Signal Processing 152 (2018) 373–383

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Model-driven online parameter adjustment for zero-attracting LMS[☆]

Danqi Jin[a], Jie Chen[a,*], Cédric Richard[b], Jingdong Chen[a]

[a] Centre of Intelligent Acoustics and Immersive Communications, School of Marine Science and Technology, Northwestern Polytechinical University, Xi'an, China

[b] Université de Nice - Sophia Antipolis, CNRS, France

ARTICLE INFO

Article history:
Received 11 September 2017
Revised 2 June 2018
Accepted 19 June 2018
Available online 20 June 2018

Keywords:
Sparse system identification
Transient behavior model
Variable parameter strategy
Adaptive algorithms
ZA-LMS
RZA-LMS
Complex-valued signal

ABSTRACT

Zero-attracting least-mean-square (ZA-LMS) algorithm has been widely used for online sparse system identification. Similarly to most adaptive filtering algorithms and sparsity-inducing regularization techniques, ZA-LMS appears to face a trade-off between convergence speed and steady-state performance, and between sparsity level and estimation bias. It is therefore important, but not trivial, to optimally set the algorithm parameters. To address this issue, a variable-parameter ZA-LMS algorithm is proposed in this paper, based on a model of the stochastic transient behavior of the ZA-LMS. By minimizing the excess mean-square error (EMSE) at each iteration on the basis of a white input assumption, we obtain closed-form expression of the step-size and regularization parameter. To improve the performance, we introduce the same strategy for the reweighted ZA-LMS (RZA-LMS). Simulation results illustrate the effectiveness of the proposed algorithms and highlight their performance through comparisons with state-of-the-art algorithms, in the case of white and correlated inputs.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

Adaptive filtering methods are powerful tools for online system identification [1,2]. Within the myriad of algorithms proposed in the literature, the least-mean-square (LMS) algorithm has been widely used since it is robust and provides reasonably good performance with low computational complexity. Several applications have recently shown the need for online sparse identification techniques. A driving force behind the development of such algorithms is, for instance, the channel estimation problem because, although the number of coefficients of the impulse response can be large, only a few of them may have significant values. It is therefore important to endow the conventional LMS algorithm with the ability to provide enhanced performance for such scenarios.

In recent years, several algorithms based on the LMS were proposed to promote the sparsity of the estimate. The proportionate normalized LMS (PNLMS) [3] and its variant called improved PNLMS (IPNLMS) [4] update each filter coefficient independently by adjusting the adaptation step-size in proportion to the estimated filter coefficient. Another family of sparsity-inducing algorithms is motivated by the compressive sensing theory, which provides a unified framework for estimating sparse signals [5,6]. In place of the ℓ_0-norm, which provides an exact count of the non-zero coefficients but leads to NP-hard optimization problems (non-deterministic polynomial-time solvable decision problems), other sparsity-inducing norms can be used as a surrogate to overcome this difficulty [7]. The use of the ℓ_1-norm is a popular choice [8]. For instance, the authors in [9] consider an ℓ_1-norm regularizer, and introduce the zero-attracting LMS and the reweighted zero-attracting LMS for sparse system identification. It is shown that the ZA-LMS and the RZA-LMS perform better than the LMS in sparse scenarios. However adjusting the algorithm parameters, including the step size and the regularization parameter, remains a tricky task. On the one hand, as for usual adaptive algorithms, the step-size plays a crucial role to control the trade-off between the convergence speed and the asymptotic performance. A small step-size leads to slower convergence but improved asymptotic performance, while a large step-size leads to faster convergence but at the cost of a higher power of the residual error, or even instability of the algorithm [1,2]. On the other hand, the regularization parameter controls the trade-off between the sparsity of the estimate and the estimation bias. A large regularization parameter associated with the ℓ_1-norm strongly promotes the sparsity of the solution. This however causes a larger bias of the non-zero parameter vector entries. Reweighted ℓ_1-regularization allows to reduce this bias. However, an improper value of the regularization parameter

☆ The work of Jie Chen was supported in part by NSFC grants 61671382 and 61811530283, and 111 project (B18041). The work of Cédric Richard was supported in part by ANR and in part by DGA under Grant ANR-13-ASTR-0030 (ODISSEE Project). The work of Jingdong Chen was supported in part by NSFC grant 61425005.

* Corresponding author.

E-mail addresses: danqijin@mail.nwpu.edu.cn (D. Jin), dr.jie.chen@ieee.org (J. Chen), cedric.richard@unice.fr (C. Richard), jingdongchen@ieee.org (J. Chen).

https://doi.org/10.1016/j.sigpro.2018.06.020
0165-1684/© 2018 Elsevier B.V. All rights reserved.

may even worsen the estimation performance. Though techniques such as regularization path and cross validation help characterize the influence of this parameter [10], they are inappropriate for online learning settings.

Variable parameter strategies provide simple but efficient solutions for optimizing the trade-off between fast convergence and low misadjustment [11]. For LMS, several variable step-size strategies have been proposed in the literature to address this issue. In most cases, the step-size adapts over time depending on the estimation error. Related works include [11–13]. A variable step-size version of the PNLMS, called NPVSS-IPNLMS, is proposed in [11]. It combines the IPNLMS and a variable step-size NLMS (VSS-NLMS) strategy [14]. However, while achieving a lower misadjustment, the convergence speed of NPVSS-IPNLMS slows down significantly after an initial phase. The zero-attracting variable step-size LMS (ZA-VSSLMS) and the reweighted zero-attracting variable step-size LMS (RZA-VSSLMS) introduced in [12] use the variable step-size strategy reported in [15]. A significant improvement in the convergence rate as well as in the misadjustment error can be observed. Another variable step-size RZA-LMS strategy based on a nonlinear relationship between the step-size and the power of the noise-free prior error, called VSS-RZA-LMS, is considered in [13]. Nevertheless, the misadjustment improvement appears to be limited. It is worth noting that some extra parameters are introduced into all these algorithms, but setting their proper values is a nontrivial task, similar to the selection of an appropriate step size.

Motivated by our recent work [16], where a new model is derived for the transient behavior of the ZA-LMS algorithm, we propose in this paper to design a variable-parameter ZA-LMS (VP-ZA-LMS) algorithm where the step-size and the regularization parameter are both adjusted in an online manner. Unlike heuristic strategies considered in the literature, our method is based on an optimization step that minimizes the EMSE at each iteration. Indeed, it turns out to be a quadratic function of the step-size and the regularization parameter when considering the transient model in [16] under a white input assumption. This yields closed-form expressions of the step-size and regularization parameter at each iteration, leading to a faster convergence as well as a lower misadjustment. To further improve the performance, we apply this strategy to the RZA-LMS, leading to a variable-parameter RZA-LMS (VP-RZA-LMS) algorithm. Simulation results illustrate the enhanced performance of our algorithms compared with ZA-LMS, RZA-LMS and other variable step-size algorithms used in sparse system identification applications. We summarize the contributions of this work as follows:

1. Compared to the existing literatures, this work is the first one that derives a variable-parameter strategy based on a theoretical model of the filter performance. The proposed algorithm jointly adjust the step-size and regularization parameter in some optimal sense.
2. Unlike existing works on ZA-LMS that focus on the real-valued data case, we derive an extension to complex-valued systems.
3. While working well for ZA-LMS/RZA-LMS, the proposed framework can be extended to several other adaptive filters having similar structure, such as the LMS with ℓ_0-norm penalty, the group ZA-LMS, etc.

Before proceeding, note that this work and [16] are both related to the transient behavior of the ZA-LMS algorithm but they address different issues. The analysis in [16] focuses on how deriving an accurate model for the transient behavior of ZA-LMS. The current work uses an approximate model that allows us to automatically adjust the algorithm parameters in an online way.

The rest of this paper is organized as follows. Section 2 reviews the ZA-LMS and RZA-LMS algorithms. The VP-ZA-LMS and VP-RZA-LMS algorithms are derived in Sections 3 and 4, respectively. In Section 5, computer simulations are performed to validate the proposed algorithms and to show their superior performance. Section 6 concludes the paper.

Notation. Normal font x denotes scalars. Boldface small letters $\boldsymbol{x}$ denote column vectors. All vectors are column vectors. Boldface capital letters $\boldsymbol{X}$ denote matrices. The superscript $(\cdot)^\top$ denotes the transpose of a matrix or a vector. The inverse of a square matrix is denoted by $(\cdot)^{-1}$. All-zero vector and all-one vector of length N are denoted by $\boldsymbol{0}_N$ and $\boldsymbol{1}_N$, respectively. The Gaussian distribution with mean μ and variance σ^2 is denoted by $\mathcal{N}(\mu, \sigma^2)$. The operator $\text{sgn}\{\cdot\}$ takes the sign of the entries of its argument. The operator $\text{tr}\{\cdot\}$ takes the trace of its matrix argument. The operator $|\cdot|$ takes the absolute value of the entries of its argument. The mathematical expectation is denoted by $\mathbb{E}\{\cdot\}$. The operators $\max\{\cdot, \cdot\}$ and $\min\{\cdot, \cdot\}$ take the maximum and minimum value of their arguments, respectively.

2. System model and zero-attracting LMS

2.1. System model and zero-attracting LMS

To be consistent with ZA-LMS/RZA-LMS framework, and for the sake of simplicity, we start by deriving our parameter adjustment strategies in the case of real-valued signals. In Appendix C, we extend ZA-LMS and RZA-LMS to complex-valued data, and then derive the associated parameter adjustment strategies in a concise manner. Consider an unknown system with input-output relation characterized by the linear model

$$y_n = \boldsymbol{x}_n^\top \boldsymbol{w}^\star + z_n \tag{1}$$

with $\boldsymbol{w}^\star \in \mathbb{R}^L$ denoting an unknown parameter vector, and $\boldsymbol{x}_n \in \mathbb{R}^L$ a regression vector with a positive definite covariance matrix $\boldsymbol{R}_x = \mathbb{E}\{\boldsymbol{x}_n\boldsymbol{x}_n^\top\} > 0$ at instant n. The regression vector $\boldsymbol{x}_n$ and the output signal y_n are assumed to be zero mean. The error signal z_n is assumed to be stationary, independent and identically distributed (i.i.d.), with zero mean and variance σ_z^2, and independent of any other signal. Let $J(\boldsymbol{w})$ denote the mean-square-error (MSE) cost, namely,

$$J(\boldsymbol{w}) = \frac{1}{2}\,\mathbb{E}\left\{[y_n - \boldsymbol{w}^\top \boldsymbol{x}_n]^2\right\}. \tag{2}$$

It is clear from (1) that $J(\boldsymbol{w})$ is minimized at $\boldsymbol{w}^\star$.

