Hindawi Publishing Corporation
EURASIP Journal on Audio, Speech, and Music Processing
Volume 2007, Article ID 34242, 10 pages
doi:10.1155/2007/34242

Research Article

Set-Membership Proportionate Affine Projection Algorithms

Stefan Werner,[1] José A. Apolinário, Jr.,[2] and Paulo S. R. Diniz[3]

[1] *Signal Processing Laboratory, Helsinki University of Technology, Otakaari 5A, 02150 Espoo, Finland*
[2] *Department of Electrical Engineering, Instituto Militar de Engenharia, 2229-270 Rio de Janeiro, Brazil*
[3] *Signal Processing Laboratory, COPPE/Poli/Universidade Federal do Rio de Janeiro, 21945-970 Rio de Janeiro, Brazil*

Received 30 June 2006; Revised 15 November 2006; Accepted 15 November 2006

Recommended by Kutluyil Dogancay

Proportionate adaptive filters can improve the convergence speed for the identification of sparse systems as compared to their conventional counterparts. In this paper, the idea of proportionate adaptation is combined with the framework of set-membership filtering (SMF) in an attempt to derive novel computationally efficient algorithms. The resulting algorithms attain an attractive faster converge for both situations of sparse and dispersive channels while decreasing the average computational complexity due to the data discerning feature of the SMF approach. In addition, we propose a rule that allows us to automatically adjust the number of past data pairs employed in the update. This leads to a set-membership proportionate affine projection algorithm (SM-PAPA) having a variable data-reuse factor allowing a significant reduction in the overall complexity when compared with a fixed data-reuse factor. Reduced-complexity implementations of the proposed algorithms are also considered that reduce the dimensions of the matrix inversions involved in the update. Simulations show good results in terms of reduced number of updates, speed of convergence, and final mean-squared error.

1. INTRODUCTION

Frequently used adaptive filtering algorithms like the least mean square (LMS) and the normalized LMS (NLMS) algorithms share the features of low computational complexity and proven robustness. The LMS and the NLMS algorithms have in common that the adaptive filter is updated in the direction of the input vector without favoring any particular direction. In other words, they are well suited for *dispersive*-type systems where the energy is uniformly distributed among the coefficients in the impulse response. On the other hand, if the system to be identified is *sparse*, that is, the impulse response is characterized by a few dominant coefficients (see [1] for a definition of a measure of sparsity), using different step sizes for each adaptive filter coefficient can improve the initial convergence of the NLMS algorithm. This basic concept is explored in *proportionate adaptive filters* [2–10], which incorporates the importance of the individual components by assigning weights proportional to the magnitude of the coefficients.

The conventional proportionate NLMS (PNLMS) algorithm [2] experiences fast initial adaptation for the dominant coefficients followed by a slower second transient for the remaining coefficients. Therefore, the slow convergence of the PNLMS algorithm after the initial transient can be circumvented by switching to the NLMS algorithm [11].

Another problem related to the conventional PNLMS algorithm is the poor performance in dispersive or semi-dispersive channels [3]. Refinements of the PNLMS have been proposed [3, 4] to improve performance in a dispersive medium and to combat the slowdown after the initial adaptation. The PNLMS++ algorithm in [3] approaches the problem by alternating the NLMS update with a PNLMS update. The improved PNLMS (IPNLMS) algorithm [4] combines the NLMS and PNLMS algorithms into one single updating expression. The main idea of the IPNLMS algorithm was to establish a rule for automatically switching from one algorithm to the other. It was further shown in [6] that the IPNLMS algorithm is a good approximation of the exponentiated gradient algorithm [1, 12]. Extension of the proportionate adaptation concept to affine projection (AP) type algorithms, proportionate affine projection (PAP) algorithms, can be found in [13, 14].

Using the PNLMS algorithm instead of the NLMS algorithm leads to 50% increase in the computational complexity. An efficient approach to reduce computations is to employ set-membership filtering (SMF) techniques [15, 16], where the filter is designed such that the output estimation error is upper bounded by a predetermined threshold.[1] Set-membership adaptive filters (SMAF) feature data-selective (sparse in time) updating, and a time-varying data-dependent step size that provides fast convergence as well as low steady-state error. SMAFs with low computational complexity per update are the set-membership NLMS (SM-NLMS) [15], the set-membership binormalized data-reusing (SM-BNDRLMS) [16], and the set-membership affine projection (SM-AP) [17] algorithms. In the following, we combine the frameworks of proportionate adaptation and SMF. A set-membership proportionate NLMS (SM-PNLMS) algorithm is proposed as a viable alternative to the SM-NLMS algorithm [15] for operation in sparse scenarios. Following the ideas of the IPNLMS algorithm, an efficient weight-scaling assignment is proposed that utilizes the information provided by the data-dependent step size. Thereafter, we propose a more general algorithm, the set-membership proportionate affine projection algorithm (SM-PAPA) that generalizes the ideas of the SM-PNLMS to reuse constraint sets from a fixed number of past input and desired signal pairs in the same way as the SM-AP algorithm [17]. The resulting algorithm can be seen as a set-membership version of the PAP algorithm [13, 14] with an optimized step size. As with the PAP algorithm, a faster convergence of the SM-PAPA algorithm may come at the expense of a slight increase in the computational complexity per update that is directly linked to the amount of reuses employed, or *data-reuse factor*. To lower the overall complexity, we propose to use a time-varying data-reuse factor. The introduction of the variable data-reuse factor results in an algorithm that close to convergence takes the form of the simple SM-PNLMS algorithm. Thereafter, we consider an efficient implementation of the new SM-PAPA algorithm that reduces the dimensions of the matrices involved in the update.

The paper is organized as follows. Section 2 reviews the concept of SMF while the SM-PNLMS algorithm is proposed in Section 3. Section 4 derives the general SM-PAPA algorithm where both cases of fixed and time-varying data-reuse factor are studied. Section 5 provides the details of an SM-PAPA implementation using reduced matrix dimensions. In Section 6, the performances of the proposed algorithms are evaluated through simulations which are followed by conclusions.

2. SET-MEMBERSHIP FILTERING

This section reviews the basic concepts of set-membership filtering (SMF). For a more detailed introduction to the concept of SMF, the reader is referred to [18]. Set-membership filtering is a framework applicable to filtering problems that are linear in parameters.[2] A specification on the filter parameters $\mathbf{w} \in \mathbb{C}^N$ is achieved by constraining the magnitude of the output estimation error, $e(k) = d(k) - \mathbf{w}^H\mathbf{x}(k)$, to be smaller than a deterministic threshold γ, where $\mathbf{x}(k) \in \mathbb{C}^N$ and $d(k) \in \mathbb{C}$ denote the input vector and the desired output signal, respectively. As a result of the bounded error constraint, there will exist a set of filters rather than a single estimate.

