

Steady-State Analysis of the Deficient Length Incremental LMS Adaptive Networks

Ghanbar Azarnia · Mohammad Ali Tinati

Received: 9 March 2014 / Revised: 19 January 2015 / Accepted: 20 January 2015 /
Published online: 6 February 2015
© Springer Science+Business Media New York 2015

Abstract The recently proposed distributed incremental least mean-square (DILMS) adaptive networks assume that the length of the adaptive filter at each node is equal to that of the unknown parameter; in other words, a sufficient length adaptive filter is assumed for each node. However, in many practical situations, the length of the employed adaptive filter at each node is less than that of the unknown parameter. In other words, at each node, a deficient length adaptive filter is employed. Since the analysis results for the sufficient length DILMS algorithm are not necessarily applicable to the deficient length case, so in this paper, we extend existing analysis to study the performance of the DILMS algorithm in this realistic case. More precisely, we derive a closed-form expression for the mean-square deviation (MSD) to explain the steady-state performance at each individual node. Simulation results support the theoretical analysis.

Keywords Adaptive networks · Deficient length · Distributed estimation · DILMS algorithm

1 Introduction

A wireless sensor network (WSN) is composed of a large number of small, low-cost sensors with integrated sensing, processing, and communication abilities [1]. These capabilities make it possible to use WSNs in many applications such as physiological monitoring, environmental monitoring, conditional-based maintenance, smart spaces,

G. Azarnia (✉) · M. A. Tinati
Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran
e-mail: g.azarnia@tabrizu.ac.ir

M. A. Tinati
e-mail: tinati@tabrizu.ac.ir

military, precision agriculture, transportation, factory instrumentation, and inventory tracking [7]. In most of these applications, we deal with the estimation of an unknown parameter using noisy measurements that are collected across sensors. Such a problem can be solved by either a centralized approach or a distributed one. In the centralized solution, every node in the network transmits its measurements to a central fusion center for processing. The central processor would then perform the required estimation tasks and send the results back to the other node. This approach has a critical point of failure at the fusion center and requires large amount of energy for communication. On the other hand, in distributed solution, each node communicates only with its closest neighbors and processing is carried out locally at every node. It is generally believed that in a WSN, the energy required for local computation is much less than what is used for communications. Furthermore, energy consumption is a critical issue in WSNs, so distributed solutions are much desirable for such situations. Moreover, in many applications, it is necessary to perform estimation task in a constantly changing environment without having a statistical model for the underlying processes of interest. This issue requires the distributed processing to be adaptive. Networks in which all nodes are equipped with the adaptive capabilities are called adaptive networks, networks with distributed adaptive estimation algorithms. Using cooperative processing in conjunction with adaptive filtering per node not only enables the tracking of the variations in the environment but also the topology of the network.

Distributed adaptive estimation algorithms could be categorized based on the mode of cooperation between the nodes and the adaptive filter that they use, such as incremental least mean-square (LMS) algorithm [12], incremental recursive least squares (RLS) algorithm [18], incremental affine projection-based adaptive (APA) algorithm [11], diffusion LMS algorithm [13], and diffusion RLS algorithm [4]. In an incremental mode of cooperation, information is sequentially circulated from one node to the adjacent node, but in a diffusion implementation, each node communicates with all its neighbors as dictated by the network topology. Each of these schemes has their own advantages and disadvantages. From the adaptive algorithm point of view which is used in the nodes, LMS algorithm has low computational cost, is robust, and has good tracking performance, but it suffers from slow initial convergence. On the other hand, RLS algorithm has faster convergence rate but the major drawback of RLS is surely its high computational complexity. Compared to the RLS algorithm, the APA algorithm has less computational cost at each node, has reduced internode communications, and memory cost is low while it retains an acceptable steady-state performance. In addition, this algorithm shows improved performance as compared to LMS algorithm in the highly correlated input case. From the cooperation mode approach point of view, the main advantage of the incremental mode of cooperation is its ability to reduce the required energy and communication resources, but it has one major drawback. This algorithm needs to establish a cyclic path through the network, which in large-size networks is a NP-hard problem, and is not guaranteed to exist in general case. In addition, cyclic paths are not robust to node or link failure. In comparison, in diffusion strategies, no cyclic path is required, and these schemes are scalable, robust to node or link failure, and are more amenable to distributed implementation. Nevertheless, diffusion-based methods need more communication resources and suffer from low convergence problem. However, some work has been done to overcome the dis-

advantages of the incremental and diffusion methods. For example, [14] relaxes the requirement to establish a cyclic path over the network by defining a random cooperation walk over the nodes, while keeping nearly the same mean-square performance. In order to save energy and communication resources in diffusion cooperation mode, in [15], a probabilistic diffusion algorithm is developed that lets the nodes to communicate only with a subset of their direct neighbors chosen at random. [17] rather than picking nodes at random proposes a dynamic technique that enables the node to choose from among its neighbors those that are likely to lead to the best MSD performance. Also, hierarchical diffusion algorithm [5], multilevel diffusion algorithm [6], and diffusion algorithm with adaptive combiner [19] are proposed in order to improve the performance of diffusion algorithm. The effect of noisy links on the performance of incremental and diffusion adaptive networks is studied in [9] and [10]. The importance of such study stems from this fact that the performance of distributed adaptive estimation algorithm can drastically be deteriorated in the presence of noisy links [9, 10].

In all of the previous works [4–6, 9–15, 17–19], it is assumed that the length of the adaptive filter at each node is equal to that of the unknown parameter. Actually, the length of the unknown parameter is unknown similar to its coefficients. So, in many practical situations, at each node, a deficient length adaptive filter, whose length is less than that of the unknown parameter, is employed.

In principle, the minimum mean-squared error (MMSE) is a monotonic non-increasing function of the tap-length, but the decrease in the MMSE due to the tap-length increase always becomes trivial when the tap-length is long enough. Obviously, it is not suitable to have too long a filter, as it not only unnecessarily increases the complexity but also introduces more adaption noise. Therefore, there exists an optimum tap-length, L_{opt} , that best balances the steady-state performance and complexity. Usually, one does not know what is the optimum filter length; therefore, each node is equipped with an adaptive filter with M coefficients, where we assume $M < L_{opt}$. In other words, at each node, a deficient length adaptive filter is employed. Since the theoretical results in the sufficient length case do not necessarily apply to the realistic deficient length situation, so in this paper we study the performance of the deficient length DILMS algorithm. More precisely, we derive a closed-form expression for the MSD to explain the steady-state performance at each individual node. However, for a non-distributive adaptive filter case, such an analysis has been done [2, 3, 8, 16], but this analysis in the distributive adaptive network domain is challenging due to the fact that the nodes in every neighborhood interact with one another, and therefore, a successful analysis must take into account both the temporal and spatial interconnectiveness of the data. Different nodes will converge to different MSD levels, reflecting the statistical diversity of the data and the different noise levels. Although the authors in [20–22] compensated the unstructured uncertainty (i.e., nonlinear friction, external disturbances, and/or unmodeled dynamics), they did not consider the unknown length of the unknown parameter. In these works, the goal was to have the inertia load to track any specified smooth motion trajectory x_{1d} as close as possible, where x_{1d} is a one-dimensional (time-varying) scalar; in other words, the dimension of x_{1d} is known. However, in our paper, we assume that the dimensions of parameters like that are also unknown. Simulation results support our theoretical analysis.

