# A Stochastic Majorize-Minimize Subspace Algorithm for Online Penalized Least Squares Estimation

Emilie Chouzenoux, *Member, IEEE,* and Jean-Christophe Pesquet, *Fellow, IEEE*

***Abstract*—Stochastic approximation techniques play an important role in solving many problems encountered in machine learning or adaptive signal processing. In these contexts, the statistics of the data are often unknown a priori or their direct computation is too intensive, and they have thus to be estimated online from the observed signals. For batch optimization of an objective function being the sum of a data fidelity term and a penalization (e.g. a sparsity promoting function), Majorize-Minimize (MM) methods have recently attracted much interest since they are fast, highly flexible, and effective in ensuring convergence. The goal of this paper is to show how these methods can be successfully extended to the case when the data fidelity term corresponds to a least squares criterion and the cost function is replaced by a sequence of stochastic approximations of it. In this context, we propose an online version of an MM subspace algorithm and we study its convergence by using suitable probabilistic tools. Simulation results illustrate the good practical performance of the proposed algorithm associated with a memory gradient subspace, when applied to both non-adaptive and adaptive filter identification problems.**

*Keywords*: stochastic approximation, optimization, subspace algorithms, memory gradient methods, descent methods, recursive algorithms, majorization-minimization, filter identification, Newton method, sparsity, machine learning, adaptive filtering.

## I. Introduction

A classical problem in data sciences consists of inferring the structure of a linear model linking some observed random variables $(\mathbf{X}_n)_{n\geqslant 1}$ in $\mathbb{R}^{N\times Q}$ to some other observed random variables $(\mathbf{y}_n)_{n\geqslant 1}$ in $\mathbb{R}^Q$. Unless otherwise specified, we will assume in this work that the following wide-sense stationarity properties hold:

$$(\forall n\in\mathbb{N}^*)\qquad \mathsf{E}(\|\mathbf{y}_n\|^2)=\varrho \tag{1}$$

$$\mathsf{E}(\mathbf{X}_n\mathbf{y}_n)=\boldsymbol{r} \tag{2}$$

$$\mathsf{E}(\mathbf{X}_n\mathbf{X}_n^\top)=\boldsymbol{R}, \tag{3}$$

where $\varrho\in]0,+\infty[$, $\boldsymbol{r}\in\mathbb{R}^N$, $\boldsymbol{R}\in\mathbb{R}^{N\times N}$ is a symmetric positive semi-definite matrix, $\mathsf{E}(\cdot)$ denotes the mathematical expectation, and $\|\cdot\|$ is the Euclidean norm. We will then be interested in the following optimization formulation:

$$\underset{\boldsymbol{h}\in\mathbb{R}^N}{\text{minimize}}\;\; F(\boldsymbol{h}), \tag{4}$$

with

$$(\forall \boldsymbol{h}\in\mathbb{R}^N)\quad F(\boldsymbol{h})=\frac{1}{2}\mathsf{E}\big(\|\mathbf{y}_n-\mathbf{X}_n^\top\boldsymbol{h}\|^2\big)+\Psi(\boldsymbol{h}), \tag{5}$$

where $\Psi$ is a function from $\mathbb{R}^N$ to $\mathbb{R}$, playing the role of a regularization function. In particular, this function may be useful to incorporate some prior knowledge about the sought parameter vector $\boldsymbol{h}$, e.g. some sparsity requirement, possibly in some transformed domain. Problem (4) is encountered in numerous applications such as system identification, channel equalization, linear prediction or interpolation, echo cancellation, interference removal, and supervised classification. In the latter area, $(\mathbf{X}_n)_{n\geqslant 1}$ are vectors ($Q=1$) which may correspond to features obtained through some nonlinear mapping of the data to be classified in a given training sequence, and $(\mathbf{y}_n)_{n\geqslant 1}$ may be the associated (discrete-valued) class index vector. Although some other measures (e.g. the logistic regression function) are often more effective in this context, the use of a least squares criterion may still be competitive for simplicity reasons, while the regularization term serves here to avoid overfitting which could arise when the number of extracted features is large. Signal reconstruction constitutes another application field of interest. Then, the vector $\boldsymbol{h}$ corresponds to an unknown signal related to some measurements $(\mathbf{y}_n)_{n\geqslant 1}$ obtained through products with matrices $(\mathbf{X}_n^\top)_{n\geqslant 1}$, and additionally corrupted by some noise process. Each matrix $\mathbf{X}_n^\top$ with $n\in\mathbb{N}^*$ corresponds to $Q$ lines of the full acquisition matrix and it is here considered as random. Under suitable stationarity assumptions, the classical least squares data fidelity term can be modeled as $\mathsf{E}\big(\|\mathbf{y}_n-\mathbf{X}_n^\top\boldsymbol{h}\|^2\big)/2$, whereas due to the ill-posedness of the great majority of such inverse problems, a regularization term $\Psi$ needs to be introduced so as to obtain reliable estimates.

Many optimization algorithms can be devised to solve Problem (4) depending on the assumptions made on $\Psi$ [2]–[5]. In this work, we will be interested in Majorize-Minimize (MM) algorithms [6], [7]. In such approaches, the iterates result from successive minimizations of simple surrogates (e.g. quadratic surrogates) majorizing the cost-function. MM algorithms are very flexible and benefit from good theoretical and practical convergence properties. However, the computation load resulting from the minimization of the majorant function may be prohibitive in the context of large scale problems. The strategy we will adopt in this work is to account for subspace acceleration [8], i.e., to constrain the inner minimization step to a subspace of low dimension, typically restricted to the

E. Chouzenoux and J.-C. Pesquet (corresponding author) are with the Laboratoire d'Informatique Gaspard Monge, UMR CNRS 8049, Université Paris-Est, 77454 Marne la Vallée Cedex 2, France. E-mail: first.last@univ-paris-est.fr.

This work was supported by the CNRS Imag'in project under grant 2015 OPTIMISME.

Part of it was presented at the EUSIPCO 2014 conference [1].

gradient computed at the current iterate and to a memory part (e.g. the difference between the current iterate and a previous one). In a number of recent works [9]–[11], MM subspace algorithms have shown to provide fast numerical solutions to optimization problems involving smooth functions, in particular in the case of large-scale problems. Note that, although our approach will require that $\Psi$ is a differentiable function, it has been shown that tight approximations of nonsmooth penalizations such as $\ell_1$ (resp. $\ell_0$) functions, namely $\ell_2 - \ell_1$ (resp. $\ell_2 - \ell_0$) functions, can be employed and are often quite effective in practice [10], [11]. Another advantage of the class of optimization methods under investigation is that their convergence can be established under some technical assumptions, even in the case when $\Psi$ is a nonconvex function (see [10] for more details).

One of the difficulties encountered in machine learning or adaptive processing is that Problem (4) cannot be directly solved since the second-order statistical moments $\varrho$, $\boldsymbol{r}$ and $\boldsymbol{R}$ are often unknown a priori or their direct computation is too intensive, and they have thus to be estimated online from the related time series. In the simple case when $\Psi = 0$, the classical Recursive Least Squares (RLS) algorithm can be used for this purpose [12]. When $\Psi$ is nonzero, stochastic approximation algorithms have been developed such as the celebrated stochastic gradient descent (SGD) algorithm [13]–[16] and some of its proximal extensions [17]–[20]. These algorithms have been at the origin of a tremendous amount of works. SGD is known to be robust and easy to implement, but its convergence speed may be relatively slow. Various extensions of this algorithm have been developed to alleviate this problem (see [21]–[24] and the references therein). Many efforts have also been devoted to developing adaptive variants of this algorithm [25], [26], in particular when identifying filters having sparse impulse responses (see e.g. [27]–[33]). In addition, in [34], a set theoretic approach is adopted for online sparse estimation based on projections onto weighted $\ell_1$ balls, which is extended in [35] by making use of generalized thresholding mappings. It is worth noting that a sparse RLS algorithm was proposed in [36] for complex-valued signals in the case when $\Psi$ is an $\ell_1$ norm. This method theoretically requires an inner loop implementing an iterative soft-thresholding technique. An online variant of the RLS algorithm corresponding to a time weighted LASSO estimator was also designed in [37] which relies on a coordinate descent approach. A similar problem was also addressed in [38] by adopting a novel Bayes variational approach, for which weak theoretical convergence guarantees however exist. If we except [39] where an adaptive primal-dual splitting is employed to deal with a total variation penalization, in almost all the works on sparse adaptive filtering, the sparsity is directly imposed on the filter coefficients, without introducing any linear transform of them.

Designing Majorize-Minimize optimization algorithms in a stochastic context constitutes a challenging task since most of the existing works concerning these methods have been focused on batch optimization procedures, and the related convergence proofs usually rely on deterministic tools. We can however mention a few recent works [40]–[42] where stochastic MM algorithms are investigated for general loss functions under specific assumptions (e.g. the independence of the involved random variables [40], [41]), but without introducing any search subspace. Works which are more closely related to ours are those based on Newton or quasi-Newton stochastic algorithms [43]–[46], in particular the approaches in [45], [46] provide extensions of BFGS algorithm, but proving the convergence of these algorithms requires some specific assumptions. Like BFGS approaches, MM subspace methods use a memory of previous estimates so as to accelerate the convergence.

In Section II, we show how Problem (4) can be reformulated in a learning context. The MM strategy which is proposed in this work is described in Section III-A. In Section III-B, we give the form of the resulting recursive algorithm and, in Section III-C, we evaluate its computational complexity. A convergence analysis of the proposed stochastic Majorize-Minimize subspace algorithm is performed in Section IV. In Section V, two simulation examples in the context of filter identification show the good performance of our algorithm when a memory gradient subspace is employed. Some conclusions are drawn in Section VII.

## II. Problem formulation

In a learning context, function $F$ can be replaced by a sequence $(\mathrm{F}_n)_{n\geqslant 1}$ of stochastic approximations of it, which are defined as follows: for every $n \in \mathbb{N}^*$,

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \quad \mathrm{F}_n(\boldsymbol{h}) = \frac{1}{2\overline{\vartheta}_n}\sum_{k=1}^{n} \vartheta^{n-k}\|\mathbf{y}_k - \mathbf{X}_k^\top \boldsymbol{h}\|^2 + \Psi(\boldsymbol{h}) = \frac{1}{2}\rho_n - \mathbf{r}_n^\top \boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^\top \mathbf{R}_n \boldsymbol{h} + \Psi(\boldsymbol{h}), \tag{6}$$

where $\vartheta \in ]0,1]$,

$$\overline{\vartheta}_n = \sum_{k=0}^{n-1}\vartheta^k = \begin{cases} n & \text{if } \vartheta = 1 \\ \dfrac{1-\vartheta^n}{1-\vartheta} & \text{if } \vartheta \in ]0,1[, \end{cases} \tag{7}$$

and $\rho_n$, $\mathbf{r}_n$, and $\mathbf{R}_n$ are given by

$$\rho_n = \frac{1}{\overline{\vartheta}_n}\sum_{k=1}^{n}\vartheta^{n-k}\|\mathbf{y}_k\|^2 \tag{8}$$

$$\mathbf{r}_n = \frac{1}{\overline{\vartheta}_n}\sum_{k=1}^{n}\vartheta^{n-k}\mathbf{X}_k\mathbf{y}_k \tag{9}$$

$$\mathbf{R}_n = \frac{1}{\overline{\vartheta}_n}\sum_{k=1}^{n}\vartheta^{n-k}\mathbf{X}_k\mathbf{X}_k^\top. \tag{10}$$

In the case when $\vartheta = 1$, we retrieve the classical sample estimates of $\varrho$, $\boldsymbol{r}$, and $\boldsymbol{R}$. When $\vartheta \in ]0,1[$, it can be interpreted as an exponential forgetting factor [12] which may be useful in adaptive processing scenarios (see Section VI).