Let $\mathcal{S}$ denote the set of all possible input-desired data pairs $(\mathbf{x}, d)$ of interest. Let Θ denote the set of all possible vectors $\mathbf{w}$ that result in an output error bounded by γ whenever $(\mathbf{x}, d) \in \mathcal{S}$. The set Θ referred to as the *feasibility set* is given by

$$\Theta = \bigcap_{(\mathbf{x},d)\in\mathcal{S}} \{\mathbf{w} \in \mathbb{C}^N : |d - \mathbf{w}^H\mathbf{x}| \le \gamma\}. \tag{1}$$

Adaptive SMF algorithms seek solutions that belong to the *exact membership set* $\psi(k)$ constructed by input-signal and desired-signal pairs,

$$\psi(k) = \bigcap_{i=1}^{k} \mathcal{H}(i), \tag{2}$$

where $\mathcal{H}(k)$ is referred to as the *constraint set* containing all vectors $\mathbf{w}$ for which the associated output error at time instant k is upper bounded in magnitude by γ:

$$\mathcal{H}(k) = \{\mathbf{w} \in \mathbb{C}^N : |d(k) - \mathbf{w}^H\mathbf{x}(k)| \le \gamma\}. \tag{3}$$

It can be seen that the *feasibility set* Θ is a subset of the *exact membership set* ψ_k at any given time instant. The *feasibility set* is also the *limiting set* of the *exact membership set*, that is, the two sets will be equal if the training signal traverses all signal pairs belonging to $\mathcal{S}$. The idea of set-membership adaptive filters (SMAF) is to find adaptively an estimate that belongs to the feasibility set or to one of its members. Since $\psi(k)$ in (2) is not easily computed, one approach is to apply one of the many optimal bounding ellipsoid (OBE) algorithms [18, 20–24], which tries to approximate the exact membership set $\psi(k)$ by tightly outer bounding it with ellipsoids. Adaptive approaches leading to algorithms with low peak complexity, $\mathcal{O}(N)$, compute a point estimate through projections using information provided by past constraint sets [15–17, 25–27]. In this paper, we are interested in algorithms derived from the latter approach.

3. THE SET-MEMBERSHIP PROPORTIONATE NLMS ALGORITHM

In this section, the idea of proportionate adaptation is applied to SMF in order to derive a data-selective algorithm, the set-membership proportionate normalized LMS (SM-PNLMS), suitable for sparse environments.

3.1. Algorithm derivation

The SM-PNLMS algorithm uses the information provided by the constraint set $\mathcal{H}(k)$ and the coefficient updating to solve the optimization problem employing the criterion

$$\mathbf{w}(k+1) = \arg\min_{\mathbf{w}} \|\mathbf{w} - \mathbf{w}(k)\|^2_{\mathbf{G}^{-1}(k)} \quad \text{subject to: } \mathbf{w} \in \mathcal{H}(k), \tag{4}$$

[1] For other reduced-complexity solutions, see, for example, [11] where the concept of partial updating is applied.

[2] This includes nonlinear problems like Volterra modeling, see, for example, [19].

where the norm employed is defined as $\|\mathbf{b}\|_{\mathbf{A}}^2 = \mathbf{b}^{\mathrm{H}}\mathbf{A}\mathbf{b}$. Matrix $\mathbf{G}(k)$ is here chosen as a diagonal weighting matrix of the form

$$\mathbf{G}(k) = \operatorname{diag}\{g_1(k), \ldots, g_N(k)\}. \tag{5}$$

The elements values of $\mathbf{G}(k)$ will be discussed in Section 3.2. The optimization criterion in (4) states that if the previous estimate already belongs to the constraint set, $\mathbf{w}(k) \in \mathcal{H}(k)$, it is a feasible solution and no update is needed. However, if $\mathbf{w}(k) \notin \mathcal{H}(k)$, an update is required. Following the principle of *minimal disturbance*, a feasible update is made such that $\mathbf{w}(k+1)$ lies up on the nearest boundary of $\mathcal{H}(k)$. In this case the updating equation is given by

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \alpha(k)\frac{e^*(k)\mathbf{G}(k)\mathbf{x}(k)}{\mathbf{x}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{x}(k)}, \tag{6}$$

where

$$\alpha(k) = \begin{cases} 1 - \dfrac{\gamma}{|e(k)|} & \text{if } |e(k)| > \gamma \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

is a time-varying data-dependent step size, and $e(k)$ is the a priori error given by

$$e(k) = d(k) - \mathbf{w}^{\mathrm{H}}(k)\mathbf{x}(k). \tag{8}$$

For the proportionate algorithms considered in this paper, matrix $\mathbf{G}(k)$ will be diagonal. However, for other choices of $\mathbf{G}(k)$, it is possible to identify from (6) different types of SMAF available in literature. For example, choosing $\mathbf{G}(k) = \mathbf{I}$ gives the SM-NLMS algorithm [15], setting $\mathbf{G}(k)$ equal to a weighted covariance matrix will result in the BEACON recursions [28], and choosing $\mathbf{G}(k)$ such that it extracts the $P \leq N$ elements in $\mathbf{x}(k)$ of largest magnitude gives a partial-updating SMF [26]. Next we consider the weighting matrix used with the SM-PNLMS algorithm.

3.2. Choice of weighting matrix $\mathbf{G}(k)$

This section proposes a weighting matrix $\mathbf{G}(k)$ suitable for operation in sparse environments.

Following the same line of thought as in the IPNLMS algorithm, the diagonal elements of $\mathbf{G}(k)$ are computed to provide a good balance between the SM-NLMS algorithm and a solution for sparse systems. The goal is to reduce the length of the initial transient for estimating the dominant peaks in the impulse response and, thereafter, to emphasize the input-signal direction to avoid a slow second transient. Furthermore, the solution should not be sensitive to the assumption of a sparse system. From the expression for $\alpha(k)$ in (7), we observe that, if the solution is far from the constraint set, we have $\alpha(k) \to 1$, whereas close to the steady state $\alpha(k) \to 0$. Therefore, a suitable weight assignment rule emphasizes dominant peaks when $\alpha(k) \to 1$ and the input-signal direction (SM-PNLMS update) when $\alpha(k) \to 0$. As $\alpha(k)$ is a good indicator of how close a steady-state solution is, we propose to use

$$g_i(k) = \frac{1 - \kappa\alpha(k)}{N} + \frac{\kappa\alpha(k)|w_i(k)|}{\|\mathbf{w}(k)\|_1}, \tag{9}$$

where $\kappa \in [0, 1]$ and $\|\mathbf{w}(k)\|_1 = \sum_i |w_i(k)|$ denotes the l_1 norm [2, 4]. The constant κ is included to increase the robustness for estimation errors in $\mathbf{w}(k)$, and from the simulations provided in Section 6, $\kappa = 0.5$ shows good performance for both sparse and dispersive systems. For $\kappa = 1$, the algorithm will converge faster but will be more sensitive to the sparse assumption. The IPNLMS algorithm uses similar strategy, see [4] for details. The updating expressions in (9) and (6) resemble those of the IPNLMS algorithm except for the time-varying step size $\alpha(k)$. From (9) we can observe the following: (1) during initial adaptation (i.e., during transient) the solution is far from the steady-state solution, and consequently $\alpha(k)$ is large, and more weight will be placed at the stronger components of the adaptive filter impulse response; (2) as the error decreases, $\alpha(k)$ gets smaller, all the coefficients become equally important, and the algorithm behaves as the SM-NLMS algorithm.

4. THE SET-MEMBERSHIP PROPORTIONATE AFFINE-PROJECTION ALGORITHM

In this section, we extend the results from the previous section to derive an algorithm that utilizes the $L(k)$ most recent constraint sets $\{\mathcal{H}(i)\}_{i=k-L(k)+1}^{k}$. The algorithm derivation will treat the most general case where $L(k)$ is allowed to vary from one updating instant to another, that is, the case of a variable data reuse factor. Thereafter, we provide algorithm implementations for the case of fixed number of data-reuses (i.e., $L(k) = L$), and the case of $L(k) \leq L_{\max}$ (i.e., $L(k)$ is upper bounded but allowed to vary). The proposed algorithm, SM-PAPA, includes the SM-AP algorithm [17, 29] as a special case and is particularly useful whenever the input signal is highly correlated. As with the SM-PNLMS algorithm, the main idea is to allocate different weights to the filter coefficients using a weighting matrix $\mathbf{G}(k)$.