Notation: For the ease of reference, the main symbols used in this paper are listed below

$col\{\cdot\}$	Column vector;
$E\{\cdot\}$	Statistical expectation;
$\ \cdot\ ^2$	Squared Euclidean norm operation;
$ \cdot ^2$	Absolute squared operation;
$\mathbf{0}_{M \times 1}$	$M \times 1$ Zero vector;
$\mathbf{0}_{1 \times M}$	$1 \times M$ Zero vector;
$\mathbf{0}_{M \times L}$	$M \times L$ Zero matrix;
I_L	$L \times L$ Identity matrix;
$(\cdot)^*$	Conjugation for scalars and Hermitian transpose for matrices;

2 Deficient Length DILMS Algorithm

Consider a set of N sensors that are randomly distributed over a region. The purpose was to estimate an $L_{\text{opt}} \times 1$ unknown vector $\mathbf{w}_{L_{\text{opt}}}^o$ from multiple measurements collected at N nodes in the network. We assume that both the coefficients and length L_{opt} of the unknown parameter $\mathbf{w}_{L_{\text{opt}}}^o$ are unknown. Since the length L_{opt} is unknown, so a conjectural length M for the unknown parameter is considered at each node, where $M < L_{\text{opt}}$. Also, suppose that each node k has access to time realizations $\{d_k(i), \mathbf{u}_{k,i}\}$ of zero-mean spatial data $\{d_k, \mathbf{u}_k\}$, where each d_k is a scalar measurement and each \mathbf{u}_k is a $1 \times M$ row regression vector.

Collecting the regression and measurement data into global matrices results

$$\mathbf{U} \triangleq col\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N\} \quad (N \times M) \quad (1)$$

$$\mathbf{d} \triangleq col\{d_1, d_2, \dots, d_N\} \quad (N \times 1) \quad (2)$$

The objective was to estimate the $M \times 1$ vector \mathbf{w} that solves

$$\arg \min_{\mathbf{w}} J(\mathbf{w}) \quad (3)$$

where

$$J(\mathbf{w}) = E \left\{ \|\mathbf{d} - \mathbf{U}\mathbf{w}\|^2 \right\} \quad (4)$$

The optimal solution \mathbf{w}^o satisfies the normal equations [12]

$$\mathbf{r}_{du} = \mathbf{R}_u \mathbf{w}^o \quad (5)$$

where \mathbf{w}^o has length M , and

$$\mathbf{R}_u = E \{ \mathbf{U}^* \mathbf{U} \}, \quad \mathbf{r}_{du} = E \{ \mathbf{U}^* \mathbf{d} \}. \quad (6)$$

Note that in order to use (5) to compute \mathbf{w}^o , each node must have access to the global statistical information $\{\mathbf{r}_{du}, \mathbf{R}_u\}$, which in turn requires more communications

between nodes and computational resources. Moreover, such an approach does not enable the network to respond to changes in statistical properties of data. In [12], a distributed incremental LMS (DILMS) strategy with a cyclic estimator structure is proposed as

$$\boldsymbol{\psi}_k^{(i)} = \boldsymbol{\psi}_{k-1}^{(i)} + \mu_k \mathbf{u}_{k,i}^* \left(d_k(i) - \mathbf{u}_{k,i} \boldsymbol{\psi}_{k-1}^{(i)} \right), \quad k \in \mathcal{N} \tag{7}$$

where $\boldsymbol{\psi}_k^{(i)}$ is the $M \times 1$ local estimate at node k and time i . For each time i , each node k utilizes the local data $d_k(i)$, $\mathbf{u}_{k,i}$ and $\boldsymbol{\psi}_{k-1}^{(i)}$ received from the $(k - 1)$ th node to obtain $\boldsymbol{\psi}_k^{(i)}$. At the end of this cycle, $\boldsymbol{\psi}_N^{(i)}$ is employed as the initial condition for the next time instant at node $k = 1$. In (7), μ_k denotes the local step size at node k , which is a positive small real number.

3 Performance Analysis of Deficient Length DILMS

As mentioned, we assume that there is an optimal length for $\mathbf{w}_{L_{\text{opt}}}^o$ as L_{opt} , where the presumed filter length M is not equal to it. We are going to analyze this condition in the context of incremental adaptive network.

3.1 Data Model and Assumptions

To perform the performance analysis, it is necessary to assume a model for the data as it is commonly done in the literature of adaptive algorithms. In the subsequent analysis, the following assumptions will be considered.

- 1) We assume a linear measurement model as

$$d_k(i) = \mathbf{u}_{L_{\text{opt}}k,i} \mathbf{w}_{L_{\text{opt}}}^o + v_k(i). \tag{8}$$

where $v_k(i)$ is some temporal and spatial white noise sequence with zero mean and variance $\sigma_{v,k}^2$ and is independent of $\mathbf{u}_{L_{\text{opt}}\ell,j}$ and $d_\ell(j)$ for all ℓ, j . The linear model (8) is also called stationary model in which the unknown parameter $\mathbf{w}_{L_{\text{opt}}}^o$ is fixed and statistics of the various input and noise signals are time-invariant. In (8), $\mathbf{u}_{L_{\text{opt}}k,i}$ is a vector with length equal to L_{opt} as

$$\mathbf{u}_{L_{\text{opt}}k,i} = [u_k(i), u_k(i - 1), \dots, u_k(i - L_{\text{opt}} + 1)] \tag{9}$$

and $\mathbf{u}_{k,i}$ consists of M first coefficients of $\mathbf{u}_{L_{\text{opt}}k,i}$ as

$$\mathbf{u}_{k,i} = [u_k(i), u_k(i - 1), \dots, u_k(i - M + 1)] \tag{10}$$

- 2) $\mathbf{u}_{L_{\text{opt}}k,i}$ is independent of $\mathbf{u}_{L_{\text{opt}}\ell,i}$ for $k \neq \ell$.
- 3) $\mathbf{u}_{L_{\text{opt}}k,i}$ is independent of $\mathbf{u}_{L_{\text{opt}}k,j}$ for $i \neq j$.

4) The components of $\mathbf{u}_{L_{opt}k,i}$ are drawn from a zero-mean white Gaussian process with variance $\sigma_{u,k}^2$, in other words the covariance matrix of $\mathbf{u}_{L_{opt}k,i}$ is $\mathbf{R}_{u,k} = \sigma_{u,k}^2 \mathbf{I}$.

According to these assumptions, we are interested in the evaluation of the MSD for the steady-state condition for every node k .