Hereafter, we will assume that the regularization function $\Psi$ has the following form:

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \quad \Psi(\boldsymbol{h}) = \frac{1}{2}\boldsymbol{h}^\top \boldsymbol{V}_0 \boldsymbol{h} - \boldsymbol{v}_0^\top \boldsymbol{h} + \sum_{s=1}^{S}\psi_s(\|\boldsymbol{V}_s\boldsymbol{h} - \boldsymbol{v}_s\|) \tag{11}$$

where $\boldsymbol{v}_0 \in \mathbb{R}^N$, $\boldsymbol{V}_0 \in \mathbb{R}^{N\times N}$ is a symmetric positive semi-definite matrix, and, for every $s \in \{1,\ldots,S\}$, $\boldsymbol{v}_s \in \mathbb{R}^{P_s}$, $\boldsymbol{V}_s \in \mathbb{R}^{P_s\times N}$, and $\psi_s\colon \mathbb{R}\to\mathbb{R}$ is a smooth function. The first term $\boldsymbol{h} \mapsto \frac{1}{2}\boldsymbol{h}^\top \boldsymbol{V}_0 \boldsymbol{h} - \boldsymbol{v}_0^\top \boldsymbol{h}$ can be viewed as an elastic net penalty [47], while various choices can be made for the second one. As shown in Table I, in addition to quadratic regularization functions (obtained when $S = 1$ and $\psi_1 = 0$), $\ell_2 - \ell_1$ functions and smoothed $\ell_2 - \ell_0$ functions constitute standard choices. The matrices $(\boldsymbol{V}_s)_{1\leqslant s\leqslant S}$ may be set to identity or they may serve to model possible transforms or discrete differentiation operators, and vectors $(\boldsymbol{v}_s)_{1\leqslant s\leqslant S}$ may be used to define reference values.

Note that the regularization strategy adopted in [37] amounts to replacing $\Psi$ in (6) by $\lambda_n\overline{\Psi}$ where $\overline{\Psi}$ is a (possibly weighted) $\ell_1$ norm and $\lambda_n \in [0,+\infty[$. Consistency results can then be established under the assumption that $\vartheta = 1$ and $\lim_{n\to+\infty}\lambda_n = 0$. Our approach here is different, not only because we are interested in a wide class of regularization functions, but also in the sense that we are looking for a solution to the fully regularized problem (4) instead of a solution to the mean square criterion.

Our objective in the next section will be to propose an efficient recursive method for minimizing functions $(\mathrm{F}_n)_{n\geqslant 1}$.

## III. Proposed method

### A. Majorization property

At each iteration $n \in \mathbb{N}^*$, we propose to replace $\mathrm{F}_n$ by a surrogate function $\Theta_n(\cdot,\mathbf{h}_n)$ based on the current estimate $\mathbf{h}_n$ (computed at the previous iteration). More precisely, a tangent majorant function is chosen such that

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \qquad \mathrm{F}_n(\boldsymbol{h}) \leqslant \Theta_n(\boldsymbol{h},\mathbf{h}_n) \tag{12}$$

$$\mathrm{F}_n(\mathbf{h}_n) = \Theta_n(\mathbf{h}_n,\mathbf{h}_n). \tag{13}$$

For the so-defined MM strategy to be worthwhile, the surrogate function has to be built in such a way that its minimization is simple. For this purpose, the following assumptions will be made on the regularization function $\Psi$ defined in (11):

**Assumption 1.**

(i) *For every $s \in \{1,\ldots,S\}$, $\psi_s$ is an even lower-bounded function, which is continuously differentiable, and $\lim_{\substack{t\to 0\\ t\neq 0}} \dot{\psi}_s(t)/t \in \mathbb{R}$, where $\dot{\psi}_s$ denotes the derivative of $\psi_s$.*

(ii) *For every $s \in \{1,\ldots,S\}$, $\psi_s(\sqrt{\cdot})$ is concave on $[0,+\infty[$.*

(iii) *There exists $\overline{\nu} \in [0,+\infty[$ such that $(\forall s \in \{1,\ldots,S\})$ $(\forall t \in [0,+\infty[)$ $0 \leqslant \nu_s(t) \leqslant \overline{\nu}$, where $\nu_s(t) = \dot{\psi}_s(t)/t$.*[1]

These assumptions are satisfied by a wide class of functions $\Psi$, in particular those corresponding to the choices of the potential functions $(\psi_s)_{1\leqslant s\leqslant S}$ listed in Table I.

Note that it follows from (6), (11), and the above definition of functions $(\nu_s)_{1\leqslant s\leqslant S}$ that, for every $n \in \mathbb{N}^*$, the gradient of $\mathrm{F}_n$ is given by

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \qquad \nabla \mathrm{F}_n(\boldsymbol{h}) = \mathbf{A}_n(\boldsymbol{h})\boldsymbol{h} - \mathbf{c}_n(\boldsymbol{h}), \tag{14}$$

where

$$\mathbf{A}_n(\boldsymbol{h}) = \mathbf{R}_n + \boldsymbol{V}_0 + \boldsymbol{V}^\top \mathrm{Diag}\big(\boldsymbol{b}(\boldsymbol{h})\big)\boldsymbol{V} \in \mathbb{R}^{N\times N} \tag{15}$$

$$\mathbf{c}_n(\boldsymbol{h}) = \mathbf{r}_n + \boldsymbol{v}_0 + \boldsymbol{V}^\top \mathrm{Diag}\big(\boldsymbol{b}(\boldsymbol{h})\big)\boldsymbol{v} \in \mathbb{R}^{N} \tag{16}$$

$$\boldsymbol{V} = [\boldsymbol{V}_1^\top \ldots \boldsymbol{V}_S^\top]^\top \in \mathbb{R}^{P\times N} \tag{17}$$

$$\boldsymbol{v} = [\boldsymbol{v}_1^\top \ldots \boldsymbol{v}_S^\top]^\top \in \mathbb{R}^{P} \tag{18}$$

with $P = P_1 + \cdots + P_S$, and $\boldsymbol{b}(\boldsymbol{h}) = \big(b_i(\boldsymbol{h})\big)_{1\leqslant i\leqslant P} \in \mathbb{R}^P$ is such that $(\forall s \in \{1,\ldots,S\})$ $(\forall p \in \{1,\ldots,P_s\})$

$$b_{P_1+\cdots+P_{s-1}+p}(\boldsymbol{h}) = \nu_s(\|\boldsymbol{V}_s\boldsymbol{h} - \boldsymbol{v}_s\|). \tag{19}$$

We have then the following result [48]:

**Proposition 1.** *Under Assumptions 1(i)-1(iii), for every $n \in \mathbb{N}^*$ and $\boldsymbol{h} \in \mathbb{R}^N$, a tangent majorant of $\mathrm{F}_n$ at $\boldsymbol{h}$ is*

$$(\forall \boldsymbol{h}' \in \mathbb{R}^N) \quad \Theta_n(\boldsymbol{h}',\boldsymbol{h}) = \mathrm{F}_n(\boldsymbol{h}) + \nabla\mathrm{F}_n(\boldsymbol{h})^\top(\boldsymbol{h}'-\boldsymbol{h}) + \frac{1}{2}(\boldsymbol{h}'-\boldsymbol{h})^\top\mathbf{A}_n(\boldsymbol{h})(\boldsymbol{h}'-\boldsymbol{h}), \tag{20}$$

*where $\mathbf{A}_n(\boldsymbol{h})$ is given by* (15).

The proposed MM subspace algorithm consists of defining the following sequence of random vectors $(\mathbf{h}_n)_{n\geqslant 1}$:

$$(\forall n \in \mathbb{N}^*) \qquad \mathbf{h}_{n+1} \in \underset{\boldsymbol{h}\in \operatorname{span}\mathbf{D}_n}{\arg\min}\ \Theta_n(\boldsymbol{h},\mathbf{h}_n), \tag{21}$$

where $\operatorname{span}\mathbf{D}_n$ is the vector subspace delineated by the columns of matrix $\mathbf{D}_n \in \mathbb{R}^{N\times M_n}$, and $\mathbf{h}_1$ has to be set to an initial value. A common assumption for subspace algorithms which will be adopted subsequently is that $\nabla\mathrm{F}_n(\mathbf{h}_n)$ belongs to $\operatorname{span}\mathbf{D}_n$. If, for every $n \in \mathbb{N}^*$, $\operatorname{rank}(\mathbf{D}_n) = N$, the algorithm is similar to a half-quadratic one [48]. Half-quadratic algorithms are known to be effective batch optimization methods, but the use of such method requires the inversion of matrix $\mathbf{A}_n(\mathbf{h}_n)$ at each iteration $n$, which may have a high computational cost. On the other hand, if for every $n \in \mathbb{N}^*$, $\mathbf{D}_n$ reduces to $[-\nabla\mathrm{F}_n(\mathbf{h}_n),\mathbf{h}_n]$, then

$$\mathbf{h}_{n+1} = \mathrm{u}_{n,2}\mathbf{h}_n - \mathrm{u}_{n,1}\nabla\mathrm{F}_n(\mathbf{h}_n), \tag{22}$$

where $(\mathrm{u}_{n,1},\mathrm{u}_{n,2})$ is a pair of real-valued random variables. In the special case when $\mathrm{u}_{n,2} = 1$, we recover the form of a SGD-like algorithm with step-size $\mathrm{u}_{n,1}$. In the machine learning literature, various forms of the step-size for SGD have been proposed [23], which often require to tune up some parameters (e.g. a multiplicative factor) so as to get the best convergence profile on the available dataset. The advantage of the proposed approach is that $(\mathrm{u}_{n,1},\mathrm{u}_{n,2})$ is automatically adjusted at each iteration following the MM rule. An intermediate size subspace matrix is obtained by choosing, for every $n \in \mathbb{N}^*$,

$$\mathbf{D}_n = \begin{cases} [-\nabla\mathrm{F}_n(\mathbf{h}_n),\mathbf{h}_n,\mathbf{h}_n-\mathbf{h}_{n-1}] & \text{if } n > 1\\ [-\nabla\mathrm{F}_n(\mathbf{h}_1),\mathbf{h}_1] & \text{if } n = 1. \end{cases} \tag{23}$$

This particular choice for the subspace yields the S3MG algorithm that will be shown in the next sections to have both good convergence properties and a low computational complexity. Note that a similar subspace choice can be found in batch optimization algorithms such as TWIST [49] or FISTA [50].

[1] The function is extended by continuity when $t = 0$.

TABLE I

SMOOTH PENALTY FUNCTIONS $\psi_s$ AND THEIR ASSOCIATED WEIGHTING FUNCTIONS $\nu_s$. ALL EXPRESSIONS ARE VALID FOR $t \in \mathbb{R}$, $(\lambda_s, \delta_s) \in ]0, +\infty[^2$ AND $\kappa_s \in [1, 2]$.

