Signal Processing 213 (2023) 109187

Contents lists available at ScienceDirect

Signal Processing

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

Short communication

An incremental noise constrained LMS algorithm

Muhammad Omer Bin Saeed [a], Azzedine Zerguine [*,b], Usman Hameed [c], Sajid Gul Khawaja [d], Oualid Hammi [e]

[a] Faculty of Engineering Sciences, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Pakistan
[b] Department of Electrical Engineering and the Center for Communication Systems and Sensing (IRC-CSS), King Fahd University of Petroleum & Minerals, Dhahran, 31261, Saudi Arabia
[c] Department of Electrical & Computer Engineering, Air University, Islamabad, Pakistan
[d] Computer Engineering Department, National University of Sciences & Technology, Islamabad, Pakistan
[e] Department of Electrical Engineering, American University of Sharjah, Sharjah, UAE

ARTICLE INFO

Keywords:
Noise constrained LMS algorithm
Incremental LMS algorithm
Distributed estimation

ABSTRACT

While a lot of research has focused on diffusion-based algorithms for distributed estimation over adaptive networks, incremental algorithms have not been studied to that extent. Here we present an incremental scheme-based distributed least mean square algorithm that uses the variance of the additive noise as a constraint. The proposed algorithm is derived and then its theoretical performance analysis is presented. Simulation results for different scenarios are then presented and it is shown that the simulation results corroborate very well the theoretical findings. More importantly, our algorithm outperforms, a recently proposed algorithm, the variable step-size incremental-based LMS (VSSILMS) algorithm.

1. Introduction

Over the last decade, significant research has been dedicated to developing new distributed adaptive algorithms for the purpose of signal estimation over adaptive networks [1]. As part of this research effort, the incremental scheme-based Least Mean Squares (LMS) algorithm was presented in [2]. This scheme traverses the entire network in a cyclic manner and each node updates the estimate based on the data provided by the previous node in the cycle. The incremental scheme performs the best among all distributed schemes but it is prone to node failure and may not perform well enough for large networks. However, its performance compared with other schemes is such that this trade-off is acceptable under most scenarios. Then, a fully distributed diffusion-based LMS algorithm was proposed in [2]. As opposed to the incremental scheme, the neighboring nodes all share data with each other, which makes diffusion a fully distributed scheme. However, it has been shown that the incremental scheme outperforms the diffusion scheme under different scenarios [4–8]. The authors in [4] presented a consensus-based distributed approach, which incorporates the constraint that all nodes reach the same estimate at steady-state. However, it was shown in [9] that the diffusion scheme outperforms the consensus-based scheme, in general. In order to cater for the trade-off that exists, for the conventional LMS algorithm, between the error performance and the speed of convergence [10], variable step-size distributed LMS algorithms were presented in [11] and [12]. The diffusion-based variable step-size LMS algorithm, presented in [11], was then extended in [13]. Several variants of the diffusion-based, variable step-size LMS algorithm have since been proposed in [14–18].

While constrained-based algorithms have found use in several applications (for example [19]), the first use of a noise-constrained based distributed algorithm was in [20,21]. Capitalizing on the work of [22] a diffusion-based noise-constrained LMS algorithm was presented in [20] by exploiting the variance of the additive noise to improve the performance of the diffusion-based LMS algorithm of [3]. This algorithm was then further improved in [21], with its detailed analysis presented in [23]. Note that the noise-constrained LMS (NCLMS) algorithm of [22] discusses only the NCLMS algorithm involving no diffusion whatsoever. This improved algorithm belongs to the class of variable step-size algorithms.

The incremental-based algorithms have generally been ignored because of the inherent disadvantages of their internal incremental scheme [1]. However, the advantages of using this incremental scheme

* corresponding author.
E-mail addresses: m.binsaeed.1981@ieee.org (M.O.B. Saeed), azzedine@kfupm.edu.sa (A. Zerguine), usman.hameed@mail.au.edu.pk (U. Hameed), sajid.gul@ceme.nust.edu.pk (S.G. Khawaja), ohammi@aus.edu (O. Hammi).

https://doi.org/10.1016/j.sigpro.2023.109187
Received 4 October 2022; Received in revised form 21 June 2023; Accepted 11 July 2023
Available online 15 July 2023
0165-1684/

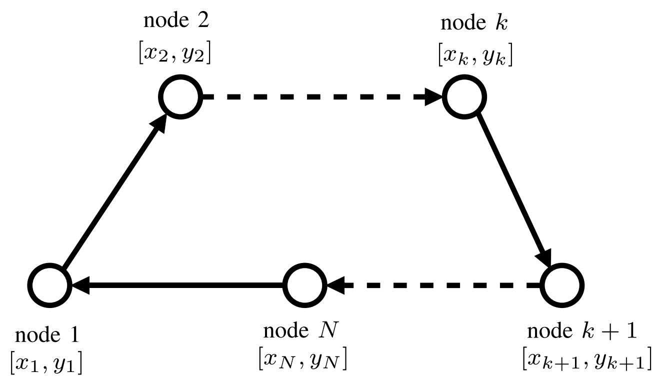

Fig. 1. Incremental scheme based network of N nodes.

Table 1
Computational complexity per node per iteration for the different algorithms.

Algorithm	Multiplications	Additions
ILMS [2]	$2M$	$2M$
VSSILMS [8]	$2M+3$	$2M+1$
IVSLMS [25]	$2M+2$	$2M+1$
Proposed	$2M+5$	$2M+3$
ZNCILMS	$2M+5$	$2M+2$

outweighs the disadvantages for small-scale networks [1]. As a result, a recent study renewed interest that has been witnessed and seen in further development and use of increment-based algorithms, such as in [8,24,25]. Other algorithms that use the noise variance to develop a variable step-size adaptive filters, such as the set-membership algorithms, include [26] and [27].

More importantly, it is assumed in the adaptive network-based algorithms that the noise level is the same for all nodes. This is not the case in practice[1]. Hence, for an optimal performance, the adaptation scheme should incorporate some local features, such as the noise statistical behaviour, that are unique to each node. To address this vitally important point, this work proposes to include the noise power of each node as a constraint in the cost function of the algorithm used. To develop such an algorithm, the authors have resorted to the noise-constrained LMS algorithm of [22] so as to derive a new adaptive incremental algorithm. Also, as will be shown in the simulation results section, our proposed algorithm outperforms an algorithm recently developed in [8]. Note here that our proposed algorithm and the one presented in [8] are both of the variable-step size type which are known to outperform algorithms with constant-step size.

Apart from its slightly higher computational complexity, the expected higher performance of the proposed algorithm is supported by the facts that the transient behaviour of the weights of the adaptive algorithm depends on the statistics of the channel noise, as pointed out in [22]. This therefore suggests the usefulness of exploiting the noise statistics when finding the optimal weight at each node of the adaptive network. Thus the use of the statistical knowledge of the channel noise in LMS-type adaptive algorithms, including the one we propose here, would therefore be justifiably expected to lead to an improvement in the convergence performance.

The contributions of this work are summarized as follows:

1) Incorporating the noise constrained strategy in the incremental scheme to formulate the proposed incremental noise constrained LMS algorithm.
2) Performing the mean and mean square analysis, the latter includes the transient analysis (with the learning behavior) and the steady state analysis.
3) Complexity comparison presented for an accurate estimate of the complexity-performance trade-off.
4) Simulation results presented for varying scenarios to demonstrate the performance enhancement of the proposed algorithm over the existing variable step-size algorithms.

The remainder of this paper is organized as follows. Section 2 introduces the system model and then derives the proposed algorithm. The complete mathematical analysis is presented in Section 3, with the experimental results and discussions presented in Section 4. Finally, Section 5 concludes this work.

