GROUP DIFFUSION LMS

Jie Chen ⋆ *Shang Kee Ting* ♮ *Cédric Richard* † *Ali H. Sayed* ‡

⋆ Center of Intelligent Acoustics and Immersive Communication
School of Marine Science and Technology, Northwestern Polytechincal University, China
♮ DSO National Laboratoires, Singapore
† Laboratoire Lagrange, Université de Nice Sophia-Antipolis, CNRS, France
‡ Department of Electrical Engineering, University of California, Los Angeles, USA
dr.jie.chen@ieee.org tshangke@dso.org.sg cedric.richard@unice.fr sayed@ee.ucla.edu

ABSTRACT

Considering groups of variables, rather than variables individually, can be beneficial for estimation accuracy if structural relationships between variables exist (e.g., spatial, hierarchical or related to the physics of the problem). Group-sparsity inducing estimators are typical examples that benefit from such type of prior knowledge. Building on this principle, we show that the diffusion LMS algorithm for distributed inference over networks can be extended to deal with structured criteria built upon groups of variables, leading to a flexible framework that can encode various structures in the parameters to estimate. We also propose an unsupervised online strategy to differentially promote or inhibit collaborations between nodes depending on the group of variables at hand.

***Index Terms*—** Diffusion adaptation, distributed optimization, multitask learning, group-based estimation

1. INTRODUCTION

Diffusion adaptation algorithms enable multi-agent networks to continuously learn and track concept drifts in a distributed manner from streaming data. They are efficient to address learning problems over multi-agent networks as well as to model the behavior of biological networks [1–3]. They have also demonstrated better stability and performance [4, 5] than incremental [6–9] and consensus [10–12] strategies.

Diffusion strategies can be implemented in single-task or multitask scenarios. In recent years, distributed learning over multitask networks has attracted increasing attention. Instead of dealing with applications where a common parameter vector is estimated by the entire network, multi-task diffusion strategies consider situations where agents need to simultaneously infer multiple parameter vectors. The performance of diffusion LMS when it is run, either intentionally or unintentionally, in a multitask environment was analyzed in [13]. An unsupervised clustering strategy that allows each node to select the neighboring nodes with which it can collaborate was also introduced in the same article. Alternative strategies were introduced in [14, 15]. A multitask counterpart of the diffusion LMS algorithm was derived in [16]. Its behavior over asynchronous networks was studied in [17]. A multitask diffusion with TV regularization was analyzed in [18]. In [19], the parameter space was decomposed into two orthogonal subspaces with one of them being common to all nodes. Other scenarios are described in [20, 21], where incremental and diffusion strategies are used to solve estimation problems with nodes that simultaneously estimate local and global parameters.

An inspection of the literature on diffusion adaptation over networks shows that, in most existing works, single-task and multitask oriented algorithms fuse information from neighboring agents via weighted combinations of estimated parameter vectors. These combinations assign the same scaling weight to all entries in the combined iterates. There are situations where different groups within the iterate vectors should be weighted differently than other groups within the same iterates. Consider an example where the top half of the entries of the parameter vectors to estimate are common across all agents, while the bottom half entries are randomly distributed without obvious relationship. Uniformly combining estimates as performed by the single-task diffusion LMS [1] may cause large estimation error due to the presence of the significantly different entries. In the same way, the multitask diffusion LMS in [13] may reduce the estimation bias by degenerating a cooperative algorithm into a non-cooperative one at each node, but may not benefit from similarities that exist among the top half of the entries.

Considering groups of variables, rather than variables individually, can be beneficial for estimation accuracy if structural relationships between variables exist (e.g., spatial, hierarchical or related to the physics of the problem). Group-sparsity inducing estimators are typical examples that benefit from such type of prior knowledge. Building on this principle, we show that the diffusion LMS algorithm can be extended to deal with structured criteria built upon groups of variables, leading to a flexible framework that can encode various structures in the parameters to estimate. We also propose an unsupervised strategy to differentially promote or inhibit collaborations between nodes in an online manner, depending on the group at hand.