| | $\lambda_s^{-1}\psi_s(t)$ | $\lambda_s^{-1}\nu_s(t)$ | Type | Name |
|---|---|---|---|---|
| Convex | $\|t\| - \delta_s \log(\|t\|/\delta_s + 1)$ | $(\|t\| + \delta_s)^{-1}$ | $\ell_2 - \ell_1$ | |
| Convex | $\begin{cases} t^2 & \text{if } \|t\| < \delta_s \\ 2\delta_s\|t\| - \delta_s^2 & \text{otherwise} \end{cases}$ | $\begin{cases} 2 & \text{if } \|t\| < \delta_s \\ 2\delta_s/\|t\| & \text{otherwise} \end{cases}$ | $\ell_2 - \ell_1$ | Huber |
| Convex | $\log(\cosh(t))$ | $\begin{cases} \tanh(t)/t & \text{if } t \neq 0 \\ 1 & \text{otherwise} \end{cases}$ | $\ell_2 - \ell_1$ | Green |
| Convex | $(1 + t^2/\delta_s^2)^{\kappa_s/2} - 1$ | $\kappa_s \delta_s^{-2}(1 + t^2/\delta_s^2)^{\kappa_s/2-1}$ | $\ell_2 - \ell_{\kappa_s}$ | |
| Nonconvex | $1 - \exp(-t^2/(2\delta_s^2))$ | $\delta_s^{-2}\exp(-t^2/(2\delta_s^2))$ | $\ell_2 - \ell_0$ | Welsch |
| Nonconvex | $t^2/(2\delta_s^2 + t^2)$ | $4\delta_s^2/(2\delta_s^2 + t^2)$ | $\ell_2 - \ell_0$ | Geman -McClure |
| Nonconvex | $\begin{cases} 1 - (1 - t^2/(6\delta_s^2))^3 & \text{if } \|t\| \leqslant \sqrt{6}\delta_s \\ 1 & \text{otherwise} \end{cases}$ | $\begin{cases} \delta_s^{-2}(1 - t^2/(6\delta_s^2))^2 & \text{if } \|t\| \leqslant \sqrt{6}\delta_s \\ 0 & \text{otherwise} \end{cases}$ | $\ell_2 - \ell_0$ | Tukey biweight |
| Nonconvex | $\tanh(t^2/(2\delta_s^2))$ | $\delta_s^{-2}(\cosh(t^2/(2\delta_s^2))^{-2}$ | $\ell_2 - \ell_0$ | Hyberbolic tangent |
| Nonconvex | $\log(1 + t^2/\delta_s^2)$ | $2/(t^2 + \delta_s^2)$ | $\ell_2 - \log$ | Cauchy |
| Nonconvex | $1 - \exp(1 - (1 + t^2/(2\delta_s^2))^{\kappa_s/2})$ | $(\kappa_s/(2\delta_s^2))(1 + t^2/(2\delta_s^2))^{\kappa_s/2-1}\exp(1 - (1 + t^2/(2\delta_s^2))^{\kappa_s/2})$ | $\ell_2 - \ell_{\kappa_s} - \ell_0$ | Chouzenoux |

### *B. Recursive MM strategy*

Since, for every $n \in \mathbb{N}^*$, $\mathbf{h}_{n+1} \in \operatorname{span} \mathbf{D}_n$, let us set

$$\mathbf{h}_{n+1} = \mathbf{D}_n \mathbf{u}_n, \tag{24}$$

where $\mathbf{u}_n$ is an $\mathbb{R}^{M_n}$-valued random vector. We deduce from (14), (20), and (21) that, for every $n \in \mathbb{N}^*$,

$$\begin{aligned} \mathbf{u}_n &= \mathbf{B}_n^\dagger \mathbf{D}_n^\top \big(\mathbf{A}_n(\mathbf{h}_n)\mathbf{h}_n - \nabla \mathrm{F}_n(\mathbf{h}_n)\big) \\ &= \mathbf{B}_n^\dagger \mathbf{D}_n^\top \mathbf{c}_n(\mathbf{h}_n), \end{aligned} \tag{25}$$

where

$$\mathbf{B}_n = \mathbf{D}_n^\top \mathbf{A}_n(\mathbf{h}_n)\mathbf{D}_n \tag{26}$$

and $(\cdot)^\dagger$ is the pseudo-inverse operation. It is important to note that, as $\mathbf{B}_n$ is of dimension $M_n \times M_n$ where $M_n$ is small (typically $M_n = 3$ for the choice of the subspace in (23) when $n > 1$), this pseudo-inversion is light. This constitutes the key advantage of the proposed approach.

By using (7), (9) and (10), the following recursive updates of $(\mathbf{r}_n)_{n\geqslant 1}$ and $(\mathbf{R}_n)_{n\geqslant 1}$, can be performed

$$(\forall n \in \mathbb{N}^*) \qquad \mathbf{r}_n = \mathbf{r}_{n-1} + \frac{1}{\overline{\vartheta}_n}(\mathbf{X}_n \mathbf{y}_n - \mathbf{r}_{n-1}) \tag{27}$$

$$\mathbf{R}_n = \mathbf{R}_{n-1} + \frac{1}{\overline{\vartheta}_n}(\mathbf{X}_n \mathbf{X}_n^\top - \mathbf{R}_{n-1}), \tag{28}$$

where we have set $\mathbf{r}_0 = \mathbf{0}$ and $\mathbf{R}_0 = \boldsymbol{O}_N$ and we have used the identity: $\vartheta\overline{\vartheta}_{n-1}/\overline{\vartheta}_n = 1 - \overline{\vartheta}_n^{-1}$.

Let us now introduce the intermediate variables:

$$(\forall n \in \mathbb{N}^*) \qquad \mathbf{D}_n^{\mathbf{R}} = \mathbf{R}_n \mathbf{D}_n \in \mathbb{R}^{N \times M_n} \tag{29}$$

$$\mathbf{D}_n^{\boldsymbol{V}_0} = \boldsymbol{V}_0 \mathbf{D}_n \in \mathbb{R}^{N \times M_n} \tag{30}$$

$$\mathbf{D}_n^{\boldsymbol{V}} = \boldsymbol{V} \mathbf{D}_n \in \mathbb{R}^{P \times M_n}. \tag{31}$$

It follows from (15) and (26) that

$$\begin{aligned} (\forall n \in \mathbb{N}^*)\ \mathbf{B}_n = \mathbf{D}_n^\top \big(\mathbf{D}_n^{\mathbf{R}} + \mathbf{D}_n^{\boldsymbol{V}_0}\big) \\ + \big(\mathbf{D}_n^{\boldsymbol{V}}\big)^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_n)\big)\mathbf{D}_n^{\boldsymbol{V}}. \end{aligned} \tag{32}$$

Without loss of generality, it can be assumed that the algorithm is initialized with $\mathbf{h}_1 = \mathbf{D}_0 \mathbf{u}_0$, where $\mathbf{D}_0 \in \mathbb{R}^{N \times M_0}$ and $\mathbf{u}_0 \in \mathbb{R}^{M_0}$. Then, (14) and (24) yield

$$(\forall n \in \mathbb{N}^*) \qquad \nabla \mathrm{F}_n(\mathbf{h}_n) = \mathbf{D}_{n-1}^{\mathbf{A}} \mathbf{u}_{n-1} - \mathbf{c}_n(\mathbf{h}_n), \tag{33}$$

where we have set

$$(\forall n \in \mathbb{N}) \qquad \mathbf{D}_n^{\mathbf{A}} = \mathbf{A}_{n+1}(\mathbf{h}_{n+1})\, \mathbf{D}_n \in \mathbb{R}^{N \times M_n}. \tag{34}$$

By using (15) and (28)-(31), the latter variable can be reexpressed as

$$\begin{aligned} \mathbf{D}_n^{\mathbf{A}} &= \mathbf{R}_{n+1}\mathbf{D}_n + \mathbf{D}_n^{\boldsymbol{V}_0} + \boldsymbol{V}^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_{n+1})\big)\mathbf{D}_n^{\boldsymbol{V}} \\ &= (1 - \frac{1}{\overline{\vartheta}_{n+1}})\mathbf{D}_n^{\mathbf{R}} + \frac{1}{\overline{\vartheta}_{n+1}}\mathbf{X}_{n+1}(\mathbf{X}_{n+1}^\top \mathbf{D}_n) + \mathbf{D}_n^{\boldsymbol{V}_0} \\ &\quad + \boldsymbol{V}^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_{n+1})\big)\mathbf{D}_n^{\boldsymbol{V}}. \end{aligned} \tag{35}$$

The resulting relations are summarized in Algorithm 1.

**Algorithm 1:** Stochastic MM subspace method

$\mathbf{r}_0 = \mathbf{0}, \mathbf{R}_0 = \boldsymbol{O}_N$
Initialize $\mathbf{D}_0, \mathbf{u}_0$
$\mathbf{h}_1 = \mathbf{D}_0 \mathbf{u}_0, \mathbf{D}_0^{\mathbf{R}} = \boldsymbol{O}_{N \times M_n}, \mathbf{D}_0^{\boldsymbol{V}_0} = \boldsymbol{V}_0 \mathbf{D}_0, \mathbf{D}_0^{\boldsymbol{V}} = \boldsymbol{V}\mathbf{D}_0$
**for** $n = 1, \ldots$ **do**
- $\mathbf{r}_n = \mathbf{r}_{n-1} + \frac{1}{\overline{\vartheta}_n}(\mathbf{X}_n \mathbf{y}_n - \mathbf{r}_{n-1})$
- $\mathbf{c}_n(\mathbf{h}_n) = \mathbf{r}_n + \boldsymbol{v}_0 + \boldsymbol{V}^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_n)\big)\boldsymbol{v}$
- $\mathbf{D}_{n-1}^{\mathbf{A}} = (1 - \frac{1}{\overline{\vartheta}_n})\mathbf{D}_{n-1}^{\mathbf{R}} + \frac{1}{\overline{\vartheta}_n}\mathbf{X}_n(\mathbf{X}_n^\top \mathbf{D}_{n-1}) + \mathbf{D}_{n-1}^{\boldsymbol{V}_0} + \boldsymbol{V}^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_n)\big)\mathbf{D}_{n-1}^{\boldsymbol{V}}$
- $\nabla \mathrm{F}_n(\mathbf{h}_n) = \mathbf{D}_{n-1}^{\mathbf{A}} \mathbf{u}_{n-1} - \mathbf{c}_n(\mathbf{h}_n)$
- $\mathbf{R}_n = \mathbf{R}_{n-1} + \frac{1}{\overline{\vartheta}_n}(\mathbf{X}_n \mathbf{X}_n^\top - \mathbf{R}_{n-1})$
- Set $\mathbf{D}_n$ using $\nabla \mathrm{F}_n(\mathbf{h}_n)$
- $\mathbf{D}_n^{\mathbf{R}} = \mathbf{R}_n \mathbf{D}_n, \mathbf{D}_n^{\boldsymbol{V}_0} = \boldsymbol{V}_0 \mathbf{D}_n, \mathbf{D}_n^{\boldsymbol{V}} = \boldsymbol{V}\mathbf{D}_n$
- $\mathbf{B}_n = \mathbf{D}_n^\top \big(\mathbf{D}_n^{\mathbf{R}} + \mathbf{D}_n^{\boldsymbol{V}_0}\big) + \big(\mathbf{D}_n^{\boldsymbol{V}}\big)^\top \operatorname{Diag}\big(\boldsymbol{b}(\mathbf{h}_n)\big)\mathbf{D}_n^{\boldsymbol{V}}$
- $\mathbf{u}_n = \mathbf{B}_n^\dagger \mathbf{D}_n^\top \big(\mathbf{c}_n(\mathbf{h}_n)\big)$
- $\mathbf{h}_{n+1} = \mathbf{D}_n \mathbf{u}_n$

**end**

TABLE II
COMPLEXITY IN TERMS OF MULTIPLICATIONS FOR ITERATION $n$ OF ALGORITHM 1.

| Step | Complexity for $\boldsymbol{V} \in \mathbb{R}^{P\times N}$ arbitrary | Complexity when $\boldsymbol{V} = \boldsymbol{I}_N$ |
|---|---|---|
| 1 | $N(Q+1)$ | |
| 2 | $(N+1)P$ | $N$ |
| 3 | $M_{n-1}\big(N(2Q+P+1)+P+Q\big)$ | $M_{n-1}\big(N(2Q+1)+Q\big)$ |
| 4 | $NM_{n-1}$ | |
| 5 | $N(N+1)Q/2$ | |
| 7 | $NM_n(2N+P)$ | $2N^2M_n$ |
| 8 | $M_n\big((M_n+1)(N+P)/2+P\big)$ | $NM_n(M_n+3)/2$ |
| 9 | $O(M_n^3)+M_n(N+M_n)$ | |
| 10 | $NM_n$ | |

### C. Complexity

As shown in Table II, provided that the subspace dimensions $(M_n)_{n\in\mathbb{N}}$ are small, the proposed algorithm has a low complexity. Indeed, the global complexity of a direct implementation of the algorithm, evaluated in terms of multiplications at iteration $n$, is of the order of

$$N\big(P(M_n+M_{n-1}+1)+N(4M_n+Q)/2\big),$$

if we assume that $N \gg \max\{M_n, M_{n-1}, Q\}$. The first term $NP(M_n+M_{n-1}+1)$ corresponds to an upper bound on the complexity induced by the use of matrices $(\boldsymbol{V}_s)_{1\leqslant s\leqslant S}$ within the regularization term. These matrices often have a sparse structure (in particular when discrete derivative operators are employed) which leads to a much lower computational cost. When $\boldsymbol{V}=\boldsymbol{I}_N$, the identity matrix of $\mathbb{R}^N$, which is a scenario frequently encountered in adaptive filtering, this term merely vanishes in the evaluation of the global complexity.