2. The proposed algorithm

Let us consider a network with N nodes, connected through a Hamiltonian cycle, as shown in Fig. 1. Each node k attempts to estimate a vector of unknown parameters of interest $\mathbf{w}^0 \in \mathbb{R}^{M\times 1}$, by using a scalar measurement, $y_k(i)$, which is connected with $\mathbf{w}^0$ as follows:

$$y_k(i) = \mathbf{x}_k^T(i)\mathbf{w}^0 + n_k(i), \tag{1}$$

where $\mathbf{x}_k(i) \in \mathbb{R}^{M\times 1}$ is the regressor input vector for node k, $n_k(i)$ is a zero-mean additive white noise with variance $\sigma_{n,k}^2$ and i denotes the iteration.

The incremental LMS (ILMS) algorithm from [2] is derived by minimizing the following cost function:

$$J_k(\mathbf{w}_k) = \frac{1}{2}\mathbb{E}\left[\left|y_k - \mathbf{x}_k^T\mathbf{w}_k\right|^2\right], \tag{2}$$

where $\mathbf{w}_k$ is the estimate for $\mathbf{w}^0$ at node k. The resulting algorithm is given by [2]

$$\mathbf{\Phi}_0(i) = \mathbf{w}(i-1), \tag{3}$$

$$e_k(i) = y_k(i) - \mathbf{x}_k^T(i)\mathbf{\Phi}_{k-1}(i), \tag{4}$$

$$\mathbf{\Phi}_k(i) = \mathbf{\Phi}_{k-1}(i) + \mu_k\mathbf{x}_k(i)e(i), \tag{5}$$

$$\mathbf{w}(i) = \mathbf{\Phi}_N(i), \tag{6}$$

where $\mathbf{\Phi}_k(i)$ is the intermediate update at node k for the i^{th} iteration and μ_k is its fixed step-size.

2.1. Proposed algorithm

For the proposed algorithm, the noise variance is incorporated into the cost function as a constraint and the cost function per node is modified as follows[2]:

$$J_{k,new}(\mathbf{w}_k) = J_k(\mathbf{w}_k) + \gamma_k\lambda_k\left[J_k(\mathbf{w}_k) - \sigma_{n,k}^2\right] - \gamma_k\lambda_k^2, \tag{7}$$

where λ_k is the Lagrangian multiplier for node k and γ_k is a positive control parameter. The cost function has been adopted from [22] and modified for the current application. The algorithm is derived using the following recursions:

[1] Note here that this scheme will not work for noises with a pathological distribution, such as Cauchy or Lorentz noise, for which both the mean and the variance are undefined. It may also not work for alpha-stable type of noises which, like Cauchy's distribution, have slow-decaying fat tails.

[2] The noise power at node k, $\sigma_{n,k}^2$, can be estimated using the method of [28].

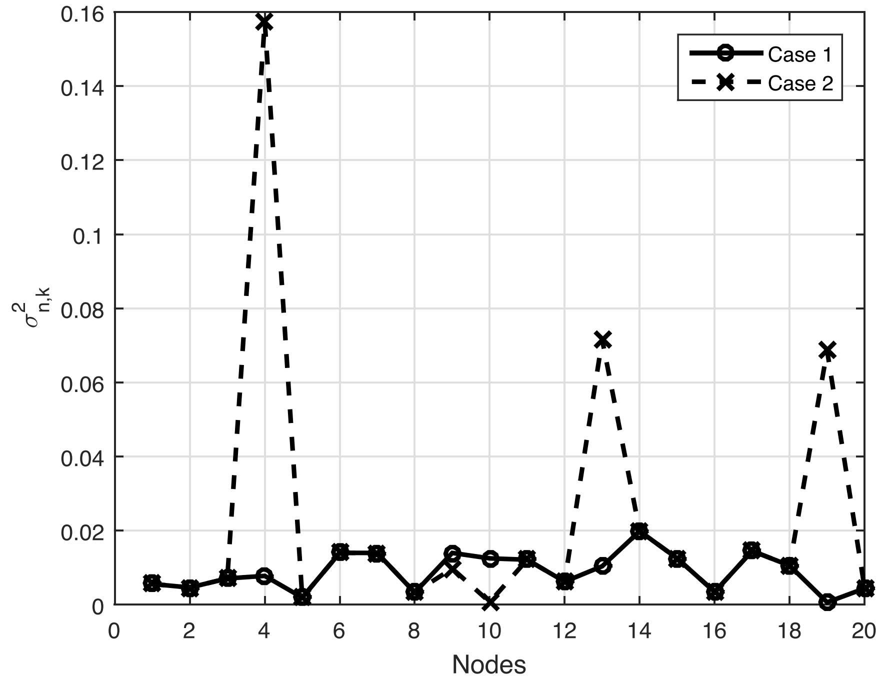

Fig. 2. The variances of the noise.

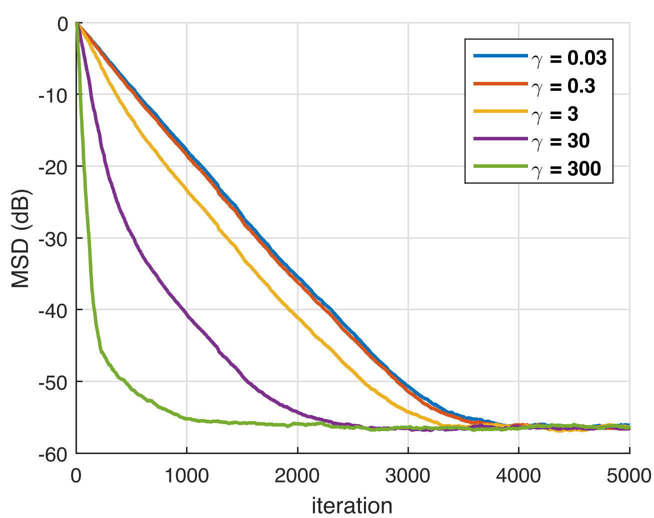

Fig. 3. Performance comparison for varying values of γ.

$$\mathbf{w}_{k+1}(i) = \mathbf{w}_k(i) - \alpha_k \frac{\partial}{\partial \mathbf{w}_k} J_{k,new}(\mathbf{w}_k) \tag{8}$$

$$\lambda_{k+1} = \lambda_k + \beta_k \frac{\partial}{\partial \lambda_k} J_k(\mathbf{w}_k), \tag{9}$$

where α_k and β_k are the respective step-size parameters for the two recursions. Drawing a parallel with (3)-(6) and solving (8)-(9) yields the proposed noise constrained incremental LMS (NCILMS) algorithm

$$\mathbf{\Phi}_0(i) = \mathbf{w}(i-1), \tag{10}$$

$$e_k(i) = y_k(i) - \mathbf{x}_k^T(i)\mathbf{\Phi}_{k-1}(i), \tag{11}$$

$$\alpha_k(i) = \mu_k[1 + \gamma_k \lambda_k(i)], \tag{12}$$

$$\mathbf{\Phi}_k(i) = \mathbf{\Phi}_{k-1}(i) + \alpha_k(i)\mathbf{x}_k(i)e_k(i), \tag{13}$$

$$\lambda_k(i+1) = (1-\beta_k)\lambda_k(i) + \frac{\beta_k}{2}\left(e_k^2(i) - \sigma_{n,k}^2\right), \tag{14}$$

$$\mathbf{w}(i) = \mathbf{\Phi}_N(i), \tag{15}$$

and as shown in (12), α_k is a time-varying step-size parameter.

