IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 23, 2018, accepted September 24, 2018, date of publication September 28, 2018, date of current version October 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2872719

Data-Reserved Periodic Diffusion LMS With Low Communication Cost Over Networks

JAE-WOO LEE[1], JUN-TAEK KONG[2], WOO-JIN SONG[2], (Member, IEEE), AND SEONG-EUN KIM[3], (Member, IEEE)

[1]Samsumg Advanced Institute of Technology, Samsung Electronics, Suwon 443-803, South Korea
[2]Department of Electrical Engineering, Pohang University of Science and Technology, Pohang 790-784, South Korea
[3]Department of Electronics and Control Engineering, Hanbat National University, Daejeon 34158, South Korea

Corresponding author: Seong-Eun Kim (sekim@hanbat.ac.kr)

This work was supported in part by the MSIT, South Korea, under the ITRC support program supervised by the IITP under Grant IITP-2017-2012-0-00628 and in part by the National Research Foundation of Korea (NRF) grants funded by the Korean Government (MSIP) under Grant 2017R1C1B5017254 and Grant 2018M3C1B8017549.

ABSTRACT In this paper, we analyze diffusion strategies in which all nodes attempt to estimate a common vector parameter for achieving distributed estimation in adaptive networks. Under diffusion strategies, each node essentially needs to share processed data with predefined neighbors. Although the use of internode communication has contributed significantly to improving convergence performance based on diffusion, such communications consume a huge quantity of power in data transmission. In developing low-power consumption diffusion strategies, it is very important to reduce the communication cost without significant degradation of convergence performance. For that purpose, we propose a data-reserved periodic diffusion least-mean-squares (LMS) algorithm in which each node updates and transmits an estimate periodically while reserving its measurement data even during non-update time. By applying these reserved data in an adaptation step at update time, the proposed algorithm mitigates the decline in convergence speed incurred by most conventional periodic schemes. For a period p, the total cost of communication is reduced to a factor of $1/p$ relative to the conventional adapt-then-combine (ATC) diffusion LMS algorithm. The loss of combination steps in this process leads naturally to a slight increase in the steady-state error as the period p increases, as is theoretically confirmed through mathematical analysis. We also prove an interesting property of the proposed algorithm, namely, that it suffers less degradation of the steady-state error than the conventional diffusion in a noisy communication environment. Experimental results show that the proposed algorithm outperforms related conventional algorithms and, in particular, outperforms ATC diffusion LMS over a network with noisy links.

INDEX TERMS Adaptive networks, distributed estimation, data-reserved periodic diffusion LMS, reduction of communication, robust to noisy communication.

I. INTRODUCTION

Adaptive networks have been widely studied for applications in distributed learning contexts in which sets of nodes cooperatively attempt to achieve a global objective. In adaptive networks, each node has the ability to process data and communicate with a subset of its neighbor nodes, making such networks robust to node failure, scalable, and suitable for implementing decentralized strategies [1]. These inherent advantages have led to a wide range of applications for adaptive networks, including distributed estimation [2]–[5], distributed detection [6], [7], distributed machine learning [8]–[12], beamforming [13], cognitive radio [14], [15], and self-organized behavior in biological systems [16]–[18].

In this paper, we consider the distributed estimation problem in which an entire network collectively estimates a vector parameter of interest. A number of distributed strategies have been developed to implement distributed estimation, including incremental [19]–[25], diffusion [3], [4], [10], [26], and consensus strategies [2], [27]–[36]. Incremental strategies require a predefined cyclic path in which each node cooperates only with one adjacent node. However, determination of this path is not feasible in practice and, furthermore, any link or node failures will cause the entire network to lose its processing ability. In diffusion strategies, each node can communicate with a set of neighbor nodes while updating its estimate, making diffusion based algorithms insensitive to node and link failure. Diffusion strategies are also scalable and capable of real-time learning. Consensus strategies attempt to induce each node in the network to converge to a common estimate, which requires implementation at two time-scales: one for measurement updating and another for sufficient repetition of spatial combinations until agreement is reached. As such, these strategies are not viable for use with streaming measurement data, although single time-scale implementation has been proposed to overcome this weakness [37], [38]. Nevertheless, diffusion algorithms have been shown to have better convergence performance than single time-scale consensus algorithms [39].

A diffusion strategy based on the least mean square (LMS) algorithm (so-called diffusion LMS) and a number of variants have been proposed [3], [4], [9], [40]–[52]. Diffusion LMS algorithms operate over two steps: an adaptation step in which each node updates its estimate using measurement data, and a combination step in which each node exchanges its estimate with neighbor nodes and the estimates are aggregated through linear combination with appropriate weights. In this process, information is refined at each node and effectively diffused over the network, resulting in a significant increase in convergence speed and steady-state accuracy. Depending on the order in which these two steps are taken, combine-then-adapt (CTA) and adapt-then-combine (ATC) diffusion LMS algorithms have been proposed [4]. It has been shown that the ATC structure always achieves lower steady-state error than the CTA structure [4], which suggests that the combination step has a greater impact on steady-state accuracy than the adaptation step. Unlike stand-alone (no cooperation) LMS algorithms, diffusion LMS algorithms require wireless communication between nodes for exchange of data. In wireless *ad hoc* networks, each node often has limited power resources for data processing and communication, and communication is the most power-consuming task [53]. Accordingly, there have been many attempts to reduce internode communications without significant degradation of convergence performance [53]–[71].

This paper proposes an ATC periodic diffusion LMS algorithm in which each node performs both adaptation and combination only at periodic update times to reduce the frequency of data transmission and, ultimately, the cost of communication. Such periodic schemes are very simple and have been widely applied in reducing the computational complexity of adaptive filters. However, the loss of measurement data at non-update times generally slows the convergence speed in direct proportion to the period. To overcome this degradation, we develop a novel periodic diffusion algorithm that reserves all measurement data sensed during non-update times and applies them during the adaptation step at update times.

In the diffusion LMS algorithm, we can easily change the update scheme: the combination step is performed every p steps while the adaptation step alone is updated every iteration. This simple periodic diffusion LMS method can be interpreted as a decimated diffusion: the combination step is decimated to one in every p steps. The decimated diffusion method is simple and intuitive. However, since it has been proven that carrying out only the adaptation step in the steady-state increases the steady-state error relative to performing both adaptation and combination steps [1]. The steady-state error of the decimated diffusion thus increases during adaptation steps and decreases when the combination step is performed together every p iteration. We need to develop a heuristic remedy to mitigate the fluctuations of steady-state errors, but it is generally tough to quantitatively assess which solution is optimal. Furthermore, the modification of the decimated diffusion makes it difficult to analyze its convergence performance mathematically.

To avoid heuristic approaches, we introduce a global cost function based on the sum of the mean-square errors with a periodicity constraint that contains all of the reserved measurements available at periodic update time. Using this cost function, we derive the data-reserved periodic diffusion LMS (DR-PDLMS) algorithm. The cost function can be twice differentiable for a general framework, and many of the results in this work can be extended to those of the twice differentiable cost function as well. Technical differences arise when using different costs instead of the mean-square error cost. These differences are beyond the scope of this paper, they are addressed in [1] and [10], along with other relevant topics. It is sufficient for our purposes here to convey the main ideas by limiting the presentation to the mean-square error cost without much loss in generality.

When the proposed algorithm is implemented with period p, the total cost of communication is reduced to $1/p$. We also analyze the mean-square performance of the proposed algorithm and mathematically compare the steady-state errors of DR-PDLMS with different periods. We also show a noteworthy property of the periodic strategy, namely, that DR-PDLMS is less sensitive to communication noise (link noise) that occurs when data are transmitted between nodes. We mathematically prove that the amount of degradation in steady-state error as a result of communication noise decreases when the period increases. When the communication noise variance is high, DR-PDLMS can outperform the conventional diffusion LMS algorithm in terms of both convergence performance and cost of communication (Fig. 7).

The main contributions of this work can be summarized specifically as follows:

- For low communication overhead, we formulate a periodic global optimization problem and derive an optimal distributed implementation that is the DR-PDLMS algorithm.
- We analyze the mean and mean square behavior of the proposed algorithm to examine how well the proposed algorithm performs and how close the local estimates converge to the desired estimate w^o; we provide the stability condition in the mean and mean-square sense, and derive mathematical expressions of theoretic steady-state mean-square deviation (MSD) and excess mean-square error (EMSE).
- We perform a comparative evaluation of the proposed algorithms with different periods. Although it may

appear intuitive that the convergence performance of the proposed algorithm can deteriorate because information is less diffused over the network, it is difficult to prove this proposition. Through mathematical analysis, we show that degradation of steady-state error always increases when the period p of the proposed algorithm is increased regardless of network environments such as network topology and noise variances.

- Further analysis shows that the proposed algorithm is less sensitive to communication noise than the conventional diffusion LMS algorithm. By comparing the steady-state MSDs of the proposed algorithm over range of periods, we show that the degradation of steady-state MSD decreases as the period decreases. This result highlights the noteworthy fact that the proposed algorithm can reduce both cost of communication and the steady-state network MSD relative to the conventional diffusion LMS algorithm.

In the remainder of this paper, we derive the proposed algorithm and illustrate its behavior as follows. In Section II, we review previous works. In Section III, we introduce a cost function containing a periodicity constraint that aggregates all measurements over p time intervals. In Section IV, we derive the ATC data-reserved periodic diffusion LMS (DR-PDLMS) algorithm. In Section V, we discuss the results of mean-square performance analysis of the algorithm, and in Section VI we compare the steady-state errors of the proposed algorithm with different periods. In Section VII, the simulation results are presented and discussed.

Notation: We use boldface letters, e.g., $\boldsymbol{w}_{k,i}$, for random variables and normal letters, e.g., $w_{k,i}$, for deterministic quantities. We write $\|\cdot\|$ to refer to the Euclidean norm of a vector and E[·] to denote expectations. The superscript $(\cdot)^*$ represents Hermitian transposition, the notation col$\{\cdots\}$ denotes a column vector, and diag $\{\cdots\}$ denotes a diagonal matrix.

II. RELATED WORK

Probabilistic diffusion LMS algorithms [54], [55] consider a changing topology in which pairs of nodes are randomly connected with a probability determined in a manner that reduces the total communication load. In the algorithms developed in [56]–[58], each node exchanges information regarding its predefined quality and uses this information to select a subset of neighbors. The algorithm in [56] applies a scaled product of the noise variance and the regression variance as a selection criterion, with each node selecting neighbor nodes with the minimal value of this criterion. In [57], each node estimates its current mean-square deviation (MSD) and exchanges it with its neighbors: the exchanged MSDs are used to compute the costs of the neighbor nodes and then select the node with the lowest cost. In [58], each node receives the intermediate estimates of a subset of its neighbors that is selected arbitrarily with equal probability.

Another class of algorithms is based on the partial-update process in which a subset of estimates is transmitted [53], [59]–[62], or is based on the set-membership method which transmit the estimates sparsely [62]–[64]. In the algorithms in [65]–[67], each node reduces the dimension of the estimate prior to transmission to reduce communication cost. In [68], each node updates and transmits its estimate only when the current measurement data contribute to a decrease of the MSD. A game-theoretic approach was proposed in [69], in which each node makes its activation decision based on a utility function that captures the trade-off between the node's energy expenditure and contribution.

