Article

Transient Analysis of a Selective Partial-Update LMS Algorithm

Newton N. Siqueira [1], Leonardo C. Resende [2], Fabio A. A. Andrade [3,4]*, Rodrigo M. S. Pimenta [1], Diego B. Haddad [1] and Mariane R. Petraglia [5]

1. Graduate Program in Instrumentation and Applied Optics, Federal Center for Technological Education of Rio de Janeiro (CEFET/RJ), Rio de Janeiro 20271-110, Brazil; newton.siqueira@cefet-rj.br (N.N.S.); rodrigo.pimenta@cefet-rj.br (R.M.S.P.); diego.haddad@cefet-rj.br (D.B.H.)
2. Campus Paracambi, Federal Institute of Rio de Janeiro (IFRJ), Rio de Janeiro 20061-002, Brazil; leonardo.resende@ifrj.edu.br
3. Department of Microsystems, Faculty of Technology, Natural Sciences and Maritime Sciences, University of South-Eastern Norway (USN), 3184 Borre, Norway
4. Drones and Autonomous Systems, NORCE Norwegian Research Centre, 9294 Tromsø, Norway
5. Laboratory for the Processing of Analog and Digital Signals, Federal University of Rio de Janeiro (UFRJ), Rio de Janeiro 21941-630, Brazil; mariane@pads.ufrj.br
* Correspondence: fabio@ieee.org

Abstract: In applications where large-order filters are needed, the computational load of adaptive filtering algorithms can become prohibitively expensive. In this paper, a comprehensive analysis of a selective partial-update least mean squares, named SPU-LMS-M-min, is developed. By employing the partial-update strategy for a non-normalized adaptive scheme, the designer can choose an appropriate number of update blocks considering a trade-off between convergence rate and computational complexity, which can result in a more than 40% reduction in the number of multiplications in some configurations compared to the traditional LMS algorithm. Based on the principle of minimum distortion, a selection criterion is proposed that is based on the input signal's blocks with the lowest energy, whereas typical Selective Partial Update (SPU) algorithms use a selection criterion based on blocks with highest energy. Stochastic models are developed for the mean weights and mean and mean squared behaviour of the proposed algorithm, which are further extended to accommodate scenarios involving time-varying dynamics and suboptimal filter lengths. Simulation results show that the theoretical predictions are in good agreement with the experimental outcomes. Furthermore, it is demonstrated that the proposed selection criterion can be easily extended to active noise cancellation algorithms as well as algorithms utilizing variable filter length. This allows for the reduction of computational costs for these algorithms without compromising their asymptotic performance.

Keywords: adaptive algorithms; least mean squares; selective partial-update algorithms

1. Introduction

Digital communications have seen rapid development, largely due to research in adaptive signal processing [1]. Other important applications of such techniques are system identification, acoustic and network echo cancelers, among others [2,3]. Especially in acoustic and channel equalization scenarios, the use of high-order adaptive filter is necessary. Thus, the required computational complexity can become the bottleneck in obtaining the expected performance [1]. In this context, strategies that reduce the computational load of adaptive filtering schemes are of great interest.

The ability of adaptive filters to adjust to variations in the environment in which they operate is what gives them flexibility and sophistication. The LMS algorithm is the most popular adaptive filtering approach. In general terms, it implements an online estimator

that employs the observable data pair $\{\boldsymbol{x}(k), d(k)\}$, where $d(k) \in \mathbb{R}$ is the reference signal and $\boldsymbol{x}(k) \in \mathbb{R}^N$ denotes the input vector, defined as:

$$\boldsymbol{x}(k) \triangleq \begin{bmatrix} x(k) & x(k-1) & \ldots & x(k-N+1) \end{bmatrix}^T. \tag{1}$$

where N denotes the filter length.

The LMS belongs to the class of supervised adaptive algorithms whose learning dynamics is driven by a feedback mechanism based on the error signal

$$e(k) \triangleq d(k) - \boldsymbol{w}^T(k)\boldsymbol{x}(k), \tag{2}$$

where $y(k) \triangleq \boldsymbol{w}^T(k)\boldsymbol{x}(k)$ is the filter output at the k-th iteration and $\boldsymbol{w}(k) \in \mathbb{R}^N$ contains the N adaptive coefficients $\{w_0(k), w_1(k), \ldots, w_{N-1}(k)\}$. Moreover, the LMS update equation is traditionally derived from the stochastic gradient of the mean squared error, thereby yielding

$$\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \beta\boldsymbol{x}(k)e(k), \tag{3}$$

where $\beta \in \mathbb{R}_+$ is the step size, whose choice implies a trade-off between asymptotic performance and convergence rate [4].

In a system identification task, the block diagram of the adaptive filter functioning is depicted in Figure 1, where the desired signal $d(k)$ sometimes is acquired by ingenious ways and signal $\nu(k)$ models the impact of inaccuracies in the measurement process, such as quantization noise.
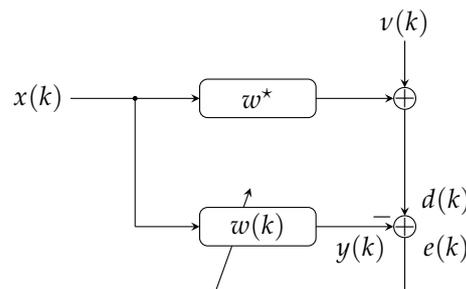
Figure 1. Block diagram of an adaptive filtering algorithm applied to systems identification.

Although the LMS is one of the simplest adaptive filtering algorithms, its update equation requires computational complexity proportional to the length of the adaptive coefficient vector $\boldsymbol{w}(k) \in \mathbb{R}^N$ (see Equations (2) and (3)). Unfortunately, as already discussed, long transfer functions that occur in important real applications might demand a prohibitively large computational effort. Several schemes have been presented in the open literature to address this issue, such as sign-error, sign-data, and selective partial-update (SPU) strategies [5–9]. The latter schemes are the focus of this paper. The SPU-NLMS algorithm, one of the most popular SPU-based adaptive algorithms, is typically presented as a solution of a deterministic optimization problem [10]. The resulting update equation implies a reduced computational burden by decreasing the amount of tap modifications at every iteration. Concisely, the SPU-NLMS groups a set of adaptive coefficients in a single or multiple blocks, and selects the significant excitation data blocks according to their squared Euclidean norms. By a proper selection of input data blocks, the reduction of the computational burden can be attained.

In this paper, a new deterministic optimization approach is proposed, which avoids the normalization steps commonly required by SPU-based schemes. Based on it, a novel SPU-based LMS algorithm, referred to as SPU-LMS-M-min, is introduced. Furthermore, stochastic models are advanced to predict its mean weight dynamics and mean square error performance. In order to obtain a comprehensive analytic characterization of the learning abilities of the SPU-LMS-M-min algorithm, tracking and deficient-length analyses were also carried out.

The proposed coefficient selection method can be easily adapted to mitigate the computational cost of other strategies. For instance, Variable Filter-Length (VFL) algorithms [11–16] dynamically update the filter size throughout iterations, thereby reducing computational cost during transients. However, such algorithms update all adaptive coefficients, leading to an increased computational cost, especially in steady-state conditions. Additionally, these algorithms commonly asymptotically demand a longer filter length than optimal [17], implying unnecessary computational overhead. Both drawbacks can be alleviated through coefficient selection techniques. Thus, it is possible to reduce the computational cost of these algorithms without compromising their asymptotic performance, albeit with a controllable loss (to be judiciously chosen based on the application requirements) in convergence rate.

This paper is organized as follows. In Section 2, the proposed framework is derived, covering the single-block case for didactic purposes. A generalized algorithm that engenders a selection of several blocks in each update is the focus of Section 2.2. Section 3.1 introduces concepts for first-order analysis, which precedes the second-order analysis described at Section 3.2. Generalizations of the advanced stochastic model for both time-variant and deficient-length scenario approaches are presented on Sections 3.3 and 3.4, respectively. Section 4 presents extensions of the proposed selection methodology to variable tap length and active noise cancellation (ANC) algorithms. Results and discussion are the focus of Section 5. At last, Section 6 contains the final considerations of the paper.

2. Proposed SPU-LMS-M-Min for a Single Block

Consider the following partition of the input vector $x(k)$ and the adaptive weights vector $w(k)$ into M equal-length blocks:

$$x(k) = \begin{bmatrix} x_0^T(k) & x_1^T(k) & \dots & x_{M-1}^T(k) \end{bmatrix}^T, \tag{4}$$

$$w(k) = \begin{bmatrix} w_0^T(k) & w_1^T(k) & \dots & w_{M-1}^T(k) \end{bmatrix}^T, \tag{5}$$

where the vectors $x_i(k)$ and $w_i(k)$ contains $N/M = L \in \mathbb{N}$ coefficients (For simplicity, L is supposed to be an integer number.) (for $i \in \{0, 1, \dots, M-1\}$). In order to reduce the computational burden, under the selective partial-update paradigm, only B blocks (for $B \in \{1, 2, \dots, M\}$) of the adaptive weights are updated in each iteration. For the sake of simplicity, in this section it is assumed that only one block is updated in each iteration (i.e., $B = 1$). Note that for the existence of a block algorithm, it is necessary that $M > 1$. Therefore, as $L = N/M$, we have $N > L$, an inequality valid for any value of B (including $B = 1$).