3. Theoretical analysis

This section presents the mathematical analysis. Before presenting the analysis, a weight-error is introduced and is given by

$$\widetilde{\mathbf{w}}_k(i) = \mathbf{w}^0 - \mathbf{w}_k(i). \tag{16}$$

3.1. Mean analysis

Inserting (1) into (11) and then (16) and (11) into (13), with some simplifications, results in

$$\mathbf{\Phi}_k(i) = \left[\mathbf{I}_M - \alpha_k(i)\mathbf{x}_k(i)\mathbf{x}_k^T(i)\right]\mathbf{\Phi}_{k-1}(i) - \alpha_k(i)\mathbf{x}_k(i)n_k(i). \tag{17}$$

Now, taking the expectation of both sides of (17) results in the analysis of the mean. Since the noise term, $n_k(i)$ is independent and zero-mean, the expectation of the last term of (17) becomes zero. Also, using the assumption of data independence [10] and simplifying yields

$$\mathbb{E}[\mathbf{\Phi}_k(i)] = \left[\mathbf{I}_M - \mathbb{E}[\alpha_k(i)]\mathbf{R}_{\mathbf{x},k}\right]\mathbb{E}[\mathbf{\Phi}_{k-1}(i)], \tag{18}$$

where $\mathbf{R}_{\mathbf{x},k} = \mathbb{E}[\mathbf{x}_k(i)\mathbf{x}_k^T(i)]$ is the input data autocorrelation matrix for node k. Here, we assume, that the input regressor data at each node has the same statistical characteristics, therefore, $\mathbf{R}_{\mathbf{x},k} = \mathbf{R}_\mathbf{x}$. Next, we decompose the matrix $\mathbf{R}_\mathbf{x}$ using $\mathbf{R}_\mathbf{x} = \mathbf{G}\mathbf{\Lambda}\mathbf{G}^T$, where $\mathbf{\Lambda}$ is a diagonal matrix containing the eigenvalues of $\mathbf{R}_\mathbf{x}$ and $\mathbf{G}$ contains the corresponding eigenvectors. Also, $\mathbf{G}\mathbf{G}^T = \mathbf{G}^T\mathbf{G} = \mathbf{I}_M$. Using $\mathbf{G}$, we get $\overline{\mathbf{\Phi}}_k = \mathbf{G}^T\mathbf{\Phi}_k$. Thus, pre-multiplying (18) by $\mathbf{G}^T$ and simplifying results in

$$\mathbb{E}[\overline{\mathbf{\Phi}}_k(i)] = [\mathbf{I}_M - \mathbb{E}[\alpha_k(i)]\mathbf{\Lambda}]\mathbb{E}[\overline{\mathbf{\Phi}}_{k-1}(i)]. \tag{19}$$

The matrices in (19) are all diagonal so each term in $\mathbb{E}[\overline{\mathbf{\Phi}}_k(i)]$ is individually linked with $\mathbb{E}[\overline{\mathbf{\Phi}}_{k-1}(i)]$ and is given by

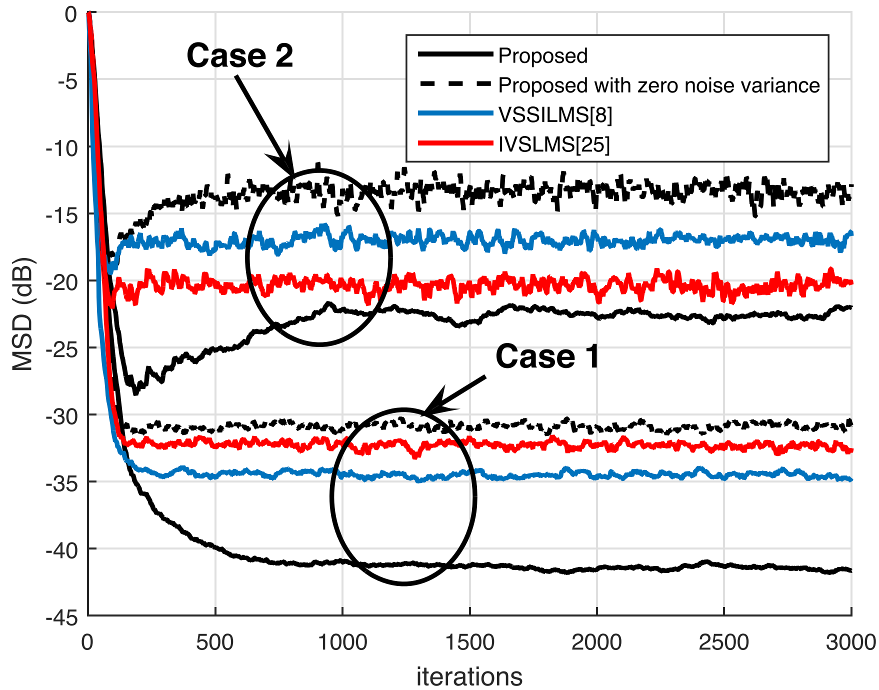

Fig. 4. Performance comparison between VSSILMS [8], IVSLMS [25] and the proposed NCILMS and its zero noise variance version for white input data.

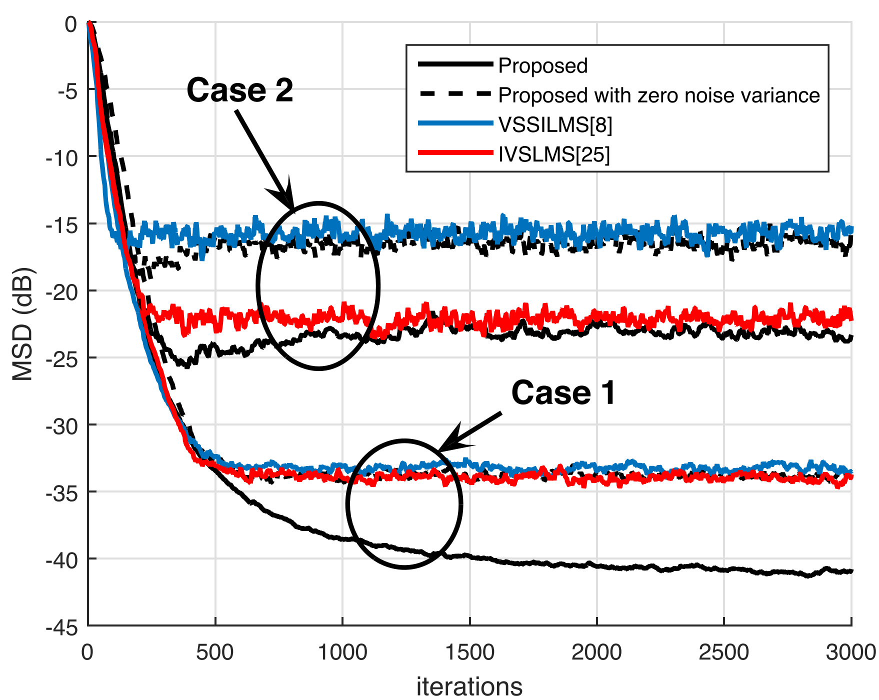

Fig. 5. Performance comparison between VSSILMS [8], IVSLMS [25] and the proposed NCILMS and its zero noise variance version for correlated input data.

$$\mathbb{E}[\overline{\boldsymbol{\Phi}}_{k,m}(i)] = [1-\mathbb{E}[\alpha_k(i)]\bar{\lambda}_{k,m}]\,\mathbb{E}[\overline{\boldsymbol{\Phi}}_{k-1,m}(i)], \tag{20}$$

where $m = 1,2,\ldots,M$. Eventually, we can prove that the mean stability of the proposed algorithm is conditioned by the following bound:

$$\left|[1-\mathbb{E}[\alpha_k(i)]\bar{\lambda}_{k,m}]\right| < 1. \tag{21}$$

Inserting (12) into (21) and simplifying gives the stability bounds for the step-size parameter value for the proposed algorithm, given by

$$0 < \mu_k < \frac{2}{\bar{\lambda}_{max}(1+\gamma_k\mathbb{E}[\lambda_k(i)])}, \tag{22}$$

where $\bar{\lambda}_{\max}$ is the maximum eigenvalue of $\mathbf{R}_{\mathbf{x}}$.