3.2 Performance Analysis

To proceed, we define the following local error signal at each node k as

$$e_k(i) = d_k(i) - \mathbf{u}_{k,i} \boldsymbol{\psi}_{k-1}^{(i)} \tag{11}$$

This signal measures the estimation error in approximating $d_k(i)$ by using information available locally, i.e., $\mathbf{u}_{k,i} \boldsymbol{\psi}_{k-1}^{(i)}$. By this definition, (7) can be written as

$$\boldsymbol{\psi}_k^{(i)} = \boldsymbol{\psi}_{k-1}^{(i)} + \mu_k \mathbf{u}_{k,i}^* e_k(i) \quad k \in \mathcal{N} \tag{12}$$

To carry out the performance analysis, we partition the unknown parameter $\mathbf{w}_{L_{opt}}^o$ as

$$\begin{bmatrix} \widehat{\mathbf{w}}_M \\ \widetilde{\mathbf{w}}_{L_{opt}-M} \end{bmatrix} \tag{13}$$

where $\widehat{\mathbf{w}}_M$ is the M first coefficients of $\mathbf{w}_{L_{opt}}^o$; in other words, $\widehat{\mathbf{w}}_M$ is the part of $\mathbf{w}_{L_{opt}}^o$ that is modeled by $\boldsymbol{\psi}_k^{(i)}$ in each node, and $\widetilde{\mathbf{w}}_{L_{opt}-M}$ is the part of $\mathbf{w}_{L_{opt}}^o$ that is excluded in the estimation of $\mathbf{w}_{L_{opt}}^o$ in each node. This partitioning makes it easy to work with vectors that have different lengths. By this partitioning, (11) could be written as

$$\begin{aligned} e_k(i) &= \mathbf{u}_{L_{opt}k,i} \mathbf{w}_{L_{opt}}^o - \mathbf{u}_{k,i} \boldsymbol{\psi}_{k-1}^{(i)} + v_k(i) \\ &= \mathbf{u}_{L_{opt}k,i} \begin{bmatrix} \widehat{\mathbf{w}}_M \\ \widetilde{\mathbf{w}}_{L_{opt}-M} \end{bmatrix} - \mathbf{u}_{L_{opt}k,i} \begin{bmatrix} \boldsymbol{\psi}_{k-1}^{(i)} \\ \mathbf{0}_{(L_{opt}-M) \times 1} \end{bmatrix} + v_k(i) \\ &= -\mathbf{u}_{L_{opt}k,i} \begin{bmatrix} \boldsymbol{\psi}_{k-1}^{(i)} - \widehat{\mathbf{w}}_M \\ -\widetilde{\mathbf{w}}_{L_{opt}-M} \end{bmatrix} + v_k(i) = -\mathbf{u}_{L_{opt}k,i} \begin{bmatrix} \bar{\boldsymbol{\psi}}_{M,k-1}^{(i)} \\ -\widetilde{\mathbf{w}}_{L_{opt}-M} \end{bmatrix} + v_k(i) \\ &= -\mathbf{u}_{L_{opt}k,i} \bar{\boldsymbol{\psi}}_{L_{opt},k-1}^{(i)} + v_k(i) \end{aligned} \tag{14}$$

The vector $\bar{\boldsymbol{\psi}}_{L_{opt},k-1}^{(i)}$ measures the difference between the weight estimate at node $k-1$ and the desired solution $\mathbf{w}_{L_{opt}}^o$. The vector $\bar{\boldsymbol{\psi}}_{M,k-1}^{(i)}$ is also a measure of difference between weights, but it only uses the coefficients of $\mathbf{w}_{L_{opt}}^o$ that are modeled by $\boldsymbol{\psi}_k^{(i)}$.

Substituting (14) into (12) results

$$\boldsymbol{\psi}_k^{(i)} = \boldsymbol{\psi}_{k-1}^{(i)} - \mu_k \mathbf{u}_{k,i}^* \mathbf{u}_{L_{opt}k,i} \bar{\boldsymbol{\psi}}_{L_{opt},k-1}^{(i)} + \mu_k v_k(i) \mathbf{u}_{k,i}^* \tag{15}$$

Padding the vectors with length M in (15) by $L_{\text{opt}} - M$ zeros results vectors with length L_{opt} that allows subtraction of $\mathbf{w}_{L_{\text{opt}}}^o$ from both sides of (15), so we have

$$\begin{bmatrix} \boldsymbol{\psi}_k^{(i)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} - \mathbf{w}_{L_{\text{opt}}}^o = \begin{bmatrix} \boldsymbol{\psi}_{k-1}^{(i)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} - \mathbf{w}_{L_{\text{opt}}}^o - \mu_k \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} + \mathbf{u}_{L_{\text{opt}}k,i} \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} + \mu_k v_k(i) \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} \tag{16}$$

and according to the definition of $\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)}$, we have

$$\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)} = \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} - \mu_k \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} + \mathbf{u}_{L_{\text{opt}}k,i} \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} + \mu_k v_k(i) \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} \tag{17}$$

by defining

$$\Lambda_k(i) = I_{L_{\text{opt}}} - \mu_k \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} + \mathbf{u}_{L_{\text{opt}}k,i} \tag{18}$$

The Eq. (17) can be written as

$$\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)} = \Lambda_k(i) \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} + \mu_k v_k(i) \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} \tag{19}$$

We are interested in the evaluation of the MSD in the steady-state condition for every node k . This quantity is defined as

$$\text{MSD}_k = E \left\{ \left\| \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} \right\|^2 \right\} \tag{20}$$

In order to derive an expression for this quantity, we write $\left\| \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)} \right\|^2$ as

$$\begin{aligned} \left\| \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)} \right\|^2 &= \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{*(i)} \Lambda_k^*(i) \Lambda_k(i) \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} \\ &\quad + \mu_k v_k(i) \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{*(i)} \Lambda_k^*(i) \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{\text{opt}}-M) \times 1} \end{bmatrix} \\ &\quad + \mu_k v_k(i) \begin{bmatrix} \mathbf{u}_{k,i} & \mathbf{0}_{1 \times (L_{\text{opt}}-M)} \end{bmatrix} \Lambda_k(i) \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} \\ &\quad + \mu_k^2 v_k^2(i) \left\| \mathbf{u}_{k,i} \right\|^2 \end{aligned} \tag{21}$$

To proceed, we should take expectations from both sides of (21). For this, we use the assumptions (1)–(4), and from these assumptions, several results can be inferred as follows

Result 1 Since the matrix $\Lambda_k(i)$ only consists of the regression vectors, it is independent of measurement noise $v_k(i)$.

Result 2 From $\psi_{k-1}^{(i)} = \psi_{k-2}^{(i)} + \mu_{k-1} \mathbf{u}_{k-1,i}^* \left(d_{k-1}(i) - \mathbf{u}_{k-1,i} \psi_{k-2}^{(i)} \right)$, it is clear that $\psi_{k-1}^{(i)}$ only depends on the measurements $d_\ell(j)$ of node $k - 1$ and its previous nodes in current iteration, and also on the measurements of its previous iterations. So, from assumption (1), the noise $v_k(i)$ will be independent of $\psi_{k-1}^{(i)}$ and $\bar{\psi}_{L_{opt},k-1}^{(i)}$.