4.1. General algorithm derivation

The SM-PAPA is derived so that its coefficient vector after updating belongs to the set $\psi^{L(k)}(k)$ corresponding to the intersection of $L(k) < N$ past constraint sets, that is,

$$\psi^{L(k)}(k) = \bigcap_{i=k-L(k)+1}^{k} \mathcal{H}(i). \tag{10}$$

The number of data-reuses $L(k)$ employed at time instant k is allowed to vary with time. If the previous estimate belongs to the $L(k)$ past constraint sets, that is, $\mathbf{w}(k) \in \psi^{L(k)}(k)$, no coefficient update is required. Otherwise, the SM-PAPA performs an update according to the following optimization criterion:

$$\begin{aligned} \mathbf{w}(k+1) &= \arg\min_{\mathbf{w}} \|\mathbf{w} - \mathbf{w}(k)\|_{\mathbf{G}^{-1}(k)}^2 \\ &\text{subject to: } \mathbf{d}(k) - \mathbf{X}^{\mathrm{T}}(k)\mathbf{w}^* = \mathbf{p}(k), \end{aligned} \tag{11}$$

where vector $\mathbf{d}(k) \in \mathbb{C}^{L(k)}$ contains the desired outputs related to the $L(k)$ last time instants, vector $\mathbf{p}(k) \in \mathbb{C}^{L(k)}$ has components that obey $|p_i(k)| < \gamma$ and so specify a point in $\psi^{L(k)}(k)$, and matrix $\mathbf{X}(k) \in \mathbb{C}^{N\times L(k)}$ contains the corresponding input vectors, that is,

$$\begin{aligned}\mathbf{p}(k) &= \left[p_1(k)p_2(k)\cdots p_{L(k)}(k)\right]^{\mathrm{T}},\\ \mathbf{d}(k) &= \left[d(k)d(k-1)\cdots d(k-L(k)+1)\right]^{\mathrm{T}},\\ \mathbf{X}(k) &= \left[\mathbf{x}(k)\mathbf{x}(k-1)\cdots \mathbf{x}(k-L(k)+1)\right].\end{aligned} \tag{12}$$

Applying the method of Lagrange multipliers for solving the minimization problem of (11), the update equation of the most general SM-PAPA version is obtained as

$$\mathbf{w}(k+1) = \begin{cases}\mathbf{w}(k)+\mathbf{G}(k)\mathbf{X}(k)\left[\mathbf{X}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{X}(k)\right]^{-1} & \\ \quad\times\left[\mathbf{e}^*(k)-\mathbf{p}^*(k)\right], & \text{if } |e(k)|>\gamma\\ \mathbf{w}(k) & \text{otherwise,}\end{cases} \tag{13}$$

where $\mathbf{e}(k) = \mathbf{d}(k) - \mathbf{X}^{\mathrm{T}}(k)\mathbf{w}^*(k)$. The recursion above requires that matrix $\mathbf{X}^{\mathrm{H}}(k)\mathbf{X}(k)$, needed for solving the vector of Lagrange multipliers, is nonsingular. To avoid problems, a regularization factor can be included in the inverse (common in conventional AP algorithms), that is, $[\mathbf{X}^{\mathrm{H}}(k)\mathbf{X}(k)+\delta\mathbf{I}]^{-1}$ with $\delta \ll 1$. The choice of $p_i(k)$ can fit each problem at hand.

4.2. *SM-PAPA with fixed number of data reuses,* $L(k) = L$

Following the ideas of [17], a particularly simple SM-PAPA version is obtained if $p_i(k)$ for $i \neq 1$ corresponds to the a posteriori error $\epsilon(k-i+1) = d(k-i+1) - \mathbf{w}^{\mathrm{H}}(k)\mathbf{x}(k-i+1)$ and $p_1(k) = \gamma e(k)/|e(k)|$. The simplified SM-PAPA version has recursion given by

$$\begin{aligned}\mathbf{w}(k+1) = \mathbf{w}(k) &+ \mathbf{G}(k)\mathbf{X}(k)\\ &\times\left[\mathbf{X}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{X}(k)\right]^{-1}\alpha(k)e^*(k)\mathbf{u}_1,\end{aligned} \tag{14}$$

where $\mathbf{u}_1 = [10\cdots 0]^{\mathrm{T}}$ and $\alpha(k)$ is given by (7).

Due to the special solution involving the $L\times 1$ vector $\mathbf{u}_1$ in (14), a computationally efficient expression for the coefficient update is obtained by partitioning the input signal matrix as[3]

$$\mathbf{X}(k) = \left[\mathbf{x}(k)\mathbf{U}(k)\right], \tag{15}$$

where $\mathbf{U}(k) = [\mathbf{x}(k-1)\cdots\mathbf{x}(k-L+1)]$. Substituting the partitioned input matrix in (14) and carrying out the multiplications, we get after some algebraic manipulations (see [9])

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\alpha(k)e^*(k)}{\boldsymbol{\phi}^{\mathrm{H}}(k)\mathbf{G}(k)\boldsymbol{\phi}(k)}\mathbf{G}(k)\boldsymbol{\phi}(k), \tag{16}$$

SM-PAPA

for each k
{
$e(k) = d(k) - \mathbf{w}^{\mathrm{H}}(k)\mathbf{x}(k)$
if $\lvert e(k)\rvert > \gamma$
{
$\alpha(k) = 1 - \gamma/\lvert e(k)\rvert$
$g_i(k) = \frac{1-\kappa\alpha(k)}{N} + \frac{\kappa\alpha(k)\lvert w_i(k)\rvert}{\sum_{i=1}^{N}\lvert w_i(k)\rvert},\quad i = 1,\ldots,N$
$\mathbf{G}(k) = \mathtt{diag}[g_1(k)\cdots g_N(k)]$
$\mathbf{X}(k) = [\mathbf{x}(k)\mathbf{U}(k)]$
$\boldsymbol{\phi}(k) = \mathbf{x}(k) - \mathbf{U}(k)[\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{U}(k)]^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{x}(k)$
$\mathbf{w}(k+1) = \mathbf{w}(k) + \alpha(k)e^*(k)\frac{1}{\boldsymbol{\phi}^{\mathrm{H}}(k)\mathbf{G}(k)\boldsymbol{\phi}(k)}\mathbf{G}(k)\boldsymbol{\phi}(k)$
}
else
{
$\mathbf{w}(k+1) = \mathbf{w}(k)$
}
}

Algorithm 1: Set-membership proportionate affine-projection algorithm with a fixed number of data reuses.

where vector $\boldsymbol{\phi}(k)$ is defined as

$$\boldsymbol{\phi}(k) = \mathbf{x}(k) - \mathbf{U}(k)\left[\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{U}(k)\right]^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{x}(k). \tag{17}$$

This representation of the SM-PAPA is computationally attractive as the dimension of the matrix to be inverted is reduced from $L\times L$ to $(L-1)\times(L-1)$. As with the SM-PNLMS algorithms, $\mathbf{G}(k)$ is a diagonal matrix whose elements are computed according to (9). Algorithm 1 shows the recursions for the SM-PAPA.