3.2. Mean square analysis

It is clear from (15) that the final estimate of an iteration is defined by the estimate of node N. Thus, the analysis for the network can be easily denoted by the analysis of the result of node N. Thus, (17) is

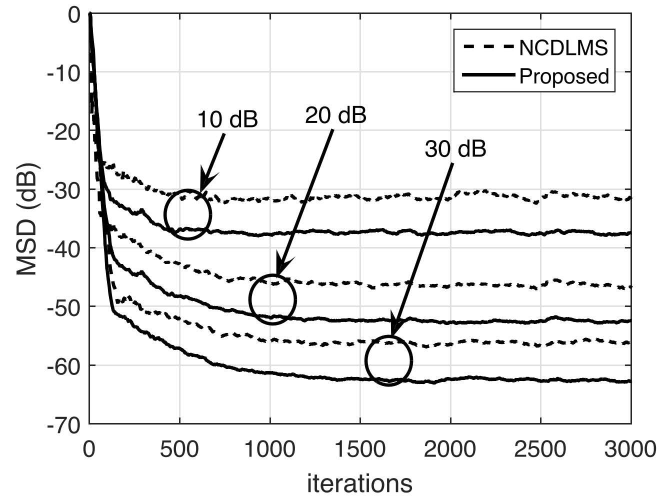

Fig. 6. Performance comparison between NCDLMS algorithm [23] and the proposed algorithm.

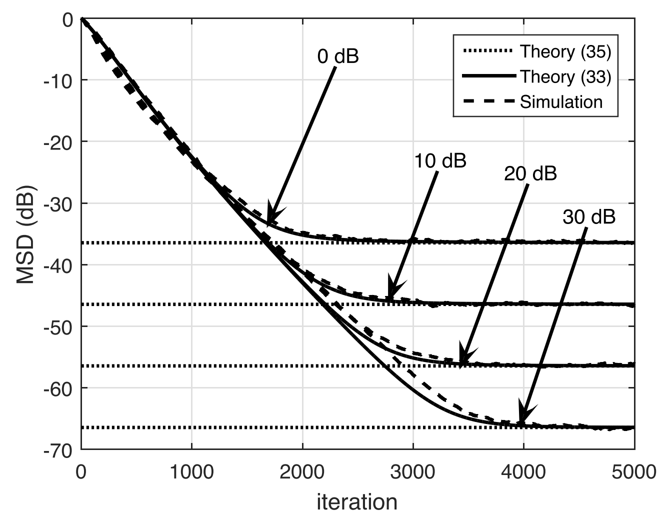

Fig. 7. Theory versus simulation performance for the proposed algorithm.

rewritten for node N after applying the transformation with $\mathbf{G}$ as in the previous section to give

$$\overline{\boldsymbol{\Phi}}_N(i) \quad = \quad \left[\mathbf{I}_M - \alpha_N(i)\overline{\mathbf{x}}_N(i)\overline{\mathbf{x}}_N^T(i)\right]\overline{\boldsymbol{\Phi}}_{N-1}(i) - \alpha_N(i)\overline{\mathbf{x}}_N(i)n_N(i). \qquad (23)$$

The first step is to take a weighted l_2-norm of (23) and then take the expectation. The cross products that include the noise value $n_N(i)$ will disappear when expectation is applied because the noise is independent and zero-mean. Assuming, the data as circular Gaussian, the remaining terms simplify to give

$$\mathbb{E}\left[\|\overline{\boldsymbol{\Phi}}_N(i)\|^2_{\overline{\Sigma}}\right] \quad = \quad \mathbb{E}\left[\|\overline{\boldsymbol{\Phi}}_{N-1}(i)\|^2_{\overline{\Sigma}'}\right] + \sigma^2_{n,N}\mathbb{E}\left[\alpha^2_N\right]Tr\{\boldsymbol{\Lambda}\overline{\Sigma}\}, \qquad (24)$$

where $\overline{\Sigma}$ is a general weighting matrix, $\sigma^2_{n,N}$ is the noise variance at node N and $\overline{\Sigma}'$ is given by

$$\overline{\Sigma}' \quad = \quad \overline{\boldsymbol{\Sigma}} - \mathbb{E}[\alpha_N(i)][\overline{\Sigma}\boldsymbol{\Lambda} + \boldsymbol{\Lambda}\overline{\Sigma}] + \mathbb{E}\left[\alpha^2_N(i)\right][\boldsymbol{\Lambda}Tr\{\overline{\Sigma}\boldsymbol{\Lambda}\} + \kappa\boldsymbol{\Lambda}\overline{\Sigma}\boldsymbol{\Lambda}], \qquad (25)$$

where κ is 1 for complex data and 2 for real data. The interested reader can go through the details of the analysis available within [10], [13–23].

Following the analysis developed in [8], recursion (24) is simplified further by using the $diag\{\}$ operator, resulting in

$$\mathbb{E}\left[\|\overline{\boldsymbol{\Phi}}_N(i)\|^2_{\overline{\boldsymbol{\sigma}}}\right] \quad = \quad \mathbb{E}\left[\|\overline{\boldsymbol{\Phi}}_{N-1}(i)\|^2_{\overline{\mathbf{F}}_N(i)\overline{\boldsymbol{\sigma}}}\right] + \sigma^2_{n,N}\overline{\alpha^2_N}(i)\mathbf{v}^T\overline{\boldsymbol{\sigma}}, \qquad (26)$$

where $\overline{\boldsymbol{\sigma}} = diag\{\overline{\Sigma}\}$, $\mathbf{v} = diag\{\boldsymbol{\Lambda}\}$ and $\overline{\mathbf{F}}_N(i)$ is given by

$$\overline{\mathbf{F}}_N(i) = \mathbf{I}_M - 2\overline{\alpha_N}(i)\Lambda + \overline{\alpha^2_N}(i)\left[\mathbf{v}\mathbf{v}^T + \kappa\Lambda^2\right], \qquad (27)$$

where $\overline{\alpha_N}(i) = \mathbb{E}[\alpha_N(i)]$ and $\overline{\alpha^2_N}(i) = \mathbb{E}[\alpha^2_N(i)] \approx \overline{\alpha_N}^2(i)$. The iterative update for $\overline{\alpha_N}(i)$ is given by

$$\overline{\alpha_N}(i) = \mu_N[1 + \gamma_N\mathbb{E}[\lambda_N(i)]], \qquad (28)$$

with

$$\mathbb{E}[\lambda_N(i+1)] = (1+\beta_N)\mathbb{E}[\lambda_N(i)]\frac{\beta_N}{2}\zeta_N(i), \qquad (29)$$

and $\zeta_N(i) = \mathbb{E}[\|\ \overline{\boldsymbol{\Phi}}_N(i)\ \|^2_{\overline{\boldsymbol{\sigma}}}]$ being the excess mean square error (EMSE) at node N for iteration i when $\overline{\boldsymbol{\sigma}} = \mathbf{v}$.

3.3. Learning behavior

Recursion (26) is not proper because of two reasons. First, the iteration updates the estimate from node $(N-1)$ to node N. A proper iterative update would improve the estimate of node N from iteration $(i-1)$ to i. Secondly, the weighting matrices on the two sides of (26) are $\overline{\Sigma}$ and $\overline{\Sigma}'$, which are transformed using the $diag\{\}$ operator. A proper iterative update should work with the same weighting matrix on both sides. Such a proper update equation is known as the learning behavior of the algorithm.