Result 3 Since $\psi_{k-1}^{(i)}$ only depends on the regression vectors $\mathbf{u}_{\ell,j}$ of node $k - 1$ and previous nodes in current iteration, and the regressions of previous iterations, so from assumptions (2) and (3), it is clear that $\mathbf{u}_{k,i}$ and so $\Lambda_k(i)$ are independent from $\psi_{k-1}^{(i)}$ and $\bar{\psi}_{L_{opt},k-1}^{(i)}$.

Using the assumptions (1)–(4) and their results, and taking expectations of both sides of (21), leads to

$$E \left\{ \left\| \bar{\psi}_{L_{opt},k}^{(i)} \right\|^2 \right\} = E \left\{ \bar{\psi}_{L_{opt},k-1}^{*(i)} \Lambda_k^*(i) \Lambda_k(i) \bar{\psi}_{L_{opt},k-1}^{(i)} \right\} + \mu_k^2 E \left\{ v_k^2(i) \right\} E \left\{ \left\| \mathbf{u}_{k,i} \right\|^2 \right\} \tag{22}$$

To proceed, we need to evaluate the moments in the right-hand side of (22). From result (3), we have

$$E \left\{ \bar{\psi}_{L_{opt},k-1}^{*(i)} \Lambda_k^*(i) \Lambda_k(i) \bar{\psi}_{L_{opt},k-1}^{(i)} \right\} = E \left\{ \bar{\psi}_{L_{opt},k-1}^{*(i)} E \left[\Lambda_k^*(i) \Lambda_k(i) \right] \bar{\psi}_{L_{opt},k-1}^{(i)} \right\} \tag{23}$$

So, first we calculate the following expectation

$$\begin{aligned} E \left\{ \Lambda_k^*(i) \Lambda_k(i) \right\} &= I_{L_{opt}} - \mu_k E \left\{ \begin{bmatrix} \mathbf{u}_{k,i}^* \\ \mathbf{0}_{(L_{opt}-M) \times 1} \end{bmatrix} \mathbf{u}_{L_{opt}k,i} \right\} \\ &\quad - \mu_k E \left\{ \mathbf{u}_{L_{opt}k,i}^* \begin{bmatrix} \mathbf{u}_{k,i} \mathbf{0}_{1 \times (L_{opt}-M)} \end{bmatrix} \right\} \\ &\quad + \mu_k^2 E \left\{ \mathbf{u}_{L_{opt}k,i}^* \mathbf{u}_{L_{opt}k,i} \left\| \mathbf{u}_{k,i} \right\|^2 \right\} \end{aligned} \tag{24}$$

As shown in the Appendix, the following relation holds true for this expectation as

$$E \left\{ \Lambda_k^*(i) \Lambda_k(i) \right\} = \begin{bmatrix} \beta_k I_M & \mathbf{0}_{M \times (L_{opt}-M)} \\ \mathbf{0}_{(L_{opt}-M) \times M} & \eta_k I_{L_{opt}-M} \end{bmatrix} \tag{25}$$

where

$$\beta_k = 1 - 2\mu_k \sigma_{u,k}^2 + \mu_k^2 \sigma_{u,k}^4 (M + 2) \tag{26}$$

and

$$\eta_k = 1 + \mu_k^2 \sigma_{u,k}^4 M \tag{27}$$

On the other hand, from $\bar{\psi}_{L_{opt},k-1}^{(i)} = \begin{bmatrix} \bar{\psi}_{M,k-1}^{(i)} \\ -\tilde{\mathbf{w}}_{L_{opt}-M} \end{bmatrix}$ we have

$$\left\| \bar{\psi}_{L_{opt},k-1}^{(i)} \right\|^2 = \left\| \bar{\psi}_{M,k-1}^{(i)} \right\|^2 + \left\| \tilde{\mathbf{w}}_{L_{opt}-M} \right\|^2 \tag{28}$$

Using (25) and (28) in (23), we get

$$\begin{aligned}
 & E \left\{ \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{*(i)} \Lambda_k^*(i) \Lambda_k(i) \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} \right\} \\
 &= E \left\{ \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{*(i)} \begin{bmatrix} \beta_k I_M & \mathbf{0}_{M \times (L_{\text{opt}}-M)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times M} & \eta_k I_{L_{\text{opt}}-M} \end{bmatrix} \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)} \right\} \\
 &= E \left\{ \begin{bmatrix} \bar{\boldsymbol{\psi}}_{M,k-1}^{*(i)} \\ -\tilde{\mathbf{w}}_{L_{\text{opt}}-M}^* \end{bmatrix} \begin{bmatrix} \beta_k I_M & \mathbf{0}_{M \times (L_{\text{opt}}-M)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times M} & \eta_k I_{L_{\text{opt}}-M} \end{bmatrix} \times \begin{bmatrix} \bar{\boldsymbol{\psi}}_{M,k-1}^{(i)} \\ -\tilde{\mathbf{w}}_{L_{\text{opt}}-M} \end{bmatrix} \right\} \\
 &= E \left\{ \begin{bmatrix} \beta_k \bar{\boldsymbol{\psi}}_{M,k-1}^{*(i)} - \eta_k \tilde{\mathbf{w}}_{L_{\text{opt}}-M}^* \\ \bar{\boldsymbol{\psi}}_{M,k-1}^{(i)} \\ -\tilde{\mathbf{w}}_{L_{\text{opt}}-M} \end{bmatrix} \right\} \\
 &= \beta_k E \left\{ \|\bar{\boldsymbol{\psi}}_{M,k-1}^{(i)}\|^2 \right\} + \eta_k \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 \\
 &= \beta_k E \left\{ \|\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)}\|^2 \right\} + (\eta_k - \beta_k) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 \tag{29}
 \end{aligned}$$

Substituting (29) into (22) results

$$E \left\{ \|\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)}\|^2 \right\} = \beta_k E \left\{ \|\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)}\|^2 \right\} + (\eta_k - \beta_k) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 + \tau_k \tag{30}$$

where

$$\tau_k = \mu_k^2 \sigma_{v,k}^2 \sigma_{u,k}^2 M \tag{31}$$

Now, we rewrite (30) in a summarized form as

$$E \left\{ \|\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(i)}\|^2 \right\} = \beta_k E \left\{ \|\bar{\boldsymbol{\psi}}_{L_{\text{opt}},k-1}^{(i)}\|^2 \right\} + f_k \tag{32}$$

where

$$f_k = (\eta_k - \beta_k) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 + \tau_k \tag{33}$$

Since we are interested in the steady-state analysis, as $(i \rightarrow \infty)$ in (32), so with assumption of $\mathbf{P}_{L_{\text{opt}},k} = \bar{\boldsymbol{\psi}}_{L_{\text{opt}},k}^{(\infty)}$, (32) can be written as

$$E \left\{ \|\mathbf{P}_{L_{\text{opt}},k}\|^2 \right\} = \beta_k E \left\{ \|\mathbf{P}_{L_{\text{opt}},k-1}\|^2 \right\} + f_k \tag{34}$$