The peak computational complexity of the SM-PAPA of Algorithm 1 is similar to that of the conventional PAP algorithm for the case of unity step size (such that the reduced dimension strategy can be employed). However, one important gain of using the SM-PAPA as well as any other SM algorithm, is the reduced number of computations for those time instants where no updates are required. The lower average complexity due to the sparse updating in time can provide substantial computational savings, that is, lower power consumption. Taking into account that the matrix inversion used in the proposed algorithm needs $\mathcal{O}([L-1]^3)$ complex operations and that $N \gg L$, the cost of the SM-PAPA is $\mathcal{O}(NL^2)$ operations *per update*. Furthermore, the variable data-reuse scheme used by the algorithm proposed in the following, the SM-REDPAPA, reduces even more the computational load by varying the complexity from the SM-PAPA to the SM-PNLMS.

[3] The same approach can be used to reduce the complexity of the Ozeki Umeda's AP algorithm for the case of unit step size [30].

4.3. SM-PAPA with variable data reuse

For the particular case when the data-reuse factor $L(k)$ is time varying, the simplified SM-PAPA version in (14) no longer guarantees that the *a posteriori* error is such that $|\epsilon(k-i+1)| \le \gamma$ for $i \ne 1$. This is the case, for example, when the number of data reuses is increased from one update instant to another, that is, $L(k) > L(k-1)$.

In order to provide an algorithm that belongs to the set $\psi^{L(k)}(k)$ in (10), we can choose the elements of vector $\mathbf{p}(k)$ to be

$$p_i(k) = \begin{cases} \gamma \dfrac{\epsilon(k-i+1)}{|\epsilon(k-i+1)|} & \text{if } |\epsilon(k-i+1)| > \gamma \\ \epsilon(k-i+1) & \text{otherwise} \end{cases} \tag{18}$$

for $i = 1, \dots, L(k)$ with $\epsilon(k) = e(k)$. With the above choice of $\mathbf{p}(k)$, the SM-AP recursions become

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) + \mathbf{G}(k)\mathbf{X}(k) \\ &\quad \times \left[\mathbf{X}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{X}(k)\right]^{-1}\mathbf{\Lambda}^*(k)\mathbf{1}_{L(k)\times 1}, \end{aligned} \tag{19}$$

where matrix $\mathbf{\Lambda}(k)$ is a diagonal matrix whose diagonal elements $[\mathbf{\Lambda}(k)]_{ii}$ are specified by

$$\begin{aligned} [\mathbf{\Lambda}(k)]_{ii} &= \alpha_i(k)\epsilon(k-i+1) \\ &= \begin{cases} \left(1 - \dfrac{\gamma}{|\epsilon(k-i+1)|}\right) & \\ \times \epsilon(k-i+1) & \text{if } |\epsilon(k-i+1)| > \gamma \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{20}$$

and $\mathbf{1}_{L(k)\times 1} = [1, \dots, 1]^{\mathrm{T}}$.

Another feature of the above algorithm is the possibility to correct previous solutions that for some reason did not satisfy the constraint $|\epsilon(k-i+1)| \le \gamma$ for $i \ne 1$. At this point $|\epsilon(k-i+1)| > \gamma$ for $i \ne 1$ could originate from a finite precision implementation or the introduction of a regularization parameter in the inverse in (19).

As can be seen from (20), the amount of zero entries can be significant if $L(k)$ is large. In Section 5, this fact is exploited in order to obtain a more computationally efficient version of the SM-AP algorithm. Next we consider how to assign a proper data-reuse factor at each time instant.

4.4. Variable data-reuse factor

This section proposes a rule for selecting the number of data-reuses $L(k)$ to be used at each coefficient update. It can be observed that the main difference in performance between the SM-PAPA and the SM-PNLMS algorithms is in the transient. Generally, the SM-PAPA algorithm has faster convergence than the SM-NLMS algorithm in colored environments. On the other hand, close to the steady state solution, their performances are comparable in terms of excess of MSE. Therefore, a suitable assignment rule increases the data-reuse factor when the solution is far from steady state and reduces to one when close to steady-state (i.e., the SM-PNLMS update).

Table 1: Quantization levels for $L_{\max} = 5$.

$L(k)$	Uniform quantizer	Using (24)
1	$\alpha_1(k) \le 0.2$	$\alpha_1(k) \le 0.2019$
2	$0.2 < \alpha_1(k) \le 0.4$	$0.2019 < \alpha_1(k) \le 0.3012$
3	$0.4 < \alpha_1(k) \le 0.6$	$0.3012 < \alpha_1(k) \le 0.4493$
4	$0.6 < \alpha_1(k) \le 0.8$	$0.4493 < \alpha_1(k) \le 0.6703$
5	$0.8 < \alpha_1(k) \le 1$	$0.6703 < \alpha_1(k) \le 1.0000$

As discussed previously, $\alpha_1(k)$ in (20) is a good indicator of how close to steady-state solution is. If $\alpha_1(k) \to 1$, the solution is far from the current constraint set which would suggest that the data-reuse factor $L(k)$ should be increased toward a predefined maximum value $L_{\max}$. If $\alpha_1(k) \to 0$, then $L(k)$ should approach one resulting in an SM-PNLMS update. Therefore, we propose to use a variable data-reuse factor of the form

$$L(k) = f[\alpha_1(k)], \tag{21}$$

where the function $f(\cdot)$ should satisfy $f(0) = 1$ and $f(1) = L_{\max}$ with $L_{\max}$ denoting the maximum number of data reuses allowed. In other words, the above expression should quantize $\alpha_1(k)$ into $L_{\max}$ regions

$$\mathcal{I}_p = \{l_{p-1} < \alpha_1(k) \le l_p\}, \quad p = 1, \dots, L_{\max} \tag{22}$$

defined by the decision levels l_p. The variable data-reuse factor is then given by the relation

$$L(k) = p \quad \text{if } \alpha_1(k) \in \mathcal{I}_p. \tag{23}$$

Indeed, there are many ways in which we could choose the decision variables l_p. In the simulations provided in Section 6, we consider two choices for l_p. The first approach consists of uniformly quantizing $\alpha_1(k)$ into $L_{\max}$ regions. The second approach is to use $l_p = e^{-\beta(P_{\max}-p)/P_{\max}}$ and $l_0 = 0$, where β is a positive constant [29]. This latter choice leads to a variable data-reuse factor on the form

$$L(k) = \max\left\{1, \left\lceil L_{\max}\left(\frac{1}{\beta}\ln\alpha_1(k) + 1\right)\right\rceil\right\}, \tag{24}$$

where the operator $\lceil(\cdot)\rceil$ rounds the element $(\cdot)$ to the nearest integer. Table 1 shows the resulting values of $\alpha_1(k)$ for both approaches in which the number of reuses should be changed for a maximum of five reuses, usually the most practical case. The values of the decision variables of the second approach provided in the table were calculated with the above expression using $\beta = 2$.

5. REDUCED COMPLEXITY VERSION OF THE VARIABLE DATA-REUSE ALGORITHM

This section presents an alternative implementation of the SM-PAPA in (19) that properly reduces the dimensions of the matrices in the recursions.