The problem considered in this paper is to estimate the unknown parameter vector $\boldsymbol{w}^\star$, which is assumed to be sparse [3,17,18]. This problem can be addressed by minimizing the following regularized MSE cost:

$$\begin{aligned} \boldsymbol{w}_{\text{ZA}}^o &= \arg\min_{\boldsymbol{w}} J_{\text{ZA}}(\boldsymbol{w}) \\ &\text{with } J_{\text{ZA}}(\boldsymbol{w}) = \frac{1}{2}\,\mathbb{E}\left\{[y_n - \boldsymbol{w}^\top \boldsymbol{x}_n]^2\right\} + \lambda\|\boldsymbol{w}\|_1, \end{aligned} \tag{3}$$

where the ℓ_1-norm term, defined as $\|\boldsymbol{w}\|_1 = \sum_{i=1}^L |w_i|$, is used to promote the sparsity of the estimate, and $\lambda \geq 0$ is the regularization parameter. A subgradient of $J_{\text{ZA}}(\boldsymbol{w})$ in problem (3) is given by:

$$\partial J_{\text{ZA}}(\boldsymbol{w}) = \boldsymbol{R}_x\boldsymbol{w} - \boldsymbol{p}_{xy} + \lambda\,\text{sgn}\{\boldsymbol{w}\} \tag{4}$$

where $\boldsymbol{p}_{xy} = \mathbb{E}\{\boldsymbol{x}_n y_n\}$ is the correlation vector between $\boldsymbol{x}_n$ and y_n. Using the instantaneous approximations $\boldsymbol{R}_x \approx \boldsymbol{x}_n\boldsymbol{x}_n^\top$ and $\boldsymbol{p}_{xy} \approx \boldsymbol{x}_n y_n$, the subgradient iteration leads to the ZA-LMS algorithm as derived in [9]:

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu\, e_n\boldsymbol{x}_n - \rho\,\text{sgn}\{\boldsymbol{w}_n\}, \tag{5}$$

where e_n is the estimation error given by:

$$e_n = y_n - \boldsymbol{w}_n^\top \boldsymbol{x}_n, \tag{6}$$

μ is a positive step-size, and $\rho = \mu\lambda$.

As the shrinkage parameter ρ in ZA-LMS algorithm does not distinguish between zero and non-zero entries of $\boldsymbol{w}_n$, the zero-attracting term $\rho\,\text{sgn}\{\boldsymbol{w}_n\}$ results in significant bias for large entries while promoting zero-valued ones. This behavior significantly degrades MSE performance [9]. To get enhanced performance in sparse system identification, the RZA-LMS was proposed to reinforce the zero attractor. Consider the following optimization problem:

$$\boldsymbol{w}^o_{\text{RZA}} = \arg\min_{\boldsymbol{w}} J_{\text{RZA}}(\boldsymbol{w})$$
$$\text{with } J_{\text{RZA}}(\boldsymbol{w}) = \frac{1}{2}\,\mathbb{E}\left\{\left[y_n - \boldsymbol{w}^\top \boldsymbol{x}_n\right]^2\right\} + \lambda \sum_{i=1}^{L} \log\left(1 + |w_i|/\varepsilon\right). \quad (7)$$

The log-sum penalty $\sum_{i=1}^{L}\log(1+|w_i|/\varepsilon)$ is considered as it behaves more similarly to the ℓ_0-norm than $\|\boldsymbol{w}\|_1$ [6]. Similarly to the ZA-LMS, using stochastic subgradient iterations yields the RZA-LMS update:

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu\, e_n \boldsymbol{x}_n - \rho\,\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}, \quad (8)$$

where the division and the absolute value operator $|\cdot|$ in the third term on r.h.s. of (8) are applied in an element-wise manner, and ε is a small positive parameter.

2.2. Motivation for using a variable-parameter strategy

For ZA-LMS and RZA-LMS algorithms, the trade-off between misadjustment and adaptation rate is mainly driven by the step-size μ, and the trade-off between sparsity and estimation bias by the regularization parameter λ, or equivalently ρ. In order that ZA-LMS and RZA-LMS can provide accurate estimation results in unknown environments, without using prior information, an efficient variable-parameter strategy can be useful. Such a strategy is expected to satisfy the following requirements:

- It should not introduce a significant number of extra parameters.
- Selecting an appropriate, if not optimal, value for a newly introduced parameter should be easier than selecting an appropriate step-size, or any other parameter in the original algorithm. Furthermore, the performance of the algorithm should not be very sensitive to these values.
- The computational complexity of evaluating the time-variant parameters should be of the same order as the adaptive algorithm. For instance, a step size determined in $\mathcal{O}(L^2)$ does not make sense for an LMS-type algorithm in $\mathcal{O}(L)$.

These requirements may rule out several strategies in the literature, though they provide an efficient solution for the conflicting requirements of fast convergence and low misadjustment. We will see that our strategy complies with these requirements.

3. Parameter design of ZA-LMS guided by a transient behavior model

3.1. Transient behavior model of ZA-LMS

Defining the weight error vector $\tilde{\boldsymbol{w}}_n$ as the difference between the estimated weight vector $\boldsymbol{w}_n$ and $\boldsymbol{w}^\star$, namely

$$\tilde{\boldsymbol{w}}_n = \boldsymbol{w}_n - \boldsymbol{w}^\star, \quad (9)$$

the analysis of ZA-LMS consists of studying the evolution of the first and second-order moments of $\tilde{\boldsymbol{w}}_n$ over time. To keep the calculations mathematically tractable, we introduce the independence assumption, which is commonly used when analyzing adaptive filtering algorithms [1]:

A1: The weight-error vector $\tilde{\boldsymbol{w}}_n$ is statistically independent of the input vector $\boldsymbol{x}_n$.

Subtracting $\boldsymbol{w}^\star$ from both sides of (5), and using $e_n = z_n - \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n$, yields the update relation of $\tilde{\boldsymbol{w}}_n$:

$$\tilde{\boldsymbol{w}}_{n+1} = \tilde{\boldsymbol{w}}_n + \mu\,\boldsymbol{x}_n z_n - \mu\,\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n - \rho\,\text{sgn}\{\boldsymbol{w}_n\}. \quad (10)$$

The condition for stability and the mean performance of ZA-LMS are analyzed in [9]. With the independence assumption **A1** and $e_n = z_n - \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n$, the mean-square-error (MSE) of the ZA-LMS is given by

$$\mathbb{E}\{e_n^2\} = \sigma_z^2 + \text{tr}\{\boldsymbol{R}_x \boldsymbol{K}_n\}, \quad (11)$$

with $\boldsymbol{K}_n = \mathbb{E}\{\tilde{\boldsymbol{w}}_n \tilde{\boldsymbol{w}}_n^\top\}$. The quantity $\text{tr}\{\boldsymbol{R}_x\boldsymbol{K}_n\}$ is the excess-mean-square-error (EMSE) at time instant n, denoted by ζ_n. The trace of $\boldsymbol{K}_n$ is the mean-square-deviation (MSD), denoted by $\xi_n = \text{tr}\{\boldsymbol{K}_n\}$. To simplify the derivation, we introduce the whiteness assumption **A2**:

A2: The input signal $\boldsymbol{x}_n$ is a zero-mean white Gaussian signal with covariance matrix $\boldsymbol{R}_x = \sigma_x^2 \boldsymbol{I}$.

Correlation of the regressors usually makes the analysis of adaptive algorithms difficult [1]. Hence some analyses in the literature restrict themselves to this white input setting. The derivation of our variable-parameter strategies would become highly challenging without assumption **A2**. However, it turns out that the resulting algorithms continue to perform well with correlated inputs even when assumption **A2** does not hold.

Under the assumption **A2**, the MSD is equal to EMSE up to a scaling factor, that is,

$$\zeta_n = \sigma_x^2\,\text{tr}\{\boldsymbol{K}_n\} = \sigma_x^2\,\xi_n. \quad (12)$$

Therefore, we need to determine a recursion for $\boldsymbol{K}_n$ in order to relate the MSD or EMSE at two consecutive time instants n and $n+1$. Post-multiplying (10) by its transpose, taking the expectation, and using assumptions **A1** and **A2**, we get:

$$\boldsymbol{K}_{n+1} = \boldsymbol{K}_n + \mu^2\sigma_z^2\boldsymbol{R}_x + \mu^2\boldsymbol{Q}_1 + \rho^2\boldsymbol{Q}_2 - \mu\left(\boldsymbol{Q}_3 + \boldsymbol{Q}_3^\top\right)$$
$$- \rho\left(\boldsymbol{Q}_4 + \boldsymbol{Q}_4^\top\right) + \mu\rho\left(\boldsymbol{Q}_5 + \boldsymbol{Q}_5^\top\right) \quad (13)$$

with

$$\boldsymbol{Q}_1 = \mathbb{E}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top\right\} \quad (14)$$

$$\boldsymbol{Q}_2 = \mathbb{E}\left\{\text{sgn}\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\text{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\right\} \quad (15)$$

$$\boldsymbol{Q}_3 = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top\right\} \quad (16)$$

$$\boldsymbol{Q}_4 = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n \text{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\right\} \quad (17)$$

$$\boldsymbol{Q}_5 = \mathbb{E}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \text{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\right\}. \quad (18)$$

As for the above matrices $\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_5$, we will sometimes drop the explicit reference to time instant n in order to keep the notation compact.

Characterizing the evolution of terms $\boldsymbol{Q}_2$, $\boldsymbol{Q}_4$ and $\boldsymbol{Q}_5$ in an exact manner is not trivial, as it is necessary to evaluate terms involving the nonlinear $\text{sgn}\{\cdot\}$ operator. To address this difficulty, the authors in [19] use a zero-order approximation of these terms by substituting the expectation of the product of two factors involving a sign function by the product of their expectations. In our recent work [16], we succeeded in evaluating these terms in an exact manner using a mild and reasonable joint Gaussian assumption. However, the aforementioned works involve cumbersome calculations that are not suitable for an online implementation with linear complexity.