A kind of block LMS approaches in the field of adaptive filtering was applied to diffusion and incremental LMS algorithms in [70] and [71] (so called block diffusion LMS (BDLMS)) where the adaptation step uses L measurements to calculate the gradient estimate and updates the weight every L iterations. BDLMS has a form similar to that of the proposed DR-PDLMS algorithm, but there are conspicuous differences between the two, with the most significant being that the proposed algorithm performs p adaptations sequentially using p respective measurements based on the incremental gradient method, while BDLMS performs one adaptation using the gradient estimate calculated from all p measurements based on the steepest-descent method. It is well known from the optimization theory that the incremental gradient method outperforms the steepest-descent method [19], [20] (details are given in Appendix A of [22]); the incremental gradient method is generally used to enhance the performance of diffusion strategies as, intuitively, $\psi_{k,i}$ contains more information than $w_{k,i-1}$ [4]. In fact, the proposed DR-PDLMS algorithm not only outperforms the BDLMS algorithm in terms of steady-state error, but also has a wider stability range, which results in more robust implementations (Fig. 8).

III. PROBLEM FORMULATION

Consider N spatially-distributed nodes. The set of nodes connected to node k (including k itself) is called the neighborhood of node k, and is denoted by $\mathcal{N}_k$. Each node k is assumed to receive scalar measurement $\boldsymbol{d}_k(i)$ and the $1 \times M$ regression vector $\boldsymbol{u}_{k,i}$ at each time instant i. To estimate an unknown $M \times 1$ parameter vector w^o in a distributed and adaptive manner, each node k shares information only within $\mathcal{N}_k$. We assume that the desired response $\boldsymbol{d}_k(i)$ is related linearly to the regression vector $\boldsymbol{u}_{k,i}$ as follows:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i), \tag{1}$$

where $\boldsymbol{v}_k(i)$ corresponds to a zero-mean measurement noise with variance $\sigma^2_{v,k}$, which is assumed to be white over time and independent over space. $\boldsymbol{u}_{k,i}$ and $\boldsymbol{v}_l(j)$ are assumed to be independent of each other for all k, l, i, j. Here, we consider all data to be complex values. With respect to $\boldsymbol{u}_{k,i}$, we assume that the autocorrelation functions can vary with time, then the covariance matrix and cross-correlation vector at time i can be defined as

$$R_{u,k,i} \triangleq \mathrm{E}\boldsymbol{u}^*_{k,i}\boldsymbol{u}_{k,i} \quad \text{and } r_{du,k,i} \triangleq \mathrm{E}\boldsymbol{d}_k(i)\boldsymbol{u}^*_{k,i}. \tag{2}$$

A. COST FUNCTION

The common objective of each node in the network is to estimate the parameter vector w^o using not only its own sensed data $\{\boldsymbol{u}_{k,i}, \boldsymbol{d}_k(i)\}$ but also information shared with its neighbor nodes. In this context, the global cost function can be expressed in various forms according to objectives, with the most general global cost function being the sum of the mean-square errors of all nodes [4]. However, use of the global cost function at each time instant requires a high communication burden; to reduce the communication cost, we propose a constraint in which each node is constrained to use the global cost function periodically with period p (i.e., only at time i satisfying $\{i \mod p = 0\}$). However, if the measurements $\{\boldsymbol{u}_{k,i}, \boldsymbol{d}_k(i)\}$ sensed at time i satisfying $\{i \mod p \neq 0\}$ are not used as a result of this periodicity constraint, information will be lost. Therefore, we propose instead a new global cost function corresponding to time i that contains all measurements from the previous p time instants (i.e., from $i-p+1$ to i) as follows:

$$J_i^{period}(w) \triangleq \begin{cases} J_i^{glob}(w) & \text{if } i \bmod p = 0 \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where

$$J_i^{glob}(w) \triangleq \sum_{j=0}^{p-1} \sum_{l=1}^{N} \mathrm{E}\left|\boldsymbol{d}_l(i-j) - \boldsymbol{u}_{l,i-j} w\right|^2. \tag{4}$$

B. LOCAL OPTIMIZATION

At the update time i that satisfies $\{i \mod p = 0\}$, each node attempts to minimize the global cost function (4); however, this task is collectively impossible because it requires that each node has access to the second-order moments $\{R_{u,k,i}, r_{du,k,i}\}$ from across the entire network. Instead, to estimate w^o the nodes must rely solely on data that are available to them locally. We now explain, following [4], how (4) can be transformed into an alternative cost function that allows a fully distributed solution.

We introduce the following individual cost function at node k that uses only measurements from node k:

$$J_{k,i}(w) \triangleq \sum_{j=0}^{p-1} \mathrm{E}\left|\boldsymbol{d}_k(i-j) - \boldsymbol{u}_{k,i-j} w\right|^2. \tag{5}$$

The optimal estimate for w^o at node k that follows from minimizing (5) is then denoted as

$$w_{k,i}^o = \Gamma_{k,i}^{-1} \Lambda_{k,i}, \tag{6}$$

where

$$\Gamma_{k,i} = \sum_{j=0}^{p-1} R_{u,k,i-j} \quad \text{and } \Lambda_{k,i} = \sum_{j=0}^{p-1} r_{du,k,i-j}. \tag{7}$$

Using a completion-of-squares-argument, (5) can be rewritten in terms of $w_{k,i}^o$ as

$$J_{k,i}(w) = \left\|w - w_{k,i}^o\right\|_{\Gamma_{k,i}}^2 + J_{k,i,\min}, \tag{8}$$

where

$$J_{k,i,\min} \triangleq \sum_{j=0}^{p-1} \mathrm{E}|\boldsymbol{d}_l(i-j)|^2 - \Lambda_{k,i}^* \Gamma_{k,i}^{-1} \Lambda_{k,i}, \tag{9}$$

and the notation $\|x\|_\Sigma^2$ denotes that the squared weighted quantity $x^*\Sigma x$. $J_{k,i,\min}$ is independent of w, allowing us to conclude that minimization of (4) is equivalent to the minimization of the following alternative cost function:

$$J_i^{\mathrm{glob}'}(w) = J_{k,i}(w) + \sum_{l \neq k}^{N} \left\|w - w_{l,i}^o\right\|_{\Gamma_{l,i}}^2. \tag{10}$$

IV. DATA-RESERVED PERIODIC DIFFUSION LMS

To minimize the proposed cost function (10), each node k still requires the global information $\Gamma_{l,i}$ and $w_{l,i}^o$ to calculate the second term of (10). Therefore, to enable each node to perform fully distributed processing, (10) should be modified to require only data from the neighborhood of node k. This requirement can be achieved using the approximations used in [4], [10], and [26], in which $\Gamma_{l,i}$ is replaced with a nonnegative scaled multiple of the identity matrix for distributed implementation as follows:

$$\Gamma_{l,i} \approx b_{l,k} I_M. \tag{11}$$

and the range of the summation on the right-hand side of (10) is approximated to the neighborhood of node k. Using these two approximations, the global cost function (10) can be modified to the following form that includes only the data that node k can utilize:

$$J_{k,i}^{\mathrm{dist}}(w) \approx J_{k,i}(w) + \sum_{l \in \mathcal{N}_k \setminus \{k\}} b_{l,k} \left\|w - w_{l,i}^o\right\|^2. \tag{12}$$

The gradient of (12) with respect to w is given by

$$\left[\nabla_w J_{k,i}^{\mathrm{dist}}(w)\right]^* = \sum_{j=0}^{p-1} \left(R_{u,k,i-j} w - r_{du,k,i-j}\right) + \sum_{l \in \mathcal{N}_k \setminus \{k\}} b_{l,k} \left(w - w_{l,i}^o\right). \tag{13}$$

The gradient vector in (13) comprises two terms: a term using p-temporal measurement information, and a term using local information from the neighbor nodes. Applying all terms in order, each node k can update its estimate $w_{k,i-1}$ to $w_{k,i}$ over the following $p+1$ steps at time i:

$$\begin{aligned} \psi_{k,i}^{(1)} &= w_{k,i-1} + \mu_k \left(r_{du,k,i-p+1} - R_{u,k,i-p+1} w_{k,i-1}\right) \\ &\vdots \\ \psi_{k,i}^{(p)} &= \psi_{k,i}^{(p-1)} + \mu_k \left(r_{du,k,i} - R_{u,k,i} w_{k,i-1}\right) \\ w_{k,i} &= \psi_{k,i}^{(p)} + \mu_k \sum_{l \in \mathcal{N}_k \setminus \{k\}} b_{l,k} \left(w_{l,i}^o - w_{k,i-1}\right). \end{aligned} \tag{14}$$

Because $w_{l,i}^o$ is unknown for node k, we replace it with the most improved estimates that are already available at the nodes:

$$w_{l,i}^o \rightarrow \psi_{l,i}^{(p)}. \tag{15}$$

Based on incremental-type arguments [19]–[22], we also replace, with the exception of the first equation, $w_{k,i-1}$ in (14) with $\psi_{k,i}^{(j)}$ for $j = 1, \cdots, p$. Then, (14) is modified as

$$\begin{cases} \psi_{k,i}^{(1)} = w_{k,i-1} + \mu_k \left(r_{du,k,i-p+1} - R_{u,k,i-p+1} w_{k,i-1}\right) \\ \vdots \\ \psi_{k,i}^{(p)} = \psi_{k,i}^{(p-1)} + \mu_k \left(r_{du,k,i} - R_{u,k,i}\psi_{k,i}^{(p-1)}\right) \\ w_{k,i} = \psi_{k,i}^{(p)} + \mu_k \sum_{l \in \mathcal{N}_k \setminus \{k\}} b_{l,k} \left(\psi_{l,i}^{(p)} - \psi_{k,i}^{(p)}\right). \end{cases} \tag{16}$$

By following the derivation of [4], the last equation in (16) becomes the combination step of the diffusion LMS algorithm. We now apply the instantaneous approximations $R_{u,k,i-j} \approx u_{k,i-j}^* u_{k,i-j}$ and $r_{du,k,i-j} \approx d_k(i-j)u_{k,i-j}^*$ using the observed realizations $\{d_k(i-j), u_{k,i-j}\}$. By considering the periodicity in (3), we have the following data-reserved periodic diffusion LMS (DR-PDLMS) algorithm as seen in (17), as shown at the top of the next page, which is summarized in Table 1. In the proposed algorithm, the estimates are updated periodically. Specifically, when time instant i satisfies $\{i \mod p \neq 0\}$, each node stops its update, i.e., $w_{k,i} = w_{k,i-1}$, and reserves its measurements. At the periodic time instant i that satisfies $\{i \mod p = 0\}$, each node performs p adaptations using the reserved measurements and then combines these with the intermediate estimates $\psi_{l,i}^{(p)}$ from the neighbor nodes. This structure enables the proposed algorithm to forgo the estimate update that cause the local estimate deterioration in the transient state without increasing time required to reach steady-state, as all measurements at all time instants are used for the adaptations (as will be shown in the simulation results, Fig. 3). When $p = 1$, the proposed algorithm reduces to the conventional ATC diffusion LMS algorithm.

The proposed algorithm reduces the total cost of communication to a factor of $1/p$ of that required by the conventional ATC diffusion LMS. Although the steady-state error increases as a result of reduced diffusion of information over the network, the proposed algorithm maintains convergence speed by utilizing all reserved measurements (Section VI). In the next section, we will discuss the mean-square performance and stability of the proposed algorithm and then, in the following Section V we will compare the steady-state errors of the proposed algorithm with different periods and discuss its robustness to noisy links by analyzing the deterioration of steady-state error in an environment contaminated by communication noise.

V. MEAN-SQUARE PERFORMANCE ANALYSIS

A. ASSUMPTIONS AND MATRICES

In this section, we examine how well the proposed algorithm performs and how closely the local estimates $w_{k,i}$ converge to the desired estimate w^o. Although this can be achieved by analyzing the mean-square performance of the algorithm, ascertaining mathematically the mean-square convergence

TABLE 1. Data-Reserved periodic diffusion LMS.