Consider that the index of the block that will be updated ($w_i(k)$) is denoted by i (later, the proposed selection procedure of such a index will be derived). The advanced SPU-LMS-M-min algorithm is the resulting solution of the following optimization problem:

$$\min_{w_i(k+1)} \frac{1}{2} \|w_i(k+1) - w_i(k)\|^2 \text{ s.t. } e_p(k) = \left(1 - \beta L \sigma_x^2\right) e(k), \tag{6}$$

where σ_x^2 denotes the variance in $x(k)$, and the posterior error $e_\mathrm{p}(k)$ is

$$\begin{aligned} e_\mathrm{p}(k) &= d(k) - w^T(k+1)x(k) \\ &= d(k) - \left[w_i^T(k+1)x_i(k) + \overline{w}_i^T(k+1)\overline{x}_i(k) \right], \end{aligned} \tag{7}$$

where $\overline{w}_i(k)$ and $\overline{x}_i(k)$ are the vectors obtained by removing $w_i(k)$ and $x_i(k)$ from $w(k)$ and $x(k)$, respectively. The equality in Equation (6) imposes a linear constraint on the solution $w(k+1)$, which requires that the *a posteriori* error (dependent on the solution $w(k+1)$) be a fraction of the *a priori* error $e(k)$.

Using the Lagrange multipliers technique, the constrained optimization problem (6) can be translated into the following *unconstrained* one:

$$\mathcal{F}_i[\boldsymbol{w}_i(k+1)] = \frac{1}{2}\|\boldsymbol{w}_i(k+1) - \boldsymbol{w}_i(k)\|^2 + \lambda\left[e_p(k) - \left(1 - \beta L\sigma_x^2\right)e(k)\right], \tag{8}$$

where $\lambda \in \mathbb{R}$ is the Lagrange multiplier. Zeroing $\nabla_{\boldsymbol{w}_i(k+1)}\mathcal{F}_i[\boldsymbol{w}_i(k+1)]$ and using the approximation $L\sigma_x^2 \approx \|\boldsymbol{x}_i(k)\|^2, \forall i \in \{0, 1, \dots, M-1\}$, yields

$$\boldsymbol{w}_i(k+1) = \boldsymbol{w}_i(k) + \beta\boldsymbol{x}_i(k)e(k), \tag{9}$$

whereas the remaining blocks are supposed to be unaltered, that is,

$$\overline{\boldsymbol{w}}_i(k+1) = \overline{\boldsymbol{w}}_i(k). \tag{10}$$

Remark 1. *It is worth mentioning that the approximation $L\sigma_x^2 \approx \|\boldsymbol{x}_i(k)\|^2, \forall i \in \{0, 1, \dots, M-1\}$ is a feature of the proposed non-normalized SPU scheme, which is not necessary when normalized adaptive filtering algorithms are adopted. For long filters, exactly where one intends to reduce the computational complexity of adaptive filtering algorithms, such an approximation is less critical (assuming a stationary input signal). In the case of non-stationary signals, real-time estimation of the input signal variance can be achieved with a minor increase in computational complexity. However, in instances of highly non-stationary input signals, the efficacy of such a mechanism may be significantly compromised.*

Note that Equation (9) represents the update equation *after* the selection of the block index i that will be updated, and that, indeed the adopted approximation makes the estimation of the variance in $x(k)$ unnecessary. The choice of such an index i can be oriented by the minimum distortion principle (MDP) [10]:

$$\begin{aligned} i &= \arg\min_{1 \le j \le M}\|\boldsymbol{w}_j(k+1) - \boldsymbol{w}_j(k)\|^2 \\ &= \arg\min_{1 \le j \le M}\|\beta\boldsymbol{x}_j(k)e(k)\|^2 = \arg\min_{1 \le j \le M}\|\boldsymbol{x}_j(k)\|^2, \end{aligned} \tag{11}$$

so that the update equation of the advanced SPU-LMS-M-min algorithm can be written as

$$\boldsymbol{w}_i(k+1) = \boldsymbol{w}_i(k) + \beta\boldsymbol{x}_i(k)e(k), \ i = \arg\min_{0 \le j \le M-1}\|\boldsymbol{x}_j(k)\|^2. \tag{12}$$

Remark 2. *Observe that the advanced criterion selects a block whose quadratic norm is the smallest among all blocks, whereas in established algorithms, the block with the largest norm is chosen [10]. For example, the selection procedure of the M-max LMS chooses the block whose ℓ_2-norm is the largest. This difference derives from the fact that in this paper, the derivation procedure relies on a deterministic and local problem induced by the minimum distortion principle, whereas normally, the stochastic-gradient interpretation of the LMS algorithm is adopted in order to motivate the derivation of SPU-based non-normalized algorithms.*

2.1. Computational Complexity

In order to guarantee an effective reduction of the computational burden, the selection of index i required by the SPU-LMS-M-min algorithm (see Equation (12)) should be carried out by fast algorithms for running ordering and max/min calculation criteria [18]. Assuming the adoption of efficient ordering algorithms (which do not require neither multiplications nor sums, but only comparisons between numbers), it is possible to evaluate the number of additions and multiplications required per iteration by the SPU-LMS-M-min.

Table 1 compares the computational complexity of the elements of an LMS algorithm bundle, taking into account SPU-NLMS (Norm. selective) and LMS-SPU-M-(max | min). It is noteworthy that update Equation (12) requires $N + BL + 1$ multiplications, whereas

the standard LMS requires $2N + 1$ multiplications per iteration. Since the designer might enforce the condition $BL < N$, a significant reduction in computational burden requirement can be attained. For example, for a configuration with $N = 4000$, $M = 8$ and $B = 1$, the number of required multiplications per iteration is reduced by 43.65%. Furthermore, no divisions are required, which is an advantage of the proposed non-normalized scheme. It is noteworthy that such complexity reduction may decrease the convergence rate of the adaptive algorithm, which may not be feasible in applications imposing stringent criteria on the number of iterations required for the learning process to reach a steady state.

Table 1. Number of multiplications, additions, comparisons (i.e., if one value is greater than another, required to find the maximum or minimum value of a vector), and divisions at each iteration of the algorithms and their respective selective partial update variants [10,19]. S denotes the period of coefficient updates (i.e., it is assumed that updates occur only every S consecutive input vectors).

Algorithm LMS-Based	Multiplication	Addition	Comparison	Division
Standard	$2N + 1$	$2N$	–	–
M-min	$N + BL + 1$	$N + BL$	$2[log_2 N] + 2$	–
Periodic	$N + (N + 1)/S$	$N + N/S$	–	–
Sequential	$N + BL + 1$	$N + BL$	–	–
Stochastic	$N + BL + 3$	$N + BL + 2$	–	–
M-max	$N + BL + 1$	$N + BL$	$2[log_2 N] + 2$	–
Norm. Selective	$N + BL + 2$	$N + BL + 2$	$2[log_2 N] + 2$	1

After the following new derivation approach of the target algorithm SPU-LMS-M-min, it is expected that presents some distinct features, such as reduction in the convergence rate, which can be predicted and elucidated by stochastic models. Some models based on conventional assumptions are described in the following sections. Note that such theoretical analysis are appreciated in the open literature due to their ability to provide performance guarantees for the algorithm designer, as well as important insights about its functioning.

2.2. SPU-LMS-M-Min for Multiple Blocks

The ensuing derivations facilitates the understanding of the generalized formulation of the algorithm, in which the designer intends to update B blocks in each iteration. Consider these indexes are denoted by $\mathcal{I}_B = \{i_0, i_1, \ldots, i_{B-1}\}$. Equation (8) can be extended by the following *generalized* optimization problem:

$$\min_{\boldsymbol{w}_{\mathcal{I}_B}(k+1)} \frac{1}{2} \|\boldsymbol{w}_{\mathcal{I}_B}(k+1) - \boldsymbol{w}_{\mathcal{I}_B}(k)\|^2$$

$$\text{s.t. } e_p(k) = \left(1 - \beta B \frac{N}{M} \sigma_x^2\right) e(k), \tag{13}$$

where $\boldsymbol{w}_{\mathcal{I}_B}(k+1) \triangleq \left[\boldsymbol{w}_{i_0}^T(k) \quad \boldsymbol{w}_{i_1}^T(k) \quad \ldots \quad \boldsymbol{w}_{i_{B-1}}^T(k)\right]^T$.