Notation. Small letters x denote scalars, and boldface small letters $\boldsymbol{x}$ denote column vectors. Boldface capital letters $\boldsymbol{R}$ represent matrices, and the operator $(\cdot)^\top$ denotes matrix transposition. $\boldsymbol{I}_N$ denotes the $N \times N$ identity matrix. $\mathcal{N}_k$ denotes the neighbors of node k, including k. Finally, $\otimes$ denotes the Kronecker product.

2. NETWORK MODEL AND DIFFUSION LMS

2.1. Network model

Consider a connected network consisting of N nodes. In this paper, we address the problem of estimating an $L \times 1$ unknown vector

The work of C. Richard was partly supported by ANR-DGA grant ANR-13-ASTR-0030 (ODISSEE project). The work of A. H. Sayed was supported in part by NSF grants CIF-1524250 and ECCS-1407712.

at each node k from collected data. Node k has access to time sequences $\{d_k(n), \boldsymbol{x}_{k,n}\}$, where $d_k(n)$ denotes the reference signal, and $\boldsymbol{x}_{k,n}$ represents an $L \times 1$ regression vector with covariance matrix $\boldsymbol{R}_{x,k} = \mathbb{E}\{\boldsymbol{x}_{k,n}\boldsymbol{x}_{k,n}^\top\} > 0$. We assume that the data are related via the linear model:

$$d_k(n) = \boldsymbol{x}_{k,n}^\top \boldsymbol{w}_k^\star + z_k(n) \tag{1}$$

for all k, with $\boldsymbol{w}_k^\star$ an unknown parameter vector, and $z_k(n)$ a zero-mean i.i.d. noise of variance $\sigma_{z,k}^2$ that is independent of every other signal. For determining the parameter vectors $\boldsymbol{w}_k$, we consider the mean-square error criterion at each node k defined as:

$$J_k(\boldsymbol{w}_k) = \mathbb{E}\big\{|d_k(n) - \boldsymbol{x}_{k,n}^\top \boldsymbol{w}_k|^2\big\}. \tag{2}$$

We refer to cases where all nodes estimate the same parameter vector, that is, $\boldsymbol{w}_k^\star = \boldsymbol{w}^\star$ for all k, as *single-task* problems. In comparison, we refer to cases where nodes may be estimating distinct parameter vectors as *multi-task* problem.

2.2. Diffusion LMS

Before introducing the diffusion strategy at the group level, we provide a brief review of standard diffusion LMS. The goal of this algorithm is to minimize the following global cost function:

$$J^{\text{glob}}(\boldsymbol{w}) = \sum_{k=1}^{N} J_k(\boldsymbol{w}) \tag{3}$$

We denote the minimizer by $\boldsymbol{w}^\star$. Minimizing (3) over $\boldsymbol{w}$ is equivalent to minimizing the following alternative cost [1–3]:

$$J^{\text{glob}'}(\boldsymbol{w}) = J_k(\boldsymbol{w}) + \sum_{\ell \neq k} \|\boldsymbol{w} - \boldsymbol{w}^\star\|^2_{\boldsymbol{R}_{x,\ell}} \tag{4}$$

In order to bypass the unknown second-order statistics $\boldsymbol{R}_{x,\ell}$, based on the Rayleigh-Ritz characterization of their eigenvalues, previous works approximated the weighted norm in (4) by a scaled unweighted norm [1–3], say as,

$$\|\boldsymbol{w} - \boldsymbol{w}^\star\|^2_{\boldsymbol{R}_{x,\ell}} \approx b_{\ell k}\, \|\boldsymbol{w} - \boldsymbol{w}^\star\|^2 \tag{5}$$

for some nonnegative coefficients $b_{\ell k}$. This leads to the following modified cost function at node k:

$$J^{\text{glob}''}(\boldsymbol{w}) = J_k(\boldsymbol{w}) + \sum_{\ell \neq k} b_{\ell k}\, \|\boldsymbol{w} - \boldsymbol{w}^\star\|^2 \tag{6}$$