Observe, however, that (34) is a coupled equation: it involves both $E \left\{ \|\mathbf{P}_{L_{\text{opt}},k}\|^2 \right\}$ and $E \left\{ \|\mathbf{P}_{L_{\text{opt}},k-1}\|^2 \right\}$, i.e., information from two spatial locations. To resolve this difficulty, we take the advantage of the ring topology [11, 12, 18]. Thus, by iterating (34), we have

$$\begin{aligned}
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},1} \right\|^2 \right\} &= \beta_1 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},N} \right\|^2 \right\} + f_1 \\
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},2} \right\|^2 \right\} &= \beta_2 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},1} \right\|^2 \right\} + f_2 \\
 &\vdots \\
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-2} \right\|^2 \right\} &= \beta_{k-2} E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-3} \right\|^2 \right\} + f_{k-2} \\
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} &= \beta_{k-1} E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-2} \right\|^2 \right\} + f_{k-1} \\
 &\vdots \\
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},N} \right\|^2 \right\} &= \beta_N E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},N-1} \right\|^2 \right\} + f_N
 \end{aligned}
 \tag{35}$$

Observe that according to (35), $E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\}$ can be expressed in terms of $E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-3} \right\|^2 \right\}$ as

$$E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} = \beta_{k-1} \beta_{k-2} E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-3} \right\|^2 \right\} + \beta_{k-1} f_{k-2} + f_{k-1}
 \tag{36}$$

By iterating in this manner, we have

$$\begin{aligned}
 E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} &= \beta_{k-1} \beta_{k-2} \dots \beta_1 \beta_N \beta_{N-1} \dots \beta_k E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} \\
 &\quad + \beta_{k-1} \beta_{k-2} \dots \beta_1 \beta_N \beta_{N-1} \dots \beta_{k+1} f_k \\
 &\quad + \beta_{k-1} \beta_{k-2} \dots \beta_1 \beta_N \beta_{N-1} \dots \beta_{k+2} f_{k+1} \\
 &\quad + \dots + \beta_{k-1} \beta_{k-2} \dots \beta_1 \beta_N f_{N-1} + \beta_{k-1} \beta_{k-2} \dots \beta_1 f_N \\
 &\quad + \beta_{k-1} \beta_{k-2} \dots \beta_2 f_1 + \beta_{k-1} \beta_{k-2} \dots \beta_4 \beta_3 f_2 \\
 &\quad + \dots + \beta_{k-1} \beta_{k-2} f_{k-3} + \beta_{k-1} f_{k-2} + f_{k-1}
 \end{aligned}
 \tag{37}$$

Now, for each node k , we define a set of N quantities as

$$\Pi_{k,\ell} \triangleq \beta_{k-1} \beta_{k-2} \dots \beta_1 \beta_N \beta_{N-1} \dots \beta_{k+\ell} \beta_{k+\ell-1}, \quad \ell = 1, \dots, N
 \tag{38}$$

and if we define s_k as

$$s_k \triangleq \Pi_{k,2} f_k + \Pi_{k,3} f_{k+1} + \dots + \Pi_{k,N-1} f_{k-3} + \Pi_{k,N} f_{k-2} + f_{k-1}
 \tag{39}$$

From (37)–(39), we have

$$E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} = \Pi_{k,1} E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} + s_k
 \tag{40}$$

From (40), we derive an expression for the desired steady-state MSD as

$$E \left\{ \left\| \mathbf{P}_{L_{\text{opt}},k-1} \right\|^2 \right\} = (1 - \Pi_{k,1})^{-1} s_k
 \tag{41}$$

3.3 Discussions on Derived Theoretical Results

Due to the complicated form of the resulting relationship for the steady-state MSD, it is not easy to achieve useful information about the use of deficient length adaptive filter in each node. In order to obtain a clear view about the effect of deficient length application, we simplify the equation (41). For this purpose, we assume that $\mu_k = \mu, \forall k \in \mathcal{N}$ and $\mathbf{R}_{u,k} = \sigma_u^2 \mathbf{I}$. Also, we assume that μ is small enough, so that the μ^2 term can be ignored in β_k . With these assumptions, β_k could be approximated as

$$\beta_k = 1 - 2\mu\sigma_u^2 \tag{42}$$

as a result, $\Pi_{k,1}$ could be approximated as

$$\Pi \triangleq \beta_1\beta_2 \dots \beta_N = \left(1 - 2\mu\sigma_u^2\right)^N \approx \left(1 - 2\mu N\sigma_u^2\right) \tag{43}$$

Similarly, using these assumption, s_k becomes

$$\begin{aligned} s_k &\approx \sum_{k=1}^N f_k \\ &= \sum_{k=1}^N \left(2\mu\sigma_u^2 \left(1 - \mu\sigma_u^2\right) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 + M\mu^2\sigma_u^2\sigma_{v,k}^2\right) \end{aligned} \tag{44}$$

Now, by substituting (43) and (44) into (41), we obtain

$$\begin{aligned} E \left\{ \|\mathbf{P}_{L_{\text{opt}},k-1}\|^2 \right\} &= \frac{1}{2\mu N\sigma_u^2} \sum_{k=1}^N \left(2\mu\sigma_u^2 \left(1 - \mu\sigma_u^2\right) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 + M\mu^2\sigma_u^2\sigma_{v,k}^2\right) \\ &= \left(1 - \mu\sigma_u^2\right) \|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2 + \frac{M\mu}{2N} \sum_{k=1}^N \sigma_{v,k}^2 \end{aligned} \tag{45}$$

Now, several results are implied from (45) as

- 1) The first term on the right-hand side of (45) is indeed the additional error due to the deficient length application. This term consists of all the omitting coefficients of unknown parameter, namely $\tilde{\mathbf{w}}_{L_{\text{opt}}-M}$ in the estimation process. Whenever the considered length for the adaptive filter in each node, namely M , is less than L_{opt} , this term will be large. In other words, as much the selected length is more deficient, the error will increase.
- 2) Unlike the full length case ($M = L_{\text{opt}}$), in which as μ approaches to zero, the steady-state MSD tends toward zero. It is observed that, in the deficient length case with $\mu \rightarrow 0$, the steady-state MSD will tend toward a constant term $\|\tilde{\mathbf{w}}_{L_{\text{opt}}-M}\|^2$.