The computational complexity can also be reduced by taking advantage of the specific form of matrices $(\mathbf{D}_n)_{n\geqslant 1}$. For example, if the subspace is chosen according to (23),

$$(\forall n>1) \quad \mathbf{D}_n^{\boldsymbol{V}} = [-\boldsymbol{V}\nabla \mathrm{F}_n(\mathbf{h}_n), \boldsymbol{V}\mathbf{h}_n, \boldsymbol{V}\mathbf{h}_n - \boldsymbol{V}\mathbf{h}_{n-1}]. \tag{36}$$

On the other hand, for every $n\geqslant 1$,

$$\boldsymbol{V}\mathbf{h}_n = \boldsymbol{V}\mathbf{D}_{n-1}\mathbf{u}_{n-1} = \mathbf{D}_{n-1}^{\boldsymbol{V}}\mathbf{u}_{n-1}, \tag{37}$$

which shows that a recursive formula holds to compute the last two components of $\mathbf{D}_n^{\boldsymbol{V}}$ in (36). The initial complexity of $3NP$ multiplications is thus reduced to $N(P+3)$. Similar recursive procedures can be employed to compute $(\mathbf{D}_n^{\boldsymbol{V}_0})_{n>1}$ allowing the complexity to be reduced to $N(N+3)$ from $3N^2$. In addition, we have, for every $n>1$,

$$\mathbf{D}_n^{\mathbf{R}} = [-\mathbf{R}_n\nabla \mathrm{F}_n(\mathbf{h}_n), \mathbf{h}_n^{\mathbf{R}}, \mathbf{h}_n^{\mathbf{R}} - \mathbf{R}_n\mathbf{h}_{n-1}], \tag{38}$$

where, by using (28),

$$\begin{aligned}\mathbf{h}_n^{\mathbf{R}} = \mathbf{R}_n\mathbf{h}_n &= (1-\frac{1}{\vartheta_n})\mathbf{R}_{n-1}\mathbf{h}_n + \frac{1}{\vartheta_n}\mathbf{X}_n\mathbf{X}_n^\top\mathbf{h}_n \\ &= (1-\frac{1}{\vartheta_n})\mathbf{D}_{n-1}^{\mathbf{R}}\mathbf{u}_{n-1} + \frac{1}{\vartheta_n}\mathbf{X}_n\mathbf{X}_n^\top\mathbf{h}_n \end{aligned} \tag{39}$$

$$\mathbf{R}_n\mathbf{h}_{n-1} = (1-\frac{1}{\vartheta_n})\mathbf{h}_{n-1}^{\mathbf{R}} + \frac{1}{\vartheta_n}\mathbf{X}_n\mathbf{X}_n^\top\mathbf{h}_{n-1}. \tag{40}$$

It can be further observed that last term $(\vartheta_n)^{-1}\mathbf{X}_n\mathbf{X}_n^\top\mathbf{h}_{n-1}$ has already been computed in Step 3 of Algorithm 1. Therefore, instead of $3N^2$ multiplications, we have now to perform $N(N+2Q+4)$ ones. With these simplifications, in the case when $\boldsymbol{V}_0$ and $\boldsymbol{V}$ are null matrices, the global complexity of the algorithm is equal to $N^2(Q+2)/2$. When $Q=1$, we thus recover the order of complexity of the classical RLS algorithm. Since the objective function then reduces to a quadratic function, Sherman-Morrison-Woodbury formula can be invoked to compute iteratively the minimizer on the whole space in an efficient manner.

Note finally that the computation of $\mathbf{X}_n\mathbf{X}_n^\top$ with $n\in\mathbb{N}^*$, which needs to be performed in Step 5, remains a main source of complexity. However, if $(\forall n>Q)$ $\mathbf{X}_n = [\mathbf{x}_{n-Q+1},\ldots,\mathbf{x}_n]$ where $\mathbf{x}_n\in\mathbb{R}^N$ (as it is the case in affine projection based algorithms for adaptive processing [51]), then a recursive computation of $\mathbf{X}_n\mathbf{X}_n^\top$ only requires $\mathbf{x}_n\mathbf{x}_n^\top$ to be computed at each iteration $n>Q$. If we further assume that the model is a one-dimensional convolutive one, i.e. $\mathbf{x}_n$ corresponds to shifted samples of a signal $\big(x(n)\big)_{n\geqslant 1}$, then $(\forall n>N)$ $\mathbf{x}_n = [x(n-N+1),\ldots,x(n)]^\top$ and $\mathbf{x}_n\mathbf{x}_n^\top$ can be itself computed recursively with a complexity of $N$ operations. Such ideas have been deeply investigated in the literature on fast RLS algorithms [52].

## IV. CONVERGENCE STUDY

Throughout this section and the related appendices, it is assumed that $\vartheta=1$ and the underlying probability space is denoted by $(\Omega,\mathcal{F},\mathsf{P})$. We will say in short that a property is $\mathsf{P}$-a.s. satisfied if this property holds almost surely.

### A. Assumptions

For every $n\in\mathbb{N}^*$, let $\mathcal{X}_n = \sigma\big((\mathbf{X}_k,\mathbf{y}_k)_{1\leqslant k\leqslant n}\big)$ be the sub-sigma algebra of $\mathcal{F}$ generated by $(\mathbf{X}_k,\mathbf{y}_k)_{1\leqslant k\leqslant n}$. In order to give a proof of convergence of the proposed stochastic MM subspace algorithm, we will make the following additional assumption:

**Assumption 2.**

(i) *$\boldsymbol{R}+\boldsymbol{V}_0$ is a positive definite matrix.*
(ii) *$\big((\mathbf{X}_n,\mathbf{y}_n)\big)_{n\geqslant 1}$ is a stationary ergodic sequence and, for every $n\in\mathbb{N}^*$, the elements of $\mathbf{X}_n$ and the components of $\mathbf{y}_n$ have finite fourth-order moments.*
(iii) *For every $n\in\mathbb{N}^*$,*

$$\mathsf{E}(\|\mathbf{y}_{n+1}\|^2\,|\,\mathcal{X}_n) = \varrho \tag{41}$$

$$\mathsf{E}(\mathbf{X}_{n+1}\mathbf{y}_{n+1}\,|\,\mathcal{X}_n) = \boldsymbol{r} \tag{42}$$

$$\mathsf{E}(\mathbf{X}_{n+1}\mathbf{X}_{n+1}^\top\,|\,\mathcal{X}_n) = \boldsymbol{R}. \tag{43}$$

(iv) *For every $n\in\mathbb{N}^*$, $\{\nabla \mathrm{F}_n(\mathbf{h}_n),\mathbf{h}_n\}\subset \operatorname{span}\mathbf{D}_n$.*
(v) *$\mathbf{h}_1$ is $\mathcal{X}_1$-measurable and, for every $n\in\mathbb{N}^*$, $\mathbf{D}_n$ is $\mathcal{X}_n$-measurable.*

The following asymptotic results will then be useful in the rest of our developments.

**Lemma 1.** *Under Assumptions 2(ii) and 2(iii), the following properties hold:*

(i) *$(\rho_n)_{n\geqslant 1}$, $(\mathbf{R}_n)_{n\geqslant 1}$, and $(\mathbf{r}_n)_{n\geqslant 1}$ converge $\mathsf{P}$-a.s. to $\varrho$, $\boldsymbol{R}$ and $\boldsymbol{r}$, respectively*

(ii) $\sum_{n=1}^{+\infty} n^{-1}|\rho_n - \varrho| < +\infty$ P-a.s.

$$\sum_{n=1}^{+\infty} n^{-1}\|\mathbf{r}_n - \boldsymbol{r}\| < +\infty \quad \text{P-a.s.}$$

$$\sum_{n=1}^{+\infty} n^{-1}|||\mathbf{R}_n - \boldsymbol{R}||| < +\infty \text{ P-a.s.},$$

*where* $|||\cdot|||$ *denotes the spectral matrix norm.*

*Proof:* See Appendix A. ■

**Remark 1.**

(i) *Assumptions 2(ii) and 2(iii) are more general than assuming that* $\big((\mathbf{X}_n, \mathbf{y}_n)\big)_{n\geqslant 1}$ *is an independent identically distributed (i.i.d.) sequence and, for every* $n \in \mathbb{N}^*$, *the elements of* $\mathbf{X}_n$ *and the components of* $\mathbf{y}_n$ *have finite fourth-order moments.*

(ii) *Assumptions 2(iv) and 2(v) are satisfied for the choice of subspace given by* (23) *if* $\mathbf{h}_1$ *is* $\mathfrak{X}_1$*-measurable (e.g.* $\mathbf{h}_1$ *is deterministic).*

### B. Almost sure convergence

Let us give the following preliminary property:

**Lemma 2.** *Under Assumptions 1 and 2(ii)-2(iii),* $(\mathbf{h}_n)_{n\geqslant 1}$ *is* P-a.s. *bounded.*[2]

*Proof:* See Appendix B. ■

Combining the previous lemma with classical results on the asymptotic behaviour of almost supermartingales, the convergence of the sequence $\big(\mathrm{F}_n(\mathbf{h}_n)\big)_{n\geqslant 1}$ can be established:

**Lemma 3.** *Under Assumptions 1 and 2,* $\big(\mathrm{F}_n(\mathbf{h}_n)\big)_{n\geqslant 1}$ *is* P-a.s. *convergent and* $\big((\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n)\big)_{n\geqslant 1}$ *is* P-a.s. *summable.*

*Proof:* See Appendix C. ■

Lemma 3 allows us to deduce the following result on the sequence of gradients computed at each iteration of the algorithm:

**Lemma 4.** *Under Assumptions 1 and 2,* $(\|\nabla \mathrm{F}_n(\mathbf{h}_n)\|)_{n\geqslant 1}$ *is* P-a.s. *square-summable.*

*Proof:* See Appendix D. ■

By gathering all the previous results, our main convergence results can now be stated:

**Proposition 2.** *Assume that Assumptions 1 and 2 hold. Then, the following hold:*

(i) *The set of cluster points of* $(\mathbf{h}_n)_{n\geqslant 1}$ *is almost surely a nonempty compact connected set.*

(ii) *Any element of this set is almost surely a critical point of* $F$.

(iii) *If the functions* $(\psi_s)_{1\leqslant s\leqslant S}$ *are convex, then* $(\mathbf{h}_n)_{n\geqslant 1}$ *converges* P-a.s. *to the unique (global) minimizer of* $F$.

[2]We say that a sequence of random vectors is almost surely bounded when the norms of all these vectors can be bounded by some random variable with probability 1.

*Proof:* See Appendix E. ■

It can be noticed that the conclusion of Proposition 2(iii) is still valid if the functions $(\psi_s)_{1\leqslant s\leqslant S}$ are nonconvex, they are twice continuously differentiable, and the regularization constants $(\lambda_s)_{1\leqslant s\leqslant S}$ as defined in Table I are small enough so that the function $F$ is strongly convex.

## V. Application to 2D filter identification

### A. Problem statement

We first demonstrate the efficiency of the proposed stochastic algorithm in a 2D filter identification problem. We consider the following observation model:

$$\boldsymbol{y} = S(\overline{\boldsymbol{h}})\boldsymbol{x} + \boldsymbol{w}, \tag{44}$$

where $\boldsymbol{x} \in \mathbb{R}^L$ and $\boldsymbol{y} \in \mathbb{R}^L$ represent the original and degraded versions of a given image, $\overline{\boldsymbol{h}} \in \mathbb{R}^N$ is the vectorized version of an unknown two-dimensional blur kernel, $S$ is the linear operator which maps the kernel to its associated Hankel-block Hankel matrix form, and $\boldsymbol{w} \in \mathbb{R}^L$ represents a realization of an additive noise. When the images $\boldsymbol{x}$ and $\boldsymbol{y}$ are of very large size, finding an estimate $\widehat{\boldsymbol{h}} \in \mathbb{R}^N$ of the blur kernel can be quite memory consuming, but one can expect good estimation performance by learning the blur kernel through a sweep of blocks in the dataset.