Calculating the gradient vector of (6), restricting communication to immediate neighbors, and using approximation (5) along with the arguments from [2], we arrive at the adapt-then-combine (ATC) strategy[1]

$$\boxed{\begin{aligned} \boldsymbol{\psi}_{k,n} &= \boldsymbol{w}_{k,n-1} + \mu_k\, \boldsymbol{x}_{k,n}\,(d_k(n) - \boldsymbol{x}_{k,n}^\top \boldsymbol{w}_{k,n-1}) && \text{(7a)} \\ \boldsymbol{w}_{k,n} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{\psi}_{\ell,n} && \text{(7b)} \end{aligned}}$$

where $\mathcal{N}_k$ denotes the set of neighbors of agent k, and the $\{a_{\ell k}\}$ coefficients are given by:

$$a_{kk} = 1 - \mu_k \sum_{\ell \in \mathcal{N}_k \backslash \{k\}} b_{\ell k} \tag{8}$$

$$a_{\ell k} = \mu_k\, b_{\ell k}, \qquad \ell \in \mathcal{N}_k \backslash \{k\} \tag{9}$$

$$a_{\ell k} = 0, \qquad \ell \notin \mathcal{N}_k \tag{10}$$

[1]We focus on the ATC form in this paper. The combine-then-adapt (CTA) form can be derived in a similar way.

and μ_k is a positive step-size. The coefficients $\{a_{\ell k}\}$ are usually treated as free weighting parameters to be chosen by the designer, it is not necessary to worry about selecting the $\{b_{\ell k}\}$ and it is sufficient to select the $\{a_{\ell k}\}$ are nonnegative convex combination coefficients satisfying

$$a_{\ell k} \geq 0, \quad \sum_{\ell \in \mathcal{N}_k} a_{\ell k} = 1, \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \tag{11}$$

The selection of the $\{a_{\ell k}\}$ has a significant impact on the performance of the algorithm.

3. GROUP DIFFUSION LMS

It is explained in [2] how approximation (5) leads to the fusion (7b) of local estimates in the neighborhood of each node. All entries are combined with the same weight. Figure 1 illustrates one possible limitation of uniform combination of the entries and the interest in grouping them. Adjacent nodes k and ℓ are estimating parameter vectors $\boldsymbol{w}_k^\star$ and $\boldsymbol{w}_\ell^\star$ structured in three groups of entries: both vectors have the same entries in the first group, they significantly differ in the second group due to sensor failure for instance, and slightly differ in the third group due to sensor drift. This scenario cannot be considered as a single-task problem, or even a multitask one, with a single set of combination weights $a_{\ell k}$. A small combination weight may not sufficiently promote the closeness of entries in the first and third groups, whereas a large combination weight may lead to a large estimation bias caused by the second group.

This motivates us to introduce a grouping strategy into distributed learning over networks. Let $\{\mathcal{G}_m\}_{m=1}^M$ be a partition of the set of indexes $\mathcal{G} = \{1, \ldots, L\}$, namely,

$$\bigcup_{m=1}^{M} \mathcal{G}_m = \mathcal{G}, \quad \mathcal{G}_m \cap \mathcal{G}_{m'} = \varnothing \text{ if } m \neq m' \tag{12}$$

and let $\boldsymbol{w}_{\mathcal{G}_m}$ or $[\boldsymbol{w}]_{\mathcal{G}_m}$ denote a sub-vector of $\boldsymbol{w}$ indexed by $\mathcal{G}_m$. For the scenario in Fig. 1, the entries of the parameter vectors can be grouped into three groups as described by Fig. 1(b). We can then assign large combination weights to the first group, small or even null-valued ones to the second group, and medium ones to the third group. Such grouping strategy can end up exploiting the structure of the parameter vectors more fully. We shall introduce an unsupervised adaptive strategy to estimate the combination weights in the next section. Since information on group structures may not be available in practice, one possible strategy is to split parameter vectors into a number of groups of preset lengths and assign a combination coefficient to each group, as illustrated in Fig. 1(c). Note that the parameter vector entries within each group need not be necessarily contiguous.