Assume that, at time instant k, the diagonal of $\mathbf{\Lambda}(k)$ specified by (20) has $P(k)$ nonzero entries (i.e., $L(k) - P(k)$ zero entries). Let $\mathbf{T}(k) \in \mathbb{R}^{L(k)\times L(k)}$ denote the permutation matrix that permutes the columns of $\mathbf{X}(k)$ such that the resulting input vectors corresponding to nonzero values in $\mathbf{\Lambda}(k)$ are shifted to the left, that is, we have

$$\overline{\mathbf{X}}(k) = \mathbf{X}(k)\mathbf{T}(k) = [\widetilde{\mathbf{X}}(k)\mathbf{U}(k)], \tag{25}$$

where matrices $\widetilde{\mathbf{X}}(k) \in \mathbb{C}^{N\times P(k)}$ and $\mathbf{U}(k) \in \mathbb{C}^{N\times[L(k)-P(k)]}$ contain the vectors giving nonzero and zero values on the diagonal of $\mathbf{\Lambda}(k)$, respectively. Matrix $\mathbf{T}(k)$ is constructed such that the column vectors of matrices $\widetilde{\mathbf{X}}(k)$ and $\mathbf{U}(k)$ are ordered according to their time index.

Using the relation $\mathbf{T}(k)\mathbf{T}^{\mathrm{T}}(k) = \mathbf{I}_{L(k)\times L(k)}$, we can rewrite the SM-PAPA recursion as

$$\begin{aligned}
&\mathbf{w}(k+1)\\
&\quad= \mathbf{w}(k) + \mathbf{G}(k)\mathbf{X}(k)\\
&\qquad\times[\mathbf{T}(k)\mathbf{T}^{\mathrm{T}}(k)\mathbf{X}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{X}(k)\mathbf{T}(k)\mathbf{T}^{\mathrm{T}}(k)]^{-1}\mathbf{\Lambda}^*(k)\mathbf{1}_{L(k)\times 1}\\
&\quad= \mathbf{w}(k) + \mathbf{G}(k)\mathbf{X}(k)\\
&\qquad\times[\mathbf{T}(k)\overline{\mathbf{X}}^{\mathrm{H}}(k)\mathbf{G}(k)\overline{\mathbf{X}}(k)\mathbf{T}^{\mathrm{T}}(k)]^{-1}\mathbf{\Lambda}^*(k)\mathbf{1}_{L(k)\times 1}\\
&\quad= \mathbf{w}(k) + \mathbf{G}(k)\overline{\mathbf{X}}(k)[\overline{\mathbf{X}}^{\mathrm{H}}(k)\mathbf{G}(k)\overline{\mathbf{X}}(k)]^{-1}\boldsymbol{\lambda}^*(k),
\end{aligned} \tag{26}$$

where vector $\boldsymbol{\lambda}(k) \in \mathbb{C}^{L(k)\times 1}$ contains the $P(k)$ nonzero adaptive step sizes of $\mathbf{\Lambda}(k)$ as the first elements (ordered in time) followed by $L(k) - P(k)$ zero entries, that is,

$$\boldsymbol{\lambda}(k) = \begin{bmatrix} \overline{\boldsymbol{\lambda}}(k) \\ \mathbf{0}_{[L(k)-P(k)]\times 1} \end{bmatrix}, \tag{27}$$

where the elements of $\overline{\boldsymbol{\lambda}}(k)$ are the $P(k)$ nonzero adaptive step sizes (ordered in time) of the form $\overline{\lambda}_i(k) = (1-\gamma/|\epsilon(k)|)\epsilon(k)$.

Due to the special solution involving $\boldsymbol{\lambda}(k)$ in (27), the following computationally efficient expression for the coefficient update is obtained using the partition in (25) (see the appendix)

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{G}(k)\mathbf{\Phi}(k)[\mathbf{\Phi}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{\Phi}(k)]^{-1}\overline{\boldsymbol{\lambda}}^*(k), \tag{28}$$

where matrix $\mathbf{\Phi}(k) \in \mathbb{C}^{N\times P(k)}$ is defined as

$$\mathbf{\Phi}(k) = \widetilde{\mathbf{X}}(k) - \mathbf{U}(k)[\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{U}(k)]^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\widetilde{\mathbf{X}}(k). \tag{29}$$

This representation of the SM-PAPA is computationally attractive as the dimension of the matrices involved is lower than that of the version described by (19)-(20). Algorithm 2 shows the recursion for the reduced-complexity SM-PAPA, where the $L(k)$ can be chosen as described in the previous section.

SM-REDPAPA

for each k
{
  $\epsilon(k) = d(k) - \mathbf{w}^{\mathrm{H}}(k)\mathbf{x}(k)$
  if $|\epsilon(k)| > \gamma$
  {
    $\widetilde{\mathbf{X}}(k) = [\mathbf{x}(k)]$; $\mathbf{U}(k) = []$; $\overline{\boldsymbol{\lambda}} = []$;
    $\alpha_1(k) = 1 - \gamma(k)/|\epsilon(k)|$
    $g_i(k) = \dfrac{1-\kappa\alpha_1(k)}{N} + \dfrac{\kappa\alpha(k)|w_i(k)|}{\sum_{i=1}^{N}|w_i(k)|}, \quad i = 1,\dots,N$
    $\mathbf{G}(k) = \mathtt{diag}[g_1(k)\cdots g_N(k)]$
    $L(k) = f[\alpha_1(k)]$
    for $i = 1$ to $L(k) - 1$
    {
      if $|\epsilon(k-i)| > \gamma$
      {
        $\widetilde{\mathbf{X}}(k) = [\widetilde{\mathbf{X}}(k)\mathbf{x}(k-i)]$ % Expand matrix
        $\overline{\boldsymbol{\lambda}}(k) = [\overline{\boldsymbol{\lambda}}^{\mathrm{T}}(k)\alpha_{i+1}(k)\epsilon(k-i)]^T$ % Expand vector
      }
      else
      {
        $\mathbf{U}(k) = [\mathbf{U}(k)\mathbf{x}(k-i)]$ % Expand matrix
      }
    $\mathbf{\Phi}(k) = \widetilde{\mathbf{X}}(k) - \mathbf{U}(k)[\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{U}(k)]^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\widetilde{\mathbf{X}}(k)$
    $\mathbf{w}(k+1) = \mathbf{w}(k) + \mathbf{G}(k)\mathbf{\Phi}(k)[\mathbf{\Phi}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{\Phi}(k)]^{-1}\overline{\boldsymbol{\lambda}}^*(k)$
  }
  else
  {
    $\mathbf{w}(k+1) = \mathbf{w}(k)$
  }
}

Algorithm 2: Reduced-complexity set-membership proportionate affine-projection algorithm with variable data reuse.

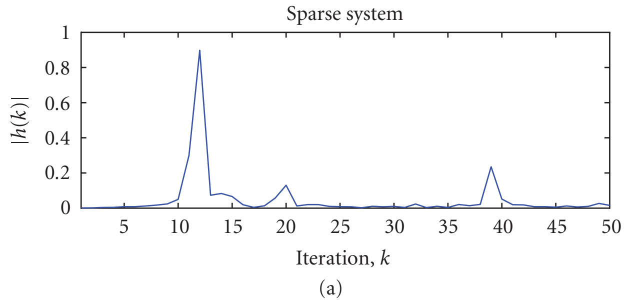
6. SIMULATION RESULTS

In this section, the performances of the SM-PNLMS algorithm and the SM-PAPA are evaluated in a system identification experiment. The performance of the NLMS, the IPNLMS, the SM-NLMS, and the SM-AP algorithms are also compared.