Let us denote by $\boldsymbol{X} \in \mathbb{R}^{L\times N}$ the matrix such that $S(\boldsymbol{h})\boldsymbol{x} = \boldsymbol{X}\boldsymbol{h}$. Then, we propose to define $\widehat{\boldsymbol{h}}$ as a solution to (4), where, for every $n \in \mathbb{N}^*$, $\mathbf{y}_n \in \mathbb{R}^Q$ and $\mathbf{X}_n^\top \in \mathbb{R}^{Q\times N}$, are subparts of $\boldsymbol{y}$ and $\boldsymbol{X}$, respectively, corresponding to $Q \in \{1,\ldots,L\}$ lines of this vector/matrix. For the regularization term $\Psi$, we consider, for every $s \in \{1,\ldots,N\}$ ($S = N$), an isotropic penalization on the gradient between neighboring coefficients of the blur kernel, i.e., $P_s = 2$ and $\boldsymbol{V}_s = \left[\boldsymbol{\Delta}_s^{\mathrm{h}}\ \boldsymbol{\Delta}_s^{\mathrm{v}}\right]^\top$, where $\boldsymbol{\Delta}_s^{\mathrm{h}} \in \mathbb{R}^N$ (resp. $\boldsymbol{\Delta}_s^{\mathrm{v}} \in \mathbb{R}^N$) is the horizontal (resp. vertical) gradient operator applied at pixel $s$. The smoothness of $\boldsymbol{h}$ is then enforced by choosing, for every $s \in \{1,\ldots,S\}$ and $u \in \mathbb{R}$, $\psi_s(u) = \lambda\sqrt{1 + u^2/\delta^2}$ with $(\lambda,\delta) \in ]0,+\infty[^2$. Finally, in order to guarantee the existence of a unique minimizer, the strong convexity of $F$ is imposed by taking $\boldsymbol{v}_0 = \mathbf{0}$ and $\boldsymbol{V}_0 = \tau\boldsymbol{I}_N$, where $\tau$ is a small positive value (typically $\tau = 10^{-10}$).

### B. Simulation results

The original image, presented in Figure 1(a), is a satellite image, of size $4096 \times 4096$ pixels. The original blur kernel $\overline{\boldsymbol{h}}$ with size $21 \times 21$, and the resulting blurred image, which has been corrupted with a zero-mean white Gaussan noise with standard deviation $\sigma = 0.03$ (the blurred signal-to-noise ratio equals 25.7 dB), are displayed in Figures 1(b)(c). Figure 1(d) presents the estimated kernel, using Algorithm1, with the subspace given by (23), leading to the so-called S3MG algorithm. Parameters $(\lambda,\delta)$ were adjusted so as to minimize the normalized root mean square estimation error, here equal to $0.064$. Figure 2 illustrates the variations of this estimation error with respect to the computation time for the proposed algorithm, the SGD algorithm with a decreasing stepsize proportional to $n^{-1/2}$, the regularized dual averaging

(RDA) method with a constant stepsize from [40], and the accelerated stochastic gradient averaging Finito method with a constant stepsize from [53] when running tests on an Intel(R) Xeon(R) E5-2630 @ 2.6GHz using a Matlab 7 implementation. Note that for the latter three algorithms, the stepsize parameter was optimized manually so as to obtain the best performance in terms of convergence speed. Finally, note that all tested algorithms were observed to provide asymptotically the same estimation quality, whatever the size of the blocks. In this example, as illustrated in Figure 3, the best trade-off in terms of convergence speed is obtained for $Q = 256 \times 256$.

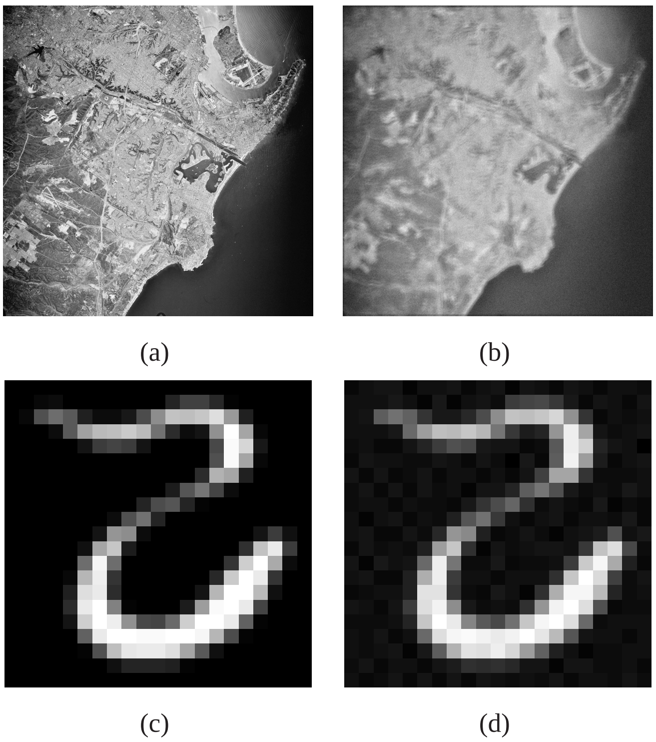


Fig. 1. (a) Original image. (b) Blurred and noisy image. (c) Original blur kernel. (d) Estimated blur kernel, with relative error 0.064.

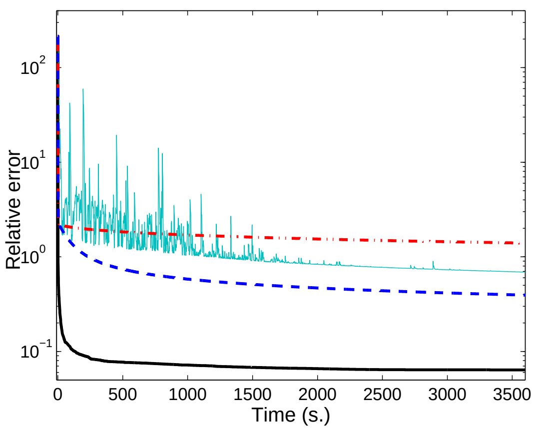


Fig. 2. Comparison of S3MG algorithm (solid black line), SGD algorithm with decreasing stepsize $\propto n^{-1/2}$ (dashed-dotted red line), RDA algorithm with constant stepsize (dashed blue line) and Finito algorithm with constant stepsize (turquoise thin line).

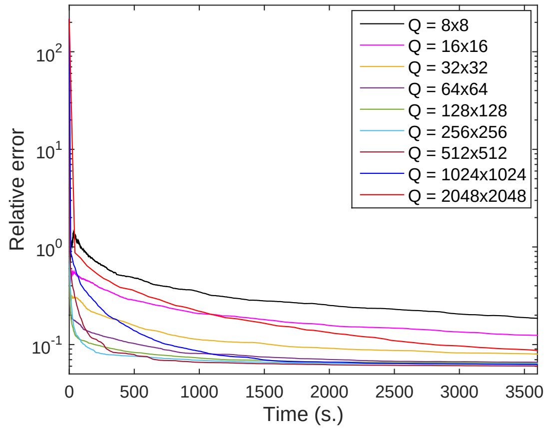


Fig. 3. Effect of the block size $Q$ on the convergence speed of S3MG.

## VI. Application to Sparse Adaptive Filtering

### A. Problem statement

As emphasized in Sections II and III, one of the advantages of Algorithm 1 compared with some other online optimization algorithms is that it is able to deal with adaptive data processing problems. In this section, we apply the S3MG algorithm to the identification of a sparse time-varying system. Given a real-valued discrete-time input signal $\big(x(n)\big)_{n\in\mathbb{Z}}$, the output of the system at time $n \geqslant 1$ is defined as

$$y_n = \boldsymbol{X}_n^\top \overline{\boldsymbol{h}}_n + w_n, \tag{45}$$

where $\boldsymbol{X}_n = [x(n-N+1), \ldots, x(n)]^\top$, $w_n$ models some measurement noise, and $\overline{\boldsymbol{h}}_n \in \mathbb{R}^N$ gathers the unknown filter taps at time $n$. Then, the objective is to provide an estimate of the vector $\overline{\boldsymbol{h}}_n$ at each time by solving Problem (4) where the regularization function $\Psi$ is chosen in order to promote the sparsity of the impulse response of the time-varying filter.

### B. Simulation results

We generate data according to Model (45) where the input signal $\big(x(n)\big)_{n\in\mathbb{Z}}$ consists of identically and independent random binary values $\{-1,+1\}$. The measurement noise $(w_n)_{n\in\mathbb{Z}}$ is white Gaussian with zero mean and variance 0.05. In order to evaluate the tracking capability of the proposed S3MG method, the following time-varying linear system is considered:

$$\overline{\boldsymbol{h}}_n = \begin{cases} \overline{\boldsymbol{h}}_1 & \text{if } n \leqslant L/2, \\ \overline{\boldsymbol{h}}_{L/2+1} & \text{if } n \geqslant L/2+1. \end{cases} \tag{46}$$

The filter length $N$ is equal to 200 and the output of the system is observed at every time $n \in \{1, \ldots, L\}$ with $L = 5000$. The sparse impulse responses corresponding to vectors $\overline{\boldsymbol{h}}_1$ and $\overline{\boldsymbol{h}}_{L/2+1}$ are represented in Figure 4.

We compute, for every $n \in \{1, \ldots, L\}$, the Euclidean norm of the error between the current estimate $\boldsymbol{h}_n$ and the true filter coefficient vector $\overline{\boldsymbol{h}}_n$. The minimal estimation error is obtained for the nonconvex Welsch penalty function (see Table I) and a smoothed $\ell_2 - \ell_0$ regularization function is thus employed by setting $S = N$, $\boldsymbol{v}_0 = \boldsymbol{0}$, $\boldsymbol{V}_0 = \boldsymbol{O}_N$, and, for every $s \in$

$\{1, \ldots, N\}$, $P_s = 1$, $\boldsymbol{v}_s = 0$, while $\boldsymbol{V}_s \in \mathbb{R}^{1\times N}$ is the $s$-th vector of the canonical basis of $\mathbb{R}^N$.

We present the results generated by S3MG in Figure 5 for two values of the forgetting factor $\vartheta$, namely $\vartheta = 1$ which corresponds to a non adaptive strategy, and $\vartheta = 0.995$ which appears to be the best choice in terms of tracking properties for this example.

We also show the results obtained with several state-of-the-art approaches in the context of sparse adaptive filtering, namely SPAL [34], RLMS [54], RZAAPA [32] and SM-PAPA [55]. Note that, for each tested method, the involved parameters (stepsize, regularization weight, blocksize) have been tuned manually in order to optimize the performance in terms of error decay.

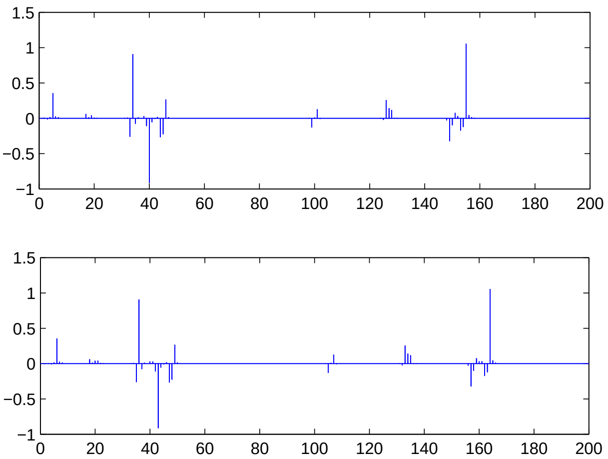


Fig. 4. Values of the coefficients of the considered sparse filters $\overline{\boldsymbol{h}}_1$ (top) and $\overline{\boldsymbol{h}}_{L/2+1}$ (bottom).