3.1. Group diffusion LMS algorithm

We now derive the group diffusion LMS. Inspecting (5), we assign a scaling factor to each group of entries instead of using a single factor for scaling the unweighted norm:

$$\|\boldsymbol{w} - \boldsymbol{w}^\star\|^2_{\boldsymbol{R}_{x,\ell}} \approx \sum_{m=1}^{M} b_{\ell k,m} \|\boldsymbol{w}_{\mathcal{G}_m} - \boldsymbol{w}^\star_{\mathcal{G}_m}\|^2 \tag{13}$$

where $b_{\ell k,m}$ is the weight for group m. The global cost (6) is then relaxed as follows:

$$J^{\text{glob}''}(\boldsymbol{w}) = J_k(\boldsymbol{w}) + \sum_{\ell \neq k} \sum_{m=1}^{M} b_{\ell k,m} \|\boldsymbol{w}_{\mathcal{G}_m} - \boldsymbol{w}^\star_{\mathcal{G}_m}\|^2 \tag{14}$$

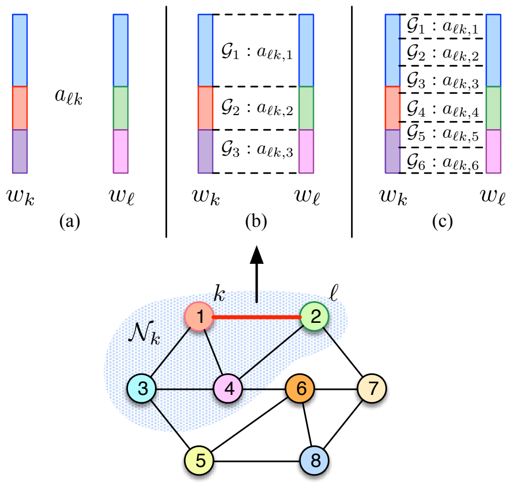


Fig. 1. Network model for group diffusion LMS. (a) Weight vector structures for nodes k and ℓ, where three sections coded by colors exhibit different levels of similarity; (b) optimal groups; (c) groups with uniform sizes.

Calculating the gradient vector of (14), following the same steps as for diffusion LMS, and introducing the following combination weights $a_{\ell k,m}$ for each group m

$$a_{kk,m} = 1 - \mu_k \sum_{\ell \in \mathcal{N}_k \backslash \{k\}} b_{\ell k,m} \tag{15}$$

$$a_{\ell k,m} = \mu_k \, b_{\ell k,m}, \qquad \ell \in \mathcal{N}_k \backslash \{k\} \tag{16}$$

$$a_{\ell k,m} = 0, \qquad \ell \notin \mathcal{N}_k, \tag{17}$$

we arrive at the group diffusion LMS algorithm:

$$\boldsymbol{\psi}_{k,n} = \boldsymbol{w}_{k,n-1} + \mu_k \, \boldsymbol{x}_{k,n} \, (d_k(n) - \boldsymbol{x}_{k,n}^\top \boldsymbol{w}_{k,n-1}) \tag{18a}$$

$$[\boldsymbol{w}_{k,n}]_{\mathcal{G}_m} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k,m} [\boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}, \quad \text{for } m = 1, \dots, M. \tag{18b}$$

Parameters $\bigcup_{m=1}^{M} \{a_{\ell k,m}\}$ can be adjusted by users, and each subset $\{a_{\ell k,m}\}$ forms a left-stochastic matrix $\boldsymbol{A}_m$ with $m = 1, \dots, M$. Appropriate selection of these coefficients can enhance the performance of diffusion LMS, especially for scenarios with structural relationships between optimums. In Section 4, we shall introduce an unsupervised strategy to adjust these weights in an online manner. A version of the group diffusion strategy (18a)–(18b) was introduced in [22] and applied to the tuning of A/D converters. The combination weights $\{a_{\ell k,m}\}$ were selected there in proportion to the SNR conditions within relevant frequency bands.

3.2. Network behavior with constant $a_{\ell k,m}$

We provide now a brief discussion on the behavior of the group diffusion LMS algorithm (18) with constant combination weights $a_{\ell k,m}$. Define the square matrix $\boldsymbol{\mathcal{A}}$ consisting of $N \times N$ diagonal matrices. The diagonal entries of the (ℓ, k)-th block $\boldsymbol{\mathcal{A}}^{(\ell k)}$ are given by $[\boldsymbol{\mathcal{A}}^{(\ell k)}]_{\mathcal{G}_m, \mathcal{G}_m} = a_{\ell k,m}$ for $m = 1, \dots, M$. Following the analysis in [13], it can be found that $\boldsymbol{\mathcal{A}}$ plays the same role as $\boldsymbol{\mathcal{A}}_2 = \boldsymbol{A}_2 \otimes \boldsymbol{I}$ in [13, Sec. III]. Since $\boldsymbol{\mathcal{A}}$ is also a left-stochastic matrix, the stability condition and performance measurement expressions remain unchanged.