For this purpose, we need to follow the iterations from the beginning. Initializing with $\mathbb{E}[\|\ \overline{\mathbf{w}}(0)\ \|^2_{\overline{\boldsymbol{\sigma}}}] = \|\ \mathbf{w}^0\ \|^2_{\overline{\boldsymbol{\sigma}}}$, the first iteration begins at node 1 as follows:

$$\begin{aligned}\mathbb{E}[\|\ \overline{\boldsymbol{\Phi}}_1(1)\ \|^2_{\overline{\boldsymbol{\sigma}}}] \quad &= \quad \mathbb{E}\left[\|\ \overline{\mathbf{w}}(0)\ \|^2_{\overline{\mathbf{F}}_1(1)\overline{\boldsymbol{\sigma}}}\right] + \sigma^2_{n,1}\overline{\alpha^2_1}(1)\mathbf{v}^T\overline{\boldsymbol{\sigma}} \\ &= \quad \|\ \mathbf{w}^0\ \|^2_{\overline{\mathbf{F}}_1(1)\overline{\boldsymbol{\sigma}}} + \sigma^2_{n,1}\overline{\alpha^2_1}(1)\mathbf{v}^T\overline{\boldsymbol{\sigma}}.\end{aligned}$$

For node 2, we get

$$\begin{aligned}\mathbb{E}[\|\ \overline{\boldsymbol{\Phi}}_2(1)\ \|^2_{\overline{\boldsymbol{\sigma}}}] \quad &= \quad \|\ \mathbf{w}^0\ \|^2_{\overline{\mathbf{F}}_1(1)\overline{\mathbf{F}}_2(1)\overline{\boldsymbol{\sigma}}} + \sigma^2_{n,1}\overline{\alpha^2_1}(1)\mathbf{v}^T\overline{\mathbf{F}}_2(1)\overline{\boldsymbol{\sigma}} + \sigma^2_{n,2}\overline{\alpha^2_2}(1)\mathbf{v}^T\overline{\boldsymbol{\sigma}} \\ &= \quad \|\ \mathbf{w}^0\ \|^2_{\overline{\mathbf{F}}_1(1)\overline{\mathbf{F}}_2(1)\overline{\boldsymbol{\sigma}}} + \mathbf{g}_1^T(1)\overline{\mathbf{F}}_2(1)\overline{\boldsymbol{\sigma}} + \mathbf{g}_2^T(1)\overline{\boldsymbol{\sigma}},\end{aligned}$$

where $\mathbf{g}_k(i) = \sigma^2_{n,k}\overline{\alpha^2_k}(i)\mathbf{v}$. Thus, for node N we get

$$\begin{aligned}\mathbb{E}[\|\ \overline{\boldsymbol{\Phi}}_N(1)\ \|^2_{\overline{\boldsymbol{\sigma}}}] \quad &= \quad \|\ \mathbf{w}^0\ \|^2_{\prod_{m_1=1}^{N}\overline{\mathbf{F}}_{m_1}(1)\overline{\boldsymbol{\sigma}}} + \mathbf{g}_N^T(1)\overline{\boldsymbol{\sigma}} + \left[\sum_{m=1}^{N-1}\mathbf{g}_m^T(1)\prod_{m_2=m+1}^{N}\overline{\mathbf{F}}_{m_2}(1)\right]\overline{\boldsymbol{\sigma}} \\ &= \quad \|\ \mathbf{w}^0\ \|^2_{\mathbf{F}_N(1)\overline{\boldsymbol{\sigma}}} + \mathbf{B}_N(1)\overline{\boldsymbol{\sigma}},\end{aligned}$$

where $\mathbf{F}_N(1) = \prod_{m_1=1}^{N}\overline{\mathbf{F}}_{m_1}(1)$ and $\mathbf{B}_N(1) = \mathbf{g}_N^T(1)\mathbf{I}_M + \sum_{m=1}^{N-1}\mathbf{g}_m^T(1)\prod_{m_2=m+1}^{N}\overline{\mathbf{F}}_{m_2}(1)$. For iteration 2, we have

$$\begin{aligned}\mathbb{E}[\|\ \overline{\boldsymbol{\Phi}}_N(2)\ \|^2_{\overline{\boldsymbol{\sigma}}}] \quad &= \quad \|\ \mathbf{w}^0\ \|^2_{\mathbf{F}_N(1)\mathbf{F}_N(2)\overline{\boldsymbol{\sigma}}} + [\mathbf{B}_N(2) + \mathbf{B}_N(1)\mathbf{F}_N(2)]\overline{\boldsymbol{\sigma}} \\ &= \quad \|\ \mathbf{w}^0\ \|^2_{\mathbf{A}_N(2)\overline{\boldsymbol{\sigma}}} + \mathbf{C}_N(2)\overline{\boldsymbol{\sigma}},\end{aligned}$$

where $\mathbf{A}_N(2) = \mathbf{F}_N(1)\mathbf{F}_N(2) = \mathbf{A}_N(1)\mathbf{F}_N(2)$ and $\mathbf{C}_N = \mathbf{B}_N(2) + \mathbf{B}_N(1)\mathbf{F}_N(2) = \mathbf{B}_N(2) + \mathbf{C}_N(1)\mathbf{F}_N(2)$. Generalizing this for the i^{th} iteration and realizing that the update for node N is in fact the final update for the network, we can write

$$\mathbb{E}[\|\ \overline{\mathbf{w}}(i)\ \|^2_{\overline{\boldsymbol{\sigma}}}] = \|\ \mathbf{w}^0\ \|^2_{\mathbf{A}_N(i)\overline{\boldsymbol{\sigma}}} + \mathbf{C}_N(i)\overline{\boldsymbol{\sigma}}. \qquad (30)$$

Similarly, for iteration $(i+1)$, we have

$$\mathbb{E}[\|\ \overline{\mathbf{w}}(i+1)\ \|^2_{\overline{\boldsymbol{\sigma}}}] = \|\ \mathbf{w}^0\ \|^2_{\mathbf{A}_N(i+1)\overline{\boldsymbol{\sigma}}} + \mathbf{C}_N(i+1)\overline{\boldsymbol{\sigma}}. \qquad (31)$$

Subtracting (30) from (31) and simplifying, we get

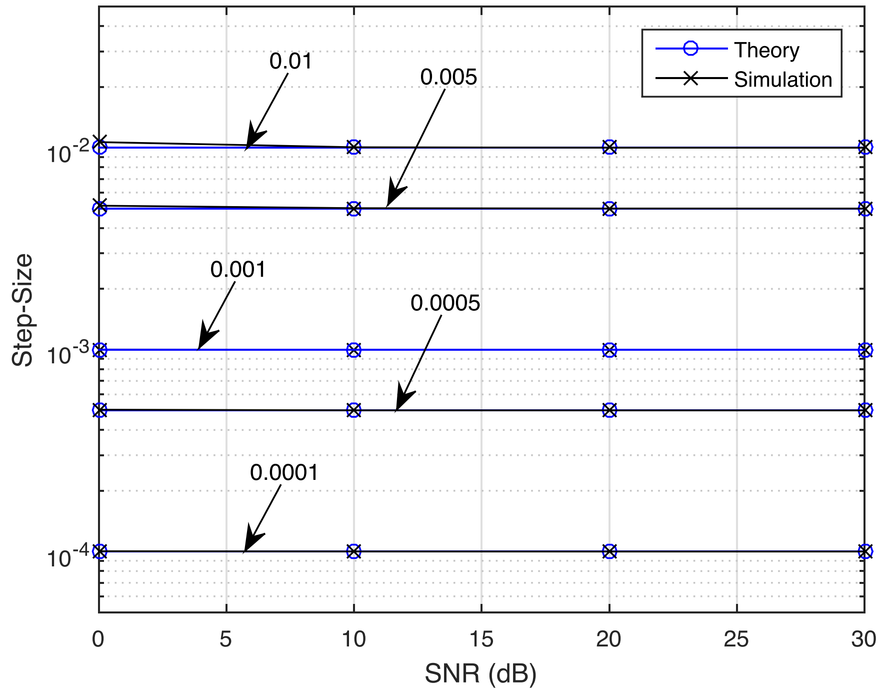

Fig. 8. Theory versus simulation steady-state step-size values.