3.2. Parameter design using a transient behavior model

We now derive a parameter design strategy for ZA-LMS based on its transient behavior model. Given the MSD ξ_n at time instant n, we seek the parameters that minimize the MSD at time instant $n+1$, that is,

$$\{\mu_n^\star, \rho_n^\star\} = \arg\min_{\mu,\rho} \xi_{n+1}. \tag{19}$$

Using the recursion (13), and considering that $\xi_n = \mathrm{tr}\{\boldsymbol{K}_n\}$, the above optimization problem becomes:

$$\begin{aligned}\{\mu_n^\star, \rho_n^\star\} &= \arg\min_{\mu,\rho} \mathrm{tr}\{\boldsymbol{K}_{n+1}\} \\ &= \arg\min_{\mu,\rho} \mathrm{tr}\{\boldsymbol{K}_n\} + \mu^2\big[\sigma_z^2\mathrm{tr}\{\boldsymbol{R}_x\} + \mathrm{tr}\{\boldsymbol{Q}_1\}\big] + \rho^2\,\mathrm{tr}\{\boldsymbol{Q}_2\} \\ &\quad - 2\mu\,\mathrm{tr}\{\boldsymbol{Q}_3\} - 2\rho\,\mathrm{tr}\{\boldsymbol{Q}_4\} + 2\mu\rho\,\mathrm{tr}\{\boldsymbol{Q}_5\}.\end{aligned} \tag{20}$$

For the sake of notation, we define the following quantities:

$$a = \sigma_z^2\mathrm{tr}\{\boldsymbol{R}_x\} + \mathrm{tr}\{\boldsymbol{Q}_1\} \tag{21}$$

$$b = \mathrm{tr}\{\boldsymbol{Q}_2\} \tag{22}$$

$$c = \mathrm{tr}\{\boldsymbol{Q}_5\} \tag{23}$$

$$p_1 = \mathrm{tr}\{\boldsymbol{Q}_3\} \tag{24}$$

$$p_2 = \mathrm{tr}\{\boldsymbol{Q}_4\}. \tag{25}$$

The objective function in matrix form can be written as:

$$\xi_{n+1} = [\mu\ \rho]\,\boldsymbol{H}\,[\mu\ \rho]^\top - 2\,[p_1\ p_2]\,[\mu\ \rho]^\top + \xi_n \tag{26}$$

with

$$\boldsymbol{H} = \begin{bmatrix} a & c \\ c & b \end{bmatrix}, \tag{27}$$

which is a quadratic function of $[\mu\ \rho]$.

Lemma 1. *The Hessian matrix* $\boldsymbol{H}$ *of (26) is positive semidefinite.*

As shown in Appendix A, matrix $\boldsymbol{H}$ can be written as the sum of a covariance matrix of arbitrary variables and a positive semidefinite matrix. Then $\boldsymbol{H}$ is positive semidefinite. For simplification purposes, we shall further assume that $\boldsymbol{H}$ is positive definite since a covariance matrix is almost always positive definite in practice [20]. Positive definiteness of $\boldsymbol{H}$ allows us to determine the optimal parameters that minimize the cost (20) via:[1]

$$[\mu_n^\star\ \rho_n^\star]^\top = \boldsymbol{H}^{-1}[p_1\ p_2]^\top, \tag{28}$$

namely,

$$\mu_n^\star = \frac{bp_1 - cp_2}{ab - c^2} \tag{29}$$

$$\rho_n^\star = \frac{ap_2 - cp_1}{ab - c^2}. \tag{30}$$

The above result cannot be used in practice since it requires statistics that are not available within an online environment. We now approximate these quantities in order to provide a parameter adjustment strategy with reasonable complexity. The subscript n in the variables a_n, b_n, c_n, p_{1n} and p_{2n} is now needed to make explicit time-dependence.

[1] Use the Moore–Penrose pseudoinverse of $\boldsymbol{H}$ otherwise.

Under assumption **A2**, the quantity a_n can be evaluated as follows:

$$a_n = \sigma_z^2\,\mathrm{tr}\{\sigma_x^2\boldsymbol{I}\} + \mathrm{tr}\Big\{2\boldsymbol{R}_x\boldsymbol{K}_n\boldsymbol{R}_x + \mathrm{tr}\{\boldsymbol{R}_x\boldsymbol{K}_n\}\boldsymbol{R}_x\Big\} \tag{31}$$

$$= \sigma_z^2\sigma_x^2 L + (2+L)\sigma_x^2\,\zeta_n. \tag{32}$$

Since the diagonal entries of $\boldsymbol{Q}_2$ are squares of a sign function, the quantity b_n is given by:

$$b_n = L. \tag{33}$$

We then evaluate the trace of $\boldsymbol{Q}_5$. Using the independence assumption **A1** yields:

$$c_n = \sigma_x^2\,\mathbb{E}\big\{\tilde{\boldsymbol{w}}_n^\top\,\mathrm{sgn}\{\boldsymbol{w}_n\}\big\}. \tag{34}$$

The weight error vector $\tilde{\boldsymbol{w}}_n = \boldsymbol{w}_n - \boldsymbol{w}^\star$ in the above relation cannot be evaluated since it requires to know $\boldsymbol{w}^\star$, namely, the minimizer of the MSE cost (2). Let us now construct an approximation of $\boldsymbol{w}^\star$ at time instant n to be used in $\tilde{\boldsymbol{w}}_n$. As already experienced successfully in another context [21], one strategy is to use a local one-step approximation of the form:

$$\hat{\boldsymbol{w}}_n^\star = \boldsymbol{w}_n - \eta_n\nabla J(\boldsymbol{w}_n) \tag{35}$$

where η_n is a positive step-size to be determined. Given the MSD ξ_n at time instant n, we seek η_n that minimizes ξ_{n+1}. Following the same reasoning as (19)–(30) leads to $\eta_n = p_{1n}/a_n$. Since the true gradient of $J(\boldsymbol{w})$ at $\boldsymbol{w}_n$ is not available in an adaptive implementation, we can approximate it by using the instantaneous value $-e_n\boldsymbol{x}_n$. Finally, we write:

$$\hat{\boldsymbol{w}}_n^\star = \boldsymbol{w}_n - \boldsymbol{g}_n \tag{36}$$

with $\boldsymbol{g}_n = -\frac{p_{1n}}{a_n}e_n\boldsymbol{x}_n$. Then, approximating the expectation in (34) by its instantaneous argument yields:

$$c_n \approx \sigma_x^2\,\boldsymbol{g}_n^\top\,\mathrm{sgn}\{\boldsymbol{w}_n\}. \tag{37}$$

The quantity p_{1n} is given by:

$$p_{1n} = \zeta_n. \tag{38}$$

Finally, as for quantity p_{2n}, we have:

$$p_{2n} \approx \boldsymbol{g}_n^\top\,\mathrm{sgn}\{\boldsymbol{w}_n\}. \tag{39}$$

As the EMSE ζ_n is not available, and depends on the unknown parameter vector $\boldsymbol{w}^\star$, we adopt the estimator $\hat{\zeta}_n$ for ζ_n:

$$\hat{e}_n = \beta\hat{e}_{n-1} + (1-\beta)e_n \tag{40}$$

$$\hat{\zeta}_n = \max\{\hat{e}_n^2 - \sigma_z^2,\ 0\}, \tag{41}$$

which provides an instantaneous approximation of the EMSE, with β being a temporal smoothing factor in the interval [0,1). To further improve the estimation accuracy of ζ_n, we set $\zeta_{n_{\min}}$ by iterating $\hat{\zeta}_{n-1}$ via (20) at iteration $n-1$ as a lower bound for ζ_n, since we have minimized $\mathrm{tr}\{\boldsymbol{K}_n\}$ with respect to $\{\mu, \rho\}$ at iteration $n-1$. Due to the approximation introduced in the theoretical derivation and the intrinsic properties of signal and noise realization, ζ_n is no less than $\sigma_x^2\,\mathrm{tr}\{\boldsymbol{K}_n\}$. Rather than (41), we then suggest to use:

$$\hat{\zeta}_n = \max\big\{\hat{e}_n^2 - \sigma_z^2,\ \zeta_{n_{\min}}\big\}. \tag{42}$$

Non-negativity of μ and ρ is required. We did not consider this constraint in (20) in order to get closed-form solutions as given by (29) and (30). To overcome undesirable behavior of the algorithm, we then need to apply the following hard thresholding operators:

$$\mu_n^\star = \max\{\mu_n^\star,\ 0\} \tag{43}$$

$$\rho_n^\star = \max\{\rho_n^\star, 0\} \tag{44}$$

We further impose a temporal smoothing with smoothing factor γ over parameters $\mu_n^\star$ and $\rho_n^\star$, as well as a possible predefined upper bound $\mu_{\max}$ on the step-size to ensure the stability of the algorithm:

$$\mu_n = \min\{\gamma\mu_{n-1} + (1-\gamma)\mu_n^\star, \mu_{\max}\} \tag{45}$$

$$\rho_n = \gamma\rho_{n-1} + (1-\gamma)\rho_n^\star. \tag{46}$$

4. Parameter design of RZA-LMS guided by a transient behavior model

4.1. Transient behavior model of RZA-LMS

In order to derive a variable-parameter strategy for RZA-LMS algorithm, we first extend the transient behavior model of ZA-LMS derived in [16] to RZA-LMS. As for ZA-LMS, we consider assumptions **A1** and **A2** to make the derivation tractable. Subtracting $\boldsymbol{w}^\star$ from both sides of (8), and using $e_n = z_n - \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n$, yields the update relation of $\tilde{\boldsymbol{w}}_n$:

$$\tilde{\boldsymbol{w}}_{n+1} = \tilde{\boldsymbol{w}}_n + \mu\,\boldsymbol{x}_n z_n - \mu\,\boldsymbol{x}_n\boldsymbol{x}_n^\top\tilde{\boldsymbol{w}}_n - \rho\,\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}. \tag{47}$$

Taking the expectation of (47), we get:

$$\mathbb{E}\{\tilde{\boldsymbol{w}}_{n+1}\} = \mathbb{E}\{\tilde{\boldsymbol{w}}_n\} - \mu\,\boldsymbol{R}_x\mathbb{E}\{\tilde{\boldsymbol{w}}_n\} - \rho\,\mathbb{E}\left\{\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}\right\}. \tag{48}$$

Similarly to (11) and (12), the MSE of RZA-LMS is given by

$$\mathbb{E}\{e_n^2\} = \sigma_z^2 + \text{tr}\{\boldsymbol{R}_x\boldsymbol{K}_n\}, \tag{49}$$

and its MSD ξ_n is equal to the EMSE ζ_n up to a scaling factor, that is,

$$\zeta_n = \sigma_x^2\,\text{tr}\{\boldsymbol{K}_n\} = \sigma_x^2\,\xi_n. \tag{50}$$