4. ADAPTIVE COMBINATION STRATEGY

We shall now derive an adaptive combination strategy for group diffusion LMS. Motivated by [13, 23], we adjust $a_{\ell k,m}$ in an online manner via instantaneous MSD optimization. Let us denote by $\boldsymbol{v}_{k,n}$ the weight error vector $\boldsymbol{w}_k^\star - \boldsymbol{w}_{k,n}$ after the combination step (18b). Considering groups $\mathcal{G}_m$ with $m = 1, \dots, M$, the instantaneous mean-square deviation (MSD) at each agent k can be expressed as a function of $a_{\ell k,m}$ by

$$\begin{aligned}\mathbb{E}\{\|\boldsymbol{v}_{k,n}\|^2\} &= \sum_{m=1}^{M} \mathbb{E}\Big\{\big\|[\boldsymbol{w}_k^\star]_{\mathcal{G}_m} - \sum_{\ell \in \mathcal{N}_k} a_{\ell k,m} \, [\boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}\big\|^2\Big\} \\ &= \sum_{m=1}^{M} \sum_{\ell \in \mathcal{N}_k} \sum_{p \in \mathcal{N}_k} a_{\ell k,m} \, a_{pk,m} \, [\boldsymbol{\Psi}_{k,n}^{(m)}]_{\ell p}\end{aligned} \tag{19}$$

The matrix $\boldsymbol{\Psi}_{k,n}^{(m)}$ is the covariance matrix of the weight error for group m at node k at instant n, with (ℓ, p)-th entry given by:

$$[\boldsymbol{\Psi}_{k,n}^{(m)}]_{\ell p} = \begin{cases} \mathbb{E}\left\{[\boldsymbol{w}_k^\star - \boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}^\top [\boldsymbol{w}_k^\star - \boldsymbol{\psi}_{p,n}]_{\mathcal{G}_m}\right\}, & \ell, p \in \mathcal{N}_k \\ 0, & \text{otherwise.} \end{cases} \tag{20}$$

To make the problem tractable, we approximate $\boldsymbol{\Psi}_{k,n}^{(m)}$ by an instantaneous value and we drop its off-diagonal entries. In addition, we approximate $\boldsymbol{w}_k^\star$ by $\widehat{\boldsymbol{w}}_k^\star$ as shown in (23). This leads to:

$$\begin{aligned} \min_{\boldsymbol{a}_{k,m}} \quad & \sum_{\ell=1}^{N} \sum_{m=1}^{M} a_{\ell k,m}^2 \, \|[\widehat{\boldsymbol{w}}_k^\star - \boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}\|^2 \\ \text{subject to} \quad & \boldsymbol{1}_N^\top \boldsymbol{a}_{k,m} = 1, \quad a_{\ell k,m} \geq 0, \\ & a_{\ell k,m} = 0 \qquad \text{if } \ell \notin \mathcal{N}_k \end{aligned} \tag{21}$$

where the vector $\boldsymbol{a}_{k,m} = [a_{1k,m}, \dots, a_{Nk,m}]^\top$. The above objective function encourages weak information exchange via small $a_{\ell k,m}$ if the estimate of group m at node ℓ is far from its counterpart at node k. The solution of (21) is given by:

$$a_{\ell k,m} = \frac{\|[\widehat{\boldsymbol{w}}_k^\star - \boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}\|^{-2}}{\sum_{j \in \mathcal{N}_k} \|[\widehat{\boldsymbol{w}}_k^\star - \boldsymbol{\psi}_{j,n}]_{\mathcal{G}_m}\|^{-2}}, \quad \text{for } \ell \in \mathcal{N}_k. \tag{22}$$