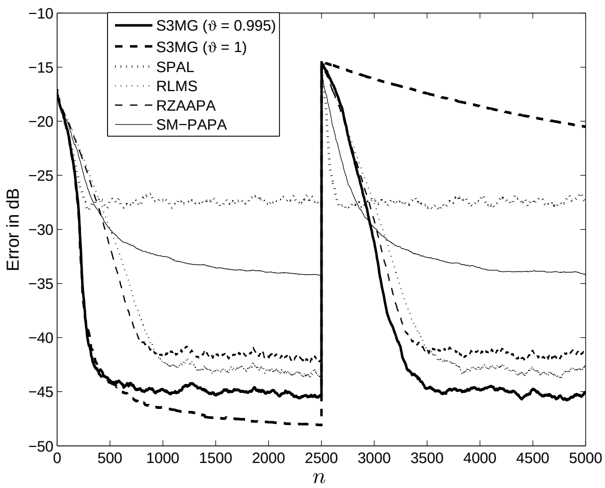


Fig. 5. Quadratic estimation error on the filter coefficients as a function of time index $n$ for various adaptive algorithms.

## VII. Conclusion

In this work, we have proposed a stochastic MM subspace algorithm for online penalized least squares estimation problems. The method makes it possible to use large-size datasets the second-order moments of which are not known a priori. We have shown that the proposed algorithm is of the same order of complexity as the classical RLS algorithm and that its computational cost can be reduced by taking advantage of specific forms of the search subspace. The choice of a memory gradient subspace led to the S3MG algorithm whose good numerical performance has been demonstrated in the context of 2D filter identification for large scale image processing problems. In the context of sparse adaptive filtering, S3MG has also been shown to be competitive with respect to recent methods. Although an analysis of the convergence of the proposed method has been carried out, it would be interesting to extend the obtained results to weaker assumptions. In addition, in a nonstationary context, a theoretical study of the tracking abilities of the algorithm should be conducted. Finally, a detailed analysis of the convergence rate of the proposed method will be undertaken in a forthcoming paper.

## Appendix A
## Proof of Lemma 1

Property (i) is a consequence of the ergodic theorem [56, Theorem 13.12]. In addition, the law of the iterated logarithm for martingale difference sequences [57] ensures that

$$\limsup_{n\to+\infty} \frac{\left|\sum_{k=1}^{n}(\|\mathbf{y}_k\|^2 - \varrho)\right|}{\big(n\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{47}$$

$$\limsup_{n\to+\infty} \frac{\left\|\sum_{k=1}^{n}(\mathbf{X}_k\mathbf{y}_k - \boldsymbol{r})\right\|}{\big(n\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{48}$$

$$\limsup_{n\to+\infty} \frac{\left|\left|\left|\sum_{k=1}^{n}(\mathbf{X}_k\mathbf{X}_k^\top - \boldsymbol{R})\right|\right|\right|}{\big(n\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{49}$$

that is

$$\limsup_{n\to+\infty} \frac{n^{1/2}|\rho_n - \varrho|}{\big(\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{50}$$

$$\limsup_{n\to+\infty} \frac{n^{1/2}\|\mathbf{r}_n - \boldsymbol{r}\|}{\big(\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{51}$$

$$\limsup_{n\to+\infty} \frac{n^{1/2}|||\mathbf{R}_n - \boldsymbol{R}|||}{\big(\log(\log n)\big)^{1/2}} < +\infty \qquad \text{P-a.s.} \tag{52}$$

Consequently, for every $n_0 \in \mathbb{N}$ with $n_0 \geqslant 2$,

$$\sum_{n=n_0}^{+\infty} n^{-1}|\rho_n - \varrho| \leqslant \sup_{n\geqslant n_0}\left(\frac{n^{1/2}|\rho_n - \varrho|}{\big(\log(\log n)\big)^{1/2}}\right)\left(\sum_{n=n_0}^{+\infty} n^{-3/2}|\log(\log n)|^{1/2}\right). \tag{53}$$

Since $\sum_{n=2}^{+\infty} n^{-3/2}|\log(\log n)|^{1/2} < +\infty$, it follows from (50) that $\sum_{n=n_0}^{+\infty} n^{-1}|\rho_n - \varrho|$ converges P-a.s. to 0 as $n_0 \to +\infty$, which means that the first line in Property (ii) is satisfied. By proceeding similarly, (51) and (52) allow us to establish the remaining two assertions in Property (ii).

## Appendix B
## Proof of Lemma 2

For every $n \in \mathbb{N}^*$, minimizing $\Theta_n(\cdot, \mathbf{h}_n)$ is equivalent to minimizing the function

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \quad \widetilde{\Theta}_n(\boldsymbol{h}, \mathbf{h}_n) = \frac{1}{2}\boldsymbol{h}^\top \mathbf{A}_n(\mathbf{h}_n)\boldsymbol{h} - \mathbf{c}_n(\mathbf{h}_n)^\top \boldsymbol{h}. \tag{54}$$

It follows from Assumption 2(ii)-2(iii) and Lemma 1(i) that there exists $\Lambda \in \mathcal{F}$ such that $\mathsf{P}(\Lambda) = 1$ and, for every $\omega \in \Lambda$,

$$\lim_{n\to+\infty} \mathbf{r}_n(\omega) = \boldsymbol{r} \tag{55}$$

$$\lim_{n\to+\infty} \mathbf{R}_n(\omega) = \boldsymbol{R}. \tag{56}$$

Let $\omega \in \Lambda$. According to Assumption 1(iii) and Eq. (19), $\boldsymbol{b}(\boldsymbol{h})$ is bounded as a function of $\boldsymbol{h}$. It is then deduced from (16) and (55) that $\big(\mathbf{c}_n(\mathbf{h}_n)(\omega)\big)_{n\geqslant 1}$ is bounded, i.e. there exists $\eta \in [0, +\infty[$ such that

$$(\forall n \in \mathbb{N}^*) \qquad \|\mathbf{c}_n(\mathbf{h}_n)(\omega)\| \leqslant \eta. \tag{57}$$

In addition, as a consequence of (19) and Assumption 1(iii), for every $n \in \mathbb{N}^*$, $\mathrm{Diag}\big(\boldsymbol{b}(\boldsymbol{h}_n)\big)$ is a positive semidefinite matrix. Hence, because of (15), Assumptions 1(iii) and 2(i), and (56), there exists $\epsilon \in ]0, +\infty[$ and $n_0 \in \mathbb{N}^*$ such that

$$(\forall n \geqslant n_0) \qquad \mathbf{A}_n(\mathbf{h}_n)(\omega) \succeq \boldsymbol{R} - \epsilon \boldsymbol{I}_N + \boldsymbol{V}_0 \succ \boldsymbol{O}_N. \tag{58}$$

(It suffices to choose $\epsilon$ lower than the minimum eigenvalue of $\boldsymbol{R} + \boldsymbol{V}_0$). As a consequence of (54), (57), (58), and the Cauchy-Schwarz inequality, we have

$$(\forall n \geqslant n_0)(\forall \boldsymbol{h} \in \mathbb{R}^N)$$
$$\frac{1}{2}\boldsymbol{h}^\top (\boldsymbol{R} - \epsilon \boldsymbol{I}_N + \boldsymbol{V}_0)\boldsymbol{h} - \eta\|\boldsymbol{h}\| \leqslant \widetilde{\Theta}_n(\boldsymbol{h}, \mathbf{h}_n). \tag{59}$$

Since $\boldsymbol{R} - \epsilon \boldsymbol{I}_N + \boldsymbol{V}_0$ is a positive definite matrix, the lower bound corresponds to a coercive function with respect to $\boldsymbol{h}$. There thus exists $\zeta \in ]0, +\infty[$ such that, for every $\boldsymbol{h} \in \mathbb{R}^N$,

$$\|\boldsymbol{h}\| > \zeta \quad \Rightarrow \quad (\forall n \geqslant n_0) \;\; \widetilde{\Theta}_n(\boldsymbol{h}, \mathbf{h}_n)(\omega) > 0. \tag{60}$$

On the other hand, since $\mathbf{0} \in \mathrm{span}\,\big(\mathbf{D}_n(\omega)\big)$, we have

$$\widetilde{\Theta}_n(\mathbf{h}_{n+1}, \mathbf{h}_n)(\omega) \leqslant \widetilde{\Theta}_n(\mathbf{0}, \mathbf{h}_n)(\omega) = 0. \tag{61}$$

The last two inequalities allow us to conclude that

$$(\forall n \geqslant n_0) \qquad \|\mathbf{h}_{n+1}(\omega)\| \leqslant \zeta. \tag{62}$$

## Appendix C
## Proof of Lemma 3

According to Assumption 2(iv), the proposed algorithm is actually equivalent to

$$(\forall n \in \mathbb{N}^*) \qquad \mathbf{h}_{n+1} = \mathbf{h}_n + \mathbf{D}_n \widetilde{\mathbf{u}}_n \tag{63}$$

$$\widetilde{\mathbf{u}}_n = \underset{\widetilde{\boldsymbol{u}} \in \mathbb{R}^M}{\arg\min}\, \Theta_n(\mathbf{h}_n + \mathbf{D}_n \widetilde{\boldsymbol{u}}, \mathbf{h}_n). \tag{64}$$

By using (20) and cancelling the derivative of the function $\widetilde{\boldsymbol{u}} \mapsto \Theta_n(\mathbf{h}_n + \mathbf{D}_n \widetilde{\boldsymbol{u}}, \mathbf{h}_n)$,

$$\mathbf{D}_n^\top \nabla \mathrm{F}_n(\mathbf{h}_n) + \mathbf{D}_n^\top \mathbf{A}_n(\mathbf{h}_n)\mathbf{D}_n \widetilde{\mathbf{u}}_n = \mathbf{0}. \tag{65}$$

Hence,

$$\begin{aligned}
&\Theta(\mathbf{h}_{n+1}, \mathbf{h}_n)\\
&= \mathrm{F}_n(\mathbf{h}_n) - \frac{1}{2}\widetilde{\mathbf{u}}_n^\top \mathbf{D}_n^\top \mathbf{A}_n(\mathbf{h}_n)\mathbf{D}_n \widetilde{\mathbf{u}}_n\\
&= \mathrm{F}_n(\mathbf{h}_n) - \frac{1}{2}(\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n).
\end{aligned} \tag{66}$$

In view of (12) and Proposition 1, this yields

$$(\forall n \in \mathbb{N}^*) \;\; \mathrm{F}_n(\mathbf{h}_{n+1}) + \frac{1}{2}(\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n) \leqslant \mathrm{F}_n(\mathbf{h}_n). \tag{67}$$

In addition, the following recursive relation holds

$$(\forall \boldsymbol{h} \in \mathbb{R}^N) \quad \mathrm{F}_{n+1}(\boldsymbol{h}) = \mathrm{F}_n(\boldsymbol{h}) + \frac{1}{2}(\rho_{n+1} - \rho_n) - (\mathbf{r}_{n+1} - \mathbf{r}_n)^\top \boldsymbol{h} + \frac{1}{2}\boldsymbol{h}^\top (\mathbf{R}_{n+1} - \mathbf{R}_n)\boldsymbol{h}. \tag{68}$$

As a consequence of Assumption 2(v), for every $n \in \mathbb{N}^*$, $\mathbf{h}_{n+1}$ is $\mathcal{X}_n$-measurable. It can thus be deduced from (67) and the previous two relations that

$$\mathsf{E}(\mathrm{F}_{n+1}(\mathbf{h}_{n+1}) \,|\, \mathcal{X}_n) + \frac{1}{2}(\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n) \leqslant \mathrm{F}_n(\mathbf{h}_n) + \chi_n \tag{69}$$

where

$$\chi_n = \frac{1}{2}\mathsf{E}(\rho_n - \rho_{n+1} \,|\, \mathcal{X}_n) - \mathsf{E}(\mathbf{r}_n - \mathbf{r}_{n+1} \,|\, \mathcal{X}_n)^\top \mathbf{h}_{n+1} + \frac{1}{2}\mathbf{h}_{n+1}^\top \mathsf{E}(\mathbf{R}_n - \mathbf{R}_{n+1} \,|\, \mathcal{X}_n)\mathbf{h}_{n+1}. \tag{70}$$