Using similar steps to those in (9), one may describe the SPU-LMS-M-min algorithm that updates B blocks by

$$\boldsymbol{w}_{\mathcal{I}_B}(k+1) = \boldsymbol{w}_{\mathcal{I}_B}(k) + \beta \boldsymbol{x}_{\mathcal{I}_B}(k) e(k), \tag{14}$$

where $\mathcal{I}_B$ can be computed by the following rule:

$$
\begin{aligned}
\mathcal{I}_B &= \arg\min_{J_B \in S} \|\boldsymbol{w}_{J_B}(k+1) - \boldsymbol{w}_{J_B}(k)\|^2 \\
&= \arg\min_{J_B \in S} \|\beta \boldsymbol{x}_{J_B}(k)e(k)\|^2 = \arg\min_{J_B \in S} \|\boldsymbol{x}_{J_B}(k)\|^2 \\
&= \arg\min_{J_B \in S} \sum_{j \in J_B} \|\boldsymbol{x}_j(k)\|^2.
\end{aligned}
\tag{15}
$$

3. Stochastic Modelling of the Proposed Algorithm

3.1. First-Order Analysis

Note that Equation (12) can be rewritten as

$$
\boldsymbol{w}(k+1) = \boldsymbol{w}(k) + \beta \boldsymbol{\Gamma}(k)\boldsymbol{x}(k)e(k),
\tag{16}
$$

where $\boldsymbol{\Gamma}(k)$ is an $N \times N$ diagonal matrix, with each element of its main diagonal denoted as $\gamma_i(k)$, where i is the associated block index. The main diagonal elements of $\boldsymbol{\Gamma}(k)$ are obtained by

$$
\gamma_i(k) = \begin{cases} 0, & \text{if } i \notin \mathcal{I}_B \\ 1, & \text{if } i \in \mathcal{I}_B \end{cases}.
\tag{17}
$$

Consider that reference signal $d(k)$ is generated by the following affine regression model:

$$
d(k) = [\boldsymbol{w}^\star]^T \boldsymbol{x}(k) + \nu(k),
\tag{18}
$$

where $\boldsymbol{w}^\star \in \mathbb{R}^N$ is a vector that contains the coefficients of the ideal (and unknown) plant and $\nu(k)$ is an additive noise. Defining the deviation coefficient vector as $\tilde{\boldsymbol{w}}(k) \triangleq \boldsymbol{w}^\star - \boldsymbol{w}(k)$, the following recursion can be obtained from Equation (16):

$$
\tilde{\boldsymbol{w}}(k+1) = \left(\boldsymbol{I} - \beta \boldsymbol{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right)\tilde{\boldsymbol{w}}(k) - \beta \boldsymbol{\Gamma}(k)\boldsymbol{x}(k)\nu(k),
\tag{19}
$$

which is a non-homogeneous stochastic difference equation, where $-\beta \boldsymbol{\Gamma}(k)\boldsymbol{x}(k)\nu(k)$ acts as a driving force that avoids the asymptotic convergence of the deviation vector energy to zero. A first-order stochastic analysis of the SPU-LMS-M-min algorithm can be performed by applying the expectation operator in $\mathbb{E}[\cdot]$ in (19), which leads (using the popular *independence assumption*-`IA` [4]) to

$$
\mathbb{E}[\tilde{\boldsymbol{w}}(k+1)] = (\boldsymbol{I} - \beta \boldsymbol{R}_{\boldsymbol{\Gamma}})\mathbb{E}[\tilde{\boldsymbol{w}}(k)],
\tag{20}
$$

where $\boldsymbol{R}_{\boldsymbol{\Gamma}} \triangleq \mathbb{E}[\boldsymbol{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)]$.

Remark 3. *Note that Equation (20) implies that the SPU-LMS-M-min, under a sufficient excitation condition, performs an asymptotically unbiased estimation [20]. Furthermore, the maximum step size that guarantees mean-weight convergence can be obtained by assuming*

$$
\beta < \frac{2}{\rho[\boldsymbol{R}_{\boldsymbol{\Gamma}}]},
\tag{21}
$$

where $\rho[\boldsymbol{R}_{\boldsymbol{\Gamma}}]$ denotes the spectral radius of $\boldsymbol{R}_{\boldsymbol{\Gamma}}$, i.e., $\rho[\boldsymbol{R}_{\boldsymbol{\Gamma}}] \triangleq \max_i |\lambda_i(\boldsymbol{R}_{\boldsymbol{\Gamma}})|$, where λ_i is the i-th eigenvalue of $\boldsymbol{R}_{\boldsymbol{\Gamma}}$. Unfortunately, a stable-in-the-mean adaptive filter may perform an estimation with unbounded variance, so that a second-order stochastic analysis is necessary in order to achieve proper performance guarantees [20]. In the derivation of Equation (20), the statistically strong (but physically plausible) assumption that the additive noise is white and statistically independent from the remaining random variables is employed. Such a noise assumption (`NA`) is almost ubiquitous and is utilized henceforth.

3.2. Second-Order Analysis

The usage of the independence assumption implies that the expectation value of the matrix

$$\mathbf{\Phi}(k) = \tilde{\boldsymbol{w}}(k)\tilde{\boldsymbol{w}}^T(k) \tag{22}$$

has a major rule in a second-order stochastic model of adaptive algorithms [20]. Using Equation (19), a recursion for $\mathbf{\Phi}(k)$ can be obtained as

$$
\begin{aligned}
\mathbf{\Phi}(k+1) \;=\; & \mathbf{\Phi}(k) - \beta\mathbf{\Phi}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Gamma}(k) \\
& -\beta\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Phi}(k) \\
& +\beta^2\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Phi}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Gamma}(k) \\
& +\beta^2\nu^2(k)\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Gamma}(k) + \mathcal{O}[\nu(k)],
\end{aligned}
\tag{23}
$$

where $\mathcal{O}[\nu(k)]$ contains first-order noise components, which are irrelevant to the following analysis.

Recursion (23) can be rewritten in a more adequate formulation by applying the relationship

$$\text{vec}(\boldsymbol{XYZ}) = \left(\boldsymbol{Z}^T \otimes \boldsymbol{X}\right)\text{vec}(\boldsymbol{Y}), \tag{24}$$

where $\text{vec}(\boldsymbol{A})$ is an operator that stacks the columns of matrix $\boldsymbol{A}$ in order to generate a column-type vector and $\otimes$ denotes the Kronecker product. Employing such a formulation, applying the expectation operator and using the independence assumption, leads to the following time-invariant state space equation:

$$\boldsymbol{v}(k+1) = \boldsymbol{A}\boldsymbol{v}(k) + \boldsymbol{b}, \tag{25}$$

where

$$\boldsymbol{v}(k) \triangleq \mathbb{E}\{\text{vec}[\mathbf{\Phi}(k)]\}, \tag{26}$$

$$
\begin{aligned}
\boldsymbol{A} \;\triangleq\; & \boldsymbol{I} - \beta\mathbb{E}\left\{\left[\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right] \otimes \boldsymbol{I}\right\} \\
& -\beta\mathbb{E}\left\{\boldsymbol{I} \otimes \left[\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right]\right\} \\
& +\beta^2\mathbb{E}\left\{\left[\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right] \otimes \left[\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right]\right\},
\end{aligned}
\tag{27}
$$

and

$$\boldsymbol{b} \triangleq \beta^2\sigma_v^2\mathbb{E}\left\{\text{vec}\left[\mathbf{\Gamma}(k)\boldsymbol{x}(k)\boldsymbol{x}^T(k)\mathbf{\Gamma}(k)\right]\right\}, \tag{28}$$

where σ_v^2 is the variance of the additive noise.

Since $\text{vec}[\boldsymbol{X}]$ is a bijection operator, all information of the mismatch covariance matrix $\boldsymbol{R}_{\tilde{w}}(k) \triangleq \mathbb{E}[\mathbf{\Phi}(k)]$ can be extracted from $\boldsymbol{v}(k)$. From both IA and NA stochastic hypotheses, the mean square deviation (MSD) and mean square error (MSE) performance metrics can be predicted for the *k*-th iteration using [20]

$$
\begin{aligned}
\text{MSD}(k) \;\triangleq\; & \mathbb{E}\left[\|\boldsymbol{w}^\star - \boldsymbol{w}(k)\|^2\right] \approx \text{Tr}[\boldsymbol{R}_{\tilde{w}}(k)], \tag{29} \\
\text{MSE}(k) \;\triangleq\; & \mathbb{E}\left[e^2(k)\right] \approx \sigma_v^2 + \text{Tr}[\boldsymbol{R}_{\boldsymbol{x}}\boldsymbol{R}_{\tilde{w}}(k)], \tag{30}
\end{aligned}
$$

where $\text{Tr}[\boldsymbol{Y}]$ denotes the trace of matrix $\boldsymbol{Y}$ and $\boldsymbol{R}_{\boldsymbol{x}} \triangleq \mathbb{E}\left[\boldsymbol{x}(k)\boldsymbol{x}^T(k)\right]$ is the input autocorrelation matrix.