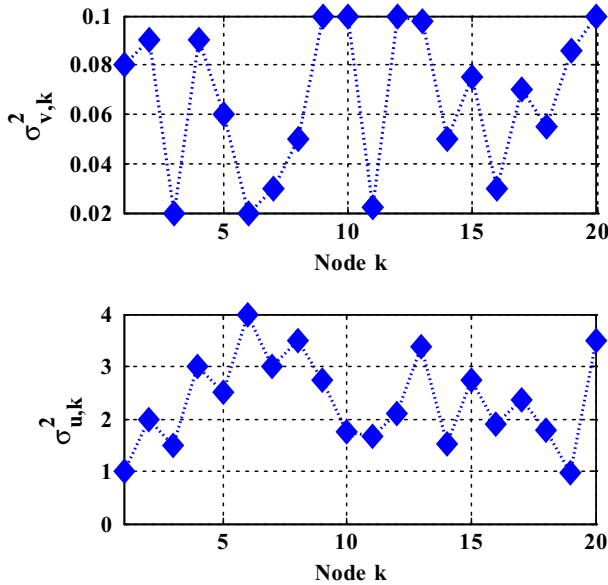


Fig. 1 Observation noise profile $\sigma_{v,k}^2$, and $\sigma_{u,k}^2$

4 Simulations

In this section, we do some simulations and compare the results with theory and show that the derived theoretical expression for MSD can predict the steady-state performance of the deficient length DILMS algorithm. To this aim, we deploy a distributed network with $N = 20$ sensors each running as an adaptive filter with M taps to estimate the $L_{\text{opt}} \times 1$ unknown parameter $\mathbf{w}_{L_{\text{opt}}}^o = \text{col}\{1, 1, \dots, 1\} / \sqrt{L_{\text{opt}}}$ where $L_{\text{opt}} = 10$. The measurement data $\{d_k(i)\}$ are generated according to model (8), and the regressors are assumed to be independent zero-mean Gaussians with covariance matrix as $\mathbf{R}_{u,k} = \sigma_{u,k}^2 \mathbf{I}$. The observation noise variances $\sigma_{v,k}^2$ and $\sigma_{u,k}^2$ are shown in Fig. 1. The steady-state curves are obtained by averaging the last 500 instantaneous samples of 3,500 iterations. All of the simulation results are averaged over 100 independent Monte Carlo runs and the theoretical curves are obtained from (41). The steady-state MSD curves in node k for different tap-length M are shown in Figs. 2 and 3 for step size $\mu_k = 0.02$ and $\mu_k = 0.005$, respectively.

Figures 2 and 3 show that there is a good match between theory and simulations. We can see from Fig. 3 that, for $M = 9, 8, 7$ and 6 , the steady-state MSD values for node $k = 1$ are $-9.665, -6.772, -5.004$, and -3.791 dB, respectively. This implies that, as much the considered length for the adaptive filter in each node, namely M , is less than L_{opt} , the steady-state MSD will increase. In fact, this is the confirmation for the first result, as we mentioned, as much the selected length is more deficient, the error will increase. In order to do a comparison, the steady-state MSD in node k for the case of no deficiency of the filter length, i.e., $M = L_{\text{opt}}$, is also shown in these figures. Comparing these figures shows that the realistic condition, i.e., deficient length, can

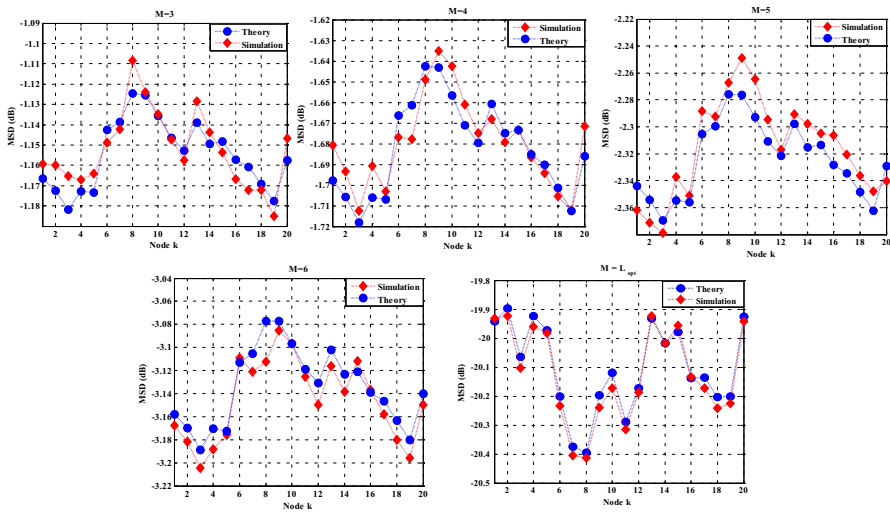


Fig. 2 Steady-state MSD versus node for different tap-length, $\mu_k = 0.02$

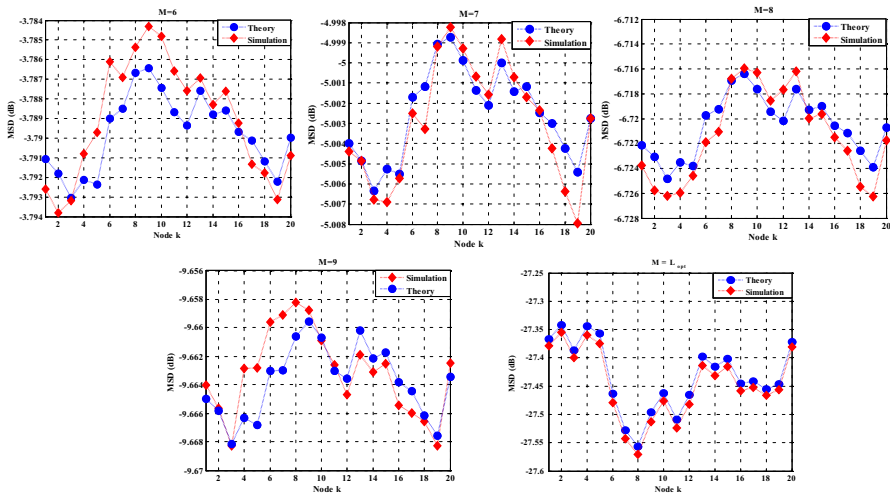


Fig. 3 Steady-state MSD versus node for different tap-length, $\mu_k = 0.005$

drastically deteriorate the performance of DILMS algorithm in term of MSD. These simulations justify the derived theoretical results.

These results could be used to assign a value for the step size for all the nodes. Nodes showing poor performance, or having high noise level, can be assigned with small step sizes; in the limit, they could become simply relay nodes.

The steady-state MSD versus different values of μ for node $k = 1$ are plotted in Figs. 4 and 5 for two cases $M = 8$ ($M < L_{opt}$) and $M = L_{opt}$, respectively. (In this case, we used $\sigma_{u,k}^2 \in (0, .4]$, but the other setup parameters are the same as those in the previous simulations). Again, as it is clear from these figures, simulation results coincide very well to the graphs of the theoretical analysis. On the other hand, as