For every node $k = 1, \ldots, N$, set $w_{k,-1} = 0$ and for every time instant $i \geq 0$, repeat:
$m = i \bmod p$
IF$\{m = 0\}$
Adaptation Step
$\psi_{k,i}^{(1)} = w_{k,i-1} + \mu_k \hat{u}_{k,1}^* \left(\hat{d}_k(1) - \hat{u}_{k,1} w_{k,i-1}\right)$
$\vdots$
$\psi_{k,i}^{(p-1)} = w_{k,i-1} + \mu_k \hat{u}_{k,p-1}^* \left(\hat{d}_k(p-1) - \hat{u}_{k,p-1}\psi_{k,i}^{(p-2)}\right)$
$\psi_{k,i}^{(p)} = \psi_{k,i}^{(p-1)} + \mu_k u_{k,i}^* \left(d_k(i) - u_{k,i}\psi_{k,i}^{(p-1)}\right)$
Combination Step
$w_{k,i} = \sum_{l \in \mathcal{N}_k} a_{l,k}\psi_{l,i}^{(p)}$
ELSE
Measurement Store
$w_{k,i} = w_{k,i-1}$
$\hat{u}_{k,m} = u_{k,i}$
$\hat{d}_k(m) = d_k(i)$
END

of even a single adaptive filter is known to be challenging because adaptive filters are non-linear, time-variant, and stochastic systems [75]. In an adaptive network, multiple nodes are strongly interconnected as they mutually influence each other. To address this difficulty, we apply the energy conservation approach to the context of adaptive networks [4], [26], which allows us to study the flow of error variances throughout the network. To proceed with this analysis, we assume that the measurements $\{\boldsymbol{u}_{k,i}, \boldsymbol{d}_k(i)\}$ satisfy the linear relation (1) and the following assumptions:

Assumption 1: Regression vectors $\boldsymbol{u}_{k,i}$ are spatially independent and temporally white.

Assumption 2: Noise $\boldsymbol{v}_k(i)$ is a spatially independent and temporally white zero-mean random process with variance $\sigma_{v,k}^2$, and is independent of $\boldsymbol{u}_{l,j}$ for all l and j.

We then define the weight error vectors at node k as

$$\tilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i}, \quad \tilde{\boldsymbol{\psi}}_{k,i}^{(l)} \triangleq w^o - \boldsymbol{\psi}_{k,i}^{(l)} \tag{18}$$

for $l = 1, \ldots, p$. We also define the global weight error vectors containing all weight error vectors over the entire network as

$$\tilde{\boldsymbol{w}}_i = \begin{bmatrix} \tilde{\boldsymbol{w}}_{1,i} \\ \vdots \\ \tilde{\boldsymbol{w}}_{N,i} \end{bmatrix}, \quad \tilde{\boldsymbol{\psi}}_i^{(l)} = \begin{bmatrix} \tilde{\boldsymbol{\psi}}_{1,i}^{(l)} \\ \vdots \\ \tilde{\boldsymbol{\psi}}_{N,i}^{(l)} \end{bmatrix} \tag{19}$$

for $l = 1, \ldots, p$. For future use, we define the following matrices:

$$\mathcal{M} = \text{diag}\{\mu_1 I_M, \cdots, \mu_N I_M\} \tag{20}$$

$$\boldsymbol{\mathcal{D}}_i = \text{diag}\left\{\boldsymbol{u}_{1,i}^*\boldsymbol{u}_{1,i}, \cdots, \boldsymbol{u}_{N,i}^*\boldsymbol{u}_{N,i}\right\} \tag{21}$$

$$\begin{aligned}
&\text{If } \{i \bmod p = 0\}\\
&\qquad \psi_{k,i}^{(1)} = w_{k,i-1} + \mu_k u_{k,i-p+1}^{*}\left(d_k(i-p+1) - u_{k,i-p+1} w_{k,i-1}\right)\\
&\qquad \vdots\\
&\qquad \psi_{k,i}^{(p)} = \psi_{k,i}^{(p-1)} + \mu_k u_{k,i}^{*}\left(d_k(i) - u_{k,i}\psi_{k,i}^{(p-1)}\right)\\
&\qquad w_{k,i} = \sum_{l\in\mathcal{N}_k} a_{l,k}\psi_{l,i}^{(p)}\\
&\text{Else}\\
&\qquad w_{k,i} = w_{k,i-1}.
\end{aligned} \tag{17}$$

$$\boldsymbol{g}_i = \text{col}\left\{\boldsymbol{u}_{1,i}^{*}v_1(i), \cdots, \boldsymbol{u}_{N,i}^{*}v_N(i)\right\} \tag{22}$$

$$\mathcal{D} = \text{E}\{\boldsymbol{\mathcal{D}}_i\} \tag{23}$$

$$\mathcal{G} = \text{E}\left\{\boldsymbol{g}_i\boldsymbol{g}_i^{*}\right\} \tag{24}$$

$$\mathcal{A} = A \otimes I_M, \tag{25}$$

where $\otimes$ denotes the Kronecker product operation. We also define the transition matrix

$$\boldsymbol{\Phi}(i, i-1) = I - \mathcal{M}\boldsymbol{\mathcal{D}}_i, \tag{26}$$

whose transition is defined as

$$\boldsymbol{\Phi}(i, i) \triangleq I, \quad \boldsymbol{\Phi}(i, j) \triangleq \boldsymbol{\Phi}(i, k)\boldsymbol{\Phi}(k, j) \tag{27}$$

for $i \geq k \geq j \geq 0$. Based on (17), the recursion of the global weight error vector at update time instant i is expressed as

$$\begin{aligned}
\tilde{\boldsymbol{\psi}}_{k,i}^{(1)} &= (I - \mathcal{M}\boldsymbol{\mathcal{D}}_{i-p+1})\tilde{\boldsymbol{w}}_{k,i-1} - \mathcal{M}\boldsymbol{g}_{i-p+1}\\
\tilde{\boldsymbol{\psi}}_{k,i}^{(2)} &= (I - \mathcal{M}\boldsymbol{\mathcal{D}}_{i-p+2})\tilde{\boldsymbol{\psi}}_{k,i}^{(1)} - \mathcal{M}\boldsymbol{g}_{i-p+2}\\
&\vdots\\
\tilde{\boldsymbol{\psi}}_{k,i}^{(p)} &= (I - \mathcal{M}\boldsymbol{\mathcal{D}}_i)\tilde{\boldsymbol{\psi}}_{k,i}^{(p-1)} - \mathcal{M}\boldsymbol{g}_i\\
\tilde{\boldsymbol{w}}_{k,i} &= \mathcal{A}^T\tilde{\boldsymbol{\psi}}_{k,i}^{(p)}.
\end{aligned} \tag{28}$$

Using transition matrix (26), we can then derive the update equation between the two time instants $i-p$ and i at which the periodic updates are performed:

$$\tilde{\boldsymbol{w}}_i = \mathcal{A}^T\boldsymbol{\Phi}(i, i-p)\tilde{\boldsymbol{w}}_{i-p} - \sum_{m=0}^{p-1}\mathcal{A}^T\boldsymbol{\Phi}(i, i-m)\mathcal{M}\boldsymbol{g}_{i-m}. \tag{29}$$

B. WEIGHTED VARIANCE RELATION

To analyze the mean-square behavior of the proposed algorithm, we use the energy conservation approach of [4]. First, we let Σ be an $M \times M$ Hermitian positive semi-definite matrix that can be defined as desired; the choice of this matrix will provide us with different quantities, such as the mean-square deviation (MSD) or excess mean-square error (EMSE) further along in the process. At time i, the MSD and EMSE of node k are defined as:

$$\text{MSD}_{k,i} \triangleq \text{E}\left\|\boldsymbol{w}_{k,i} - w^o\right\|^2, \quad \text{EMSE}_{k,i} \triangleq \text{E}\left|\boldsymbol{u}_{k,i}\tilde{\boldsymbol{w}}_{k,i-1}\right|^2. \tag{30}$$

Taking a weighted norm with the Hermitian matrix Σ in (29) yields the following weighted variance relation:

$$\begin{aligned}
\text{E}\|\tilde{\boldsymbol{w}}_i\|_\Sigma^2 &= \text{E}\left\|\tilde{\boldsymbol{w}}_{i-p}\right\|^2_{\boldsymbol{\Phi}^*(i,i-p)\mathcal{A}\Sigma\mathcal{A}^T\boldsymbol{\Phi}(i,i-p)}\\
&\quad + \text{E}\Bigg[\left(\sum_{m=0}^{p-1}\mathcal{A}^T\boldsymbol{\Phi}(i, i-m)\mathcal{M}\boldsymbol{g}_{i-m}\right)^{*}\\
&\quad \times\Sigma\left(\sum_{m=0}^{p-1}\mathcal{A}^T\boldsymbol{\Phi}(i, i-m)\mathcal{M}\boldsymbol{g}_{i-m}\right)\Bigg].
\end{aligned} \tag{31}$$

From assumptions 1 and 2, the regression $\boldsymbol{u}_{k,i-m}$ is independent of $\tilde{\boldsymbol{w}}_{i-p}$ for $m = 0, \cdots, p-1$, allowing the first term in the right side of (31) to be rewritten as

$$\text{E}\left\|\tilde{\boldsymbol{w}}_{i-p}\right\|^2_{\boldsymbol{\Phi}^*(i,i-p)\mathcal{A}\Sigma\mathcal{A}^T\boldsymbol{\Phi}(i,i-p)} = \text{E}\left\|\tilde{\boldsymbol{w}}_{i-p}\right\|^2_{\Sigma'}, \tag{32}$$

where

$$\Sigma' = \text{E}\{\boldsymbol{\Phi}^*(i, i-p)\mathcal{A}\Sigma\mathcal{A}^T\boldsymbol{\Phi}(i, i-p)\}. \tag{33}$$

We define the vectorization operation vec($\cdot$) to replace an $M \times M$ diagonal matrix by an $M^2 \times 1$ column vector by stacking the columns of the matrix on top of each other, with the vectorization versions of Σ' and Σ denoted as σ' and σ, respectively. Then, using the property $\text{vec}(U\Sigma V) = (V^T \otimes U)\text{vec}(\Sigma)$,

$$\sigma' = \mathcal{F}\sigma \tag{34}$$

where

$$\begin{aligned}
\mathcal{F} &= \text{E}\left\{\boldsymbol{\Phi}^T(i, i-p)\mathcal{A} \otimes \boldsymbol{\Phi}^*(i, i-p)\mathcal{A}\right\}\\
&= \text{E}\left\{\boldsymbol{\Phi}^T(i, i-p) \otimes \boldsymbol{\Phi}^*(i, i-p)\right\}(\mathcal{A} \otimes \mathcal{A})\\
&= \left\{I - p(I \otimes \mathcal{D}\mathcal{M}) - p(\mathcal{D}^T\mathcal{M} \otimes I) + O(\mathcal{M}^2)\right\}\\
&\quad \times (\mathcal{A} \otimes \mathcal{A}),
\end{aligned} \tag{35}$$

where $O(\mathcal{M}^2)$ includes terms of the order $\mathcal{M}^2$ and higher orders. Here, we introduce the small step-size assumption [4], [10] to render the term $O(\mathcal{M}^2)$ negligible.

Assumption 3: The step sizes are sufficiently small, i.e., $\mu_k \ll 1$, to allow the terms related to the higher-order powers of the step-sizes to be ignored.