Remark 4. *It is noteworthy that Equation* (25) *is able to yield a closed-form estimate for both asymptotic MSD and MSE, provided the algorithm is stable. In this case, the steady-state vector* $v_\infty \triangleq \lim_{k \to \infty} v(k)$ *can be computed by* [20]

$$v_\infty = (I - A)^{-1} b. \tag{31}$$

3.3. Tracking Analysis

Since adaptive filters that present good performance in stationary environments do not necessarily exhibit good learning behavior in the identification of time-variant plants [21], obtaining a stochastic model that offers guarantees with respect to the algorithm learning capabilities in a tracking scenario is crucial.

A popular model that accounts for temporal variation in $w^\star(k)$ can be described as a first-order Markov process [22]

$$w^\star(k+1) = w^\star(k) + q(k), \tag{32}$$

where the statistical characterization of the perturbation vector $q(k)$ is established by the assumption stated below.

Assumption 1. *Tracking assumption (TA): vector* $q(k_1)$ *is statistically independent from the input data and from* $q(k_2)$, $\forall k_2 \neq k_1$. *Furthermore, its autocorrelation matrix is given by* $R_q \triangleq \mathbb{E}[q(k)q^T(k)]\sigma_q^2 I_N$, *where* I_N *denotes the N-order identity matrix.*

In order to create coefficient error vector in a deviation form, let us consider Equation (32) *compared to Equation* (16):

$$\begin{aligned} w^\star(k+1) - w(k+1) &= w^\star(k+1) - w(k) - \beta\Gamma(k)x(k)e(k) \\ \tilde{w}(k+1) &= q(k) + w^\star(k) - w(k) - \beta\Gamma(k)x(k)e(k) \\ \tilde{w}(k+1) &= \tilde{w}(k) - \beta\Gamma(k)x(k)e(k) + q(k), \end{aligned} \tag{33}$$

By also replacing $e(k)$ *by Equation* (2) *at Equation* (34), *after some arrangements, the following recursion deviation vector* $\tilde{w}(k)$ *can be rewritten:*

$$\begin{aligned} \tilde{w}(k+1) = \Big(I - \beta\Gamma(k)x(k)x^T(k)\Big)\tilde{w}(k) - \\ -\beta\Gamma(k)x(k)\nu(k) + q(k) \end{aligned} \tag{34}$$

In order to reach a mean-square performance model (following similar steps than those employed to obtain Equation (25)*), taking into account Equation* (34), *which can be used to derive the non-homogeneous stochastic difference equation by performing the product between Equation* (34) *and its transposed version, reaching this equation:*

$$\begin{aligned} \Phi(k+1) \quad = \quad & \Phi(k) - \beta\Phi(k)x(k)x^T(k)\Gamma(k) \\ & -\beta\Gamma(k)x(k)x^T(k)\Phi(k) \\ & +\beta^2\Gamma(k)x(k)x^T(k)\Phi(k)x(k)x^T(k)\Gamma(k) \\ & +\beta^2\nu^2(k)\Gamma(k)x(k)x^T(k)\Gamma(k) \\ & +q(k)q^T(k) + \mathcal{O}[\nu(k)] + \mathcal{O}[q(k)], \end{aligned} \tag{35}$$

where $\Phi(k)$ *is as defined by Equation* (22), $\mathcal{O}[\nu(k)]$ *denotes noise-related terms that will not interfere with the hereinafter analysis, and* $\mathcal{O}[q(k)]$ *is related to the first-order perturbation vector, which would also not be part of that balance. Aiming to make the mathematics tractable, the application of the expectation operator from Equation* (36) *should be performed alongside some assumptions, which are described below.*

$$v(k+1) = Av(k) + b + c, \tag{36}$$

where $c \triangleq \mathbb{E}\{vec[q(k)q^T(k)]\}$.

Remark 5. *Note that under the considered tracking scenario, the stability region of the algorithm remains unaltered, since the time-invariant transition matrix A is the same as the stationary configuration [20].*

3.4. Deficient-Length Analysis

There are several practical situations to motivate that analysis: Usually we do not know in advance the transfer function to be identified, no how limited the computational resources would be. In addition, the length of the adaptive filter N is often less than the length L of the transfer function to be identified. In this deficient-length setting, according to [4], it is possible to model the transfer function coefficients vector and the reference signal as:

$$w_o^\star = [w^\star, \bar{w}^\star]^T, \tag{37}$$

$$d_{\text{def}}(k) = [w^\star]^T x(k) + [\bar{w}^\star]^T \bar{x}(k) + \nu(k), \tag{38}$$

where

$$\bar{x}(k) \triangleq \begin{bmatrix} x(k-N) & x(k-N-1) & \cdots & x(k-N-L+1) \end{bmatrix} \tag{39}$$

and $\bar{w}^\star \in \mathbb{R}^L$ is the L-length component of the ideal transfer function, $w^\star$, whose total length is $N + L$, that the adaptive filter is not able to emulate. Considering Equation (2) and the previous definitions, it is possible to write $e_{\text{def}}(k)$ in the deficient-length approach:

$$
\begin{aligned}
e_{\text{def}}(k) \quad &= d_{\text{def}}(k) - \tilde{w}^T(k) x(k) \\
&= (w^\star)^T x(k) + (\bar{w}^\star)^T \bar{x}(k) + \nu(k) - \tilde{w}^T(k) x(k) \\
&= \tilde{w}^T(k) x(k) + (\bar{w}^\star)^T \bar{x}(k) + \nu(k),
\end{aligned}
\tag{40}
$$

Subsequently, defining input, reference signals, and error, it is necessary to rewrite Equation (16) for the context of the deficient-length analysis with Equation (41):

$$
\begin{aligned}
w^\star(k+1) - w(k+1) \quad &= \quad w^\star(k+1) - w(k) - \beta \Gamma(k) x(k) e_{\text{def}}(k) \\
\tilde{w}(k+1) \quad &= \quad \tilde{w}(k) - \beta \Gamma(k) x(k) e_{\text{def}}(k) \\
\tilde{w}(k+1) \quad &= \quad \tilde{w}(k) - \beta \Gamma(k) x(k) [\tilde{w}^T(k) x(k) \\
&\quad + \quad (\bar{w}^\star)^T \bar{x}(k) + \nu(k)] \\
\tilde{w}(k+1) \quad &= \quad \tilde{w}(k) - \beta \Gamma(k) x(k) x^T(k) \tilde{w}(k) \\
&\quad - \quad \beta \Gamma(k) x(k) \bar{x}^T(k) \bar{w}^\star - \beta \Gamma(k) x(k) \nu(k)
\end{aligned}
\tag{41}
$$

Calculating the product between Equation (41) and its respective transposed version, applying the same steps as than those employed to obtain Equation (25), also to make more comprehensible the mathematics, the following assumption will be considered in the deficient-length setup:

Assumption 2. *Whiteness assumption (`WA`). The excitation data is white.*

Remark 6. *Note that `WA` is a common assumption in the field of adaptive filtering analysis, even when the input signal is non-stationary (see, e.g., [23]). A more evolved analysis is necessary if such an assumption should be circumvented [24]. Using `WA` and the previous assumptions, it can be demonstrated that in the suboptimal scenario (38) the vector $v(k)$ is updated according to*

$$v_{k+1} = A v_k + b + d \tag{42}$$

where

$$d \triangleq \beta^2 \mathbb{E}\left\{ \left[\Gamma(k) x(k) x^T(k) \right] \otimes \left[\Gamma(k) x(k) x^T(k) \right] \right\} \bar{v} \tag{43}$$

and $\overline{v} \triangleq vec\left[\overline{w}\overline{w}^T\right].$

4. Extensions of the Proposed Framework

Although the entire discussion up to this point has been focused on the LMS algorithm and the system identification problem, the proposed framework can be easily extended to VTL strategies and active noise cancellation (ANC) problems. This section details the extension of the proposed methodology for both cases. It is observed that the computational cost reduction capacity of the proposed methodology can be easily extended to other schemes and scenarios.

The computational complexity of an adaptive filter can be reduced through the use of VTL schemes [11–16], which adjust the tap length in real time. These algorithms share with the SPU technique the ability to present reduced computational cost during transients. However, under steady-state conditions, they update all filter coefficients. This, together with the tendency of these algorithms to overestimate the optimal steady-state adaptive filter length, can lead to excessively high steady-state computational costs [17]. Such an issue can be circumvented by using the proposed selection update scheme.