6.1. *Fixed number of data reuses*

The first experiment was carried out with an unknown plant with sparse impulse response that consisted of an $N = 50$ truncated FIR model from a digital microwave radio channel.[4] Thereafter, the algorithms were tested for a dispersive channel, where the plant was a complex FIR filter whose co-

[4] The coefficients of this complex-valued baseband channel model can be downloaded from http://spib.rice.edu/spib/microwave.html.



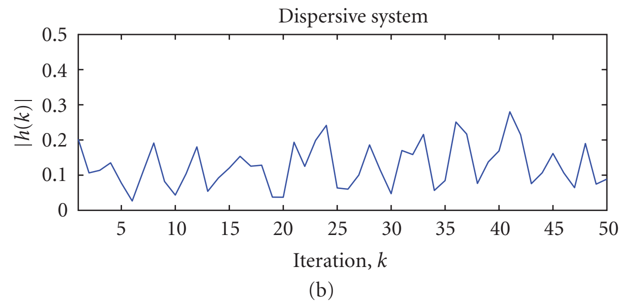


FIGURE 1: The amplitude of two impulse responses used in the simulations: (a) sparse microwave channel (see Footnote 4), (b) dispersive channel.

efficients were generated randomly. Figure 1 depicts the absolute values of the channel impulse responses used in the simulations. For the simulation experiments, we have used the following parameters: $\mu = 0.4$ for the NLMS and the PAP algorithms, $\gamma = \sqrt{2\sigma_n^2}$ for all SMAF, and $\kappa = 0.5$ for all proportionate algorithms. Note that for the IPNLMS and the PAP algorithms, $g_i(k) = (1-\kappa)/N + \kappa|w_i(k)|/\|\mathbf{w}(k)\|_{-1}$ corresponds to the same updating as in [4] when $\kappa \in [0,1]$. The parameters were set in order to have fair comparison in terms of final steady-state error. The input signal $x(k)$ was a complex-valued noise sequence, colored by filtering a zero-mean white complex-valued Gaussian noise sequence $n_x(k)$ through the fourth-order IIR filter $x(k) = n_x(k) + 0.95x(k-1) + 0.19x(k-2) + 0.09x(k-3) - 0.5x(k-4)$, and the SNR was set to 40 dB.

The learning curves shown in Figures 2 and 3 are the result of 500 independent runs and smoothed by a low pass filter. From the learning curves in Figure 2 for the sparse system, it can be seen that the SMF algorithms converge slightly faster than their conventional counterparts to the same level of MSE. In addition to the faster convergence, the SMF algorithms will have a reduced numbers of updates. In 20000 iterations, the number of times an update took place for the SM-PNLMS, the SM-PAPA, and the SM-AP algorithms were 7730 (39%), 6000 (30%), and 6330 (32%), respectively. This should be compared with 20000 updates required by the IPNLMS and PAP algorithms. From Figure 2, we also observe that the proportionate SMF algorithms converge faster than those without proportionate adaptation.

Figure 3 shows the learning curves for the dispersive channel identification, where it can be observed that the

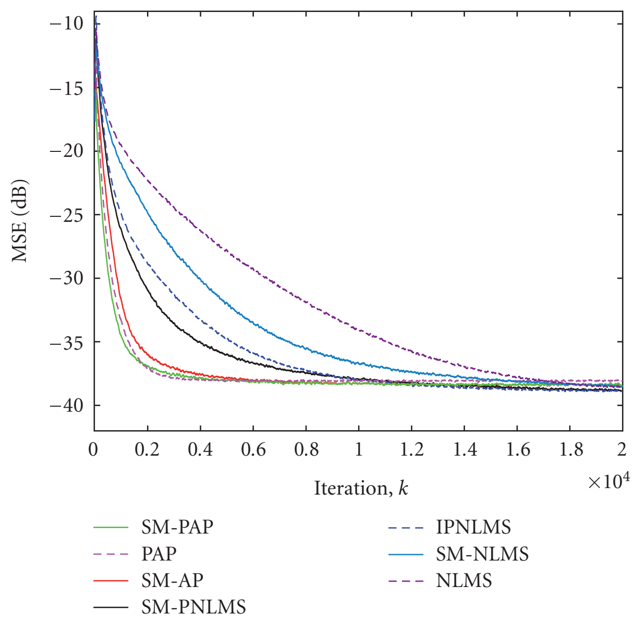


FIGURE 2: Learning curves in a *sparse system* for the SM-PNLMS, the SM-PAPA ($L = 2$), the SM-NLMS, the NLMS, the IPNLMS, and the PAP ($L = 2$) algorithms. SNR = 40 dB, $\gamma = \sqrt{2}\sigma_n$, and $\mu = 0.4$.

performances of the SM-PNLMS and SM-PAPA algorithms are very close to the SM-AP and SM-NLMS algorithms, respectively. In other words, the SM-PNLMS algorithm and the SM-PAPA are not sensitive to the assumption of having a sparse impulse response. In 20000 iterations, the SM-PAPA

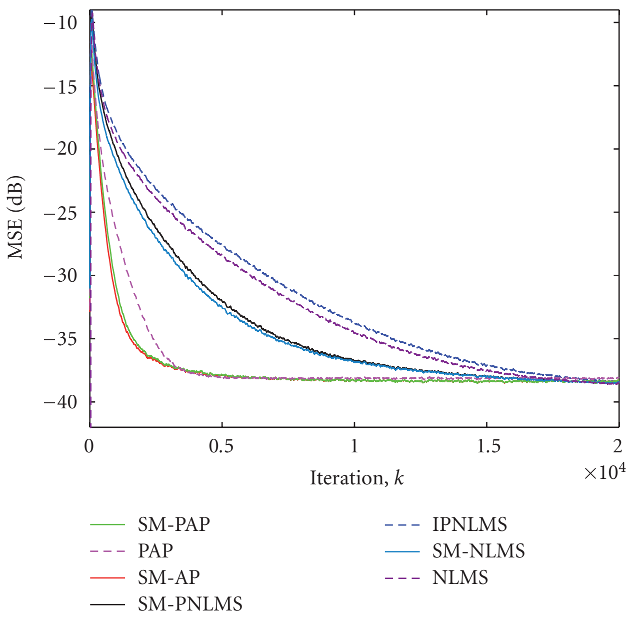


FIGURE 3: Learning curves in a *dispersive system* for the SM-PNLMS, the SM-PAPA ($L = 2$), the SM-NLMS, the NLMS, the IPNLMS, and the PAP ($L = 2$) algorithms. SNR = 40 dB, $\gamma = \sqrt{2}\sigma_n$, and $\mu = 0.4$.