Likewise, $\boldsymbol{K}_n$ can be calculated recursively as follows:

$$\begin{aligned}\boldsymbol{K}_{n+1} = \boldsymbol{K}_n + \mu^2\sigma_z^2\boldsymbol{R}_x + \mu^2\boldsymbol{Q}_1 + \rho^2\boldsymbol{Q}_2 - \mu\left(\boldsymbol{Q}_3 + \boldsymbol{Q}_3^\top\right)\\ - \rho\left(\boldsymbol{Q}_4 + \boldsymbol{Q}_4^\top\right) + \mu\rho\left(\boldsymbol{Q}_5 + \boldsymbol{Q}_5^\top\right)\end{aligned} \tag{51}$$

where $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_3$ are given by (14) and (16), respectively, and the reweighting terms in $\boldsymbol{Q}_2$, $\boldsymbol{Q}_4$ and $\boldsymbol{Q}_5$ by:

$$\boldsymbol{Q}_2 = \mathbb{E}\left\{\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}\,\frac{\text{sgn}^\top\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n^\top|}\right\} \tag{52}$$

$$\boldsymbol{Q}_4 = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n\,\frac{\text{sgn}^\top\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n^\top|}\right\} \tag{53}$$

$$\boldsymbol{Q}_5 = \mathbb{E}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^\top\tilde{\boldsymbol{w}}_n\,\frac{\text{sgn}^\top\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n^\top|}\right\}. \tag{54}$$

4.2. Parameter design with transient behavior model

We derive now a parameter design strategy for RZA-LMS based on a simplified model of its transient behavior model. Minimizing the MSD ξ_{n+1} with respect to parameters μ and ρ, based on recursion (51), leads to:

$$\{\mu_n^\star, \rho_n^\star\} = \arg\min_{\mu,\rho}\,\xi_{n+1} \tag{55}$$

$$= \arg\min_{\mu,\rho}\,\text{tr}\{\boldsymbol{K}_{n+1}\} \tag{56}$$

$$\begin{aligned}= \arg\min_{\mu,\rho}\,\text{tr}\{\boldsymbol{K}_n\} + \mu^2\left[\sigma_z^2\text{tr}\{\boldsymbol{R}_x\} + \text{tr}\{\boldsymbol{Q}_1\}\right] + \rho^2\,\text{tr}\{\boldsymbol{Q}_2\}\\ - 2\mu\,\text{tr}\{\boldsymbol{Q}_3\} - 2\rho\,\text{tr}\{\boldsymbol{Q}_4\} + 2\mu\rho\,\text{tr}\{\boldsymbol{Q}_5\}.\end{aligned} \tag{57}$$

Following the reasoning as in Section 3.2 leads to:

$$\mu_n^\star = \frac{bp_1 - cp_2}{ab - c^2} \tag{58}$$

$$\rho_n^\star = \frac{ap_2 - cp_1}{ab - c^2}. \tag{59}$$

Likewise, we approximate the expectations that are not accessible to provide a parameter adjustment strategy with appropriate computational complexity. The subscript n in the variables a_n, b_n, c_n, p_{1n} and p_{2n} is now needed to make explicit time-dependence. Since $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_3$ have the same expressions as for the ZA-LMS, we approximate a_n and p_{1n} by (32) and (38), respectively. Using (52) and approximating the expectation by its instantaneous argument, b_n is given by:

$$b_n \approx \sum_{i=1}^{L}\left(\frac{1}{\varepsilon + |w_{n,i}|}\right)^2, \tag{60}$$

where $w_{n,i}$ is the i-th entry of vector $\boldsymbol{w}_n$. We then evaluate the trace of $\boldsymbol{Q}_5$. Using the independence assumption yield:

$$c_n = \sigma_x^2\,\mathbb{E}\left\{\frac{\text{sgn}^\top\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n^\top|}\,\tilde{\boldsymbol{w}}_n\right\}. \tag{61}$$

As for ZA-LMS, we approximate $\tilde{\boldsymbol{w}}_n$ by the stochastic gradient of the cost function at time instant n, that is, $\boldsymbol{g}_n = -\frac{p_{1n}}{a_n}\,e_n\boldsymbol{x}_n$. Then, approximating the expectation by its instantaneous argument yields:

$$c_n \approx \sigma_x^2\,\boldsymbol{g}_n^\top\,\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}. \tag{62}$$

In a similar way to the evaluation of c_n, we have:

$$p_{2n} \approx \boldsymbol{g}_n^\top\,\frac{\text{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}. \tag{63}$$

Next, we adopt the same approximation for ζ_n as in (40)–(42). As in (43), (44), we impose $\mu_n^\star$ and $\rho_n^\star$ to be nonnegative. Finally, we apply a temporal smoothing as in (45)–(46), as well as possibly a predefined upper $\mu_{\max}$ on the step-size to ensure the stability of the algorithm:

$$\mu_n = \min\{\gamma\mu_{n-1} + (1-\gamma)\mu_n^\star, \mu_{\max}\} \tag{64}$$

$$\rho_n = \gamma\rho_{n-1} + (1-\gamma)\rho_n^\star. \tag{65}$$

We now summarize the proposed VP-ZA-LMS and VP-RZA-LMS in **Algorithm 1**. Since several steps of both algorithms are identical, we combine their presentations for compactness. As can be seen in **Algorithm 1**, we introduce little extra parameters in VP-ZA-LMS and VP-RZA-LMS. More importantly, these parameters can be set pragmatically to some typical values.

5. Simulation results

Considering stationary and non-stationary system identification problems, we present now simulation results to illustrate the effectiveness of our algorithms. The input signal was a first-order AR process defined by $x_n = \alpha\,x_{n-1} + v_n$, where v_n is an i.i.d. zero-mean Gaussian variable with variance $\sigma_v^2 = 1 - \alpha^2$ (so that $\sigma_x^2 = 1$), and α is the correlation coefficient of x_n. By varying the value of α, we obtained processes x_n with different levels of correlation. The additive noise z_n was an i.i.d. zero-mean white Gaussian noise with

Algorithm 1: Variable-Parameter ZA-LMS (VP-ZA-LMS) and Variable-Parameter RZA-LMS (VP-RZA-LMS).

1 Initialize parameters with typical values, e.g.:

$$\boldsymbol{w}_1 = \boldsymbol{0}_L, \quad \rho_0 = 0, \quad \mu_0 = 0.01, \quad \bar{e}_0 = e_1, \quad \zeta_{1_{\min}} = e_1^2, \quad \gamma = 0.95, \quad \beta = 0.95, \quad \mu_{\max}$$

and repeat the following steps (iteration $n = 1, \ldots, N$);

2 Calculate the estimation error e_n of adaptive filter:

$$e_n = y_n - \boldsymbol{w}_n^\top \boldsymbol{x}_n;$$

Calculate the estimate $\hat{\zeta}_n$ at iteration n:

$$\bar{e}_n = \beta \bar{e}_{n-1} + (1-\beta) e_n, \quad \hat{\zeta}_n = \max\{\bar{e}_n^2 - \sigma_z^2, \, \zeta_{n_{\min}}\};$$

Calculate the values of a_n, and p_{1n}, at iteration n:

$$a_n = \sigma_z^2 \sigma_x^2 \, L + (2+L)\sigma_x^2 \, \hat{\zeta}_n, \quad p_{1n} = \hat{\zeta}_n;$$

Calculate the estimate $\boldsymbol{g}_n$ at iteration n:

$$\boldsymbol{g}_n = -\frac{p_{1n}}{a_n} \, e_n \, \boldsymbol{x}_n;$$

Calculate the values of b_n, c_n and p_{2n} for VP-ZA-LMS and VP-RZA-LMS at iteration n:

$$\text{VP-ZA-LMS:} \quad b_n = L, \quad c_n = \sigma_x^2 \, \boldsymbol{g}_n^\top \operatorname{sgn}\{\boldsymbol{w}_n\}, \quad p_{2n} = \boldsymbol{g}_n^\top \operatorname{sgn}\{\boldsymbol{w}_n\};$$

$$\text{VP-RZA-LMS:} \quad b_n = \sum_{i=1}^{L} \left(\frac{1}{\varepsilon + |w_{n,i}|}\right)^2, \quad c_n = \sigma_x^2 \, \boldsymbol{g}_n^\top \frac{\operatorname{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}, \quad p_{2n} = \boldsymbol{g}_n^\top \frac{\operatorname{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|};$$

Calculate the values of μ and ρ:

$$\mu_n^\star = \frac{b_n p_{1n} - c_n p_{2n}}{a_n b_n - c_n^2}, \quad \rho_n^\star = \frac{a_n p_{2n} - c_n p_{1n}}{a_n b_n - c_n^2};$$

Constrain the values of μ and ρ to be nonnegative:

$$\mu_n^\star = \max\left\{\mu_n^\star, \, 0\right\}, \quad \rho_n^\star = \max\left\{\rho_n^\star, \, 0\right\};$$

Apply temporal smoothing over parameters $\mu_n^\star$ and $\rho_n^\star$, and possibly truncate μ_n:

$$\mu_n = \min\{\gamma \mu_{n-1} + (1-\gamma)\mu_n^\star, \, \mu_{\max}\}, \quad \rho_n = \gamma \rho_{n-1} + (1-\gamma)\rho_n;$$

Update the filter coefficients and the minimal EMSE:

$$\zeta_{n+1_{\min}} = \hat{\zeta}_n + \sigma_x^2 \, (\mu_n^2 a_n + \rho_n^2 b_n - 2\mu_n p_{1n} - 2\rho_n p_{2n} + 2\mu_n \rho_n c_n),$$

$$\text{VP-ZA-LMS:} \quad \boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu_n \, e_n \boldsymbol{x}_n - \rho_n \operatorname{sgn}\{\boldsymbol{w}_n\},$$

$$\text{VP-RZA-LMS:} \quad \boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu_n \, e_n \boldsymbol{x}_n - \rho_n \frac{\operatorname{sgn}\{\boldsymbol{w}_n\}}{\varepsilon + |\boldsymbol{w}_n|}.$$

variance $\sigma_z^2 = 0.01$. In all the experiments, the initial weight vector was set to the all-zero vector. The MSD learning curves were obtained by averaging results over 100 Monte-Carlo runs.