As a proof of concept, we consider the VTL algorithm proposed in [12]. The algorithm adjusts the pseudo fractional tap-length $n_f(k) \in \mathbb{R}_+$ through the following updating rule:

$$n_f(k+1) = (n_f(k) - \alpha) - \gamma\left\{\left[e^{(N(k))}(k)\right]^2 - \left[e^{(N(k)-\Delta)}(k)\right]^2\right\},\tag{44}$$

where α, γ, and Δ are adjustable parameters, the choice of which has been widely discussed in the literature (see, e.g., [17]), $N(k)$ is the length of the adaptive filter in the kth iteration, and $e^{(L)}(k)$ is the error calculated with a filter of length L. The algorithm adjusts the tap length using the rule:

$$N(k+1) = \begin{cases} \lfloor n_f(k) \rfloor, & |N(k) - n_f(k)| \geq \delta, \\ N(k), & \text{otherwise} \end{cases},\tag{45}$$

where $\delta \in \mathbb{R}_+$ is also a tuning parameter. Since the size of the adaptive filter in VTL schemes varies dynamically, it is challenging to choose a block size M for an SPU technique insertion such that the current size of the adaptive filter is a multiple of M. To overcome this issue, we consider the choice $M = N(k)$. Thus, each sample of the input vector corresponds mathematically to a block of the SPU methodology. One can then choose B to ensure that a certain percentage of the adaptive coefficients in the VTL technique are updated in a given iteration, using the criterion proposed in Equation (15) for the coefficients to be updated. This ensures that the B adaptive coefficients to be updated are associated with the B samples of the input vector $x(k)$ with the *lowest* magnitudes. The resulting algorithm exhibits asymptotic performance equivalent to that of the original VTL algorithm, with lower computational cost. However, due to the coefficient selection, there is also a loss in convergence rate, which is inevitable when constraints on computational complexity are strict.

ANC algorithms can also benefit from adopting the coefficient selection strategy. Consider, for example, the Filtered-X LMS (FX-LMS) algorithm [15,25,26], whose block diagram is presented in Figure 2. The FX-LMS algorithm is used in active noise control applications to minimize noise in a primary input signal. It operates by subtracting the output of an adaptive filter from a reference signal, aiming to estimate and cancel unwanted noise in the primary input. The algorithm adjusts the adaptive filter coefficients iteratively using the LMS approach, with each update being proportional to the product of the error signal (the difference between the reference signal and the filtered input) and the input signal. The process continues iteratively until convergence, where the adaptive filter effectively minimizes the mean square error, resulting in reduced noise in the primary input signal. The filtering of the input signal is performed through an estimate $\hat{S}(z)$ of the secondary path $S(z)$. The computational cost of the FX-LMS algorithm can be mitigated by

selectively updating a fraction of its coefficients using the proposed criterion. Similar to the case of VTL techniques, a maintenance of the asymptotic performance of the FX-LMS algorithm is observed, along with a controllable loss in the convergence rate.
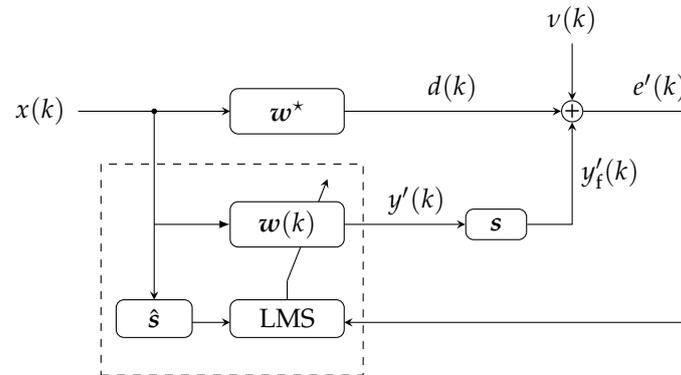
Figure 2. Block diagram of the FX-LMS algorithm.

5. Results

In this section, simulation results are presented in order to assess the quality of theoretical predictions using simulated curves obtained through 1000 independent Monte Carlo trials (except when a different value is explained). In all of the following scenarios, the additive noise derives from a Gaussian and white process.

5.1. First-Order Analysis

In this scenario, the proposed model of the algorithm is evaluated with regard to the average evolution of the coefficients. The plant to be emulated consists of the first 64 coefficients from Model 4 of [27]. The total number of blocks is $M = 8$. To evaluate the accuracy of the proposed model's behavior, we consider a configuration in which the input signal is white Gaussian and of unitary variance. The SPU-LMS-M-min parameters are: $\beta = 10^{-2}$, $\sigma_\nu^2 = 10^{-6}$. The results shown in Figure 3 reveal good correspondence with the experimental results and confirm the expectation that an increase in B implies a faster convergence of the algorithm. It is important to point out that Figure 3a evaluates the evolution of the coefficient $w_8(k)$, among the 64 coefficients, while Figure 3b presents the evolution of the coefficient $w_{34}(k)$, among the 64 adaptive coefficients. In both figures, an almost perfect adherence can be seen, with updates of one, two, three, or four blocks. Such adherence allows us to verify that the first-order theoretical modeling is validated by the results obtained via simulation, indicating that the hypotheses described do not project significant discrepancies in relation to the real behavior of the algorithm in this scenario.
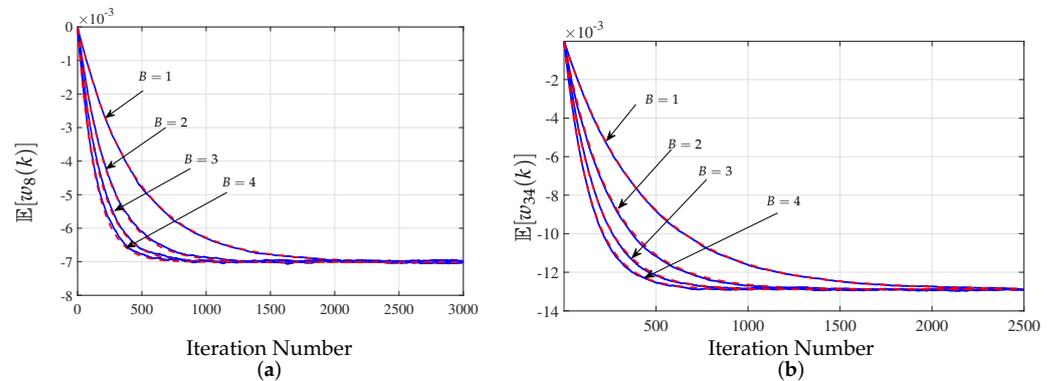
Figure 3. Theoretical (red dashed line) and empirical (blue solid line) evolution of the adaptive filter coefficients for $B \in \{1, 2, 3, 4\}$. (**a**) Coefficient $w_8(k)$; (**b**) Coefficient $w_{34}(k)$.

5.2. Second-Order Analysis

In this scenario, the proposed model of the algorithm is evaluated with respect to the MSD and MSE metrics. The plant to be emulated contains the first 32 coefficients from Model 4 of [27]. The total number of blocks is $M = 8$. The input signal is white and the SPU-LMS-M-min parameters are: $\beta = 10^{-2}$, $\sigma_v^2 = 10^{-6}$. The results shown in Figure 4a reveal good correspondence between the theoretical and experimental curves of the MSD, both in steady-state and transient regimes. There is also confirmation that an increase in B implies a higher convergence rate of the algorithm. Such an increase also implies greater adherence between theoretical and experimental results (see the curve with $B = 4$). The results shown in Figure 4b reveal good correspondence between the theoretical and experimental curves of the MSE, both in steady-state and transient regimes.

5.3. Tracking Analysis

The results presented in Figure 5 shows the ability of the proposed stochastic model to predict the tracking ability of the SPU-LMS-M-min algorithm. The plant to be emulated contains the first 32 coefficients from Model 2 of [28]. The additive noise, as well as the perturbation $q(k)$, are distributed according to a white Gaussian process. Table 2 presents the parameters used in the simulations.

The results shown in Figure 5 reveal the asymptotic MSD performance of the algorithm as a function of β. The excellent adherence between the experimental results and the theoretical forecast stands out. Note that there is a specific value of β that optimizes the asymptotic performance, which is duly estimated by the elaborated theoretical analysis. This value of β is high enough to allow the filter to follow the non-stationarity of the plant varying in time, but at the same time, it is not high enough to generate a lot of variability in the estimation process. Such high variability has repercussions on the second-order statistics (such as variance), which are captured by performance metrics.

Table 2. Parameters used in tracking simulations scenario.