By using (8)-(10) with $\vartheta = 1$ and Assumption 2(iii), we have

$$\begin{aligned}
\chi_n &= \frac{1}{2(n+1)}\big(\rho_n - \mathsf{E}(\|\mathbf{y}_{n+1}\|^2 \,|\, \mathcal{X}_n)\big)\\
&\quad - \frac{1}{n+1}\big(\mathbf{r}_n - \mathsf{E}(\mathbf{X}_{n+1}\mathbf{y}_{n+1} \,|\, \mathcal{X}_n)\big)^\top \mathbf{h}_{n+1}\\
&\quad + \frac{1}{2(n+1)}\mathbf{h}_{n+1}^\top \big(\mathbf{R}_n - \mathsf{E}(\mathbf{X}_{n+1}\mathbf{X}_{n+1}^\top \,|\, \mathcal{X}_n)\big)\mathbf{h}_{n+1}\\
&= \frac{1}{2(n+1)}\big(\rho_n - \varrho\big) - \frac{1}{n+1}\big(\mathbf{r}_n - \boldsymbol{r}\big)^\top \mathbf{h}_{n+1}\\
&\quad + \frac{1}{2(n+1)}\mathbf{h}_{n+1}^\top \big(\mathbf{R}_n - \boldsymbol{R}\big)\mathbf{h}_{n+1}
\end{aligned} \tag{71}$$

which yields

$$|\chi_n| \leqslant \frac{1}{2(n+1)}|\rho_n - \varrho| + \frac{1}{n+1}\|\mathbf{r}_n - \boldsymbol{r}\|\|\mathbf{h}_{n+1}\| + \frac{1}{2(n+1)}|||\mathbf{R}_n - \boldsymbol{R}|||\,\|\mathbf{h}_{n+1}\|^2. \tag{72}$$

According to Lemma 2, $(\mathbf{h}_n)_{n\geqslant 1}$ is $\mathsf{P}$-a.s. bounded, and Assumptions 2(ii)-2(iii) and Lemma 1(ii) thus guarantee that

$$\sum_{n=1}^{+\infty} |\chi_n| < +\infty \qquad \mathsf{P}\text{-a.s.} \tag{73}$$

Assumption 1(i) entails that, for every $n \in \mathbb{N}^*$, $\mathrm{F}_n$ is lower bounded by $\inf \Psi > -\infty$. Furthermore, (69) leads to

$$\begin{aligned}&\mathsf{E}(\mathrm{F}_{n+1}(\mathbf{h}_{n+1}) - \inf \Psi \,|\, \mathfrak{X}_n)\\&+ \frac{1}{2}(\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n)\\&\leqslant \mathrm{F}_n(\mathbf{h}_n) - \inf \Psi + |\chi_n|.\end{aligned} \tag{74}$$

Since, for every $n \in \mathbb{N}^*$, $\mathrm{F}_n(\mathbf{h}_n) - \inf \Psi$ and $(\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n)$ are nonnegative, $(\mathrm{F}_n(\mathbf{h}_n) - \inf \Psi)_{n \geqslant 1}$ is a nonnegative almost supermartingale [58]. By invoking now Siegmund-Robbins lemma [59], it can be deduced from (73) that the desired convergence results hold.

## Appendix D
## Proof of Lemma 4

According to (20), we have, for every $\phi \in \mathbb{R}$ and $n \in \mathbb{N}^*$,

$$\begin{aligned}\Theta_n\big(\mathbf{h}_n - \phi \nabla \mathrm{F}_n(\mathbf{h}_n), \mathbf{h}_n\big) &= \mathrm{F}_n(\mathbf{h}_n) - \phi \|\nabla \mathrm{F}_n(\mathbf{h}_n)\|^2\\&+ \frac{\phi^2}{2}\big(\nabla \mathrm{F}_n(\mathbf{h}_n)\big)^\top \mathbf{A}_n(\mathbf{h}_n) \nabla \mathrm{F}_n(\mathbf{h}_n).\end{aligned} \tag{75}$$

Let

$$\Phi_n \in \operatorname*{Argmin}_{\phi \in \mathbb{R}} \; \Theta_n\big(\mathbf{h}_n - \phi \nabla \mathrm{F}_n(\mathbf{h}_n), \mathbf{h}_n\big). \tag{76}$$

The following optimality condition holds:

$$\big(\nabla \mathrm{F}_n(\mathbf{h}_n)\big)^\top \mathbf{A}_n(\mathbf{h}_n) \nabla \mathrm{F}_n(\mathbf{h}_n)\, \Phi_n = \|\nabla \mathrm{F}_n(\mathbf{h}_n)\|^2. \tag{77}$$

As a consequence of Assumption 2(iv), $(\forall \phi \in \mathbb{R})\ \mathbf{h}_n - \phi \nabla \mathrm{F}_n(\mathbf{h}_n) \in \operatorname{span} \mathbf{D}_n$. It then follows from (21) and (77) that

$$\begin{aligned}\Theta_n\big(\mathbf{h}_{n+1}, \mathbf{h}_n\big) &\leqslant \Theta_n\big(\mathbf{h}_n - \Phi_n \nabla \mathrm{F}_n(\mathbf{h}_n), \mathbf{h}_n\big)\\&\leqslant \mathrm{F}_n(\mathbf{h}_n) - \frac{\Phi_n}{2}\|\nabla \mathrm{F}_n(\mathbf{h}_n)\|^2\end{aligned} \tag{78}$$

which, by using (66), leads to

$$\Phi_n \|\nabla \mathrm{F}_n(\mathbf{h}_n)\|^2 \leqslant (\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n). \tag{79}$$

Let $\epsilon > 0$. Assumption 1(iii) and (15) yield, for every $n \in \mathbb{N}^*$,

$$\mathbf{A}_n(\mathbf{h}_n) \preceq (|||\mathbf{R}_n + \boldsymbol{V}_0||| + \overline{\nu} |||\boldsymbol{V}|||^2) \boldsymbol{I}_N. \tag{80}$$

Therefore, according to Assumptions 2(i) and 2(ii), and Lemma 1(i), there exists $\Lambda \in \mathcal{F}$ such that $\mathsf{P}(\Lambda) = 1$ and, for every $\omega \in \Lambda$,

$$(\exists n_0 \in \mathbb{N}^*)(\forall n \geqslant n_0) \quad \boldsymbol{O}_N \prec \mathbf{A}_n(\mathbf{h}_n)(\omega) \preceq \alpha_\epsilon^{-1} \boldsymbol{I}_N \tag{81}$$

where

$$\alpha_\epsilon = (|||\boldsymbol{R} + \boldsymbol{V}_0||| + \overline{\nu} |||\boldsymbol{V}|||^2 + \epsilon)^{-1} > 0. \tag{82}$$

Let $\omega \in \Lambda$. By using now (77), it can be deduced from (81) that, if $n \geqslant n_0$ and $\nabla \mathrm{F}_n(\mathbf{h}_n)(\omega) \neq \mathbf{0}$, then

$$\Phi_n(\omega) \geqslant \alpha_\epsilon. \tag{83}$$

Then, it follows from (79) that

$$\begin{aligned}&\alpha_\epsilon \sum_{n=n_0}^{+\infty} \|\nabla \mathrm{F}_n(\mathbf{h}_n)(\omega)\|^2\\&\leqslant \sum_{n=n_0}^{+\infty} \big(\mathbf{h}_{n+1}(\omega) - \mathbf{h}_n(\omega)\big)^\top \mathbf{A}_n(\mathbf{h}_n)(\omega)\big(\mathbf{h}_{n+1}(\omega) - \mathbf{h}_n(\omega)\big).\end{aligned} \tag{84}$$

By invoking Lemma 3, we can conclude that $(\|\nabla \mathrm{F}_n(\mathbf{h}_n)\|^2)_{n \geqslant 1}$ is $\mathsf{P}$-a.s. summable.

## Appendix E
## Proof of Proposition 2

It follows from Lemma 3 that $\big((\mathbf{h}_{n+1} - \mathbf{h}_n)^\top \mathbf{A}_n(\mathbf{h}_n)(\mathbf{h}_{n+1} - \mathbf{h}_n)\big)_{n \geqslant 1}$ converges $\mathsf{P}$-a.s. to 0. In addition, we have seen in the proof of Lemma 2 that there exists $\Lambda \in \mathcal{F}$ such that $\mathsf{P}(\Lambda) = 1$ and, for every $\omega \in \Lambda$, (58) holds with $\epsilon \in ]0, +\infty[$ and $n_0 \in \mathbb{N}^*$. This implies that, for every $n \geqslant n_0$,

$$\begin{aligned}&|||\boldsymbol{R} - \epsilon \boldsymbol{I}_N + \boldsymbol{V}_0|||\, \|\mathbf{h}_{n+1}(\omega) - \mathbf{h}_n(\omega)\|^2\\&\leqslant \big(\mathbf{h}_{n+1}(\omega) - \mathbf{h}_n(\omega)\big)^\top \mathbf{A}_n(\mathbf{h}_n)(\omega)\big(\mathbf{h}_{n+1}(\omega) - \mathbf{h}_n(\omega)\big)\end{aligned} \tag{85}$$

where $|||\boldsymbol{R} - \epsilon \boldsymbol{I}_N + \boldsymbol{V}_0||| > 0$. Consequently, $(\mathbf{h}_{n+1} - \mathbf{h}_n)_{n \geqslant 1}$ converges $\mathsf{P}$-a.s. to $\mathbf{0}$. In addition, according to Lemma 2, $(\mathbf{h}_n)_{n \geqslant 1}$ belongs almost surely to a compact set. The result is then obtained by invoking Ostrowski's theorem [60, Theorem 26.1].

(ii) By using (14)-(16), we have

$$(\forall n \in \mathbb{N}^*) \quad \nabla \mathrm{F}_n(\mathbf{h}_n) - \nabla F(\mathbf{h}_n) = (\mathbf{R}_n - \boldsymbol{R})\mathbf{h}_n - \mathbf{r}_n + \boldsymbol{r}. \tag{86}$$

Since $(\mathbf{h}_n)_{n \geqslant 1}$ is almost surely bounded, it follows from Lemma 1(i) that $\big(\nabla \mathrm{F}_n(\mathbf{h}_n) - \nabla F(\mathbf{h}_n)\big)_{n \geqslant 1}$ converges $\mathsf{P}$-a.s. to $\mathbf{0}$. Since Lemma 4 ensures that $\big(\nabla \mathrm{F}_n(\mathbf{h}_n)\big)_{n \geqslant 1}$ converges $\mathsf{P}$-a.s. to $\mathbf{0}$, $\big(\nabla F(\mathbf{h}_n)\big)_{n \geqslant 1}$ also converges $\mathsf{P}$-a.s. to $\mathbf{0}$. There thus exists $\Lambda \in \mathcal{F}$ such that $\mathsf{P}(\Lambda) = 1$ and, for every $\omega \in \Lambda$, $\nabla F\big(\mathbf{h}_n(\omega)\big) \to \mathbf{0}$. Let $\widehat{\boldsymbol{h}}$ be a cluster point of $\big(\mathbf{h}_n(\omega)\big)_{n \geqslant 1}$. There exists a subsequence $\big(\mathbf{h}_{k_n}(\omega)\big)_{n \geqslant 1}$ such that $\mathbf{h}_{k_n}(\omega) \to \widehat{\boldsymbol{h}}$. As we have assumed that the regularization functions $(\psi_s)_{1 \leqslant s \leqslant S}$ are continuously differentiable (see Assumption 1(i)), $F$ is also continuously differentiable, and

$$\nabla F(\widehat{\boldsymbol{h}}) = \lim_{n \to +\infty} \nabla F\big(\mathbf{h}_{k_n}(\omega)\big) = \mathbf{0}. \tag{87}$$

This means that $\widehat{\boldsymbol{h}}$ is a critical point of $F$.