The VP-ZA-LMS and VP-RZA-LMS were compared with the standard LMS, ZA-LMS [9], RZA-LMS [9], ZA-VSSLMS [12], WZA-VSSLMS [12] and M-VSS-RZA-LMS[2][13] algorithms. Note that ZA-LMS, ZA-VSSLMS and VP-ZA-LMS operate with a fixed regularization parameter, while RZA-LMS, WZA-VSSLMS, M-VSS-RZA-LMS and VP-RZA-LMS are provided with a reweighting strategy. The

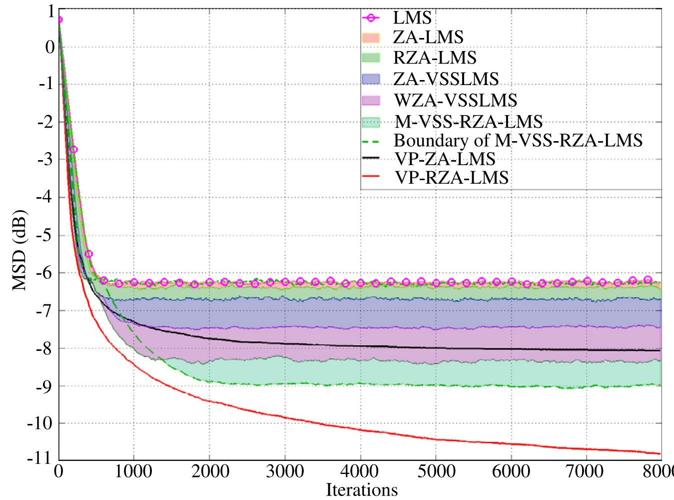

Fig. 1. Learning curves of LMS, ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS, M-VSS-RZA-LMS, VP-ZA-LMS and VP-RZA-LMS algorithms in the case of a white input Gaussian signal. For the sake of clarity, the learning curves of the M-VSS-RZA-LMS are shown in dashed green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

LMS algorithm with fixed step-size ($\mu = 0.01$) was used as a baseline. First, we set the parameters of each algorithm, except the LMS, so that *its initial convergence speed was almost the same as the LMS* and its steady-state MSD as small as possible. This gave us a first set of learning curves, one curve for each algorithm. Then, we set the parameters of each algorithm to obtain *the same steady-state MSD as the LMS* and the fastest convergence rate. This gave us a second set of learning curves, one for each algorithm. For each algorithm, these two learning curves bound a region that was used to characterize its convergence behavior compared to the LMS.

Three experiments were designed to illustrate the performance of all the algorithms with uncorrelated and correlated input signals. Time-varying systems were also considered to characterize their tracking performance. All the parameters used in the experiments are listed in Tables 2–5 of Appendix B.

5.1. Performance under a stationary system

In the first experiment, we considered an unknown system of order $L = 32$ with weights defined by:

$$\boldsymbol{w}^\star = \quad [0.8, \ 0.5, \ 0.3, \ 0.2, \ 0.1, \ 0.05, \ \boldsymbol{0}_{20}, \ -0.05, \ -0.1, \ -0.2, \ -0.3, \ -0.5, \ -0.8]^\top, \tag{66}$$

with 20 zero entries over 32. The correlation coefficient α was set to 0 so that the input signal x_n was uncorrelated and Gaussian which is consistent with our design assumption **A2**. The learning curves and performance regions of all the algorithms are provided in Fig. 1. The ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS and M-VSS-RZA-LMS outperformed the LMS.

The VP-ZA-LMS and VP-RZA-LMS algorithms are characterized by single learning curves in Fig. 1. The evolution of their step-size and regularization parameter over time is represented in Figs. 3 and 4, respectively. It is observed in Fig. 1 that the proposed VP-RZA-LMS algorithm significantly outperformed the other algorithms, by reducing the steady-state MSD and increasing the convergence speed. Though the proposed VP-ZA-LMS algorithm did not outperform all the RZA-LMS-type algorithms, it performed significantly better than the ZA-LMS-type algorithms with a faster convergence speed and a lower steady-state MSD. Moreover, the learning curves of VP-ZA-LMS and VP-RZA-LMS algorithms were still decreasing when they were stopped. It is seen in Figs. 3 and

[2] M-VSS-RZA-LMS denotes Modified-VSS-RZA-LMS algorithm. We slightly modified the original VSS-RZA-LMS algorithm [13] by bounding the calculated step-size with a maximum value. We found in simulations that properly bounding the result of [13] can be crucial for ensuring the stability and robustness of this algorithm.

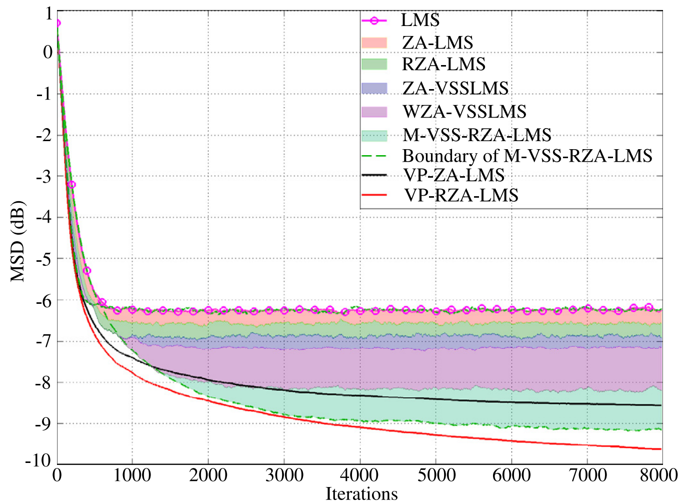

Fig. 2. Learning curves of LMS, ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS, M-VSS-RZA-LMS, VP-ZA-LMS and VP-RZA-LMS algorithms in the case of a correlated Gaussian input signal ($\alpha = 0.5$). For the sake of clarity, the learning curves of the M-VSS-RZA-LMS are shown in dashed green. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

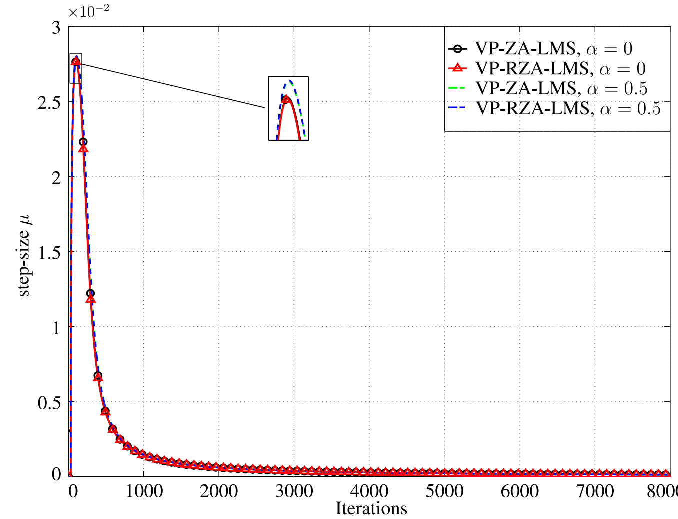

Fig. 3. Step-sizes of VP-ZA-LMS and VP-RZA-LMS algorithms in the case of white and correlated ($\alpha = 0.5$) input signals.

4 that the VP-ZA-LMS and VP-RZA-LMS algorithms set the step-size and the regularization parameter to large values at the initial phase of the adaptation to ensure fast convergence. Next, they gradually decreased the step-size and used nearly constant regularization parameter to reach a low misadjustment error as well as a reasonable level of sparsity.

In the second experiment, we used the same setting except that the correlation coefficient α was set to 0.5. The learning curves and performance regions of all the algorithms are provided in Fig. 2. The ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS and M-VSS-RZA-LMS outperformed the LMS.

The VP-ZA-LMS and VP-RZA-LMS algorithms are characterized by single learning curves in Fig. 2. The evolution of their step-size and regularization parameter over time is represented in Figs. 3 and 4, respectively. Though there was some performance degradation of VP-RZA-LMS algorithm compared with the first experiment, the VP-RZA-LMS algorithm still yielded the lowest steady-state MSD along with the fastest convergence speed among all the competing algorithms. Interestingly, the VP-ZA-LMS algorithm maintained the performance improvement observed in the first experiment, and outperformed all the competing ZA-LMS-type algorithms both with its convergence speed and steady-state performance. Despite the loss of the whiteness assumption **A2**, the VP-ZA-LMS and VP-RZA-LMS algorithms still work well with correlated inputs. Besides, observe that both algorithms set the step-size and the regularization parameter to large values at the initial phase, next changed them to reach a compromise between convergence rate, MSD and estimation bias (sparsity).

5.2. *Tracking performance in a non-stationary environment*

In the third experiment, we compared the tracking performance of VP-ZA-LMS and VP-RZA-LMS algorithms with the other competing algorithms. The order of the unknown time-varying system was set to $L = 32$.

From instant $n = 1$ to 8000, we set the system parameter vector to $\boldsymbol{w}_1^\star$, with 20 null entries over 32. At $n = 8001$, the system parameter vector was changed to the non-sparse one $\boldsymbol{w}_2^\star$. At $n = 16001$, we changed the system parameter vector to $\boldsymbol{w}_3^\star$, with 12 null entries over 32. The parameter vectors $\boldsymbol{w}_1^\star$, $\boldsymbol{w}_2^\star$ and $\boldsymbol{w}_3^\star$

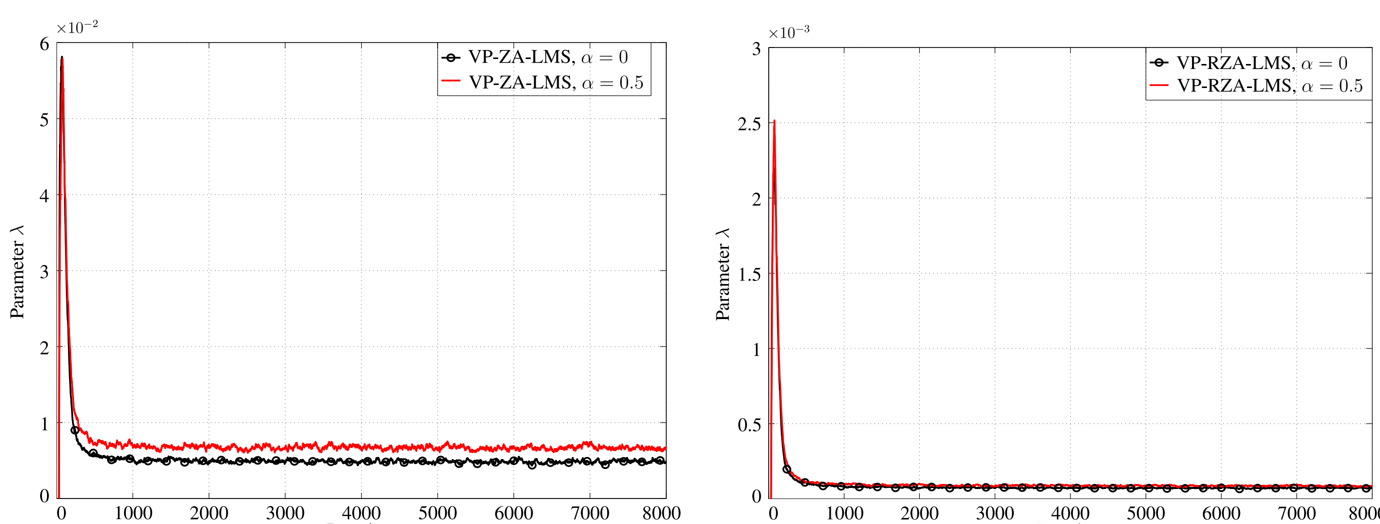

Fig. 4. Regularization parameter λ of (a) VP-ZA-LMS and (b) VP-RZA-LMS algorithms in the case of white and correlated input signal ($\alpha = 0.5$).