Using assumption 3, $\mathcal{F}$ can be approximated as

$$\mathcal{F} \simeq \left\{I - p(I \otimes \mathcal{D}\mathcal{M}) - p(\mathcal{D}^T\mathcal{M} \otimes I)\right\}(\mathcal{A} \otimes \mathcal{A}). \quad (36)$$

Next, using the property $\mathrm{Tr}(\Sigma X) = \mathrm{vec}(X^T)^T\sigma$, the second term in the right side of (31) can be rearranged as

$$\begin{aligned}
&\mathrm{E}\Bigg[\left(\sum_{m=0}^{p-1}\mathcal{A}^T\boldsymbol{\Phi}(i,i-m)\mathcal{M}\boldsymbol{g}_{i-m}\right)^* \\
&\quad\times\Sigma\left(\sum_{m=0}^{p-1}\mathcal{A}^T\boldsymbol{\Phi}(i,i-m)\mathcal{M}\boldsymbol{g}_{i-m}\right)\Bigg] \\
&= \sum_{m=0}^{p-1}\mathrm{Tr}\Big[\mathrm{E}\{\Sigma\mathcal{A}^T\boldsymbol{\Phi}(i,i-m)\mathcal{M}\mathcal{G}\mathcal{M}\boldsymbol{\Phi}^*(i,i-m)\mathcal{A}\}\Big] \\
&= \sum_{m=0}^{p-1}\Big[\mathrm{vec}(\mathrm{E}\{\mathcal{A}^T\boldsymbol{\Phi}^{*T}(i,i-m)\mathcal{M}\mathcal{G}^T\mathcal{M}\boldsymbol{\Phi}^T(i,i-m)\mathcal{A}\})\Big]^T\sigma.
\end{aligned} \quad (37)$$

Using the small-step size assumption, we can rearrange the vectorization term in (37) as

$$\begin{aligned}
&\mathrm{vec}(\mathrm{E}\{\mathcal{A}^T\boldsymbol{\Phi}^{*T}(i,i-m)\mathcal{M}\mathcal{G}^T\mathcal{M}\boldsymbol{\Phi}^T(i,i-m)\mathcal{A}\}) \\
&\quad= (\mathcal{A}^T\otimes\mathcal{A}^T)\mathrm{E}\Big\{\boldsymbol{\Phi}(i,i-m)\otimes\boldsymbol{\Phi}^{*T}(i,i-m)\Big\}\mathrm{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M}) \\
&\quad\approx (\mathcal{A}^T\otimes\mathcal{A}^T)\Bigg(I - \mathrm{E}\left\{\sum_{h=0}^{m-1}\mathcal{M}\boldsymbol{\mathcal{D}}_{i-h}\otimes I\right\} \\
&\qquad -\mathrm{E}\left\{\sum_{h=0}^{m-1}I\otimes\mathcal{M}\boldsymbol{\mathcal{D}}_{i-h}^T\right\}\Bigg)\mathrm{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M}) \\
&\quad= (\mathcal{A}^T\otimes\mathcal{A}^T)\Big(I - (m\mathcal{M}\mathcal{D}\otimes I) - \big(I\otimes m\mathcal{M}\mathcal{D}^T\big)\Big) \\
&\qquad\times\mathrm{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M}).
\end{aligned} \quad (38)$$

Substituting (32), (37), and (38) into (31), we reach the following weighted variance recursion:

$$\begin{aligned}
&\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_\sigma \\
&\quad= \mathrm{E}\,\|\tilde{\boldsymbol{w}}_{i-p}\|^2_{\mathcal{F}\sigma} + \sum_{m=0}^{p-1}\Big[\Big(\mathcal{A}^T\otimes\mathcal{A}^T\Big)\Big(I - \Big(m\mathcal{M}\mathcal{D}\otimes I\Big) \\
&\qquad - \Big(I\otimes m\mathcal{M}\mathcal{D}^T\Big)\Big)\,\mathrm{vec}\Big(\mathcal{M}\mathcal{G}^T\mathcal{M}\Big)\Big]^T\sigma.
\end{aligned} \quad (39)$$

C. STEADY-STATE BEHAVIOR

When the step-size is sufficiently small to make the matrix $\mathcal{F}$ stable, we obtain the following equation from (39) at steady state ($i \to \infty$):

$$\begin{aligned}
&\lim_{i\to\infty}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_{(I-\mathcal{F})\sigma} \\
&\quad= \sum_{m=0}^{p-1}\Big[\Big(\mathcal{A}^T\otimes\mathcal{A}^T\Big)\Big(I - \big(m\mathcal{M}\mathcal{D}\otimes I\big) \\
&\qquad - \Big(I\otimes m\mathcal{M}\mathcal{D}^T\Big)\Big)\mathrm{vec}\Big(\mathcal{M}\mathcal{G}^T\mathcal{M}\Big)\Big]^T\sigma.
\end{aligned} \quad (40)$$

We then introduce the following two matrices for Σ to calculate the MSD and the EMSE at each node k:

$$m_k \triangleq \mathrm{diag}\{e_k\}\otimes I, \quad r_k \triangleq \mathrm{diag}\{e_k\}\otimes R_{u,k} \quad (41)$$

where e_k denotes a $N \times 1$ column vector with a unit entry at the position of k and zeroes elsewhere. The steady-state MSD and EMSE at each node k are then defined as

$$\mathrm{MSD}_k \triangleq \lim_{i\to\infty}\mathrm{E}\,\big\|\boldsymbol{w}_{k,i} - w^o\big\|^2 = \lim_{i\to\infty}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_{\mathrm{vec}\{m_k\}} \quad (42)$$

$$\mathrm{EMSE}_k \triangleq \lim_{i\to\infty}\mathrm{E}\,\big|\boldsymbol{u}_{k,i}\tilde{\boldsymbol{w}}_{k,i-1}\big|^2 = \lim_{i\to\infty}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_{\mathrm{vec}\{r_k\}}. \quad (43)$$

We can obtain the steady-state MSD and EMSE by selecting σ that satisfies $\mathcal{F}\sigma = \mathrm{vec}\{m_k\}$ and $\mathcal{F}\sigma = \mathrm{vec}\{r_k\}$ as shown in (44) and (45), as shown at the top of the next page, respectively. We define the average MSD and EMSE over the entire network as the network MSD and EMSE as follows:

$$\mathrm{MSD}^{\mathrm{network}} = \frac{1}{N}\sum_{k=1}^{N}\mathrm{MSD}_k = \frac{1}{N}\lim_{i\to\infty}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_{\mathrm{vec}\{\bar{m}\}} \quad (46)$$

$$\mathrm{EMSE}^{\mathrm{network}} = \frac{1}{N}\sum_{k=1}^{N}\mathrm{EMSE}_k = \frac{1}{N}\lim_{i\to\infty}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_{\mathrm{vec}\{\bar{r}\}}, \quad (47)$$

where

$$\bar{m} \triangleq \sum_{k=1}^{N}m_k, \quad \bar{r} \triangleq \sum_{k=1}^{N}r_k. \quad (48)$$

D. TRANSIENT BEHAVIOR

Let $b(m) \triangleq (\mathcal{A}^T\otimes\mathcal{A}^T)(I - (m\mathcal{M}\mathcal{D}\otimes I - (I\otimes\mathcal{M}\mathcal{D}^T))$ $\mathrm{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M})$. We can rewrite the weighted recursion (39) as

$$\begin{aligned}
\mathrm{E}\,\|\tilde{\boldsymbol{w}}_i\|^2_\sigma &= \mathrm{E}\,\big\|\tilde{\boldsymbol{w}}_{i-p}\big\|^2_{\mathcal{F}\sigma} + \sum_{m=0}^{p-1}b(m)^T\sigma \\
&= \mathrm{E}\,\big\|W^o\big\|^2_{\mathcal{F}^{\lfloor i/p\rfloor+1}\sigma} + \sum_{n=0}^{\lfloor i/p\rfloor}\left(\sum_{m=0}^{p-1}b(m)^T\mathcal{F}^n\sigma\right) \\
&= \mathrm{E}\,\big\|\tilde{\boldsymbol{w}}_{i-p}\big\|^2_\sigma - \mathrm{E}\,\big\|W^o\big\|^2_{\mathcal{F}^{\lfloor i/p\rfloor}(I-\mathcal{F})\sigma} \\
&\quad + \sum_{m=0}^{p-1}b(m)^T\mathcal{F}^{\lfloor i/p\rfloor}\sigma,
\end{aligned} \quad (49)$$

where $W^o = \mathbb{1}_N\otimes w^o$. For simplicity, we assume that $w_{k,-1} = 0$ for all k. By applying $\sigma = \bar{m}/N$ or $\sigma = \bar{r}/N$, the equation (49) expresses the transient behavior of the network MSD and EMSE, respectively.

E. STABILITY

1) MEAN STABILITY

Taking the expectation of (29),

$$\begin{aligned}
\mathrm{E}\tilde{\boldsymbol{w}}_i &= \mathcal{A}^T\mathrm{E}\{\boldsymbol{\Phi}(i,i-p)\}\mathrm{E}\tilde{\boldsymbol{w}}_{i-p} \\
&= \mathcal{A}^T(I-\mathcal{M}\mathcal{D})^p\mathrm{E}\tilde{\boldsymbol{w}}_{i-p},
\end{aligned} \quad (50)$$

$$\text{MSD}_k = \sum_{m=0}^{p-1} \left[(\mathcal{A}^T \otimes \mathcal{A}^T) \left(I - (m\mathcal{M}\mathcal{D} \otimes I) - \left(I \otimes m\mathcal{M}\mathcal{D}^T \right) \right) \text{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M}) \right]^T (I - \mathcal{F})^{-1} \text{vec}\{m_k\} \tag{44}$$

$$\text{EMSE}_k = \sum_{m=0}^{p-1} \left[(\mathcal{A}^T \otimes \mathcal{A}^T) \left(I - (m\mathcal{M}\mathcal{D} \otimes I) - \left(I \otimes m\mathcal{M}\mathcal{D}^T \right) \right) \text{vec}(\mathcal{M}\mathcal{G}^T\mathcal{M}) \right]^T (I - \mathcal{F})^{-1} \text{vec}\{r_k\}. \tag{45}$$

where assumptions 1 and 2 are used to divide the expectations. Recursion (50) is stable in the mean sense if $\rho\left(\mathcal{A}^T(I - \mathcal{M}\mathcal{D})^p\right) < 1$ where $\rho(\cdot)$ denotes the spectral radius of its argument. Because $\mathcal{A}^T$ is a right-stochastic matrix, $\rho\left(\mathcal{A}^T(I - \mathcal{M}\mathcal{D})^p\right) < 1$ only if $\rho\left(I - \mathcal{M}\mathcal{D}\right) < 1$ [26]. Therefore, the proposed algorithm is stable in the mean sense when the step sizes $\{\mu_k\}$ satisfy

$$0 < \mu_k < \frac{2}{\lambda_{\max}(R_{u,k})} \quad \text{for } k = 1, \cdots, N, \tag{51}$$

where $\lambda_{\max}(X)$ is the maximum eigenvalue of the Hermitian matrix X.

2) MEAN-SQUARE STABILITY

Under assumption 3, the mean-square stability of the proposed algorithm is guaranteed when the following approximate version of $\mathcal{F}$ in (35) is stable [4], [26], i.e., $\rho(\mathcal{F}) < 1$:

$$\begin{aligned} \mathcal{F} &\approx \left\{ \text{E}\boldsymbol{\Phi}^T(i, i-p) \otimes \text{E}\boldsymbol{\Phi}^*(i, i-p) \right\} (\mathcal{A} \otimes \mathcal{A}) \\ &= \left\{ (I - \mathcal{D}^T\mathcal{M})^p \otimes (I - \mathcal{D}\mathcal{M})^p \right\} (\mathcal{A} \otimes \mathcal{A}) \\ &= (I - \mathcal{D}^T\mathcal{M})^p \mathcal{A} \otimes (I - \mathcal{D}\mathcal{M})^p \mathcal{A}. \end{aligned} \tag{52}$$

When the eigenvalues of the $N \times N$ matrix A and the $M \times M$ matrix B are $\lambda_1, \cdots, \lambda_N$ and $\kappa_1, \cdots, \kappa_M$, respectively, the eigenvalues of $A \otimes B$ are $\lambda_i\kappa_j$ for $i = 1, \cdots, N$ and $j = 1, \cdots, M$. Therefore, we can conclude that the matrix $\mathcal{F}$ is stable if and only if $\rho(I - \mathcal{D}^T\mathcal{M}) < 1$ [4], [26], which is the same condition for mean-stability given by (51). Thus, by using a sufficiently small step-size, the stability of the proposed algorithm is ensured in both the mean and mean-square senses.