Parameters	Figure 5
β	10^{-2}
σ_v^2	10^{-6}
$H(z)$	$1 - 0.8z^{-1} + 0.2z^{-2}$
σ_q^2	10^{-15}
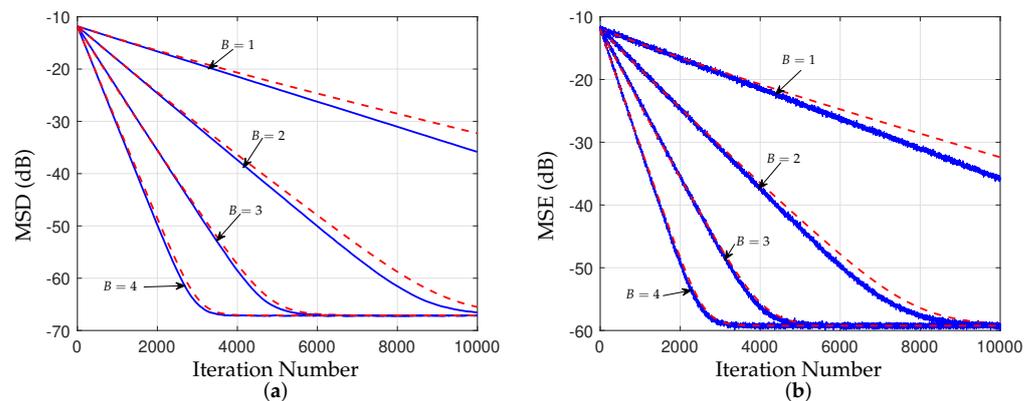
Figure 4. Theoretical (red dashed line) and empirical (blue solid line) evolution of the adaptive filter coefficients for $B \in \{1, 2, 3, 4\}$. (**a**) MSD (dB); (**b**) MSE (dB).
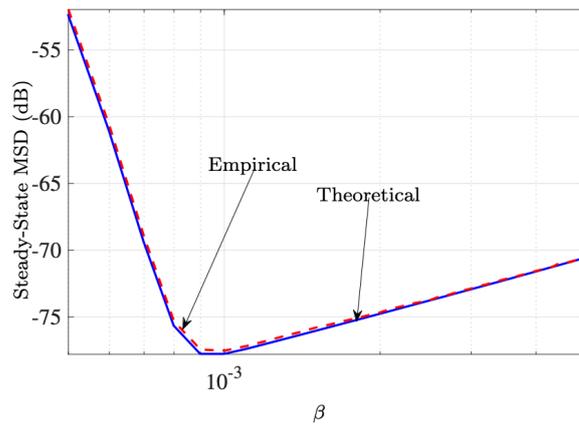
Figure 5. Theoretical (red) and simulated (blue) steady-state MSD (in dB), as a function of β. The variance in the random perturbation is $\sigma_q^2 = 10^{-15}$.

In the following scenario related to Figure 6, the target is to identify ability of the proposed stochastic model in following the evolution of the MSD and MSE under a non-stationary σ_v^2. The plant to be identified contains the first 32 coefficients of Model 2 of [28]. The total number of blocks is $M = 8$. To evaluate the accuracy of the behavior of the proposed model before a colored input signal, the input signal is generated by passing a Gaussian white noise of unitary variance through the filter $H(z) = 1 - 0.8z^{-1} + 0.6z^{-2} - 0.1z^{-3}$. The parameters of SPU-LMS-M-min are: $\beta = 8 \times 10^{-3}$ and the formula used for calculating $\sigma_v^2(k)$ is

$$\sigma_v^2(k) = \overline{\sigma}_v^2 + A \sin(2\pi f_s k), \tag{46}$$

where $\overline{\sigma}_v^2 = 10^{-2}$, $A = 10^{-1}$ and $f_s = 2 \times 10^{-3}$. Empirical results were obtained through 2×10^4 independent Monte Carlo trials. The results shown in Figure 6a reveal adequate correspondence between the theoretical and experimental MSD curves, both in steady-state and transient regimes. The results shown in Figure 6b, which take advantage of the same parameters as those of Figure 6a, reveal adequate correspondence between the theoretical and experimental MSE curves, both in steady-state and transient regimes.
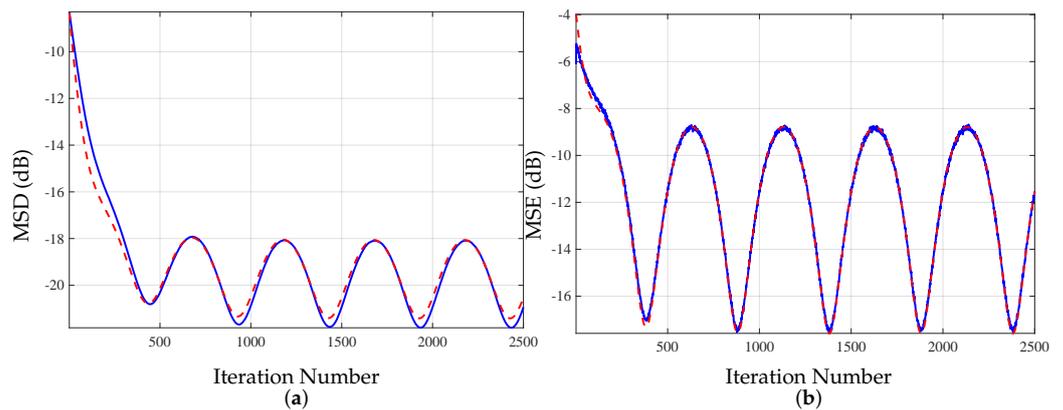
Figure 6. Comparison of empirical results (**a**) MSD (dB); (**b**) MSE (dB). Theoretical result (red) and experimental result (blue), considering the colored input signal.

5.4. Deficient-Length Analysis

The results shown in Figure 7a,b reveal adequate correspondence between the theoretical and experimental MSE curves. Note that Equation (42) is very consistent with the dynamics of the MSE, both in transient regions, and especially in asymptotic regions. In these simulations, we used the following parameters: $\beta = 10^{-2}$, $\sigma_v^2 = 10^{-4}$, and $M = 4$.

The average results were computed from 10^5 independent Monte Carlo trials. The ideal transfer function, $w_i^\star$ was generated using the following interval:

$$w_i^\star = \begin{cases} 1, & \text{for } i \in \{0, 1, ..., N-1\} \\ 0.5, & \text{for } i \in \{N, N+1, ..., N+D-1\} \end{cases} \tag{47}$$

Taking into account $N = C + D$, where $C \in \{12, 16\}$ is the fixed number of filter coefficients, and $D \in \{2, 4, 6\}$ is the number of coefficients that exceed the filter amount. Note that Equation (42) is very consistent with the empirical dynamics of the MSE both in transient and asymptotic regions.
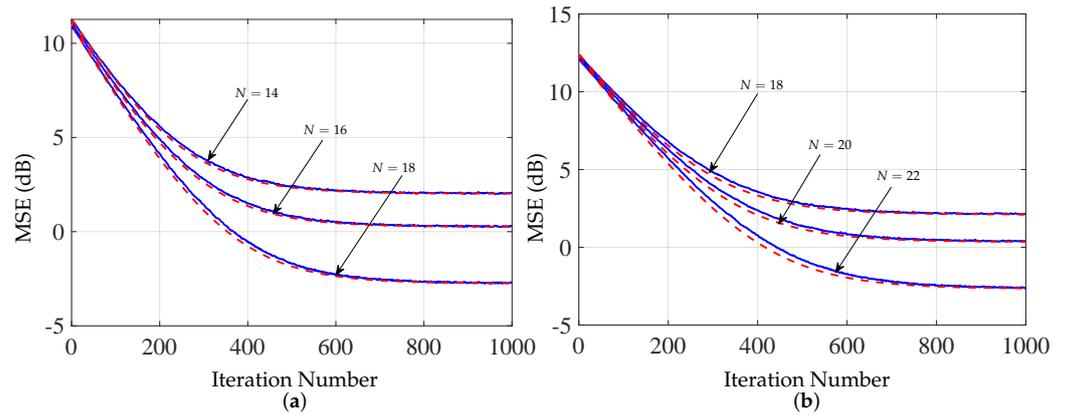
Figure 7. Comparison between theoretical MSE (dashed red line) and empirical MSE (solid blue line) for the deficient-length scenario.

5.5. SPU-LMS-M-Min versus SPU-LMS-M-Max

Accordingly with classical approach for derivation of the SPU-LMS-M-max algorithm there was a statement never discussed before related to block selection, whose quadratic norm is the *largest* chosen [10] among all blocks, whereas in this paper the block with the *smallest* norm is elected. It should be observed that the advanced criterion selects a smallest norm (M-min) and shows better and more efficient results compared with the M-max variation, even-though under very intensive impulsive noise. Figure 8a shows comparisons between empirical MSD in which the first 64 coefficients of Model 3 of [28] are employed as the ideal transfer function. All empirical results were obtained by performing 2×10^5 independent Monte Carlo trials, and the additive noise derives from a white Gaussian process. The total number of blocks is $M = 8$ and only $B = 5$ blocks were updated. The input signal is generated by passing a unity-variance white Gaussian noise through the filter $H(z) = 1 - 0.8z^{-1} + 0.6z^{-2} - 0.1z^{-3}$. The selected parameters are $\beta = 10^{-3}$, $\overline{\sigma}_\nu^2 = 10^{-8}$, and impulsive noise in a range from $\overline{\sigma}_\nu^2 = 2^{-1}$ to $\overline{\sigma}_\nu^2 = 1$ with 50% probability of occurrence. Figure 8b shows comparisons between empirical MSEs taking advantage of the same parameters as those employed in Figure 8a. In the settings depicted in Figure 8a,b, the reduction in the number of multiplications brought about by the proposed algorithm (compared to LMS) was approximately 20%. Note that a high-energy noise signal severely degrades the algorithm's performance.
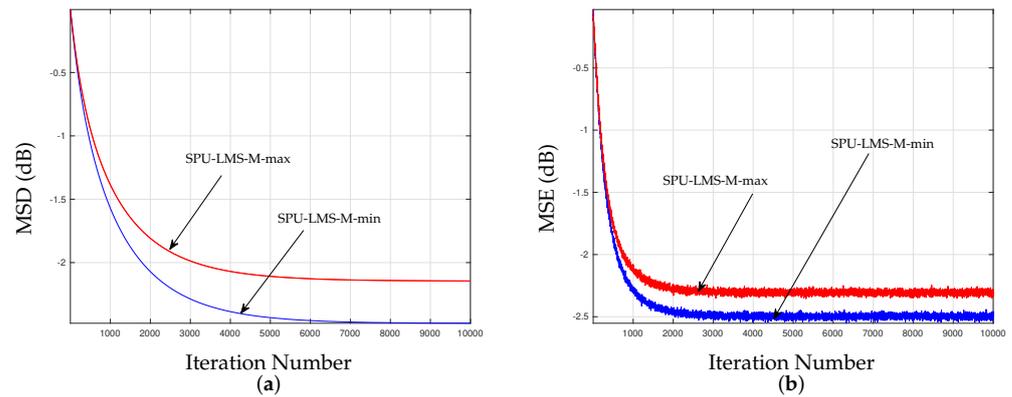
Figure 8. Comparison between empirical SPU-LMS-M-max (red) and SPU-LMS-M-min (blue) for the very intense impulsive noise scenario. (**a**) MSD (dB); (**b**) MSE (dB).