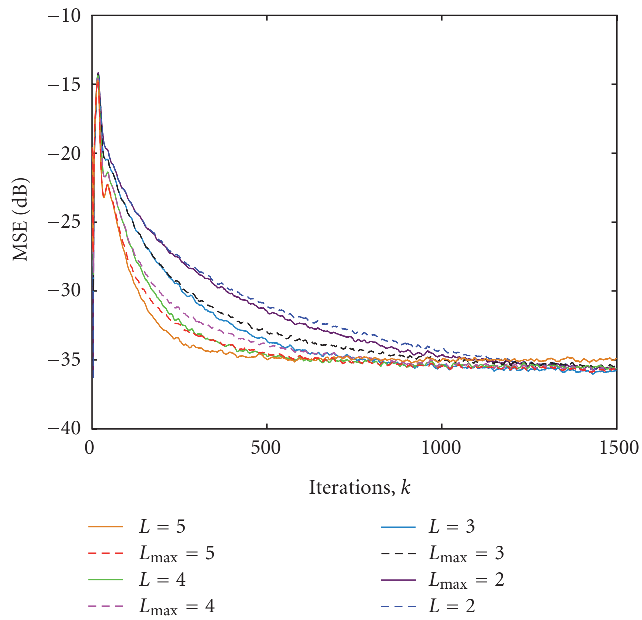


FIGURE 4: Learning curves in a *sparse system* for the SM-PAPA (L = 2 to 5), and the SM-REDPAPA (L_{max} = 2 to 5) based on a uniformly quantized $\alpha_1(k)$. SNR = 40 dB, $\gamma = \sqrt{2}\sigma_n$.

and the SM-PNLMS algorithms updated 32% and 50%, respectively, while the SM-AP and SM-NLMS algorithms updated 32% and 49%, respectively.

6.2. Variable data-reuse factor

The SM-PAPA algorithm with variable data-reuse factor was applied to the sparse system example of the previous section. Figures 4 and 5 show the learning curves averaged over 500 simulations for the SM-PAPA for $L = 2$ to $L = 5$, and SM-REDPAPA for $L_{max} = 2$ to $L_{max} = 5$. Figure 4 shows the results obtained with a uniformly quantized $\alpha_1(k)$, whereas Figure 5 shows the results obtained using (24) with $\beta = 2$. It can be seen that the SM-REDPAPA not only achieves a similar convergence speed, but is also able to reach a lower steady state using fewer updates. The approach of (24) performs slightly better than the one using a uniformly quantized $\alpha_1(k)$, which slows down during the second part of the transient. On the other hand, the latter approach has the advantage that no parameter tuning is required. Tables 2 and 3 show the number of data-reuses employed for each approach. As can be inferred from the tables, the use of variable data-reuse factor can significantly reduce the overall complexity as compared with the case of keeping it fixed.

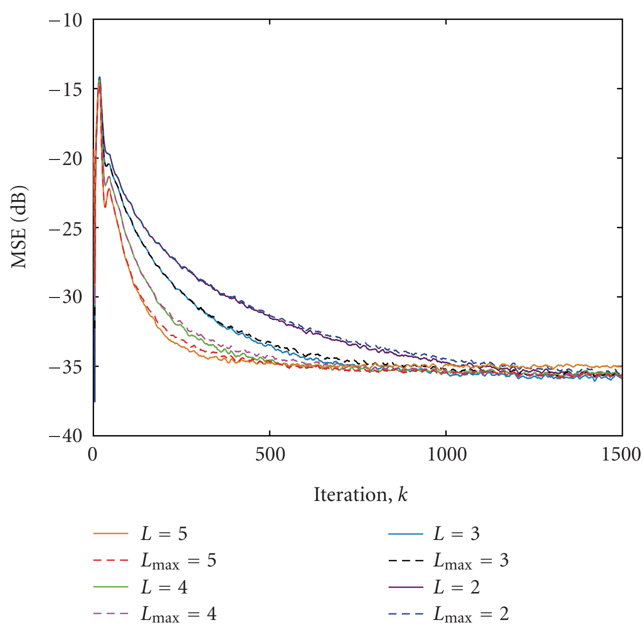


FIGURE 5: Learning curves in a *sparse system* for the SM-PAPA (L = 2 to 5), and the SM-REDPAPA (L_{max} = 2 to 5) based on (24). SNR = 40 dB, $\gamma = \sqrt{2}\sigma_n$.

7. CONCLUSIONS

This paper presented novel set-membership filtering (SMF) algorithms suitable for applications in sparse environments. The set-membership proportionate NLMS (SM-PNLMS) algorithm and the set-membership proportionate affine projection algorithm (SM-PAPA) were proposed as viable alternatives to the SM-NLMS and SM-AP algorithms. The algorithms benefit from the reduced average computational complexity from the SMF strategy and fast convergence for sparse scenarios resulting from proportionate updating. Simulations were presented for both sparse and dispersive impulse

TABLE 2: Distribution of the variable data-reuse factor $L(k)$ used in the SM-PAPA for the case when $\alpha_1(k)$ is *uniformly quantized*.

$L_{\max}$	$L(k) = 1$	$L(k) = 2$	$L(k) = 3$	$L(k) = 4$	$L(k) = 5$
1	100%	—	—	—	—
2	54.10%	45.90%	—	—	—
3	36.55%	45.80%	17.65%	—	—
4	28.80%	36.90%	26.55%	7.75%	—
5	23.95%	29.95%	28.45%	13.50%	4.15%

TABLE 3: Distribution of the variable data-reuse factor $L(k)$ used in the SM-PAPA for the case when $\alpha_1(k)$ is *quantized according to* (24), $\beta = 2$.

$L_{\max}$	$L(k) = 1$	$L(k) = 2$	$L(k) = 3$	$L(k) = 4$	$L(k) = 5$
1	100%	—	—	—	—
2	37.90%	62.90%	—	—	—
3	28.90%	35.45%	35.65%	—	—
4	28.86%	21.37%	33.51%	18.26%	—
5	25.71%	15.03%	23.53%	25.82%	9.91%

responses. It was verified that not only the proposed SMF algorithms can further reduce the computational complexity when compared with their conventional counterparts, the IPNLMS and PAP algorithms, but they also present faster convergence to the same level of MSE when compared with the SM-NLMS and the SM-AP algorithms. The weight assignment of the proposed algorithms utilizes the information provided by a time-varying step size typical for SMF algorithms and is robust to the assumption of sparse impulse response. In order to reduce the overall complexity of the SM-PAPA we proposed to employ a variable data reuse factor. The introduction of a variable data-reuse factor allows significant reduction in the overall complexity as compared to fixed data-reuse factor. Simulations showed that the proposed algorithm could outperform the SM-PAPA with fixed number of data-reuses in terms of computational complexity and final mean-squared error.

APPENDIX

The inverse in (26) can be partitioned as

$$\begin{aligned}\left[\overline{\mathbf{X}}^{\mathrm{H}}(k)\mathbf{G}(k)\overline{\mathbf{X}}(k)\right]^{-1} &= \left(\left[\widetilde{\mathbf{X}}(k)\mathbf{U}(k)\right]^{\mathrm{H}}\mathbf{G}(k)\left[\widetilde{\mathbf{X}}(k)\mathbf{U}(k)\right]\right)^{-1}\\ &= \begin{bmatrix}\mathbf{A} & \mathbf{B}^{\mathrm{H}}\\ \mathbf{B} & \mathbf{C}\end{bmatrix},\end{aligned} \tag{A.1}$$

where

$$\begin{aligned}\mathbf{A} &= \left[\mathbf{\Phi}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{\Phi}(k)\right]^{-1},\\ \mathbf{B} &= -\left[\mathbf{U}(k)^{\mathrm{H}}\mathbf{G}(k)\mathbf{U}(k)\right]^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\widetilde{\mathbf{X}}(k)\mathbf{A},\end{aligned} \tag{A.2}$$