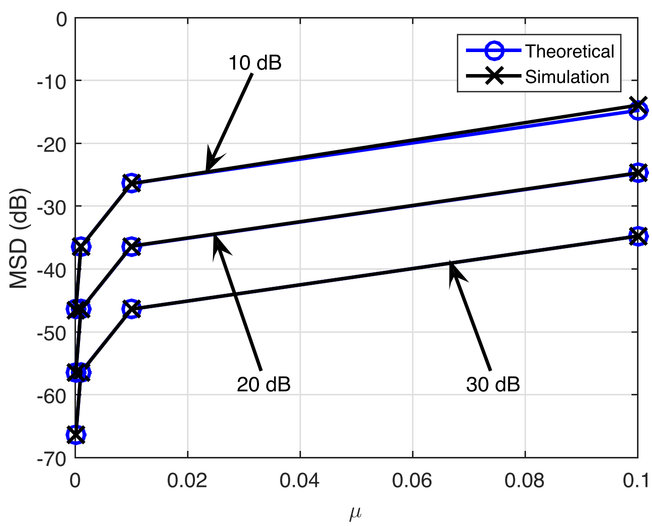

Fig. 9. Steady-state MSD for varying step-size for different SNR values.

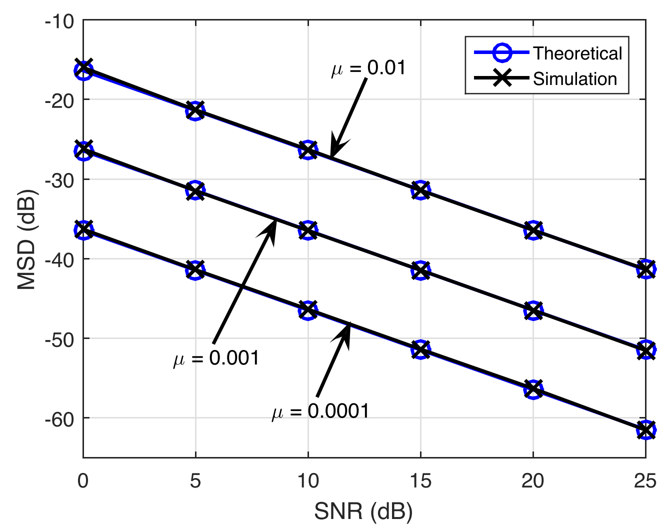

Fig. 10. Steady-state MSD for varying SNR values.

$$\begin{aligned}\mathbb{E}\left[\|\,\overline{\mathbf{w}}(i+1)\,\|^2_{\overline{\boldsymbol{\sigma}}}\right] \;=\;& \mathbb{E}\left[\|\,\overline{\mathbf{w}}(i)\,\|^2_{\overline{\boldsymbol{\sigma}}}\right] + \|\,\mathbf{w}^0\,\|^2_{\mathbf{A}_N(i)[\mathbf{F}_N(i+1)-\mathbf{I}_M]\overline{\boldsymbol{\sigma}}} \\ &+ \mathbf{C}_N(i)[\mathbf{F}_N(i+1)-\mathbf{I}_M]\overline{\boldsymbol{\sigma}} + \mathbf{B}_N(i+1)\overline{\boldsymbol{\sigma}}.\end{aligned} \tag{32}$$

The result in (32) defines the mean square transient learning behavior for the proposed algorithm. From this result, setting $\overline{\boldsymbol{\sigma}} = diag\{\mathbf{I}_M\}$ and $\overline{\boldsymbol{\sigma}} = \mathbf{v}$ in (32) result in the Mean Square Deviation (MSD) and in the Excess Mean Square Error (EMSE), respectively.

3.4. Steady state analysis

At steady-state, the behavior needs to be looked at using (26). Following the learning behavior, it can be seen that the overall update for the network is given by

$$\mathbb{E}\left[\|\,\overline{\mathbf{w}}(i+1)\,\|^2_{\overline{\boldsymbol{\sigma}}}\right] = \mathbb{E}\left[\|\,\overline{\mathbf{w}}(i)\,\|^2_{\mathbf{F}_N(i+1)\overline{\boldsymbol{\sigma}}}\right] + \mathbf{B}_N(i+1)\overline{\boldsymbol{\sigma}}. \tag{33}$$

Table 2
Steady-state MSD vs varying μ.

SNR	Step-Size	Theory	Simulation
	0.0001	– 66.45	– 66.35
30	0.001	– 56.45	– 56.48
	0.01	– 46.40	– 46.39
	0.1	– 34.77	– 34.77
	0.0001	– 56.45	– 56.50
20	0.001	– 46.45	– 46.43
	0.01	– 36.40	– 36.35
	0.1	– 24.77	– 24.71
	0.0001	– 46.45	– 46.49
10	0.001	– 36.45	– 36.49
	0.01	– 26.40	– 26.40
	0.1	– 14.77	– 13.94

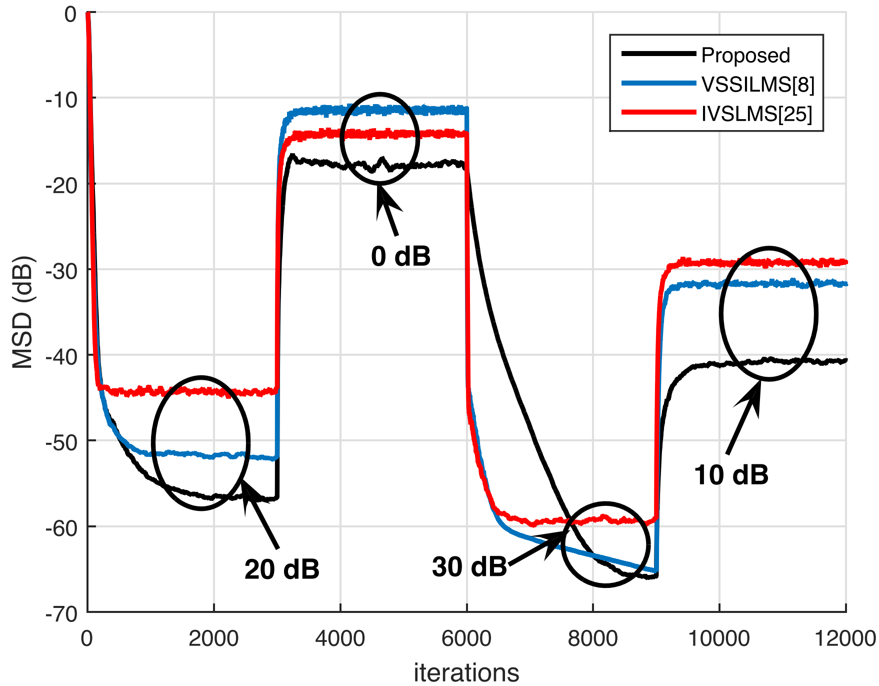

Fig. 11. Proposed algorithm versus VSSILMS algorithm [8] and IVSLMS algorithm [25] for varying SNR values.