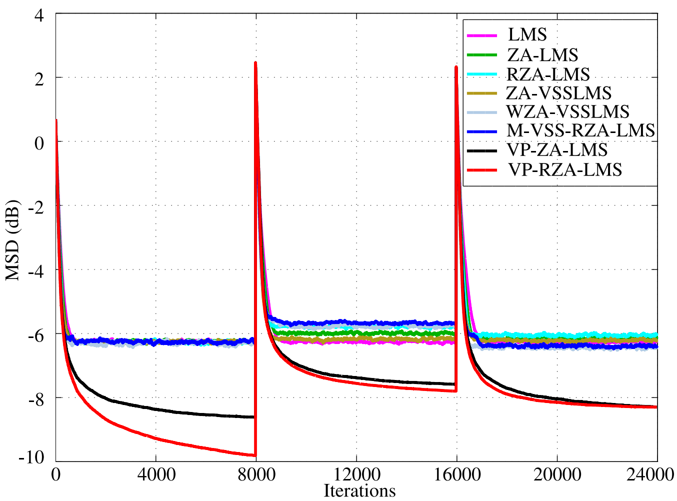

Fig. 5. Transient behaviors of LMS, ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS, M-VSS-RZA-LMS, VP-ZA-LMS and VP-RZA-LMS algorithms, in the presence of time varying systems driven by colored Gaussian input signal with correlation coefficient $\alpha = 0.5$. All the algorithms were set up to reach the same steady-state MSD as the LMS (and the fastest convergence speed) during the first 8000 iterations.

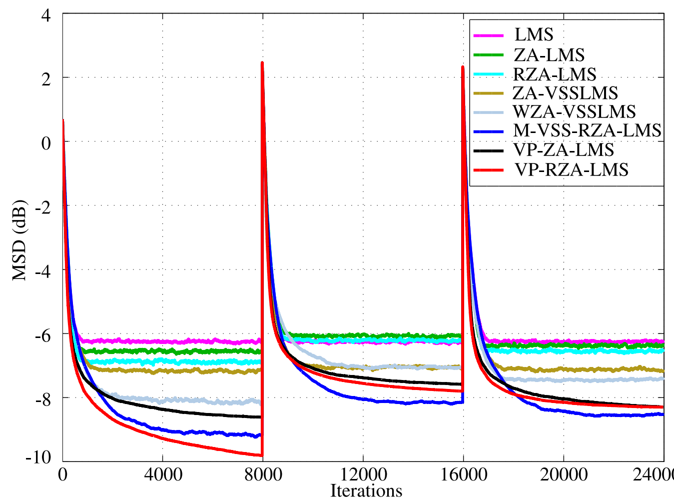

Fig. 6. Transient behaviors of LMS, ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS, M-VSS-RZA-LMS, VP-ZA-LMS and VP-RZA-LMS algorithms, in the presence of time varying systems driven by colored Gaussian input signal with correlation coefficient $\alpha = 0.5$. All the algorithms were set up to have the same initial convergence rate as the LMS (and the lowest steady-state MSD) during the first 8000 iterations.

were defined as:

$$\boldsymbol{w}_1^\star = [0.8,\ 0.5,\ 0.3,\ 0.2,\ 0.1,\ 0.05,\ \mathbf{0}_{20},\ -0.05,\ -0.1,\ -0.2,\ -0.3,\ -0.5,\ -0.8]^\top; \tag{67}$$

$$\boldsymbol{w}_2^\star = \Big[0.9,\ 0.8,\ 0.7,\ 0.6,\ 0.5,\ 0.4,\ 0.3,\ 0.2,\ 0.1,\ 0.05,\ 0.01,\ \mathbf{1}_{10},\ -0.01,\ -0.05,\ -0.1,\ -0.2,\ -0.3,\ -0.4,\ -0.5,\ -0.6,\ -0.7,\ -0.8\ -0.9\Big]^\top; \tag{68}$$

$$\boldsymbol{w}_3^\star = \Big[1.2,\ 0.9,\ 0.8,\ 0.7,\ 0.6,\ 0.5,\ 0.4,\ 0.2,\ 0.1,\ 0.01,\ \mathbf{0}_{12},\ -0.01,\ -0.1,\ -0.2,\ -0.4,\ -0.5,\ -0.6\ -0.7,\ -0.8,\ -0.9,\ -1.2\Big]^\top. \tag{69}$$

The input signal was Gaussian correlated with correlation coefficient $\alpha = 0.5$. The LMS algorithm with fixed step-size ($\mu = 0.01$) was used as a baseline.

First, we set the parameters of each algorithm, except LMS, to obtain the same steady-state MSD as LMS during the first 8000 iterations. Next, we set the parameters of each algorithm so that its initial convergence speed was almost the same as LMS. Figs. 5 and 6 plot the learning curves resulting from these two experiments, respectively. Because the hyperparameters of these two algorithms remained unchanged in the two experiments, the learning curves of the VP-ZA-LMS and VP-RZA-LMS algorithms are identical in Fig. 5 and Fig. 6. The evolution of their step-size and regularization parameter over time is provided in Figs. 7 and 8, respectively.

Results in Figs. 5 and 6 show that the VP-ZA-LMS and VP-RZA-LMS algorithms converged as fast as the other algorithms when estimating the sparse parameter vector $\boldsymbol{w}_1^\star$ while maintaining a lower misadjustement error. The estimation of the non-sparse parameter vector $\boldsymbol{w}_2^\star$ caused a moderate degradation of their performance. As shown in Fig. 5, their convergence speeds slowed down compared to the other algorithms but they reached a smaller residual error. It is seen in Fig. 6 that M-VSS-RZA-LMS performed the best. The estimation of $\boldsymbol{w}_3^\star$ confirms the good performance and tracking capability of VP-ZA-LMS and VP-RZA-LMS in practice. Results in Fig. 7 shows that VP-ZA-LMS and VP-RZA-LMS set the step-size and the regularization parameter to large values in order to ensure tracking and promote sparsity at the beginning of each estimation phase. Then they gradually reduced these values to ensure small MSD.

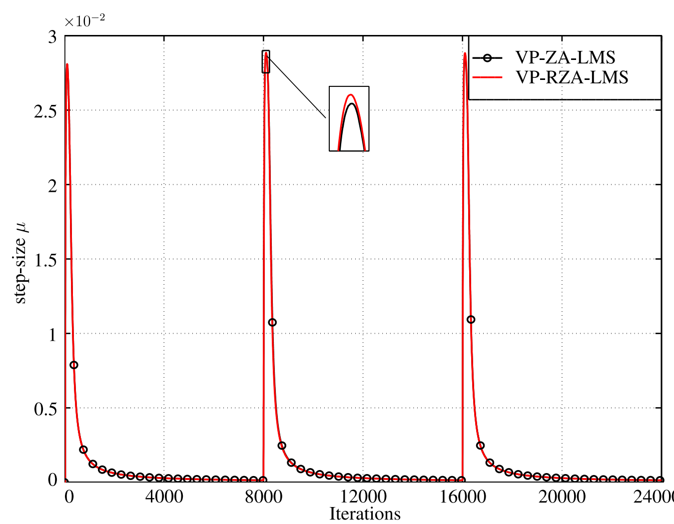

Fig. 7. Step-sizes μ of VP-ZA-LMS and VP-RZA-LMS for time varying system identification with correlation coefficient $\alpha = 0.5$.

To conclude these experiments, it should be noted that adjusting the hyperparameters of all the algorithms competing with VP-ZA-LMS and VP-RZA-LMS was not a trivial task. On the contrary, our algorithms used the same hyperparameters for all the simulations, initially set to typical values.

5.3. Computational complexity

We summarize in Table 1 the computational complexity of all the algorithms used in the experiments. We restrict our focus to real-valued identification problems. The computational complexity is measured in terms of real additions, real multiplications, and sign operations. Note that M-VSS-RZA-LMS requires evaluating the error function erf(·) function at each iteration. As this can be done

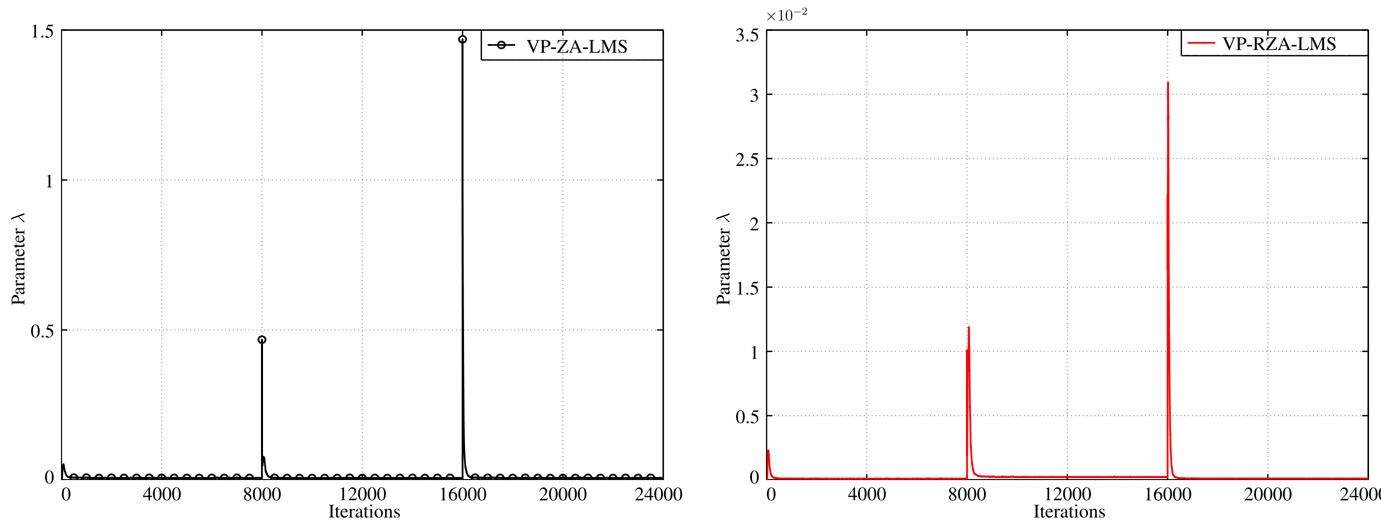

Fig. 8. Regularization parameters λ of (a) VP-ZA-LMS and (b) VP-RZA-LMS for time varying system identification with correlation coefficient $\alpha = 0.5$.