We now introduce an instantaneous approximation $\widehat{\boldsymbol{w}}_{k,n}^\star$ for $\boldsymbol{w}_k^\star$ at each node k and time instant n. In order to reduce the MSD bias that may result from an inappropriate cooperation between nodes performing distinct estimation tasks, a possible strategy is to use the local one-step ahead approximation:

$$\widehat{\boldsymbol{w}}_{k,n}^\star = \boldsymbol{\psi}_{k,n} + \mu_k' \, \boldsymbol{q}_{k,n} \tag{23}$$

where $\boldsymbol{q}_{k,n} = (d_k(n) - \boldsymbol{x}_{k,n}^\top \boldsymbol{\psi}_{k,n}) \, \boldsymbol{x}_{k,n}$ is the instantaneous approximation of the negative gradient of $J_k(\boldsymbol{w})$ at $\boldsymbol{\psi}_{k,n}$. Substituting this expression into (22) leads to the combination rule:

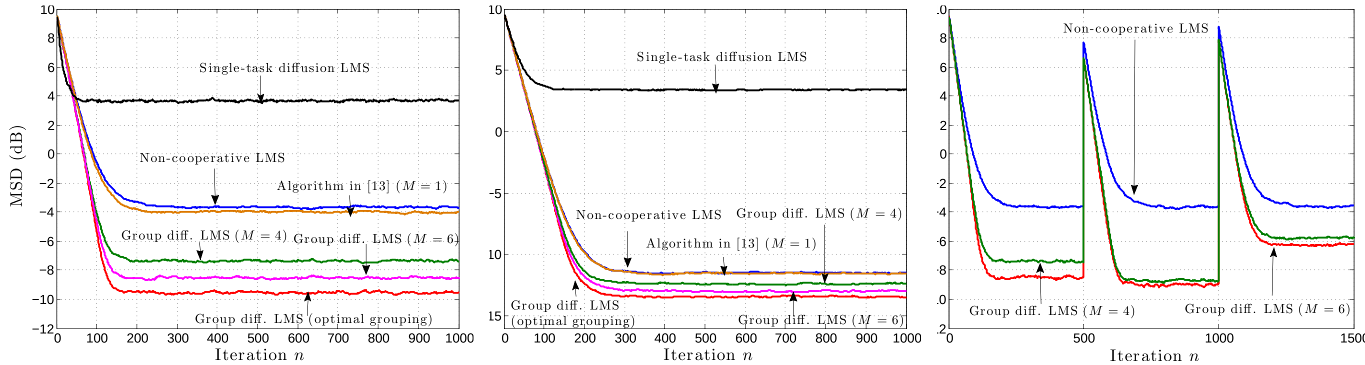
$$a_{\ell k,m}(n) = \frac{\|[\boldsymbol{\psi}_{k,n} + \mu_k' \, \boldsymbol{q}_{k,n} - \boldsymbol{\psi}_{\ell,n}]_{\mathcal{G}_m}\|^{-2}}{\sum_{j \in \mathcal{N}_k} \|[\boldsymbol{\psi}_{k,n} + \mu_k' \, \boldsymbol{q}_{k,n} - \boldsymbol{\psi}_{j,n}]_{\mathcal{G}_m}\|^{-2}} \tag{24}$$

for $\ell \in \mathcal{N}_k$ and $m = 1, \dots, M$. Furthermore, we observed in our experiments that the normalized gradient $\boldsymbol{q}_{k,n} \leftarrow \boldsymbol{q}_{k,n} / (\|\boldsymbol{q}_{k,n}\| + \epsilon)$ with ϵ a small positive constant can increase the robustness of the resulting strategy.

5. SIMULATIONS

This section shows how the proposed strategy behaves with two illustrative scenarios. All curves were obtained by averaging over 100 runs [2].

[2]Matlab code is available at `http://www.jie-chen.com/codes`



(a) Stationary case with $\mu = 0.05$. (b) Stationary case with $\mu = 0.02$. (c) Nonstationary case with $\mu = 0.05$.

Fig. 2. Network MSD learning curves

.