VI. PERFORMANCE COMPARISON

Under assumption 3 (small step size), the equation (31) can be approximated as

$$\begin{aligned} &\text{E}\, \|\tilde{\boldsymbol{w}}_i\|_\Sigma^2 \\ &= \text{E}\left\|\tilde{\boldsymbol{w}}_{i-p}\right\|^2_{(I-\mathcal{D}\mathcal{M})^p \mathcal{A}\Sigma\mathcal{A}^T(I-\mathcal{M}\mathcal{D})^p} \\ &\quad + \sum_{m=0}^{p-1} \text{Tr}\left[\text{E}\{\Sigma\mathcal{A}^T(I - \mathcal{M}\mathcal{D})^m \mathcal{M}\mathcal{G}\mathcal{M}(I - \mathcal{D}\mathcal{M})^m \mathcal{A}\} \right]. \end{aligned} \tag{53}$$

Then, the steady-state MSD is written as

$$\frac{1}{N}\text{E}\, \|\tilde{\boldsymbol{w}}_\infty\|^2 = \frac{1}{N} \sum_{j=0}^{\infty} \left\{ \text{Tr}\left[\mathcal{B}_p^j \left(\sum_{m=0}^{p-1} \mathcal{Y}_m \right) \mathcal{B}_p^{*j} \right] \right\}, \tag{54}$$

where

$$\begin{aligned} \mathcal{B}_p &= \mathcal{A}^T(I - \mathcal{M}\mathcal{D})^p, \\ \mathcal{Y}_m &= \mathcal{A}^T(I - \mathcal{M}\mathcal{D})^m \mathcal{M}\mathcal{G}\mathcal{M}(I - \mathcal{D}\mathcal{M})^m \mathcal{A}. \end{aligned} \tag{55}$$

To simplify the analysis, we use the following uniform data profile environment [26]:

$$R_{u,k} = R_u, \quad \mu_k = \mu \quad \text{for all } k. \tag{56}$$

Because the noise variances $\{\sigma_{v,k}\}$ vary for all nodes, the signal-to-noise ratios (SNRs) differ among the nodes. We also assume that the combination matrix A is doubly stochastic and satisfies

$$A\mathbb{1} = \mathbb{1} \quad \mathbb{1}^T A = \mathbb{1}^T. \tag{57}$$

Then, by using the property $(X_1 \otimes Y_1)(X_2 \otimes Y_2) = X_1X_2 \otimes Y_1Y_2$, we can rewrite $\mathcal{B}_p$ and $\mathcal{Y}_m$ in (55) as

$$\begin{aligned} \mathcal{B}_p &= (A^T \otimes I_M)\{I_N \otimes (I_M - \mu R_u)^p\} \\ &= A^T \otimes (I_M - \mu R_u)^p \end{aligned} \tag{58}$$

$$\begin{aligned} \mathcal{Y}_m &= \mu^2(A^T \otimes I_M)\left\{I_N \otimes (I_M - \mu R_u)^m\right\} \\ &\quad \times (R_v \otimes R_u)\left\{I_N \otimes (I_M - \mu R_u)^m\right\}(A \otimes I_M) \\ &= \mu^2 \left\{ A^T R_v A \otimes (I_M - \mu R_u)^m R_u (I_M - \mu R_u)^m \right\}. \end{aligned} \tag{59}$$

Using (58) and (59), (54) is expressed as:

$$\begin{aligned} \frac{\mu^2}{N} \sum_{j=0}^{\infty} \sum_{m=0}^{p-1} &\text{Tr}[A^{(j+1)T} R_v A^{j+1} \otimes (I_M - \mu R_u)^{(jp+m)} \\ &\times R_u (I_M - \mu R_u)^{(jp+m)}]. \end{aligned} \tag{60}$$

If we use the subscript substitution $n = jp + m$, j can be expressed by the floor function for a given n, i.e., $j = \lfloor n/p \rfloor$, because $0 \leq m < p$. Then, (60) can be rewritten using the subscript n as

$$\begin{aligned} \frac{\mu^2}{N} \sum_{n=0}^{\infty} &\text{Tr}[A^{(\lfloor n/p \rfloor+1)T} R_v A^{\lfloor n/p \rfloor+1} \otimes (I_M - \mu R_u)^n \\ &\times R_u (I_M - \mu R_u)^n]. \end{aligned} \tag{61}$$

A. EFFECT OF PERIOD P ON STEADY-STATE NETWORK MSD

We now compare the performance of the data-reserved periodic diffusion LMS algorithms with different periods. Using the property $\text{Tr}[X \otimes Y] = \text{Tr}[X]\text{Tr}[Y]$, the difference between the network MSDs with different periods p_1 and p_2 ($p_1 > p_2 \geq 1$) is given by (62), as shown at the top of the next page.

$$\begin{aligned}\mathrm{MSD}^{(p_1)} - \mathrm{MSD}^{(p_2)} &= \frac{\mu^2}{N}\sum_{n=0}^{\infty}\mathrm{Tr}\left[\left(A^{(\lfloor n/p_1\rfloor+1)T}R_vA^{\lfloor n/p_1\rfloor+1} - A^{(\lfloor n/p_2\rfloor+1)T}R_vA^{\lfloor n/p_2\rfloor+1}\right)\otimes(I_M-\mu R_u)^nR_u(I_M-\mu R_u)^{nH}\right]\\ &= \frac{\mu^2}{N}\sum_{n=0}^{\infty}\mathrm{Tr}\left[A^{(\lfloor n/p_1\rfloor+1)T}R_vA^{\lfloor n/p_1\rfloor+1} - A^{(\lfloor n/p_2\rfloor+1)T}R_vA^{\lfloor n/p_2\rfloor+1}\right]\mathrm{Tr}\left[(I_M-\mu R_u)^nR_u(I_M-\mu R_u)^{nH}\right].\end{aligned} \tag{62}$$

The first trace value of (62) is arranged as

$$\mathrm{Tr}\Big[A^{\lfloor n/p_1\rfloor+1}\left(I - A^{\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor}A^{(\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor)T}\right)\times A^{(\lfloor n/p_1\rfloor+1)T}R_v\Big]. \tag{63}$$

We further assume that the matrix $A^{\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor}$ is a doubly stochastic matrix. Using the assumption, the following inequality is satisfied (property (e) of [26, Lemma C.3.]):

$$I - A^{\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor}A^{(\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor)T} \geq 0. \tag{64}$$

Lemma 1: Let P and X be arbitrary $M \times M$ matrices, where X is real symmetric. Then the matrix $Y = PXP^T$ always satisfies $Y \geq 0$ for any choice of P, if $X \geq 0$.

Proof: See Appendix A.

By Lemma 1, we find that

$$A^{\lfloor n/p_1\rfloor+1}\left(I - A^{\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor}A^{(\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor)T}\right)\times A^{(\lfloor n/p_1\rfloor+1)T} \geq 0. \tag{65}$$

Assuming R_v is Hermitian and positive semidefinite, the following inequality is always satisfied for all n:

$$\mathrm{Tr}\Big[A^{\lfloor n/p_1\rfloor+1}\left(I - A^{\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor}A^{(\lfloor n/p_2\rfloor-\lfloor n/p_1\rfloor)T}\right)\times A^{(\lfloor n/p_1\rfloor+1)T}R_v\Big] \geq 0. \tag{66}$$

By lemma 1, the second trace value of (62) is positive semidefinite. Therefore, we can conclude that

$$\mathrm{MSD}^{(p_1)} \geq \mathrm{MSD}^{(p_2)} \quad \text{for } p_1 > p_2 \geq 0. \tag{67}$$

This means that the steady-state MSD increases as p increases. This theoretical result is well matched to the practical result of the proposed algorithm (Fig. 2).

B. NOISY COMMUNICATION CONDITION

In this subsection, we consider the noise that is presented in the communication links between two nodes. In several previous papers [76]–[80], the effects of communication noise were thoroughly explored mathematically. Because the proposed algorithm does not exchange the measurements $\{\boldsymbol{u}_{k,i}$ and $\boldsymbol{d}_k(i)\}$, the communication noise is added only at the exchange of $\psi_{k,i}$ during the combination step. Following the notation in [80], we denote the communication noise added when node l transmits its intermediate estimate $\psi_{l,i}$ to node k as $v^{(\psi)}_{lk,i}$, and its covariance matrix as $R^{(\psi)}_{v,lk}$. We assume that the communication noise $v^{(\psi)}_{lk,i}$ is white Gaussian such that $R^{(\psi)}_{v,lk} = \sigma^2_{\psi,lk}I_M$. The aggregated noise in node k and its variance are then expressed as

$$v^{(\psi)}_{k,i} = \sum_{l\in\mathcal{N}_k\setminus\{k\}} a_{l,k}v^{(\psi)}_{lk,i}, \quad \sigma^2_{\psi,k} = \sum_{l\in\mathcal{N}_k\setminus\{k\}} a^2_{l,k}\sigma^2_{\psi,lk}, \tag{68}$$

and we define

$$R^{(\psi)}_v = \mathrm{diag}\{\sigma^2_{\psi,1}, \sigma^2_{\psi,2}, \cdots, \sigma^2_{\psi,N}\}. \tag{69}$$

We also introduce the block vector and its covariance matrix as

$$v^{(\psi)}_i \triangleq \mathrm{col}\left\{v^{(\psi)}_{1,i}, \ldots, v^{(\psi)}_{N,i}\right\}, \quad \mathcal{R}^{(\psi)}_v = R^{(\psi)}_v \otimes I_M. \tag{70}$$

The network steady-state MSD over noisy communication condition can then be expressed as [80]

$$\frac{1}{N}\mathrm{E}\,\|\tilde{\boldsymbol{w}}_\infty\|^2 = \frac{1}{N}\sum_{j=0}^{\infty}\left\{\mathrm{Tr}\left[\mathcal{B}^j_p\left(\sum_{m=0}^{p-1}\mathcal{Y}_m\right)\mathcal{B}^{*j}_p\right]\right\} + \frac{1}{N}\sum_{j=0}^{\infty}\mathrm{Tr}\left[\mathcal{B}^j_p\mathcal{R}^{(\psi)}_v\mathcal{B}^{*j}_p\right]. \tag{71}$$

The first term is the same as that in (54), which is independent of the communication noise; this means that only the second term is related to the noise. We next analyze the effect of the period p over noisy communication by comparing the values of the second term at period values $p_1 > p_2 \geq 1$. We define the second term of (71) with period p as

$$\mathcal{S}^{(p)} = \frac{1}{N}\sum_{j=0}^{\infty}\mathrm{Tr}\left[\mathcal{B}^j_p\mathcal{R}^{(\psi)}_v\mathcal{B}^{*j}_p\right]. \tag{72}$$