Table 1
Computational complexity of the competing algorithms.

Algorithms	# additions	# multiplications	# sign operations	# num. integrations
LMS	$2L$	$2L+1$		
ZA-LMS	$3L$	$3L+1$	L	
RZA-LMS	$4L$	$5L+1$	L	
ZA-VSSLMS	$3L+1$	$3L+5$	L	
WZA-VSSLMS	$4L+1$	$5L+5$	L	
M-VSS-RZA-LMS	$6L+1$	$8L+7$	L	1
VP-ZA-LMS	$4L+12$	$5L+30$	L	
VP-RZA-LMS	$6L+11$	$9L+29$	L	

Table 2
Parameters used in Fig. 1, to get the same initial convergence rate as the LMS.

Algorithms	$\mu \mid \rho$	$\lambda \mid \lambda'$	$\varepsilon \mid \zeta$	$\mu_{\max}$	$\mu_{\min}$	$\gamma \mid \delta$	$\alpha \mid \beta$
LMS	0.01						
ZA-LMS	0.01	0.003					
RZA-LMS	0.01	0.00005	0.0002				
ZA-VSSLMS		0.0008		$2/(3\sigma_x^2 \cdot L)$	0.0001	0.00212	0.95
WZA-VSSLMS		| 0.0002	| 0.001	$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0018	0.95
M-VSS-RZA-LMS	| 3×10^{-7}	0.995	0.05	0.01		| 0.00001	| 1.3

Table 3
Parameters used in Fig. 1, to get the same steady-state MSD as the LMS.

Algorithms	$\mu \mid \rho$	$\lambda \mid \lambda'$	$\varepsilon \mid \zeta$	$\mu_{\max}$	$\mu_{\min}$	$\gamma \mid \delta$	$\alpha \mid \beta$
LMS	0.01						
ZA-LMS	0.0113	0.003					
RZA-LMS	0.013	0.00005	0.0002				
ZA-VSSLMS		0.0008		$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0055	0.95
WZA-VSSLMS		| 0.0005	| 0.001	$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0063	0.95
M-VSS-RZA-LMS	| 1.5×10^{-5}	0.995	0.05	0.0145		| 0.00001	| 11

Table 4
Parameters used in Figs. 2 and 6, to get the same initial convergence rate as the LMS.

Algorithms	$\mu \mid \rho$	$\lambda \mid \lambda'$	$\varepsilon \mid \zeta$	$\mu_{\max}$	$\mu_{\min}$	$\gamma \mid \delta$	$\alpha \mid \beta$
LMS	0.01						
ZA-LMS	0.01	0.006					
RZA-LMS	0.01	0.00008	0.0002				
ZA-VSSLMS		0.0008		$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0028	0.95
WZA-VSSLMS		| 0.0002	| 0.001	$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0022	0.95
M-VSS-RZA-LMS	| 3×10^{-7}	0.995	0.05	0.01		| 0.00001	| 1.3

by numerical integration or using a lookup table, the corresponding computational load is not detailed in Table 1. It is observed in Table 1 that the computational complexity of the proposed VP-ZA-LMS and VP-RZA-LMS algorithms is of the same order as the ZA-LMS and RZA-LMS algorithms.

6. Conclusion

In this paper, we introduced the VP-ZA-LMS and VP-RZA-LMS algorithms to address online sparse system identification problems. Based on a stochastic model of the transient behavior of the ZA-LMS, we proposed to minimize the EMSE with respect to the step-

Table 5
Parameters used in Figs. 2 and 5, to get the same steady-state MSD as the LMS.

Algorithms	μ | ρ	λ | λ'	ε | ζ	$\mu_{\max}$	$\mu_{\min}$	γ | δ	α | β
LMS	0.01						
ZA-LMS	0.012	0.003					
RZA-LMS	0.0145	0.00008	0.0002				
ZA-VSSLMS		0.0008		$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0057	0.95
WZA-VSSLMS		| 0.00041	| 0.001	$2/(3\sigma_x^2 \cdot L)$	0.0001	0.0065	0.95
M-VSS-RZA-LMS	| 1.75×10^{-5}	0.995	0.05	0.016		| 0.00001	| 10.6

size and the regularization parameter simultaneously, at each iteration. This led to a convex optimization problem with a closed-form solution. Simulation results demonstrated the effectiveness of VP-ZA-LMS and VP-RZA-LMS algorithms over other existing variable step-size ZA-LMS-type and RZA-LMS-type algorithms, without requiring extra computational effort. Compared to the competing algorithms, VP-ZA-LMS and VP-RZA-LMS depend on a few number of hyperparameters that do not drastically affect the performance. Considering the locations of zero-valued coefficients can be clustered, a variable parameter algorithm of group ZA-LMS is further derived and provided in report [22].

Appendix A. Proof of Lemma 1

We have

$$\boldsymbol{H} = \begin{bmatrix} \sigma_z^2 \mathrm{tr}\{\boldsymbol{R}_x\} + \mathrm{tr}\{\boldsymbol{Q}_1\} & \mathrm{tr}\{\boldsymbol{Q}_5\} \\ \mathrm{tr}\{\boldsymbol{Q}_5\} & \mathrm{tr}\{\boldsymbol{Q}_2\} \end{bmatrix}. \tag{70}$$

It can be observed that $\boldsymbol{H}$ can be decomposed into two additive components $\boldsymbol{H}_1$ and $\boldsymbol{H}_2$ such that

$$\boldsymbol{H} = \boldsymbol{H}_1 + \boldsymbol{H}_2, \tag{71}$$

with

$$\boldsymbol{H}_1 = \begin{bmatrix} \mathrm{tr}\{\boldsymbol{Q}_1\} & \mathrm{tr}\{\boldsymbol{Q}_5\} \\ \mathrm{tr}\{\boldsymbol{Q}_5\} & \mathrm{tr}\{\boldsymbol{Q}_2\} \end{bmatrix} \tag{72}$$

and

$$\boldsymbol{H}_2 = \begin{bmatrix} \sigma_z^2 \mathrm{tr}\{\boldsymbol{R}_x\} & 0 \\ 0 & 0 \end{bmatrix}. \tag{73}$$

We now prove that matrix $\boldsymbol{H}_1$ is positive semidefinite. Using the definition of $\boldsymbol{Q}_1$, $\boldsymbol{Q}_2$ and $\boldsymbol{Q}_5$ in (14), (15) and (18), $\boldsymbol{H}_1$ can be written as

$$\boldsymbol{H}_1 = \mathbb{E}\left\{\begin{bmatrix} \mathrm{tr}\{\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top\} & \mathrm{tr}\{\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\} \\ \mathrm{tr}\{\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\} & \mathrm{tr}\{\mathrm{sgn}\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\} \end{bmatrix}\right\}. \tag{74}$$

Using that $\mathrm{tr}\{\boldsymbol{AB}\} = \mathrm{tr}\{\boldsymbol{BA}\}$, we have

$$\begin{aligned} \boldsymbol{H}_1 &= \mathbb{E}\left\{\begin{bmatrix} \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n & \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n \\ \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\boldsymbol{x}_n\boldsymbol{x}_n^\top \tilde{\boldsymbol{w}}_n & \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\}\mathrm{sgn}\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\} \end{bmatrix}\right\} \\ &= \mathbb{E}\{\boldsymbol{V}_n \boldsymbol{V}_n^\top\}, \end{aligned} \tag{75}$$

where

$$\boldsymbol{V}_n = \begin{bmatrix} \tilde{\boldsymbol{w}}_n^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top \\ \mathrm{sgn}^\top\{\boldsymbol{w}^\star + \tilde{\boldsymbol{w}}_n\} \end{bmatrix}. \tag{76}$$

Matrix $\boldsymbol{H}_1$ is thus positive semidefinite. Considering that the diagonal entries of $\boldsymbol{H}_2$ are nonnegative, we conclude that $\boldsymbol{H}$ is positive semidefinite.

Appendix B. Parameters used for simulations

The parameters used for LMS, ZA-LMS, RZA-LMS, ZA-VSSLMS, WZA-VSSLMS, M-VSS-RZA-LMS algorithms in the simulations are reported in Tables 2–5. Some pairs of columns, standing for different parameters, are merged into a single column for compactness. The corresponding symbols can be distinguished by the symbol "|", standing for "or" in the Tables. The parameters used for VP-ZA-LMS and VP-RZA-LMS algorithms are provided in **Algorithm 1**. They remained unchanged for all the experiments as we observed that they did not drastically affect the performance.

Appendix C. Algorithms for complex-valued signals

In the complex domain, the unknown system is characterized by:

$$y_n = \boldsymbol{w}^{\star\mathrm{H}} \boldsymbol{x}_n + z_n \tag{77}$$

with $(\cdot)^{\mathrm{H}}$ the conjugate transpose, and $\boldsymbol{w}^\star, \boldsymbol{x}_n, z_n, y_n \in \mathbb{C}^L$. Consider the regularized costs:

$$\boldsymbol{w}_{\mathrm{ZA}}^o = \arg\min_{\boldsymbol{w}} \frac{1}{2}\mathbb{E}\left\{|y_n - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{x}_n|^2\right\} + \lambda\|\boldsymbol{w}\|_1 \tag{78}$$

and

$$\boldsymbol{w}_{\mathrm{RZA}}^o = \arg\min_{\boldsymbol{w}} \frac{1}{2}\mathbb{E}\left\{|y_n - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{x}_n|^2\right\} + \lambda\sum_{i=1}^{L}\log(1 + |w_i|/\varepsilon) \tag{79}$$

where $\|\boldsymbol{w}\|_1 = \sum_{i=1}^{L}\sqrt{\mathrm{Re}\{w_i\}^2 + \mathrm{Im}\{w_i\}^2}$, with $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ the real part and the imaginary part of its complex argument, respectively. Using steepest descent method of complex-valued signals [1,20,23]:

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n - 2\mu\,[\nabla_{\boldsymbol{w}} J(\boldsymbol{w}_n)]^{\mathrm{H}}, \tag{80}$$

with instantaneous approximation for statistics, we have:

$$\boldsymbol{w}_{n+1} = \boldsymbol{w}_n + \mu\,\boldsymbol{x}_n e_n^* - \rho\left(\boldsymbol{\beta}_n \circ \boldsymbol{s}_n\right) \tag{81}$$

where $\circ$ is the Hadamard product, $(\cdot)^*$ is the complex conjugate operator, $\rho = \mu\lambda$, and $e_n = y_n - \boldsymbol{w}_n^{\mathrm{H}}\boldsymbol{x}_n$ is the estimation error. In (81), $\boldsymbol{s}_n$ and $\boldsymbol{\beta}_n$ are $L \times 1$ vectors with jth entry given by:

$$s_{n,j} = \begin{cases} \frac{w_{n,j}}{\|w_{n,j}\|_1} & \text{when } \|w_{n,j}\|_1 \neq 0 \\ 0 & \text{when } \|w_{n,j}\|_1 = 0 \end{cases} \tag{82}$$

and

$$\beta_{n,j} = \begin{cases} 1 & \text{for ZA-LMS} \\ \frac{1}{\|w_{n,j}\|_1 + \varepsilon} & \text{for RZA-LMS.} \end{cases} \tag{83}$$

It should be mentioned that in (81) the correction term associated to regularization terms are calculated in an elementwise manner via the definition of complex gradient in [1] with using subgradients for non-differential points.