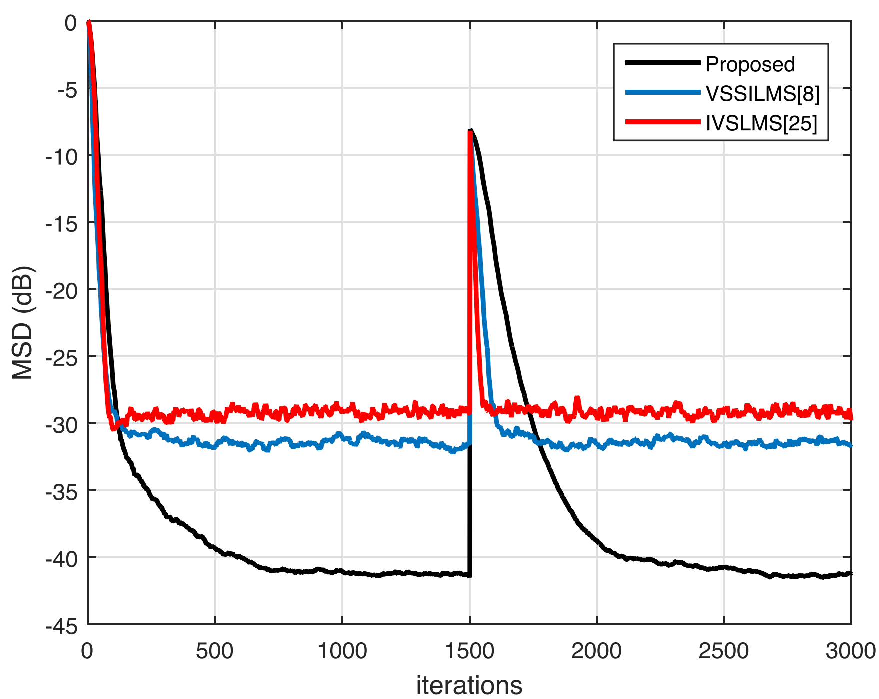

Fig. 12. Proposed algorithm versus VSSILMS algorithm [8] and IVSLMS algorithm [25] for an abrupt change in the unknown system coefficients.

At steady-state, (33) can be solved to give

$$\mathbb{E}\left[\|\overline{\mathbf{w}}_{ss}\|^2_{\overline{\boldsymbol{\sigma}}}\right] = \mathbf{B}_{N,ss}\left[\mathbf{I}_M - \mathbf{F}_{N,ss}\right]^{-1}\overline{\boldsymbol{\sigma}}, \tag{34}$$

where the subscript ss denotes steady-state, $\mathbf{B}_{N,ss} = \mathbf{g}^T_{N,ss}\mathbf{I}_M + \sum_{m=1}^{N-1}\mathbf{g}^T_{m,ss}\prod_{m_2=m+1}^{N}\overline{\mathbf{F}}_{m_2,ss}$, $\mathbf{g}_{m,ss} = \sigma^2_{n,m}\overline{\alpha^2}_{m,ss}\mathbf{v}$, $\mathbf{F}_{N,ss} = \prod_{m_1=1}^{N}\overline{\mathbf{F}}_{m_1,ss}$ and

$$\overline{\mathbf{F}}_{m,ss} = \mathbf{I}_M - 2\overline{\alpha}_{m,ss}\boldsymbol{\Lambda}_m + \overline{\alpha^2}_{m,ss}\left[\mathbf{v}\mathbf{v}^T + \kappa\boldsymbol{\Lambda}^2_m\right]. \tag{35}$$

The steady-state value for the step-size $\overline{\alpha}_{m,ss}$ is governed by the steady-state value of the Lagrangian multiplier, which evolves according to (29). Realizing that the EMSE is a very small number at steady-state, we can safely assume that the Lagrangian multiplier goes to zero at steady-state. As a result, the steady-state step-size value is given by $\overline{\alpha}_{m,ss} = \mu_m$ and $\overline{\alpha^2}_{m,ss} = \mu^2_m$.

For the case where the noise variance is unavailable, (29) changes to

$$\mathbb{E}[\lambda_m(i+1)] = (1-\beta_m)\mathbb{E}[\lambda_m(i)] + \frac{\beta_m}{2}\left[\zeta_m(i) + \sigma^2_{n,m}\right], \tag{36}$$

which, at steady-state, evolves into

$$\mathbb{E}\left[\lambda_{m,ss}\right] = \frac{1}{2}\sigma^2_{n,m}, \tag{37}$$

which is not zero. Thus, the steady-state step-size now becomes $\overline{\alpha}_{m,ss} = \mu_m\left\{1 + \frac{1}{2}\gamma_m\sigma^2_{n,m}\right\}$.

3.5. Complexity analysis

Finally, comparing the computational complexity of the proposed algorithm with those of the ILMS algorithm [2] and the VSSILMS algorithm [8], it was found that the proposed algorithm has five extra multiplications and three extra additions over the ILMS algorithm and two extra multiplications and two extra additions over the VSSILMS algorithm. The computational comparison is shown in Table 1. The table also includes the computational complexity for the IVSLMS algorithm [25]. The computations for the zero noise version (ZNCILMS) of the proposed algorithm are also shown. The ZNCILMS assumes that the noise variance value is unavailable and proceeds without incorporating the value of $\sigma^2_{n,k}$ in (14). The ZNCILMS algorithm is considered as the worst case scenario for the proposed algorithm. In case of a mismatch in the estimation of the noise variance value, the performance of the proposed algorithm exhibits a performance between that of the ideal NCILMS algorithm and that of the ZNCILMS algorithm.

4. Simulation results

This section presents the performance behaviour of the proposed algorithm in different scenarios. In all experiments, the unknown vector length is chosen as $M = 5$ and the network size is $N = 20$. The noise values $n_k(i)$ are zero-mean i.i.d. Gaussian random variables with variances $\sigma^2_{n,k}$. The noise profiles for the two cases are shown in Fig. 2.

Initially, a sensitivity analysis is performed to test the performance of the proposed algorithm for varying values of the parameter γ. Fig. 3 shows this effect. As can be seen, the algorithm reaches the same steady-state in each case. However, the speed of convergence increases as the value of γ increases. It should be noted here that a very large value for γ is not always recommended as this could result in the algorithm diverging instead of converging to the steady-state value. For our experiments, unless otherwise stated, the value for γ is chosen to be 300.

The first experiment gives a comparison in performance of the proposed algorithm with the VSSILMS algorithm [8], the IVSLMS algorithm [25] and the ZNCILMS algorithm for two scenarios, one where the input data is white and the second where the input data is correlated with a correlation factor 0.6. The parameter values are chosen such that they ensure that the speed of convergence for the algorithms is similar for both cases of the noise. Fig. 4 depicts the results for both cases of the noise for white input data. Fig. 5 depicts the results for both cases of the noise for correlated data with correlation factor 0.6. The plots indicate two important points. Firstly, the proposed algorithm clearly outperforms the other VSS algorithms in both scenarios. Secondly, the deterioration in performance for the second case is much less for the proposed algorithm compared with the other algorithms. This clearly shows that the absence of the noise variance value is undesirable and some estimate of the noise variance should be used instead. This is also showcased in the performance of the ZNCILMS algorithm, which degrades in a manner similar to that of the two VSS algorithms.

As for the comparison between the noise-constrained diffusion-based LMS (NCDLMS) algorithm [23] and the proposed algorithm, Fig. 6 reports the findings of this comparison. The results clearly show that the proposed algorithm outperforms the NCDLMS algorithm for different SNR values. Almost a 6 dB gain in favor of our proposed algorithm for an SNR of 30 dB.