Then, applying (58) yields

$$\begin{aligned}\mathcal{S}^{(p_1)} - \mathcal{S}^{(p_2)} &= \frac{1}{N}\sum_{j=0}^{\infty}\mathrm{Tr}\Big[A^{jT}R^{(\psi)}_vA^j \otimes \big\{(I_M-\mu R_u)^{2jp_1} - (I_M-\mu R_u)^{2jp_2}\big\}\Big]\\ &= \frac{1}{N}\sum_{j=0}^{\infty}\mathrm{Tr}\left[A^{jT}R^{(\psi)}_vA^j\right]\mathrm{Tr}\Big[(I_M-\mu R_u)^{2jp_1} - (I_M-\mu R_u)^{2jp_2}\Big].\end{aligned} \tag{73}$$

Because $R^{(\psi)}_v$ is positive semidefinite, the first trace value of (73) is positive from Lemma 1. If we denote the eigenvalues of $(I_M - \mu R_u)^2$ by λ_n for $n = 1, \ldots, M$, then $\lambda_n \geq 0$

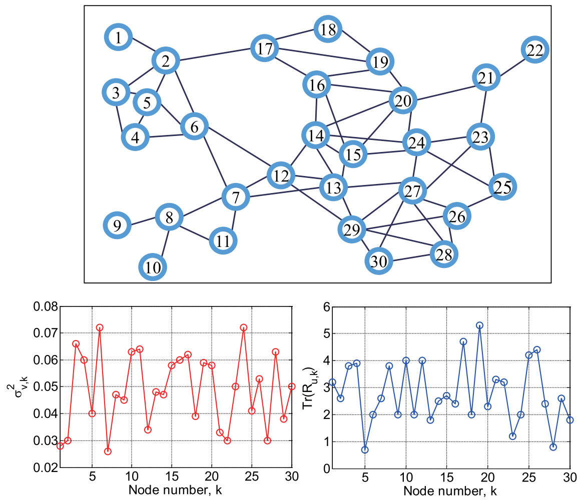

FIGURE 1. Network topology (top), noise variance $\sigma^2_{v,k}$ (left, bottom) and trace of regressor covariance $\mathrm{Tr}(R_{u,k})$ (right, bottom) for $N = 30$ nodes.

for all n because the matrix is positive semidefinite. Under the stability condition (51), we conclude that $0 \le \lambda_n < 1$ for all n. Then the eigenvalues of $(I_M - \mu R_u)^{2jp_1} - (I_M - \mu R_u)^{2jp_2}$ are $-1 < \lambda_n^{jp_1} - \lambda_n^{jp_2} \le 0$ for all n, because $\lambda_n^{jp_1} \le \lambda_n^{jp_2}$. Thus, the second trace value of (73) is non-positive and we conclude that

$$\mathcal{S}^{(p_1)} \le \mathcal{S}^{(p_2)}. \tag{74}$$

In (67), we proved that the proposed algorithm has a higher steady-state network MSD level than the conventional diffusion LMS in ideal channel environments; however, in the presence of communication noise the second term in (71), which is related to the communication noise, decreases when the period p increases, indicating that the proposed algorithm is less degraded than the diffusion LMS over the communication noise. Because the proposed algorithm exchanges the estimates more sparsely, the communication noise is less contaminated on average. Furthermore, even though the proposed algorithm has a higher steady-state network MSD than the conventional diffusion LMS under ideal channel condition, the gap between the two decreases in a noisy communication environment, and the proposed algorithm can have a lower steady-state network MSD than the diffusion LMS, even in environments in which communication noise variance is very high (Fig. 7).

VII. SIMULATION RESULTS

In our simulations, we assumed a channel identification scenario involving a finite impulse response (FIR) model with a channel length of $M = 8$. We used a network topology with $N = 30$ nodes with different regression and noise variances (Fig. 1). The regressions were zero-mean Gaussian, and independent over space. All simulations were obtained by

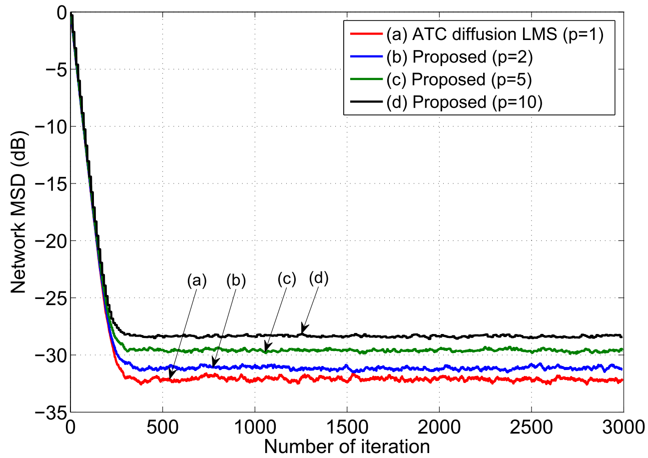

FIGURE 2. Transient network MSD curves for the conventional ATC diffusion LMS and the proposed algorithms with three different periods $p = 2, 5, 10$.

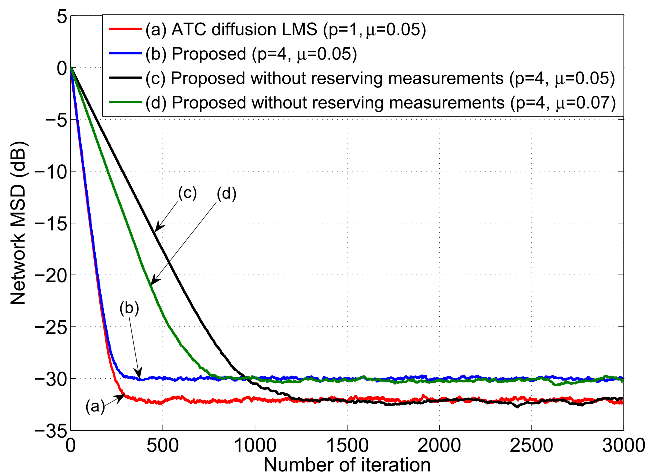

FIGURE 3. Transient network MSD curves for the proposed algorithm and the simple periodic diffusion scheme without measurement reserving.

taking the ensemble average of the network MSD:

$$\mathrm{MSD}^{\mathrm{Network}}(i) = \frac{1}{N}\sum_{k=1}^{N} \mathrm{E}\left\|w^o - \mathbf{w}_{k,i}\right\|^2 \tag{75}$$

over 200 trials. The step-size μ_k was set to 0.05 and the metropolis rule [29] was used to obtain the combination weight $a_{l,k}$.

From the results, we obtained network MSD curves (Fig. 2) for the conventional diffusion LMS and the proposed algorithms with different periods $p = 2, 5, 10$. The proposed DR-PDLMS algorithm has a trade-off between steady-state error and the communication cost that changes with the period p. As p increases, the frequency of data transmission decreases, resulting in a scaling-down of the total cost of communication to $1/p$, but an increased network MSD. For example, when $p = 5$, the required cost of communication reduces to only 20% of the full diffusion LMS; at the same

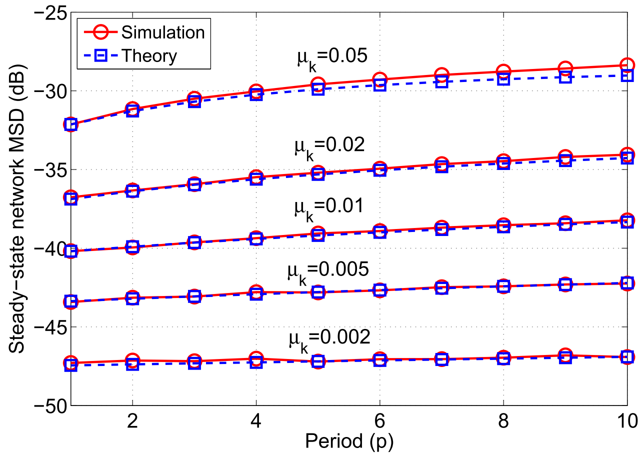

FIGURE 4. Theoretical and experimental steady-state network MSDs plotted against period p for different step-sizes in the proposed DR-PDLMS algorithm.

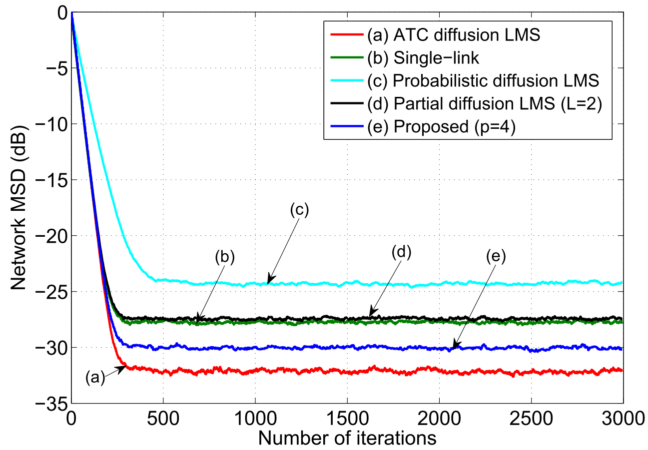

FIGURE 5. Transient network MSD curves for the proposed algorithms with $p = 2, 3, 4$ in an environment in which the unknown parameter w^o is abruptly changed.

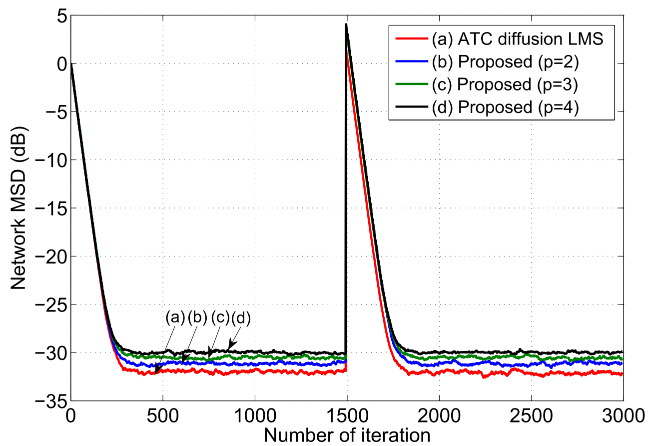

FIGURE 6. Transient network MSD curves for the conventional diffusion LMS and the proposed algorithm with $p = 4$ in noisy communication environments with three variance levels $\sigma_\psi^2 = 0, 0.0003, 0.0009$.

time, the reduction of diffusion of the information over the network increases the steady-state network MSD.

We then compared the proposed algorithm with a simple scheme that only performs the adaptation and combination steps periodically without reserving measurements (Fig. 3), i.e., it performs the adaptation step with measurements only at a periodic time instant. In the simulation, $p = 4$ was used for all algorithms. It has been seen from Fig. 3(c) that the simplified algorithm has a steady-state network MSD that did not increase and a convergence speed that was much slower than that of the ATC diffusion LMS (p=1). When the step-size was adjusted to $\mu = 0.07$ to match the steady-state network MSD with that of the proposed algorithm (Fig. 3 (b)), the convergence speed was still slower. These results confirm the superior convergence performance of the proposed algorithm resulting from its utilization of more information from the measurements than the simple periodic algorithm. The results can be elaborated by the effects of adaptation and combination steps on performance. The adaptation step increases the convergence speed in the transient stage, but in steady-state, the adaptation step makes noise while the combination step removes it, and they reach an equilibrium. Our proposed algorithm maintains all the adaptation steps and converges at the same speed as the conventional ATC LMS algorithm in the transient stage, but p-1 combination steps are skipped so that the steady-state network MSD is higher than the conventional algorithm. Fig. 3(c) can be interpreted as a lower version of the conventional algorithm.

The steady-state network MSD was obtained as a function of p for different values of node-consistent step-sizes (Fig. 4). The simulated steady-state values were obtained by averaging over 300 samples in the steady state, while the theoretical values were calculated using (44) and (46). A comparison of the results indicates that the simulated and theoretical values coincide closely over the ranges of step sizes and periods.