Using assumptions **A1** and **A2**, and following a similar derivation as in (9)–(13), we have:

$$\begin{aligned} \boldsymbol{K}_{n+1} &= \boldsymbol{K}_n + \mu^2\left(\sigma_z^2\boldsymbol{R}_x + \boldsymbol{Q}_1\right) + \rho^2\boldsymbol{Q}_2 - \mu\left(\boldsymbol{Q}_3 + \boldsymbol{Q}_3^{\mathrm{H}}\right) \\ &\quad - \rho\left(\boldsymbol{Q}_4 + \boldsymbol{Q}_4^{\mathrm{H}}\right) + \mu\rho\left(\boldsymbol{Q}_5 + \boldsymbol{Q}_5^{\mathrm{H}}\right) \end{aligned} \tag{84}$$

with

$$\boldsymbol{K}_n = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n\tilde{\boldsymbol{w}}_n^{\mathrm{H}}\right\} \tag{85}$$

$$\boldsymbol{Q}_1 = \mathbb{E}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^{\mathrm{H}}\tilde{\boldsymbol{w}}_n\tilde{\boldsymbol{w}}_n^{\mathrm{H}}\boldsymbol{x}_n\boldsymbol{x}_n^{\mathrm{H}}\right\} \tag{86}$$

$$\boldsymbol{Q}_2 = \mathbb{E}\left\{(\boldsymbol{\beta}_n \circ \boldsymbol{s}_n)\cdot(\boldsymbol{\beta}_n \circ \boldsymbol{s}_n)^{\mathrm{H}}\right\} \tag{87}$$

$$\boldsymbol{Q}_3 = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n\tilde{\boldsymbol{w}}_n^{\mathrm{H}}\boldsymbol{x}_n\boldsymbol{x}_n^{\mathrm{H}}\right\} \tag{88}$$

$$\boldsymbol{Q}_4 = \mathbb{E}\left\{\tilde{\boldsymbol{w}}_n\cdot(\boldsymbol{\beta}_n \circ \boldsymbol{s}_n)^{\mathrm{H}}\right\} \tag{89}$$

$$\boldsymbol{Q}_5 = \mathbb{E}\left\{\boldsymbol{x}_n\boldsymbol{x}_n^{\mathrm{H}}\tilde{\boldsymbol{w}}_n(\boldsymbol{\beta}_n \circ \boldsymbol{s}_n)^{\mathrm{H}}\right\}. \tag{90}$$

Based on the transient behavior model, we then minimize $\mathrm{tr}\{\boldsymbol{K}_{n+1}\}$ with respect to parameters μ and ρ:

$$\{\mu_n^\star, \rho_n^\star\} = \arg\min_{\mu,\rho}\ \mathrm{tr}\{\boldsymbol{K}_n\} + \mu^2 a + \rho^2 b - \mu(p_1 + p_1^*) - \rho(p_2 + p_2^*) + \mu\rho(c + c^*), \tag{91}$$

where:

$$a = \sigma_z^2\mathrm{tr}\{\boldsymbol{R}_x\} + \mathrm{tr}\{\boldsymbol{Q}_1\} \tag{92}$$

$$b = \mathrm{tr}\{\boldsymbol{Q}_2\} \tag{93}$$

$$c = \mathrm{tr}\{\boldsymbol{Q}_5\} \tag{94}$$

$$p_1 = \mathrm{tr}\{\boldsymbol{Q}_3\} \tag{95}$$

$$p_2 = \mathrm{tr}\{\boldsymbol{Q}_4\}. \tag{96}$$

It can be checked that coefficients a and b are real. The objective function can then be written as follows:

$$\xi_{n+1} = [\mu\ \rho]\,\boldsymbol{H}\,[\mu\ \rho]^\top - [p_1 + p_1^*\ \ p_2 + p_2^*]\,[\mu\ \rho]^\top + \xi_n, \tag{97}$$

with

$$\boldsymbol{H} = \begin{bmatrix} a & \dfrac{c + c^*}{2} \\ \dfrac{c + c^*}{2} & b \end{bmatrix} \tag{98}$$

Considering that $\boldsymbol{H}$ is a positive definite Hessian matrix, the optimal parameters are given by:

$$[\mu_n^\star\ \rho_n^\star]^\top = \boldsymbol{H}^{-1}\left[\frac{p_1 + p_1^*}{2}\ \ \frac{p_2 + p_2^*}{2}\right]^\top, \tag{99}$$

namely,

$$\mu_n^\star = \frac{b\mathrm{Re}\{p_1\} - \mathrm{Re}\{c\}\mathrm{Re}\{p_2\}}{ab - \mathrm{Re}\{c\}^2} \tag{100}$$

$$\rho_n^\star = \frac{a\mathrm{Re}\{p_2\} - \mathrm{Re}\{c\}\mathrm{Re}\{p_1\}}{ab - \mathrm{Re}\{c\}^2}. \tag{101}$$

Approximating the quantities a via $a = \sigma_z^2\sigma_x^2 L + (1 + L)\sigma_x^2\zeta_n$ and b, c, p_1, p_2 in the same manner as in the real-valued case completes the algorithm.

References

[1] A. H. Sayed, Adaptive Filters, John Wiley & Sons, Inc., 10.1002/9780470374122.ch33.
[2] B. Widrow, S.D. Stearns, Adaptive Signal Processing, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1985.
[3] D.L. Duttweiler, Proportionate normalized least-mean-squares adaptation in echo cancelers, IEEE Trans. Speech Audio Process. 8 (5) (2000) 508–518, doi:10.1109/89.861368.
[4] J. Benesty, S.L. Gay, An improved PNLMS algorithm, in: Proceedings of the IEEE ICASSP, 2, 2002, doi:10.1109/ICASSP.2002.5744994.
[5] D.L. Donoho, Compressed sensing, IEEE Trans. Inf. Theory 52 (4) (2006) 1289–1306.
[6] E.J. Candès, M.B. Wakin, S.P. Boyd, Enhancing sparsity by reweighted l_1 minimization, J. Fourier Anal. Appl. 14 (5) (2008) 877–905, doi:10.1007/s00041-008-9045-x.
[7] G. Su, J. Jin, Y. Gu, J. Wang, Performance analysis of l_0 norm constraint least mean square algorithm, IEEE Trans. Signal Process. 60 (5) (2012) 2223–2235.
[8] E.J. Candès, M.B. Wakin, An introduction to compressive sampling, IEEE Signal Process. Mag. 25 (2) (2008) 21–30.
[9] Y. Chen, Y. Gu, A.O. Hero, Sparse LMS for system identification, Taipei, China, in: Proceedings of the IEEE ICASSP, 2009, pp. 3125–3128.
[10] K.P. Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press, 2012.
[11] C. Paleologu, J. Benesty, S. Ciochină, A variable step-size proportionate NLMS algorithm for echo cancellation, Revue Roumaine des Sciences Techniques – Serie Electrotechnique et Energetique 53 (2008) 309–317.
[12] M.S. Salman, M.N.S. Jahromi, A. Hocanin, O. Kukrer, A zero-attracting variable step-size LMS algorithm for sparse system identification, in: Proceedings of the Ninth International Symposium on Telecommunications (BIHTEL), 2012, pp. 1–4, doi:10.1109/BIHTEL.2012.6412087.
[13] T. Fan, Y. Lin, A variable step-size strategy based on error function for sparse system identification, Circuits Syst. Signal Process. 36 (3) (2017) 1301–1310, doi:10.1007/s00034-016-0344-1.
[14] J. Benesty, H. Rey, L.R. Vega, S. Tressens, A nonparametric VSS NLMS algorithm, IEEE Signal Process. Lett. 13 (10) (2006) 581–584, doi:10.1109/LSP.2006.876323.
[15] R.H. Kwong, E.W. Johnston, A variable step size LMS algorithm, IEEE Trans. Signal Process. 40 (7) (1992) 1633–1642, doi:10.1109/78.143435.
[16] J. Chen, C. Richard, Y. Song, D. Brie, Transient performance analysis of zero-attracting LMS, IEEE Signal Process. Lett. 23 (12) (2016) 1786–1790.
[17] S.F. Cotter, B.D. Rao, Sparse channel estimation via matching pursuit with application to equalization, IEEE Trans. Commun. 50 (3) (2002) 374–377.
[18] W.F. Schreiber, Advanced television systems for terrestrial broadcasting: some problems and some proposed solutions, Proc. IEEE 83 (6) (1995) 958–981, doi:10.1109/5.387095.
[19] S. Zhang, J. Zhang, Transient analysis of zero attracting NLMS algorithm without gaussian inputs assumption, Signal Process. 97 (2014) 100–109.
[20] S. Haykin, Adaptive Filter Theory, Fourth, Pearson Education India, 2005.
[21] J. Chen, C. Richard, A.H. Sayed, Diffusion LMS over multitask networks, IEEE Trans. Signal Process. 63 (11) (2015) 2733–2748.
[22] D. Jin, J. Chen, C. Richard, J. Chen, Adaptive parameters adjustment for group reweighted zero-attracting LMS, Proc. of IEEE ICASSP (2018) 4294–4298.
[23] A. Hjorungnes, D. Gesbert, Complex-valued matrix differentiation: techniques and key results, IEEE Trans. Signal Process. 55 (6) (2007) 2740–2746, doi:10.1109/TSP.2007.893762.