5.6. Extensions of the Proposed Framework Simulations

In accordance with the extension approach for SPU-LMS-M-max, incorporating a variable tap length algorithm, it is noteworthy that this method inherently utilizes a greater number of steady-state coefficients than strictly essential. Consequently, such an algorithm demands inefficient utilization of computational resources during steady-state operation. In response to this challenge, the amalgamation of SPU and VTL is proposed, thereby facilitating diminished computational costs while preserving an equivalent asymptotic performance. Despite the inevitable consequence of a controllable reduction in the convergence rate due to this integration, this trade-off can be effectively managed by judiciously selecting the fraction of coefficients to update. Thus, the proposed SPU methodology was applied to an Acoustic Echo Cancellation (AEC) context, showcasing sustained asymptotic performance alongside diminished computational costs. This accomplishment is achieved at the cost of a controllable loss in the convergence rate, a parameter that remains adjustable by the designer. Figure 9a shows comparisons between empirical MSE for those algorithms VTL and SPU-VTL, where the 100 coefficients of Model 2 of [27] are employed as the ideal transfer function. All empirical results were obtained by performing 2×10^4 independent Monte Carlo trials, and the additive noise derives from a white Gaussian process. The total number of blocks is $M = 8$ and only 85% of the blocks were updated. The input signal is generated by passing a unity-variance white Gaussian noise through the filter $H(z) = 0.35 - z^{-1} + 0.35z^{-2}$. The selected parameters were $\beta = 10^{-2}$, $\sigma_v^2 = 10^{-6}$, $\gamma = 20$, $\alpha = 4.10^{-4}$, $\delta = 3$, and $\Delta = 15$. While the transient regime underscores the superior convergence rate of Variable Tap Length (VTL), it is imperative to highlight that the utilization of Selective Partial Update with Variable Tap Length (SPU-VTL) concurrently diminishes computational complexity while maintaining an equivalent convergence rate during steady-state operation. Therefore, in the context of a Mean Squared Error (MSE) comparison, the judicious selection between VTL and SPU-VTL is pivotal for designers aiming to achieve specific targets. Figure 9b shows comparisons between empirical Tap Length over iterations used on VTL and SPU-VTL taking advantage of the same parameters as those employed in Figure 9a. In the configurations depicted in Figure 9a,b, the reduction in the number of multiplications resulting from the insertion of the SPU technique averaged 7.5%. Further, a detailed examination of Figure 9b reveals that in terms of energy efficiency, the application of tap length measurements by MSE calculation over up to 10^4 iterations proved to be more effective during the transient regime. Moreover, in the steady-state regime, both algorithms demonstrated comparable performance.
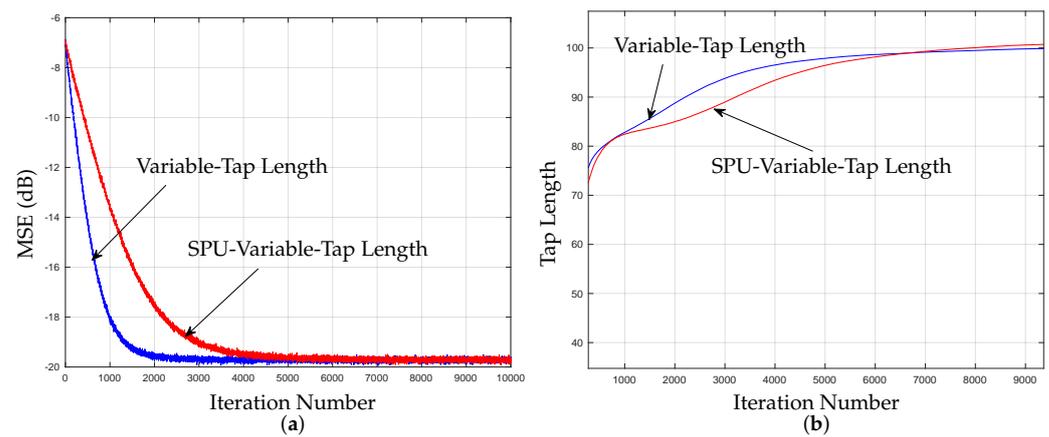
Figure 9. Comparison between empirical SPU-Variable Tap Length (red) and Variable Tap Length (blue). (**a**) MSE (dB); (**b**) Tap Length.

Figure 10a,b present comparisons, measured in terms of Mean Squared Error (MSE), between traditional LMS and SPU algorithms applied to a practical scenario of Acoustic Echo Cancellation (AEC). In both instances, the parameters utilized included 20 coefficients representing the ideal transfer function, with SPU employing only 18 coefficients. The empirical results stem from 2.10^4 independent Monte Carlo trials, incorporating white Gaussian noise as the additive element. The input signal is generated by passing unity-variance white Gaussian noise through the filter $H(z) = 1 - 0.2z^{-1} + z^{-2}$. The coefficients of the ideal plant are described by , defined as $h_k = [\cos(0.2 \times \pi \times (k+1))]^{-0.2 \times (k+1)}$. The parameters applied are $\beta = 10^{-2}$, $\sigma_v^2 = 10^{-6}$ for Figure 10a and $\sigma_v^2 = 10^{-2}$ for Figure 10b. In the settings depicted in Figure 10a,b, the computational cost reduction provided by the SPU-FX-LMS algorithm amounts to approximately 8%. Greater computational cost reductions can be achieved if the constraints of the application permit, given that such reductions entail a larger decrease in the convergence rate.

While both figures emphasize the superior convergence rate of FX-LMS during the transient regime, and in the steady-state regime where both FX-LMS and SPU-FX-LMS exhibit similarity, it is crucial to underscore that the implementation of SPU-FX-LMS concurrently reduces computational complexity while maintaining an equivalent performance during steady-state operation. Therefore, in the context of a Mean Squared Error (MSE) comparison, there is an important choice between FX-LMS and SPU-FX-LMS that becomes pivotal for designers with specific objectives in mind. Figure 10b illustrates similar comparisons of empirical MSE over iterations for FX-LMS and SPU-FX-LMS, leveraging the same parameters as in Figure 10a.

The theoretical framework expounded in the antecedent Figure 10a suggests a diminished computational demand. Nevertheless, a meticulous scrutiny of Figure 10b elucidates that, particularly in the presence of intensified noise, the augmentation of σ_v^2 values in the MSE calculations over a span of up to 10^4 iterations proves to be notably efficacious when employing SPU-FX-LMS. This effectiveness arises from the algorithm's capacity to optimize performance under conditions of high noise while concurrently exhibiting reduced computational complexity. The observed parity in steady-state results between SPU-FX-LMS and FX-LMS further underscores the former's viability. Consequently, the strategic application of SPU-FX-LMS is advocated in scenarios characterized by constrained computational resources, even in the presence of substantial ambient noise. Notably, in applications such as Acoustic Echo Cancellation (AEC), the SPU strategy demonstrates potential benefits, including potential reductions in battery usage for hands-free devices, while maintaining robust echo cancellation capabilities in comparison to conventional LMS techniques.
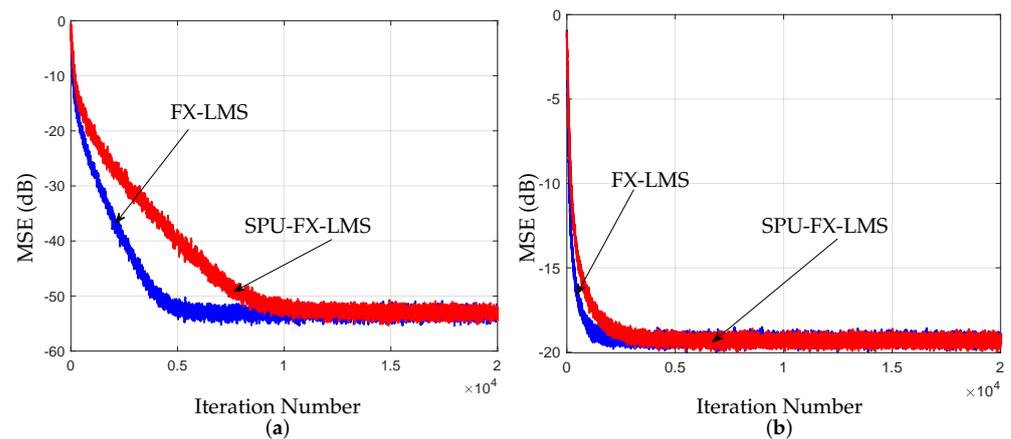
Figure 10. Comparison between empirical SPU-FX-LMS (red) and FX-LMS (blue) for different intense noise scenarios. (**a**) $\sigma_v^2 = 10^{-6}$; (**b**) $\sigma_v^2 = 10^{-2}$.