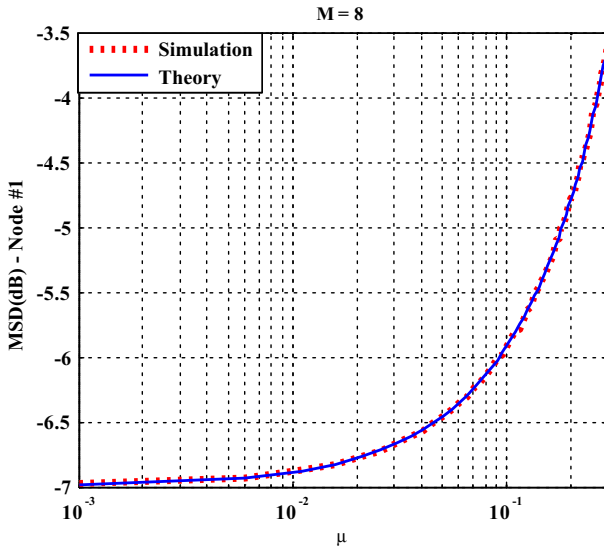


Fig. 4 Steady-state MSD versus μ for node $k = 1$, $M = 8$

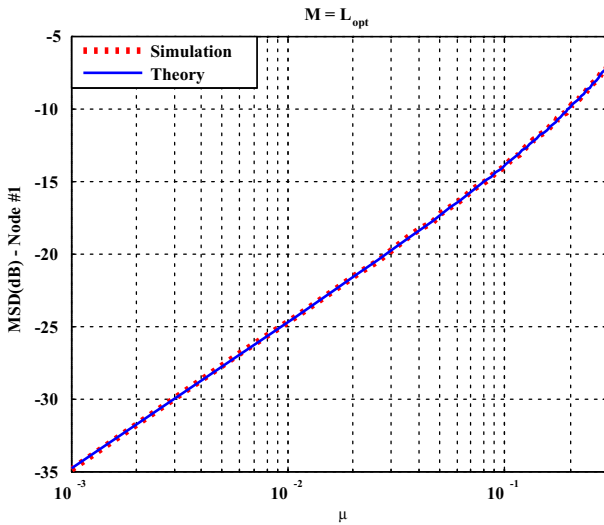


Fig. 5 Steady-state MSD versus μ for node $k = 1$, $M = L_{opt}$

mentioned in the second result, comparison of Figs. 4 and 5 shows that unlike the full length case ($M = L_{opt}$), in which as μ approach to zero, the steady-state MSD tends toward zero, it is observed that in the deficient length case, as $\mu \rightarrow 0$, the steady-state MSD will approach toward a constant term $\|\tilde{w}_{L_{opt}-M}\|^2$ (here, it is equal to -6.9897 dB).

5 Conclusions

In this paper, we have presented a theoretical analysis of the DILMS algorithm, in the case when there is a mismatch between the length of the adaptive filter at each node and the length of the unknown parameter. Based on our analysis, we have derived a closed-form expression for the MSD to explain the steady-state performance at each individual node. In practical physical systems, a conjectural length for an unknown parameter is usually considered, and indeed, our derived expression for the MSD predicts the performance of physical systems for this realistic condition. Our simulation results show there is a good match between our derived expression and computer simulations. The most important results of our analysis were that, in comparison with the sufficient length case, the steady-state MSD includes an additional term that arises from the deficient length application. This term includes all the coefficients of the unknown parameter that are omitted in the estimation process. As the length of the adaptive filter assigned to each node becomes more and more deficient, this term will become larger and larger. The results also show that, unlike the sufficient length case, as the step size tends toward zero, the steady-state MSD tends toward a nonzero constant value. It must be noted that different learning rules such as RLS can also be applied in the context of a distributed network with incremental topology.

Appendix

We show the derivation of (25) in this section. By substituting the vector components into (24), we get

$$\begin{aligned}
 E \{ \Lambda_k^*(i) \Lambda_k(i) \} &= I_{L_{opt}} - \mu_k E \left\{ \begin{bmatrix} u_k^*(i) \\ \vdots \\ u_k^*(i - M + 1) \\ 0 \\ \vdots \\ 0 \end{bmatrix} [u_k(i) \dots u_k(i - L_{opt} + 1)] \right\} \\
 &\quad - \mu_k E \left\{ \begin{bmatrix} u_k^*(i) \\ \vdots \\ u_k^*(i - L_{opt} + 1) \end{bmatrix} [u_k(i) \dots u_k(i - M + 1), 0 \dots 0] \right\} \\
 &\quad + \mu_k^2 E \left\{ \left(\sum_{n=0}^{M-1} |u_k(i - n)|^2 \right) \right. \\
 &\quad \times \left. \begin{bmatrix} u_k^*(i) \\ \vdots \\ u_k^*(i - L_{opt} + 1) \end{bmatrix} [u_k(i) \dots u_k(i - L_{opt} + 1)] \right\} \tag{46}
 \end{aligned}$$

From assumption (4), we have

$$\begin{aligned}
 E \left\{ \left[\begin{array}{c} u_k^*(i) \\ \vdots \\ u_k^*(i - M + 1) \\ 0 \\ \vdots \\ 0 \end{array} \right] [u_k(i) \dots u_k(i - L_{\text{opt}} + 1)] \right\} \\
 = \sigma_{u,k}^2 \left[\begin{array}{cc} I_M & \mathbf{0}_{M \times (L_{\text{opt}} - M)} \\ \mathbf{0}_{(L_{\text{opt}} - M) \times M} & \mathbf{0}_{(L_{\text{opt}} - M) \times (L_{\text{opt}} - M)} \end{array} \right] \tag{47}
 \end{aligned}$$

and

$$\begin{aligned}
 E \left\{ \left[\begin{array}{c} u_k^*(i) \\ \vdots \\ u_k^*(i - L_{\text{opt}} + 1) \end{array} \right] [u_k(i) \dots u_k(i - M + 1), 0 \dots 0] \right\} \\
 = \sigma_{u,k}^2 \left[\begin{array}{cc} I_M & \mathbf{0}_{M \times (L_{\text{opt}} - M)} \\ \mathbf{0}_{(L_{\text{opt}} - M) \times M} & \mathbf{0}_{(L_{\text{opt}} - M) \times (L_{\text{opt}} - M)} \end{array} \right] \tag{48}
 \end{aligned}$$

and finally, for the last expectation of the right-hand side of (46), we have

$$\begin{aligned}
 E \left\{ \left(\sum_{n=0}^{M-1} |u_k(i - n)|^2 \right) \left[\begin{array}{c} u_k^*(i) \\ \vdots \\ u_k^*(i - L_{\text{opt}} + 1) \end{array} \right] \times [u_k(i) \dots u_k(i - L_{\text{opt}} + 1)] \right\} \\
 = E \left\{ \sum_{n=0}^{M-1} \left(|u_k(i - n)|^2 \right. \right. \\
 \left. \left. \times \left[\begin{array}{ccc} |u_k(i)|^2 & \dots & u_k^*(i)u_k(i - L_{\text{opt}} + 1) \\ \vdots & \ddots & \vdots \\ u_k^*(i - L_{\text{opt}} + 1)u_k(i) & \dots & |u_k(i - L_{\text{opt}} + 1)|^2 \end{array} \right] \right) \right\} \tag{49}
 \end{aligned}$$

In (49), the non-diagonal components are zero, since for a Gaussian random process with zero mean, the odd ordered moments are zero. For the diagonal components, two cases are possible

First, for the (m, m)th component of (49), where 1 ≤ m ≤ M, we have

$$\begin{aligned}
 E \left\{ \sum_{n=0}^{M-1} |u_k(i - n)|^2 |u_k(i - m + 1)|^2 \right\} \\
 = \sum_{\substack{n=0 \\ n \neq m-1}}^{M-1} E \left\{ |u_k(i - n)|^2 \right\} E \left\{ |u_k(i - m + 1)|^2 \right\}
 \end{aligned}$$

$$\begin{aligned}
& + E \left\{ |u_k(i-m+1)|^4 \right\} \\
& = \sum_{\substack{n=0 \\ n \neq m-1}}^{M-1} \sigma_{u,k}^4 + 3\sigma_{u,k}^4 = (M-1)\sigma_{u,k}^4 + 3\sigma_{u,k}^4 = (M+2)\sigma_{u,k}^4 \quad (50)
\end{aligned}$$

where we use the fact that for a Gaussian random process, kurtosis is zero.

