computational complexity during the update process of the adaptive filter.

Acknowledgements

The authors would like to thank CAPES, CNPq, and FAPERJ agencies for funding this research work.

6. REFERENCES

[1] B. Widrow and M.E. Hoff, "Adaptive switching circuits," *IRE WESCOM Conv. Rec.*, vol. 4, pp. 96–104, 1960.

[2] P.S.R. Diniz and B. Widrow, "The history of adaptive filters," in *A Short History of Circuits and Systems*, F. Maloberti and A.C. Davies, Eds., chapter 5, pp. 85–90. River Publishers, Denmark, 2016.

[3] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1985.

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

[5] M. Rupp and F. Hausberg, "LMS algorithmic variants in active noise and vibration control," in *22th European Signal Processing Conference (EUSIPCO 2014)*, Lisbon, Portugal, Sept. 2014, pp. 691–695.

[6] S. Rebhi, R. Barrak, and M. Menif, "LMS-based digital pre-equalizer for cognitive RoF system," in *18th International Conference on Transparent Optical Networks (ICTON 2016)*, Trento, Italy, July 2016, pp. 1–4.

[7] D.T. Westwick, B. Maundy, and R. Salmeh, "Approximate LMS tuning of continuous time filters: convergence and sensitivity analysis," *IEE Proceedings - Circuits, Devices and Systems*, vol. 152, no. 1, pp. 1–6, Feb. 2005.

[8] S. Ciochină, C. Paleologu, J. Benesty, S.L. Grant, and A. Anghel, "A family of optimized LMS-based algorithms for system identification," in *24th European Signal Processing Conference (EUSIPCO 2016)*, Budapest, Hungary, Sept. 2016, pp. 1803–1807.

[9] H. Yazdanpanah, P.S.R. Diniz, and M.V.S. Lima, "A simple set-membership affine projection algorithm for sparse system modeling," in *24th European Signal Processing Conference (EUSIPCO 2016)*, Budapest, Hungary, Sept. 2016, pp. 1798–1802.

[10] 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.

[11] 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.

[12] G. Gasso, A. Rakotomamonjy, and S. Canu, "Recovering sparse signals with a certain family of nonconvex penalties and DC programming," *IEEE Transactions on Signal Processing*, vol. 57, no. 12, pp. 4686–4698, Dec. 2009.

[13] 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 (SSPD 2011)*, London, U.K., Sept. 2011, pp. 1–5.

[14] H. Yazdanpanah and P.S.R. Diniz, "Recursive least-squares algorithms for sparse system modeling," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, New Orleans, LA, USA, March 2017, pp. 3879–3883.

[15] M.V.S. Lima, W.A. Martins, and P.S.R. Diniz, "Affine projection algorithms for sparse system identification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013)*, Vancouver, Canada, May 2013, pp. 5666–5670.

[16] Y.-X. Wang, J. Sharpnack, A. Smola, and R.J. Tibshirani, "Trend filtering on graphs," *Journal of Machine Learning Research*, vol. 17, pp. 1–41, April 2016.