6. Conclusions

This paper proposes a deterministic and local optimization problem, whose approximate solution puts forward a non-normalized adaptive algorithm with selective partial updates. Such an algorithm provides the designer with the ability to operate in a region of a trade-off between computational complexity and convergence rate. A stochastic model that predicts the learning capabilities of the new algorithm is derived and then extended to address first-order Markovian perturbations of the ideal plant in an identification scenario. Theoretical predictions are shown to be in close agreement with computer simulation results. The proposed methodology has been extended to configurations of automatic noise cancellation and time-varying filter length, providing the designer with the possibility to operate at various points where the convergence rate and computational complexity vary. Extensions to adaptive algorithms that exhibit higher computational cost (such as RLS [29]) and to the double-talk scenario constitute a promising line of future investigations.

Author Contributions: Conceptualization, N.N.S. and L.C.R.; Methodology, N.N.S. and L.C.R.; Software, N.N.S. and L.C.R.; Validation, L.C.R.; Writing–original draft, N.N.S. and L.C.R.; Writing–review & editing, F.A.A.A., R.M.S.P., D.B.H. and M.R.P.; Visualization, N.N.S.; Supervision, F.A.A.A., R.M.S.P., D.B.H. and M.R.P.; Project administration, D.B.H. and M.R.P.; Funding acquisition, F.A.A.A. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article , further inquiries can be directed to the corresponding author/s.

References

1. Abadi, M.; Mehrdad, V.; Husoy, J. Combining Selective Partial Update and Selective Regressor Approaches for Affine Projection Adaptive Filtering. In Proceedings of the 2009 7th International Conference on Information, Communications and Signal Processing (ICICS), Macau, China, 8–10 December 2009; pp. 1–4.
2. Sayed, A.H. *Adaptive Filters*, 1st ed.; Wiley-IEEE Press: Newark, NJ, USA, 2008.
3. Wittenmark, B. *Adaptive Filter Theory: Simon Haykin*; Automatica, Elsevier Ltd: Amsterdam, The Netherlands, 1993; pp. 567–568.
4. Lara, P.; Igreja, F.; Tarrataca, L.D.T.J.; Barreto Haddad, D.; Petraglia, M.R. Exact Expectation Evaluation and Design of Variable Step-Size Adaptive Algorithms. *IEEE Signal Process. Lett.* **2019**, *26*, 74–78. [CrossRef]

5. Aboulnasr, T.; Mayyas, K. Complexity reduction of the NLMS algorithm via selective coefficient update. *IEEE Trans. Signal Process.* **1999**, *47*, 1421–1424. [CrossRef]

6. De Souza, J.V.G.; Henriques, F.D.R.; Siqueira, N.N.; Tarrataca, L.; Andrade, F.A.A.; Haddad, D.B. Stochastic Modeling of the Set-Membership- Sign-NLMS Algorithm. *IEEE Access* **2024**, *12*, 32739–32752. [CrossRef]

7. Wang, W.; Dogancay, K. Convergence Issues in Sequential Partial-Update LMS for Cyclostationary White Gaussian Input Signals. *IEEE Signal Process. Lett.* **2021**, *28*, 967–971. [CrossRef]

8. Wen, H.X.; Yang, S.Q.; Hong, Y.Q.; Luo, H. A Partial Update Adaptive Algorithm for Sparse System Identification. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2020**, *28*, 240–255. [CrossRef]

9. Wang, W.; Doğançay, K. Partial-update strictly linear, semi-widely linear, and widely linear geometric-algebra adaptive filters. *Signal Process.* **2023**, *210*, 109059. [CrossRef]

10. Dogancay, K.; Tanrikulu, O. Adaptive filtering algorithms with selective partial updates. *IEEE Trans. Circuits Syst. Analog. Digit. Signal Process.* **2001**, *48*, 762–769. [CrossRef]

11. Akhtar, M.T.; Ahmed, S. A robust normalized variable tap-length normalized fractional LMS algorithm. In Proceedings of the 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS), Abu Dhabi, United Arab Emirates, 16–19 October 2016; pp. 1–4. [CrossRef]

12. Gong, Y.; Cowan, C. An LMS style variable tap-length algorithm for structure adaptation. *IEEE Trans. Signal Process.* **2005**, *53*, 2400–2407. [CrossRef]

13. Li, N.; Zhang, Y.; Zhao, Y.; Hao, Y. An improved variable tap-length LMS algorithm. *Signal Process.* **2009**, *89*, 908–912. [CrossRef]

14. Zhang, Y.; Chambers, J. Convex Combination of Adaptive Filters for a Variable Tap-Length LMS Algorithm. *IEEE Signal Process. Lett.* **2006**, *13*, 628–631. [CrossRef]

15. Chang, D.C.; Chu, F.T. Feedforward Active Noise Control With a New Variable Tap-Length and Step-Size Filtered-X LMS Algorithm. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2014**, *22*, 542–555. [CrossRef]

16. Kar, A.; Swamy, M. Tap-length optimization of adaptive filters used in stereophonic acoustic echo cancellation. *Signal Process.* **2017**, *131*, 422–433. [CrossRef]

17. Zhang, Y.; Li, N.; Chambers, J.; Sayed, A. Steady-State Performance Analysis of a Variable Tap-Length LMS Algorithm. *IEEE Trans. Signal Process.* **2008**, *56*, 839–845. [CrossRef]

18. Pitas, I. Fast algorithms for running ordering and max/min calculation. *IEEE Trans. Circuits Syst.* **1989**, *36*, 795–804. [CrossRef]

19. Boudiaf, M.; Benkherrat, M.; Boudiaf, M. Partial-update adaptive filters for event-related potentials denoising. In Proceedings of the IET 3rd International Conference on Intelligent Signal Processing (ISP 2017), London, UK, 4–5 December 2017; pp. 1–6.

20. Lara, P.; Tarrataca, L.D.; Haddad, D.B. Exact expectation analysis of the deficient-length LMS algorithm. *Signal Process.* **2019**, *162*, 54–64. [CrossRef]

21. Silva, M.T.M.; Nascimento, V.H. Convex Combination of Adaptive Filters with Different Tracking Capabilities. In Proceedings of the 2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP '07, Honolulu, HI, USA, 15–20 April 2007; Volume 3, pp. III–925–III–928.

22. Ibe, O.C. Basic Concepts in Probability. In *Markov Processes for Stochastic Modeling*; Elsevier: Amsterdam, The Netherlands, 2013; pp. 1–27.

23. Bershad, N.J.; Bermudez, J.C.M. Stochastic analysis of the LMS algorithm for non-stationary white Gaussian inputs. In Proceedings of the 2011 IEEE Statistical Signal Processing Workshop (SSP), Nice, France, 28–30 June 2011; pp. 57–60.

24. Mayyas, K. Performance analysis of the deficient length LMS adaptive algorithm. *IEEE Trans. Signal Process.* **2005**, *53*, 2727–2734. [CrossRef]

25. Le, D.C.; Viet, H.H. Filtered-x Set Membership Algorithm With Time-Varying Error Bound for Nonlinear Active Noise Control. *IEEE Access* **2022**, *10*, 90079–90091. [CrossRef]

26. Mossi, M.I.; Yemdji, C.; Evans, N.; Beaugeant, C. A Comparative Assessment of Noise and Non-Linear EchoEffects in Acoustic Echo Cancellation. In Proceedings of the IEEE 10th International Conference on Signal Processing Proceedings, Beijing, China, 24–28 October 2010; pp. 223–226.

27. ITU-T-2004. *Digital Network Echo Cancellers (Recommendation)*; Technical Report G.168, ITU-T: Geneva, Switzerland, 2004.

28. ITU-T-2015. *Digital Network Echo Cancellers (Recommendation)*; Technical Report G.168, ITU-T: Geneva, Switzerland, 2015.

29. Sadigh, A.N.; Zayyani, H.; Korki, M. A Robust Proportionate Graph Recursive Least Squares Algorithm for Adaptive Graph Signal Recovery. In *IEEE Transactions on Circuits and Systems II: Express Briefs*; IEEE: Piscataway, NJ, USA, 2024; p. 1. [CrossRef]