We then compared the proposed algorithm to probabilistic diffusion LMS [54], single-link diffusion LMS [57], and partial diffusion LMS algorithms [53] (Fig. 5) by setting all the parameters of the respective algorithms to equalize the communication costs. This was done by setting the link probability in [54] to 0.25 and the forgetting factor α in [57] to 0.95. For the single-link algorithm [57], single-node selection and true noise variance values were applied. For the partial diffusion LMS [53], a stochastic scheme was used to select the two coefficients to be communicated at each iteration. To enable fair comparison, we used $p = 4$ in the proposed algorithm to match the total cost of communication in the other algorithms. The proposed algorithm was found to outperform the other algorithms, as it had a faster convergence speed and a lower steady-state error.

We then simulated the convergence behavior of the proposed algorithm when the unknown parameter w^o was abruptly changed in the middle of the maximum number of iterations (Fig. 6). It was shown that the proposed algorithm skillfully tracked the sudden weight changes without experiencing degradation either the convergence speed or steady-state error.

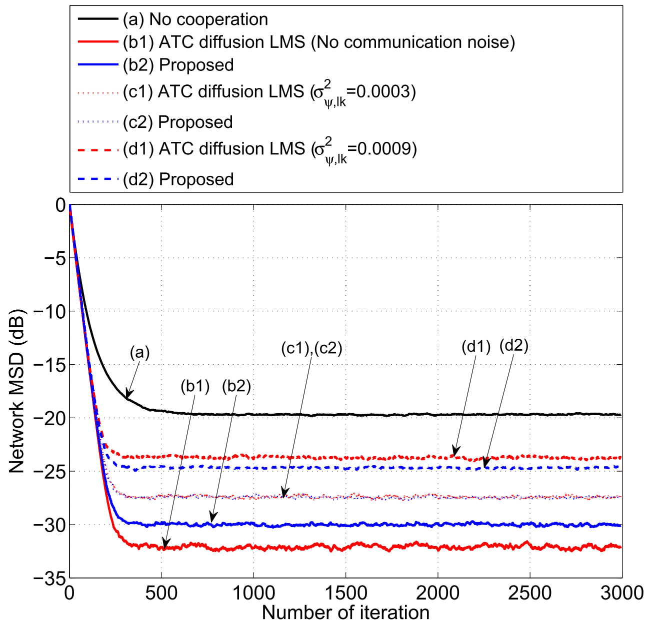

FIGURE 7. Transient network MSD curves for the conventional diffusion LMS algorithm and the proposed algorithm with $p = 4$ in the noisy communication environments at three noise variance levels $\sigma_{\psi}^2 = 0, 0.0003, 0.0009$.

We next simulated the convergence behavior of the proposed, no cooperation, and ATC diffusion LMS algorithms in a noisy communication environment (Fig. 7) in which white Gaussian communication noise $R_{v,lk}^{(\psi)} = \sigma_{\psi,lk}^2 I_M$ was applied. We assumed a simple case in which the communication noise variances were the same in all links, i.e., $\sigma_{\psi,lk}^2 = \sigma_{\psi}^2$ and considered three variance values $\sigma_{\psi}^2 = 0, 0.0003, 0.0009$. The period of the proposed algorithm was set to 4. In the case of $\sigma_{\psi}^2 = 0$, the proposed algorithm had higher steady-state error than the ATC diffusion LMS. However, for $\sigma_{\psi}^2 = 0.0003$, the algorithms had the same steady-state errors, and the proposed algorithm had lower steady-state error than the ATC diffusion LMS for $\sigma_{\psi}^2 = 0.0009$. These results indicate that, as the period p increases, the results obtained using the proposed algorithm are less degraded by communication noise. Although the periodic method slightly degrade the convergence performance of ATC diffusion LMS over the perfect network, the proposed algorithm is remarkably less sensitive to communication noise (Section V-B, (74)).

Finally, we conducted a performance comparison between the proposed algorithm and the block diffusion LMS (BDLMS) algorithm in [70] and [71] (Fig. 8). The two algorithms have the same combination step, but the adaptation steps are remarkably different; the adaptation steps of the proposed algorithm are presented in (17), while those of the BDLMS algorithm are given as follows:

$$\psi_{k,i} = w_{k,i-1} + \frac{\mu}{p} \sum_{m=0}^{p-1} u_{k,i-m}(d_k(i-m) - u_{k,i-m} w_{k,i-1}). \tag{76}$$

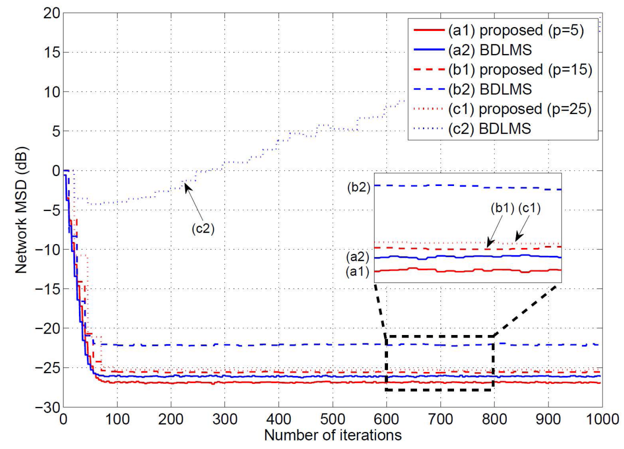

FIGURE 8. Transient network MSD curves for the proposed and the block diffusion LMS (BDLMS) algorithms at $p = 5, 15, 25$.

In the BDLMS, the step-size parameter μ is divided by the period p to average gradient value. For a fair comparison, we set $\mu = p \cdot \mu_{\text{DR-PDLMS}}$ to give both algorithms the same step-size value for the adaptation step. We note that, here, "iteration" is defined as the measurement sensing time (not the updating time after taking p measurements as in [70]). The simulation was performed for three values of period $p = 5, 15, 25$. In all cases, the proposed algorithm had a lower steady-state MSD than the BDLMS algorithm while the convergence speeds of the two algorithms are almost the same. Furthermore, at comparably large p ($p = 25$), the BDLMS algorithm unexpectedly failed to converge, while the proposed algorithm converged well.

VIII. DISCUSSION AND CONCLUSIONS

In this paper, we proposed a data-reserved periodic diffusion LMS (DR-PDLMS) algorithm in which estimates are periodically exchanged among neighbor nodes. By reserving all measurements sensed during non-update times and then utilizing them at update times, the proposed algorithm can reduce the communication cost effectively without significant degradation in either the convergence rate or steady-state error. Through intensive mathematical analysis, we analyzed the stability and mean-square behavior of the proposed algorithm, and showed theoretically that the DR-PDLMS with a larger period have a larger steady-state MSD. Our simulated results were in close agreement with these theoretical results. We proved that the proposed algorithm is less sensitive to communication noise and outperforms the original diffusion LMS algorithms in highly noisy communication environments, and also demonstrated the superiority of DR-PDLMS to related algorithms in a channel identification scenario.

In most real applications, the target parameters drifts over time, and tracking ability is very important. The internal latency caused by no-processing forces the proposed algorithm to slowly respond to the change of objective and the reaction can be maximally delayed for p iterations. We have shown that the proposed algorithm can track a sudden change of the target parameters (Fig. 6), but if the target system is constantly changing more rapidly than the period p, the proposed estimate may inaccurate. In this respect, the decimated diffusion would work better in the transient phase, but it also has fluctuations in the steady-state phase. As an alternative, unifying the proposed method with the decimated diffusion would be optimal for both transient and steady-state phases. This extension will be the topic of a future investigation.

APPENDIX
PROOF OF LEMMA 1

Assume that X is a $M \times M$ real symmetric semi-positive definite matrix. Using eigenvalue decomposition, X can be decomposed as

$$X = Q\Lambda Q^T \tag{77}$$

where $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \cdots, \lambda_M\}$ contains the eigenvalues $\lambda_k \geq 0$. Then we have

$$Y = PXP^T = PQ\Lambda Q^T P^T = R\Lambda R^T \tag{78}$$

where $R \triangleq PQ$. For an arbitrary $M \times 1$ vector a,

$$a^T Ya = a^T R\Lambda R^T a. \tag{79}$$

If we define $b = R^T a$,

$$a^T R\Lambda R^T a = b^T \Lambda b = \sum_{k=1}^{M} \lambda_k b^2 \geq 0 \tag{80}$$

where the last inequality is valid because $\lambda_k > 0$ for all k. Therefore, Y is also non-negative.

REFERENCES

[1] A. H. Sayed, "Adaptive networks," *Proc. IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.
[2] L. Xiao, S. Boyd, and S. Lall, "A space-time diffusion scheme for peer-to-peer least-squares estimation," in *Proc. ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Nashville, TN, USA, Apr. 2006, pp. 168–176.
[3] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3122–3136, Jul. 2008.
[4] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.
[5] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.
[6] F. S. Cattivelli and A. H. Sayed, "Distributed detection over adaptive networks using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 1917–1932, May 2011.
[7] D. Bajovic, D. Jakovetic, J. M. F. Moura, J. Xavier, and B. Sinopoli, "Large deviations performance of consensus+innovations distributed detection with non-Gaussian observations," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 5987–6002, Nov. 2012.
[8] S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections: A unifying framework for linear and nonlinear classification and regression tasks," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, Jan. 2011.
[9] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4692–4707, Oct. 2011.
[10] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.
[11] J. Chen, Z. J. Towfic, and A. H. Sayed, "Dictionary learning over distributed models," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 1001–1016, Feb. 2015.
[12] Z. J. Towfic, J. Chen, and A. H. Sayed, "On distributed online classification in the midst of concept drifts," *Neurocomputing*, vol. 112, pp. 139–152, Jul. 2013.
[13] X. Zhao, J. Chen, and A. H. Sayed, "Beam coordination via diffusion adaptation over array networks," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Cesme, Turkey, Jun. 2012, pp. 105–109.
[14] P. Di Lorenzo, S. Barbarossa, and A. H. Sayed, "Bio-inspired decentralized radio access based on swarming mechanisms over adaptive networks," *IEEE Trans. Signal Process.*, vol. 61, no. 12, pp. 3183–3197, Jun. 2013.
[15] R. Arroyo-Valles, S. Maleki, and G. Leus, "Distributed wideband spectrum sensing for cognitive radio networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 7263–7267.
[16] J. Chen, X. Zhao, and A. H. Sayed, "Bacterial motility via diffusion adaptation," in *Proc. 44th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2010, pp. 1930–1934.
[17] F. S. Cattivelli and A. H. Sayed, "Modeling bird flight formations using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2038–2051, May 2011.
[18] S.-Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 649–664, Aug. 2011.
[19] D. P. Bertsekas, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optim.*, vol. 7, no. 4, pp. 913–926, 1997.
[20] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM J. Optim.*, vol. 12, no. 1, pp. 109–138, 2001.
[21] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 798–808, Apr. 2005.
[22] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.
[23] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient method with a constant step size," *SIAM J. Optim.*, vol. 18, no. 1, pp. 29–51, Feb. 2007.
[24] L. Li, J. A. Chambers, C. G. Lopes, and A. H. Sayed, "Distributed estimation over an adaptive incremental network based on the affine projection algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 151–164, Jan. 2010.
[25] Y. Liu and W. K. S. Tang, "Enhanced incremental LMS with norm constraints for distributed in-network estimation," *Signal Process.*, vol. 94, pp. 373–385, Jan. 2014.
[26] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Library in Signal Processing*, vol. 3, R. Chellapa and S. Theodoridis, Eds. Oxford, U.K.: Academic, 2014, pp. 323–454.
[27] J. N. Tsitsiklis and M. Athans, "Convergence and asymptotic agreement in distributed decision problems," *IEEE Trans. Autom. Control*, vol. AC-29, no. 1, pp. 42–50, Jan. 1984.
[28] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. AC-31, no. 9, pp. 803–812, Sep. 1986.
[29] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, 2004.
[30] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.
[31] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links—Part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.
[32] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links—Part II: Distributed estimation and smoothing of random signals," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1650–1666, Apr. 2008.
[33] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.
[34] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 355–369, Jan. 2009.