(iii) Because of Assumption 2(i), when the functions $(\psi_s)_{1 \leqslant s \leqslant S}$ are convex, $F$ is a strongly convex function. It thus possesses a unique critical point $\widehat{\boldsymbol{h}}$, which is the global minimizer of $F$. It follows from (i) and (ii) that, almost surely, the unique cluster point of $(\mathbf{h}_n)_{n \geqslant 1}$ is $\widehat{\boldsymbol{h}}$, which shows that $\mathbf{h}_n \to \widehat{\boldsymbol{h}}$ $\mathsf{P}$-a.s.

## ACKNOWLEDGEMENTS

The authors would like to thank Professors Markus V. S. Lima and Paulo S. R. Diniz from Federal University of Rio de Janeiro for kindly making us accessible some of their codes. We would also like to thank Professor Anisia Florescu from Dunărea de Jos University of Galaţi for providing the initial motivation for this work.


## REFERENCES


[1] E. Chouzenoux, J.-C. Pesquet, and A. Florescu, "A stochastic 3MG algorithm with application to 2D filter identification," in *Proc. 22nd European Signal Process. Conf. (EUSIPCO 2014)*, Lisboa, Portugal, 1-5 Sep. 2014, pp. 1587–1591.

[2] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.

[3] P. L. Combettes and J.-C. Pesquet, "Proximal splitting methods in signal processing," in *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, H. H. Bauschke, R. Burachik, P. L. Combettes, V. Elser, D. R. Luke, and H. Wolkowicz, Eds., pp. 185–212. Springer-Verlag, New York, 2010.

[4] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "An augmented Lagrangian aproach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, 2011.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Machine Learn.*, vol. 8, no. 1, pp. 1–122, 2011.

[6] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Stat*, vol. 58, no. 1, pp. 30–37, Feb. 2004.

[7] Z. Zhang, J. T. Kwok, and D.-Y. Yeung, "Surrogate maximization/minimization algorithms and extensions," *Mach. Learn.*, vol. 69, pp. 1–33, 2007.

[8] M. Zibulevsky and M. Elad, "$\ell_2 - \ell_1$ optimization in signal and image processing," *IEEE Signal Process. Mag.*, vol. 27, pp. 76–88, May 2010.

[9] E. Chouzenoux, J. Idier, and S. Moussaoui, "A majorize-minimize subspace strategy for subspace optimization applied to image restoration," *IEEE Trans. Image Process.*, vol. 20, no. 18, pp. 1517–1528, Jun. 2011.

[10] E. Chouzenoux, A. Jezierska, J.-C. Pesquet, and H. Talbot, "A majorize-minimize subspace approach for $\ell_2$-$\ell_0$ image regularization," *SIAM J. Imag. Sci.*, vol. 6, no. 1, pp. 563–591, 2013.

[11] A. Florescu, E. Chouzenoux, J.-C. Pesquet, P. Ciuciu, and S. Ciochina, "A majorize-minimize memory gradient method for complex-valued inverse problem," *Signal Process.*, vol. 103, pp. 285–295, Oct. 2014, Special issue on Image Restoration and Enhancement: Recent Advances and Applications.

[12] S. O. Haykin, *Adaptive Filter Theory*, Prentice Hall, New Jersey, USA, 4th edition, 2002.

[13] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Stat.*, vol. 22, no. 3, pp. 400–407, 1951.

[14] Ju. M. Ermoliev and Z. V. Nekrylova, "The method of stochastic gradients and its application," in *Seminar: Theory of Optimal Solutions. No. 1 (Russian)*, pp. 24–47. Akad. Nauk Ukrain. SSR, Kiev, 1967.

[15] O. V. Guseva, "The rate of convergence of the method of generalized stochastic gradients," *Kibernetika (Kiev)*, , no. 4, pp. 143–145, 1971.

[16] D. P. Bertsekas and J. N. Tsitsiklis, "Gradient convergence in gradient methods with errors," *SIAM J. Optim.*, vol. 10, no. 3, pp. 627–642, 2000.

[17] Y. F. Atchadé, G. Fort, and E. Moulines, "On stochastic proximal gradient algorithms," 2014, http://arxiv.org/abs/1402.2365.

[18] L. Rosasco, S. Villa, and B. C. Vũ, "Convergence of stochastic proximal gradient algorithm," 2014, http://arxiv.org/abs/1403.5074.

[19] P. L. Combettes and J.-C. Pesquet, "Stochastic approximations and perturbations in forward-backward splitting for monotone operators," *Pure Appl. Funct. Anal.*, 2016, to appear, http://arxiv.org/abs/1507.07095.

[20] J. Konečný, J. Liu, P. Richtárik, and M. Takáč, "mS2GD: Mini-batch semi-stochastic gradient descent in the proximal setting," 2014, http://arxiv.org/pdf/1410.4744v1.pdf.

[21] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM J. Control Optim.*, vol. 30, no. 4, pp. 838–855, 1992.

[22] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1574–1609, 2008.

[23] F. Bach and E. Moulines, "Non-asymptotic analysis of stochastic approximation algorithms for machine learning," in *Proc. Ann. Conf. Neur. Inform. Proc. Syst.*, Granada, Spain, Dec. 12 - 17 2011, pp. x–x+8.

[24] M. Pereyra, P. Schniter, E. Chouzenoux, J.-C. Pesquet, J.-Y. Tourneret, A. O Hero, and S. McLaughlin, "A survey of stochastic simulation and optimization methods in signal processing," *IEEE Journal of Selected Topics in Signal Processing*, 2016, to appear, http://arxiv.org/abs/1505.00273.

[25] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, New Jersey, 1985.

[26] O. Macchi, *Adaptive Processing: The Least Mean Squares Approach With Applications in Transmission*, Wiley, Chichester, UK, 1995.

[27] O. Hoshuyama, R. A. Goubran, and A. Sugiyama, "A generalized proportionate variable step-size algorithm for fast changing acoustic environments," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP 2004)*, Montreal, Canada, May 17-21 2004, vol. 4, pp. 161–164.

[28] A. W. H. Khong and P. A Naylor, "Efficient use of sparse adaptive filters," in *Proc. Asilomar Conf. Signal, Systems and Computers*, Pacific Grove, CA, Oct. 29-Nov. 1 2006, pp. 1375–1379.

[29] Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP 2009)*, Taipei, Taiwan, Apr. 19-24 2009, pp. 3125–3128.

[30] C. Paleologu, J. Benesty, and S. Ciochină, *Sparse adaptive filters for echo cancellation*, Synthesis Lectures on Speech and Audio Processing. Morgan and Claypool, San Rafael, USA, 2010.

[31] Y. Murakami, M. Yamagishi, M. Yukawa, and I. Yamada, "A sparse adaptive filtering using time-varying soft-thresholding techniques," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP 2010)*, Dallas, Texas, Mar. 14-19 2010, pp. 3734–3737.

[32] R. Meng, R. C. De Lamare, and V. H. Nascimento, "Sparsity-aware affine projection adaptive algorithms for system identification," in *Proc. Sensor Signal Process. Defence*, London, U.K., Sept. 27-29 2011, pp. 1–5.

[33] L. V. S. Markus, W. A. Martins, and P. S. R. Diniz, "Affine projection algorithms for sparse system identification," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP 2013)*, Vancouver, Canada, May 26-31 2013, pp. 5666–5670.

[34] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online sparse system identification and signal reconstruction using projections onto weighted $\ell_1$ balls," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 936–952, Mar. 2011.

[35] K. Slavakis, Y. Kopsinis, S. Theodoridis, and S. McLaughlin, "Generalized thresholding and online sparsity-aware learning in a union of subspaces," *IEEE Trans. Signal Process.*, vol. 61, no. 15, pp. 3760–3773, Aug. 2013.

[36] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The sparse RLS algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4013–4025, Aug. 2010.

[37] D. Angelosante, J. A. Bazerque, and G. B. Giannakis, "Online adaptive estimation of sparse signals: where RLS meets the $\ell_1$ -norm," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3436–3447, Jul. 2010.

[38] K. E. Themelis, A. A. Rontogiannis, and K. D. Koutroumbas, "A variational bayes framework for sparse adaptive estimation," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4723–4736, 2014.

[39] S. Ono, M. Yamagishi, and I. Yamada, "A sparse system identification by using adaptively-weighted total variation via a primal-dual splitting approach," in *Proc. Int. Conf. Acoust., Speech Signal Process. (ICASSP 2013)*, Vancouver, Canada, 26-31 May 2013, pp. 6029–6033.

[40] L. Xiao, "Dual averaging methods for regularized stochastic learning and online optimization," *J. Mach. Learn. Res.*, vol. 11, pp. 2543–2596, Oct. 2010.

[41] J. Mairal, "Stochastic Majorization-Minimization algorithms for large-scale optimization," in *Proc. Adv. Conf. Neur. Inform. Proc. Syst.*, Lake Tahoe, Nevada, Dec. 5-8 2013, pp. x–x+8.

[42] D. Carlson, Y.-P. Hsieh, E. Collins, L. Carin, and V. Cevher, "Stochastic spectral descent for discrete graphical models," *IEEE Journal of Selected Topics in Signal Processing*, 2016, to appear, http://people.ee.duke.edu/~lcarin/SSD_for_belief_nets.pdf.

[43] J. R. Birge, X. Chen, L. Qi, and Z. Wei, "A stochastic Newton method for stochastic quadratic programs with recourse," 1995, technical report, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.49.4279.

[44] A. Bordes, L. Bottou, and P. Gallinari, "SGD-QN: Careful quasi-Newton stochastic gradient descent," *J. Mach. Learn. Res.*, vol. 10, pp. 1737–1754, Jul. 2009.

[45] J. Yu, S. V. N. Vishwanathan, S. Günter, and N. N. Schraudolph, "A quasi-Newton approach to nonsmooth convex optimization problems in machine learning," *J. Mach. Learn. Res.*, vol. 11, pp. 1145–1200, Mar. 2010.

[46] R. H. Byrd, S. L. Hansen, J. Nocedal, and Y. Singer, "A stochastic quasi-Newton method for large-scale optimization," 2014, technical report, http://arxiv.org/abs/1401.7020.

[47] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Statist. Soc. B*, vol. 67, no. 2, pp. 301–320, 2005.

[48] M. Allain, J. Idier, and Y. Goussard, "On global and local convergence of half-quadratic algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1130–1142, May 2006.

[49] J. M. Bioucas-Dias and M. A. T. Figueiredo, "A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2992–3004, Dec. 2007.

[50] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[51] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Detroit, MI, May 9-12 1995, vol. 5, pp. 3023–3026.

[52] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and Adaptive Signal Processing*, Artech House, Norwood, MA, 2005.

[53] A. Defazio, J. Domke, and T. Cartano, "Finito: A faster, permutable incremental gradient method for big data problems," in *Proc. 31st Int. Conf. Mach. Learn.*, Beijing, China, Jun. 21-26 2014, pp. 1125–1133.

[54] Y. Chen, Y. Gu, and A. O. Hero, "Regularized least-mean-square algorithms," pp. x–x+7, 2010, http://arxiv.org/pdf/1012.5066.pdf.

[55] S. Werner, J.A. Apolinario, and P. S. R. Diniz, "Set-membership proportionate affine projection algorithms," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2007, no. 1, pp. 034242, 2007.

[56] J. Davidson, *Stochastic Limit Theory*, Oxford University Press, New York, 1994.

[57] W. F. Stout, "The Hartman-Wintner law of the iterated logarithm for martingales," *Ann. Math. Stat.*, vol. 41, no. 6, pp. 2158–2160, 1970.

[58] T. L. Lai, "Martingales in sequential analysis and time series, 1945-1985," *J. Electron. Hist. Probab. Stat.*, vol. 5, no. 1, pp. x–x+30, June 2009.

[59] H. Robbins and D. Siegmund, "A convergence theorem for non negative almost supermartingales and some applications," in *Optimizing Methods in Statistics*, J. S. Rustagi, Ed., pp. 233–257. Academic Press, New York, 1971.

[60] A. M. Ostrowski, *Solution of Equations in Euclidean and Banach Spaces*, Academic Press, London, 1973.