The third experiment presents a comparison between theory and simulation. The theoretical results, (32) and (34), are compared with the simulation results and are found in close agreement as shown in Fig. 7. Similarly, Fig. 8 compares the theoretical steady-state step-size values for different SNR values with their counterparts obtained through simulations. Here too, as reported by this figure, the two results are closely matched. To assess more this issue, Figs. 9 and 10 ascertain more of the findings. In both figures, the theory corroborates very well the simulation results. Lastly, as can be seen from Table 2, the simulation results corroborate the theory very well.

Next, a comparison between the proposed algorithm, the VSSILMS algorithm [8] and the IVSLMS algorithm [25] is given in two scenarios with variations during the simulation. First is a scenario where the SNR changes after every 3000 iterations. The initial SNR value is 20 dB, which then changes to 0 dB after 3000 iterations, 30 dB after 6000 iterations and 10 dB after 9000 iterations. The parameter values are the same as in the first experiment, in order to have a sense of how the algorithms perform under changing scenarios. As depicted in Fig. 11, knowledge of the noise variance helps the proposed algorithm in performing consistently, whereas the performances of the VSSILMS and IVSLMS algorithms fluctuate with varying SNR values.

Finally, the algorithm's performance when there is an abrupt change in the unknown system coefficients (time-varying system) is presented. Here, a scenario where the unknown parameter vector $\mathbf{w}^0$ changes after half the number of iterations. The result is depicted in Fig. 12. As can be seen, the proposed algorithm consistently performs better than the other algorithms, despite the change during the simulation. A consistency in performance is achieved with the proposed algorithm in all the scenarios considered for assessment.

5. Conclusion

This work proposes a noise-constrained LMS algorithm for distributed estimation in wireless sensor networks based on the incremental scheme. The algorithm uses the variance of the additive noise as a constraint to develop a specific type of variable step-size for the LMS algorithm that improves performance at the cost of some extra computations. More importantly, the algorithm derivation is followed by a detailed theoretical analysis. Simulation results show that the proposed algorithm clearly outperforms the VSSILMS algorithm, particularly when some nodes have poor SNR compared to the rest of the network. Finally, the theoretical results corroborate very well the simulation results.

Credit author statement

The authors declare that they have all equally contributed in the production of this work.

Declaration of Competing Interest

The authors declare that they have no conflict of Interest.

Data availability

No data was used for the research described in the article.

Acknowledgment

The authors would like to thank the anonymous reviewers for their feedback which has greatly improved the clarity and readability of our manuscript. Also, the authors acknowledge the support provided by the Deanship of Research Oversight and Coordination at KFUPM under Research Grant SB191012.

References

[1] A.H. Sayed, Adaptive networks 102 (2014) 460–497.

[2] C.G. Lopes, A.H. Sayed, Incremental adaptive strategies over distributed networks, IEEE Trans. Signal Process 55 (8) (2007) 4064–4077.
[3] C.G. Lopes, A.H. Sayed, Diffusion least-mean squares over adaptive networks: formulation and performance analysis, IEEE Trans. Signal Process 56 (7) (2008) 3122–3136.
[4] I.D. Schizas, G. Mateos, G.B. Giannakis, Distributed lms for consensus-based in-network adaptive processing, IEEE Trans. Signal Process 57 (6) (2009) 2365–2382.
[5] G. Mateos, I.D. Schizas, G.B. Giannakis, Distributed recursive least-squares for consensus-based in-network adaptive estimation, IEEE Trans. Signal Process 57 (11) (2009) 4583–4588.
[6] N. Takahashi, I. Yamada, A.H. Sayed, Diffusion least-mean squares with adaptive combiners: formulation and performance analysis, IEEE Trans. Signal Process 58 (9) (2010) 4795–4810.
[7] M.O.B. Saeed, A.U.H. Sheikh, A new lms strategy for sparse estimation in adaptive networks. Proc. of IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC), Sydney, NSW, Australia, 2012, pp. 1728–1733.
[8] M.O.B. Saeed, A. Zerguine, An incremental variable step-size lms algorithm for adaptive networks, IEEE Trans. Circuits Systems II: Express Briefs 67 (10) (2020) 2264–2268.
[9] F.S. Cattivelli, A.H. Sayed, Diffusion lms strategies for distributed estimation, IEEE Trans. Signal Process 58 (3) (2010) 1035–1048.
[10] A.H. Sayed, Fundamentals of Adaptive Filtering, in: New York, Wiley, 2003.
[11] M.O.B. Saeed, A. Zerguine, S.A. Zummo, Variable step-size least mean square algorithms over adaptive networks. Proc. of 10th International Conference on Information Sciences Signal Processing and their Applications (ISSPA), Kuala Lumpur, Malaysia, 2010, pp. 381–384.
[12] M.O.B. Saeed, A. Zerguine, A new variable step-size strategy for adaptive networks. Proc. of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove, CA, 2011, pp. 312–315.
[13] M.O.B. Saeed, A. Zerguine, S.A. Zummo, A variable step-size strategy for distributed estimation over adaptive networks, EURASIP J. Adv. Sig. Process 2013 (135) (2013).
[14] H.S. Lee, S.E. Kim, J.W. Lee, W.J. Song, A variable step-size diffusion lms algorithm for distributed estimation, IEEE Trans. Signal Process 63 (7) (2015) 1808–1820.
[15] W. Xia, Y. Wang, A variable step-size diffusion LMS algorithm over networks with noisy links, Signal Process 148 (2018) 205–213.
[16] W. Huang, L. Li, Q. Li, X. Yao, Diffusion robust variable step-size lms algorithm over distributed networks, IEEE Access 6 (2018) 47511–47520.
[17] S. Ashkezari-Toussi, H. Sadoghi-Yazdi, Robust Diffusion LMS Over Adaptive Networks 158 (2019) 201–209.
[18] H. Zayyani, Robust minimum disturbance diffusion LMS for distributed estimation, IEEE Trans. Circuits Syst. II: Exp. Briefs 68 (1) (2021) 521–525.
[19] F. Albu, The constrained stability least mean square algorithm for active noise control. IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), in: Batumi, Georgia, 2018, pp. 1–5.
[20] M.O.B. Saeed, A. Zerguine, S.A. Zummo, Noise constrained diffusion least mean squares over adaptive networks. 21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, in: Istanbul, Turkey, 2010, pp. 288–292.
[21] M.O.B. Saeed, A. Zerguine, S.A. Zummo, A robust lms adaptive algorithm over distributed networks. 2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), in: Pacific Grove, CA, USA, 2011, pp. 547–550.
[22] Y. Wei, S.B. Gelfand, J.V. Krogmeier, Noise-constrained least mean squares algorithm, IEEE Trans. Signal Process 49 (9) (2001) 1961–1970.
[23] M.O.B. Saeed, A. Zerguine, S.A. Zummo, A noise-constrained algorithm for estimation over distributed networks, J. Adap. Cont. & Signal Process 27 (10) (2013) 827–845.
[24] Y. Yu, H. Zhao, Robust incremental normalized least mean square algorithm with variable step sizes over distributed networks 144 (2018) 1–6.
[25] M.O.B. Saeed, S.A. Pasha, A. Zerguine, A variable step-size incremental lms solution for low snr applications 178 (2021) 1–8.
[26] H. Yazdanpanah, M.V.S. Lima, P.S.R. Diniz, On the robustness of set-membership adaptive filtering algorithms, EURASIP J. Adv. Signal Process 2017 (72) (2017).
[27] A. Flores, R.C.D. Lamare, Set-membership adaptive kernel NLMS algorithms: design and analysis, Signal Process 154 (2019) 1–14.
[28] T. Strutz, Estimation of measurement-noise variance for variable-step-size nlms filters. 27th European Signal Processing Conference, 2019, pp. 1–5.