[35] A. Bertrand and M. Moonen, "Consensus-based distributed total least squares estimation in ad hoc wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2320–2330, May 2011.
[36] O. Hlinka, O. Sluciak, F. Hlawatsch, P. M. Djuric, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4334–4349, Aug. 2012.
[37] A. Nedić and A. Ozdaglar, "Cooperative distributed multi-agent optimization," in *Convex Optimization in Signal Processing and Communications*, Y. Eldar and D. Palomar, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2009, pp. 340–386.
[38] S. Kar and J. M. F. Moura, "Convergence rate analysis of distributed gossip (linear parameter) estimation: Fundamental limits and tradeoffs," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 5, pp. 674–690, Aug. 2011.
[39] S.-Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, Dec. 2012.
[40] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, Sep. 2010.
[41] J. W. Lee, S. E. Kim, W. J. Song, and A. H. Sayed, "Spatio-temporal diffusion strategies for estimation and detection over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4017–4034, Aug. 2012.
[42] Y. Liu, C. Li, and Z. Zhang, "Diffusion sparse least-mean squares over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4480–4485, Aug. 2012.
[43] J. Fernandez-Bes, L. A. Azpicueta-Ruiz, J. Arenas-García, and M. T. M. Silva, "Distributed estimation in diffusion networks using affine least-squares combiners," *Digit. Signal Process.*, vol. 36, pp. 1–14, Jan. 2015.
[44] H. S. Lee, S. E. Kim, J. W. Lee, and W. J. Song, "A variable step-size diffusion LMS algorithm for distributed estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 7, pp. 1808–1820, Apr. 2015.
[45] J.-T. Kong, J.-W. Lee, S.-E. Kim, and W.-J. Song, "Diffusion LMS algorithms with multi combination for distributed estimation: Formulation and performance analysis," *Digit. Signal Process.*, vol. 71, pp. 117–130, Dec. 2017.
[46] S. Al-Sayed, A. M. Zoubir, and A. H. Sayed, "Robust adaptation in impulsive noise," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2851–2865, Jun. 2016.
[47] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks with common latent representations," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 3, pp. 563–579, Apr. 2017.
[48] S. Khawatmi, A. H. Sayed, and A. M. Zoubir, "Decentralized clustering and linking by networked agents," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3526–3537, Jul. 2017.
[49] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Diffusion LMS for multitask problems with local linear equality constraints," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 4979–4993, Oct. 2017.
[50] M. J. Piggott and V. Solo, "Diffusion LMS with correlated regressors I: Realization-wise stability," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5473–5484, Nov. 2016.
[51] M. J. Piggott and V. Solo, "Diffusion LMS with correlated regressors II: Performance," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 3934–3947, Aug. 2017.
[52] W. Wang and H. Zhao, "Diffusion signed LMS algorithms and their performance analyses for cyclostationary white Gaussian inputs," *IEEE Access*, vol. 5, pp. 18876–18894, 2017.
[53] R. Arablouei, S. Werner, Y.-F. Huang, and K. Doğançay, "Distributed least mean-square estimation with partial diffusion," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 472–484, Jan. 2014.
[54] C. G. Lopes and A. H. Sayed, "Diffusion adaptive networks with changing topologies," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Las Vegas, NV, USA, Apr. 2008, pp. 3285–3288.
[55] N. Takahashi and I. Yamada, "Link probability control for probabilistic diffusion least-mean squares over resource-constrained networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Dallas, TX, USA, Mar. 2010, pp. 3518–3521.
[56] Q. L. Rortveit, J. H. Husoy, and A. H. Sayed, "Diffusion LMS with communication constraints," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Nov. 2010, pp. 1645–1649.
[57] X. Zhao and A. H. Sayed, "Single-link diffusion strategies over adaptive networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Kyoto, Japan, Mar. 2012, pp. 3749–3752.
[58] R. Arablouei, S. Werner, K. Doğançay, and Y.-F. Huang, "Analysis of a reduced-communication diffusion LMS algorithm," *Signal Process.*, vol. 117, pp. 355–361, Dec. 2015.
[59] S. Werner, T. Riihonen, and Y.-F. Huang, "Energy-efficient distributed parameter estimation with partial updates," in *Proc. Int. Conf. Green Circuits Syst.*, Shanghai, China, Jun. 2010, pp. 36–40.
[60] R. Arablouei, K. Dogancay, S. Werner, and Y.-F. Huang, "Adaptive distributed estimation based on recursive least-squares and partial diffusion," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3510–3522, Jul. 2014.
[61] V. Vahidpour, A. Rastegarnia, A. Khalili, W. M. Bazzi, and S. Sanei, "Analysis of partial diffusion LMS for adaptive estimation over networks with noisy links," *IEEE Trans. Netw. Sci. Eng.*, vol. 5, no. 2, pp. 101–112, Apr./Jun. 2018.
[62] S. Werner and Y.-F. Huang, "Time-and coefficient-selective diffusion strategies for distributed parameter estimation," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Nov. 2010, pp. 2205–2209.
[63] S. Werner, M. Mohammed, Y.-F. Huang, and V. Koivunen, "Decentralized set-membership adaptive estimation for clustered sensor networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Las Vegas, NV, USA, Apr. 2008, pp. 3573–3576.
[64] A. Malipatil, Y.-F. Huang, and S. Werner, "An SMF approach to distributed average consensus in clustered sensor networks," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Perugia, Italy, Jun. 2009, pp. 81–85.
[65] M. O. Sayin and S. S. Kozat, "Single bit and reduced dimension diffusion strategies over distributed networks," *IEEE Signal Process. Lett.*, vol. 20, no. 10, pp. 976–979, Oct. 2013.
[66] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Trading off complexity with communication costs in distributed adaptive learning via Krylov subspaces for dimensionality reduction," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 257–273, Apr. 2013.
[67] M. O. Sayin and S. S. Kozat, "Compressive diffusion strategies over distributed networks for reduced communication load," *IEEE Trans. Signal Process.*, vol. 62, no. 20, pp. 5308–5323, Oct. 2014.
[68] J.-W. Lee, S.-E. Kim, and W.-J. Song, "Data-selective diffusion LMS for reducing communication overhead," *Signal Process.*, vol. 113, pp. 211–217, Aug. 2015.
[69] O. N. Gharehshiran, V. Krishnamurthy, and G. Yin, "Distributed energy-aware diffusion least mean squares: Game-theoretic learning," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 5, pp. 821–836, Oct. 2014.
[70] T. Panigrahi, P. M. Pradhan, G. Panda, and B. Mulgrew, "Block least mean squares algorithm over distributed wireless sensor network," *J. Comput. Netw. Commun.*, vol. 2012, 2012, Art. no. 601287, doi: 10.1155/2012/601287.
[71] S. Kumar, A. K. Sahoo, and D. P. Achary, "Block diffusion adaptation over distributed adaptive networks under imperfect data transmission," in *Proc. Annu. IEEE India Conf. (INDICON)*, Pune, India, Dec. 2014, pp. 1–5.
[72] X. Zhao and A. H. Sayed, "Asynchronous adaptation and learning over networks—Part I: Modeling and stability analysis," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 811–826, Feb. 2015.
[73] X. Zhao and A. H. Sayed, "Asynchronous adaptation and learning over networks—Part II: Performance analysis," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 827–842, Feb. 2015.
[74] X. Zhao and A. H. Sayed, "Asynchronous adaptation and learning over networks—Part III: Comparison analysis," *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 842–858, Feb. 2015.
[75] A. H. Sayed, *Adaptive Filters*. New York, NY, USA: Wiley, 2008.
[76] R. Abdolee and B. Champagne, "Diffusion LMS algorithms for sensor networks over non-ideal inter-sensor wireless channels," in *Proc. IEEE Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, Barcelona, Spain, Jun. 2011, pp. 1–6.
[77] A. Khalili, M. A. Tinati, A. Rastegarnia, and J. A. Chambers, "Transient analysis of diffusion least-mean squares adaptive networks with noisy channels," *Int. J. Adapt. Control Signal Process.*, vol. 26, no. 2, pp. 171–180, Feb. 2012.
[78] S.-Y. Tu and A. H. Sayed, "Adaptive networks with noisy links," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Houston, TX, USA, Dec. 2011, pp. 1–5.

[79] A. Khalili, M. A. Tinati, A. Rastegarnia, and J. A. Chambers, "Steady-state analysis of diffusion LMS adaptive networks with noisy links," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 974–979, Feb. 2012.
[80] X. Zhao, S.-Y. Tu, and A. H. Sayed, "Diffusion adaptation over networks under imperfect information exchange and non-stationary data," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3460–3475, Jul. 2012.

JAE-WOO LEE was born in Iksan, South Korea, in 1987. He received the B.S. and Ph.D. degrees in electronic and electrical engineering from the Pohang University of Science Technology, South Korea, in 2009 and 2016, respectively. From 2010 to 2011, he was a Visiting Researcher at the Adaptive Systems Laboratory, University of California at Los Angeles. Since 2016, he has been a Research Staff Member at the Samsung Advanced Institute of Technology, Suwon, South Korea. His research interests are signal processing, adaptive filtering, adaptive networks, machine learning, and autonomous driving.

JUN-TAEK KONG was born in Pyeongtaek, South Korea, in 1989. He received the B.S. degree in electronic and electrical engineering from the Pohang University of Science Technology (POSTECH), South Korea, in 2012, where he is currently pursuing the Ph.D. degree. Since 2012, he has been a Research Assistant at the Department of Electronic and Electrical Engineering, POSTECH. His research interests are signal processing, adaptive filtering, and adaptive networks.

WOO-JIN SONG (M'86) was born in Seoul, South Korea, in 1956. He received the B.S. and M.S. degrees in electronics engineering from Seoul National University, Seoul, in 1979 and 1981, respectively, and the Ph.D. degree in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, USA in 1986. From 1981 to 1982, he was with the Electronics and Telecommunication Research Institute, Daejeon, South Korea. In 1986, he joined Polaroid Corporation as a Senior Engineer, where he was involved in digital image processing. In 1989, he was promoted to Principal Engineer at Polaroid. In 1989, he joined the faculty at the Pohang University of Science and Technology, Pohang, South Korea, where he is currently a Professor of electronic and electrical engineering. His current research interests are in the area of digital signal processing, in particular, radar signal processing, signal processing for digital television and multimedia products, and adaptive signal processing.

SEONG-EUN KIM (S'09–M'10) received the B.Sc. and Ph.D. degrees in electronics and electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea in 2004 and 2010, respectively. From 2011 to 2014, he has been a Research Staff Member at the Samsung Advanced Institute of Technology, Yongin, South Korea. From 2014 to 2017, he has been a Post-Doctoral Associate with the MIT/Harvard Neuroscience Statistics Research Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA, where he is involved in neural signal processing algorithms. In 2017, he moved to Hanbat National University, Daejeon, South Korea, where he is currently an Assistant Professor at the Department of Electronics and Control Engineering. His research interests include statistical and adaptive signal processing, biomedical signal processing, and systems and computational neuroscience.