Second, for the (m, m) th component of (49), where $M+1 \leq m \leq L_{\text{opt}}$, we have

$$\begin{aligned}
E \left\{ \sum_{n=0}^{M-1} |u_k(i-n)|^2 |u_k(i-m+1)|^2 \right\} &= \sum_{n=0}^{M-1} E \left\{ |u_k(i-n)|^2 \right\} \\
E \left\{ |u_k(i-m+1)|^2 \right\} &= M\sigma_{u,k}^4 \quad (51)
\end{aligned}$$

Substituting (47)–(51) into (46) results

$$\begin{aligned}
E \left\{ \Lambda_k^*(i) \Lambda_k(i) \right\} &= I_{L_{\text{opt}}} - 2\mu_k \sigma_{u,k}^2 \begin{bmatrix} I_M & \mathbf{0}_{M \times (L_{\text{opt}}-M)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times M} & \mathbf{0}_{(L_{\text{opt}}-M) \times (L_{\text{opt}}-M)} \end{bmatrix} \\
&+ \mu_k^2 \sigma_{u,k}^4 \begin{bmatrix} (M+2)I_M & \mathbf{0}_{M \times (L_{\text{opt}}-M)} \\ \mathbf{0}_{(L_{\text{opt}}-M) \times M} & MI_{L_{\text{opt}}-M} \end{bmatrix} \quad (52)
\end{aligned}$$

References

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey. *Computer Netw.* **38**(4), 393–422 (2002)
2. R.C. Bilcu, P. Kuosmanen, K. Egiazarian, A new variable length LMS algorithm: theoretical analysis and implementations. in *Proceedings of 9th international conference electronics, circuits, system*, vol. 3, pp. 1031–1034 (2002)
3. R.C. Bilcu, P. Kuosmanen, K. Egiazarian, On length adaptation for the least mean square adaptive filters. *Signal Process.* **86**, 3089–3094 (2006)
4. F.S. Cattivelli, C.G. Lopes, A.H. Sayed, Diffusion recursive least-squares for distributed estimation over adaptive networks. *IEEE Trans. Signal Process.* **56**(5), 1865–1877 (2008)
5. F.S. Cattivelli, A. H. Sayed, Hierarchical diffusion algorithms for distributed estimation, in *Proceedings of IEEE workshop on statistical signal processing (SSP)* (Wales, UK, 2009), pp. 537–540
6. F. S. Cattivelli, A. H. Sayed, Multilevel diffusion adaptive networks, in *Proceedings of international conference acoustics, speech, signal processing, (ICASSP)* (Taipei, Taiwan, 2009), pp. 2789–2792
7. D. Estrin, L. Girod, G. Pottie, M. Srivastava, Instrumenting the world with wireless sensor networks, in *Proceedings of IEEE international Conference on acoustics, speech and signal processing (ICASSP)* (Salt Lake City, UT, 2001), pp. 2033–2036
8. Y. Gu, K. Tang, H. Cui, W. Du, Convergence analysis of a deficient-length LMS filter and optimal-length sequence to model exponential decay impulse response. *IEEE Signal Process. Lett.* **10**(1), 4–7 (2003)
9. A. Khalili, M.A. Tinati, A. Rastegarnia, Steady-state analysis of incremental LMS adaptive networks with noisy links. *IEEE Trans. Signal Process.* **59**(5), 2416–2421 (2011)
10. A. Khalili, M.A. Tinati, A. Rastegarnia, J.A. Chambers, Steady-state analysis of diffusion LMS adaptive networks with noisy links. *IEEE Trans. Signal Process.* **60**(2), 974–979 (2012)
11. L. Li, J.A. Chambers, C.G. Lopes, A.H. Sayed, Distributed estimation over an adaptive incremental network based on the affine projection algorithm. *IEEE Trans. Signal Process.* **58**(1), 151–164 (2010)

12. C.G. Lopes, A.H. Sayed, Incremental adaptive strategies over distributed networks. *IEEE Trans. Signal Process.* **55**(8), 4064–4077 (2007)
13. C.G. Lopes, A.H. Sayed, Diffusion least-mean squares over adaptive networks: formulation and performance analysis. *IEEE Trans. Signal Process.* **56**(7), 3122–3136 (2008)
14. C. G. Lopes, A. H. Sayed, Randomized incremental protocols over adaptive networks, in *Proceedings of international conference on acoustics, speech, signal processing, (ICASSP)* (Dallas, TX, 2010), pp. 3514–3517
15. C.G. Lopes, A.H. Sayed, Diffusion adaptive networks with changing topologies, in *Proceedings of IEEE ICASSP* (Las Vegas, NV, 2008) ,pp. 3285–3288
16. K. Mayyas, Performance analysis of the deficient length LMS adaptive algorithm. *IEEE Trans. Signal Process.* **53**(8), 2727–2734 (2005)
17. O. L. Rortveit, J. H. Husoy, A. H. Sayed, Diffusion LMS with communications constraints, in *Proceedings of 44th asilomar conference on signals, systems and computers* (Pacific Grove, CA, 2010), pp. 1645–1649
18. A. H. Sayed, C. G. Lopes, Distributed recursive least-squares strategies over adaptive networks, in *Proceedings of 40th asilomar conference on signals, systems and computers* (Pacific Grove, CA, 2006), pp. 233–237
19. N. Takahashi, I. Yamada, A.H. Sayed, Diffusion least-mean-squares with adaptive combiners: formulation and performance analysis. *IEEE Trans. Signal Process.* **58**(9), 4795–4810 (2010)
20. J. Yao, Z. Jiao, D. Ma, L. Yan, High-accuracy tracking control of hydraulic rotary actuators with modeling uncertainties. *IEEE/ASME Trans. Mechatron.* **19**(2), 633–641 (2014)
21. J. Yao, Z. Jiao, D. Ma, Adaptive robust control of DC motors with extended state observer. *IEEE Trans. Ind. Electron.* **61**(7), 3630–3637 (2014)
22. J. Yao, Z. Jiao, D. Ma, Extended-state-observer-based output feedback nonlinear robust control of hydraulic systems with backstepping. *IEEE Trans. Ind. Electron.* **61**(11), 6285–6293 (2014)