with $\mathbf{\Phi}(k)$ defined as in (29). Therefore,

$$\begin{aligned}&\overline{\mathbf{X}}(k)\left[\overline{\mathbf{X}}^{\mathrm{H}}(k)\mathbf{G}(k)\overline{\mathbf{X}}(k)\right]^{-1}\boldsymbol{\lambda}^*(k)\\ &\quad= \overline{\mathbf{X}}(k)\begin{bmatrix}\mathbf{A}\\ \mathbf{B}\end{bmatrix}\overline{\boldsymbol{\lambda}}^*(k)\\ &\quad= \left[\widetilde{\mathbf{X}}(k) - \left\{\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{U}(k)\right\}^{-1}\mathbf{U}^{\mathrm{H}}(k)\mathbf{G}(k)\widetilde{\mathbf{X}}(k)\right]\\ &\qquad\times\left[\mathbf{\Phi}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{\Phi}(k)\right]^{-1}\boldsymbol{\lambda}^*(k)\\ &\quad= \mathbf{\Phi}(k)\left[\mathbf{\Phi}^{\mathrm{H}}(k)\mathbf{G}(k)\mathbf{\Phi}(k)\right]^{-1}\boldsymbol{\lambda}^*(k).\end{aligned} \tag{A.3}$$

ACKNOWLEDGMENTS

The authors would like to thank CAPES, CNPq, FAPERJ (Brazil), and Academy of Finland, Smart and Novel Radios (SMARAD) Center of Excellence (Finland), for partially supporting this work.

REFERENCES

[1] R. K. Martin, W. A. Sethares, R. C. Williamson, and C. R. Johnson Jr., "Exploiting sparsity in adaptive filters," *IEEE Transactions on Signal Processing*, vol. 50, no. 8, pp. 1883–1894, 2002.

[2] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, pp. 508–518, 2000.

[3] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," in *Proceedings of the 32nd Asilomar Conference on Signals, Systems & Computers*, vol. 1, pp. 394–398, Pacific Grove, Calif, USA, November 1998.

[4] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, vol. 2, pp. 1881–1884, Orlando, Fla, USA, May 2002.

[5] B. D. Rao and B. Song, "Adaptive filtering algorithms for promoting sparsity," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 6, pp. 361–364, Hong Kong, April 2003.

[6] A. W. H. Khong, J. Benesty, and P. A. Naylor, "An improved proportionate multi-delay block adaptive filter for packet-switched network echo cancellation," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.

[7] K. Doğançay and P. Naylor, "Recent advances in partial update and sparse adaptive filters," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.

[8] A. Deshpande and S. L. Grant, "A new multi-algorithm approach to sparse system adaptation," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.

[9] S. Werner, J. A. Apolinário Jr., P. S. R. Diniz, and T. I. Laakso, "A set-membership approach to normalized proportionate adaptation algorithms," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO '05)*, Antalya, Turkey, September 2005.

[10] H. Deng and M. Doroslovački, "Proportionate adaptive algorithms for network echo cancellation," *IEEE Transactions on Signal Processing*, vol. 54, no. 5, pp. 1794–1803, 2006.

[11] O. Tanrıkulu and K. Doğançay, "Selective-partial-update normalized least-mean-square algorithm for network echo cancellation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, vol. 2, pp. 1889–1892, Orlando, Fla, USA, May 2002.

[12] J. Kivinen and M. K. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Information and Computation*, vol. 132, no. 1, pp. 1–63, 1997.

[13] J. Benesty, T. Gänsler, D. Morgan, M. Sondhi, and S. Gay, Eds., *Advances in Network and Acoustic Echo Cancellation*, Springer, Boston, Mass, USA, 2001.

[14] O. Hoshuyama, R. A. Goubran, and A. Sugiyama, "A generalized proportionate variable step-size algorithm for fast changing acoustic environments," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, vol. 4, pp. 161–164, Montreal, Quebec, Canada, May 2004.

[15] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y.-F. Huang, "Set-membership filtering and a set-membership normalized LMS algorithm with an adaptive step size," *IEEE Signal Processing Letters*, vol. 5, no. 5, pp. 111–114, 1998.

[16] P. S. R. Diniz and S. Werner, "Set-membership binormalized data-reusing LMS algorithms," *IEEE Transactions on Signal Processing*, vol. 51, no. 1, pp. 124–134, 2003.

[17] S. Werner and P. S. R. Diniz, "Set-membership affine projection algorithm," *IEEE Signal Processing Letters*, vol. 8, no. 8, pp. 231–235, 2001.

[18] S. Gollamudi, S. Kapoor, S. Nagaraj, and Y.-F. Huang, "Set-membership adaptive equalization and an updator-shared implementation for multiple channel communications systems," *IEEE Transactions on Signal Processing*, vol. 46, no. 9, pp. 2372–2385, 1998.

[19] A. V. Malipatil, Y.-F. Huang, S. Andra, and K. Bennett, "Kernelized set-membership approach to nonlinear adaptive filtering," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 4, pp. 149–152, Philadelphia, Pa, USA, March 2005.

[20] E. Fogel and Y.-F. Huang, "On the value of information in system identification—bounded noise case," *Automatica*, vol. 18, no. 2, pp. 229–238, 1982.

[21] S. Dasgupta and Y.-F. Huang, "Asymptotically convergent modified recursive least-squares with data-dependent updating and forgetting factor for systems with bounded noise," *IEEE Transactions on Information Theory*, vol. 33, no. 3, pp. 383–392, 1987.

[22] J. R. Deller Jr., M. Nayeri, and M. S. Liu, "Unifying the Landmark developments in optimal bounding ellipsoid identification," *International Journal of Adaptive Control and Signal Processing*, vol. 8, no. 1, pp. 43–60, 1994.

[23] D. Joachim and J. R. Deller Jr., "Multiweight optimization in optimal bounding ellipsoid algorithms," *IEEE Transactions on Signal Processing*, vol. 54, no. 2, pp. 679–690, 2006.

[24] S. Gollamudi, S. Nagaraj, and Y.-F. Huang, "Blind equalization with a deterministic constant modulus cost-a set-membership filtering approach," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00)*, vol. 5, pp. 2765–2768, Istanbul, Turkey, June 2000.

[25] P. S. R. Diniz and S. Werner, "Set-membership binormalized data-reusing algorithms," in *Proceedings of the IFAC Symposium on System Identification (SYSID '00)*, vol. 3, pp. 869–874, Santa Barbara, Calif, USA, June 2000.

[26] S. Werner, M. L. R. de Campos, and P. S. R. Diniz, "Partial-update NLMS algorithms with data-selective updating," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 938–949, 2004.

[27] S. Werner, J. A. Apolinário Jr., M. L. R. de Campos, and P. S. R. Diniz, "Low-complexity constrained affine-projection algorithms," *IEEE Transactions on Signal Processing*, vol. 53, no. 12, pp. 4545–4555, 2005.

[28] S. Nagaraj, S. Gollamudi, S. Kapoor, and Y.-F. Huang, "BEACON: an adaptive set-membership filtering technique with sparse updates," *IEEE Transactions on Signal Processing*, vol. 47, no. 11, pp. 2928–2941, 1999.

[29] S. Werner, P. S. R. Diniz, and J. E. W. Moreira, "Set-membership affine projection algorithm with variable data-reuse factor," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '06)*, pp. 261–264, Island of Kos, Greece, May 2006.

[30] M. Rupp, "A family of adaptive filter algorithms with decorrelating properties," *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 771–775, 1998.