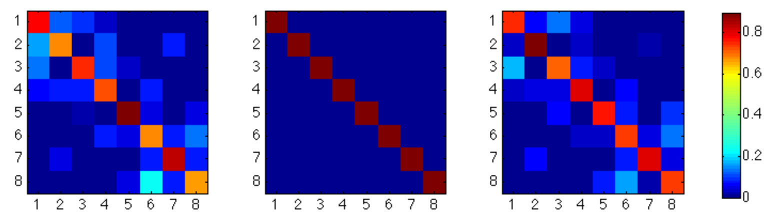


Fig. 3. Combination coefficient matrices $\{\boldsymbol{A}_m\}_{m=1}^3$ with optimal grouping. Left to right: group $\mathcal{G}_1$ to $\mathcal{G}_3$.

5.1. Stationary environment

We considered the network of $N = 8$ nodes shown in Fig. 1. The optimums $\{\boldsymbol{w}_k^\star\}_{k=1}^N$ consisted of $L = 26$ entries. The first 12 entries were common across all nodes, that is, $[\boldsymbol{w}_1^\star]_{\mathcal{G}_1} = \ldots = [\boldsymbol{w}_N^\star]_{\mathcal{G}_1}$ with $\mathcal{G}_1 = \{1, \ldots, 12\}$. These entries were sampled from a uniform distribution $\mathcal{U}(-1, 1)$. The next 8 entries were uniformly sampled from $\mathcal{U}(-1, 1)$ for each node, so that there was no relationship between the entries of group $\mathcal{G}_2 = \{13, \ldots, 20\}$. The last 6 entries were $[\boldsymbol{w}_k^\star]_i = [\boldsymbol{w}^o]_i + u_{ki}$ for $i \in \mathcal{G}_3 = \{21, \ldots, 26\}$, with $[\boldsymbol{w}^o]_i$ uniformly sampled from $\mathcal{U}(-1, 1)$ for all nodes, and independent perturbations u_{ki} sampled from $\mathcal{U}(-0.1, 0.1)$. We did not assume that nodes know this vector structure beforehand. Inputs $\boldsymbol{x}_n$ were zero-mean 26×1 random vectors governed by a Gaussian distribution with covariance matrix $\boldsymbol{R}_{x,k} = \sigma_{x,k}^2 \boldsymbol{I}_L$. The noises $z_k(n)$ were i.i.d. zero-mean Gaussian random variables, independent of any other signal with variances $\sigma_{z,k}^2$. Variances $\sigma_{x,k}^2$ and $\sigma_{z,k}^2$ used in this experiment were sampled from $\mathcal{U}(0.8, 1.2)$ and $\mathcal{U}(0.18, 0.22)$, respectively. The following algorithms were run: (i) Non-cooperative LMS ($\boldsymbol{A} = \boldsymbol{I}_N$); (ii) diffusion LMS; (iii) Multitask algorithm with combination weight adjustments in [13]; (iv) Group diffusion LMS with (iv) $M = 4$ uniform contingous groups; (v) $M = 6$ uniform contingous groups; and (vi) optimal grouping. The step sizes were successively set to $\mu = 0.05$ and $\mu = 0.02$. The prediction step size μ' in (23) was set to $\mu' = 2\mu$. Parameter ϵ in the normalized gradient $\boldsymbol{q}_{k,n}$ was set to 0.01. The MSD learning curves are shown in Figs. 2(a) and 2(b).

The non-cooperative LMS algorithm can be considered as the reference for this comparison test since it does not rely on any cooperation. Since the single-task assumption was violated, especially by the entries of group $\mathcal{G}_2$, the performance of diffusion LMS was severely degraded. The strategy in [13] can adaptively adjust the combination weights, but it cannot take possible group structures into account. It thus processed the parameter vectors $\boldsymbol{w}_k$ as if they were significantly different, in particular because of the entries in group $\mathcal{G}_2$. This algorithm thus inhibited cooperation between nodes and achieved similar performance as the non-cooperative LMS algorithm. These results show that for estimation problems involving group structures in their weight vectors, implementations beyond single-task or multi-task diffusion are called for in order to exploit the group structure more fully. The proposed algorithm with grouping strategy assigned a combination coefficient to each group, separately. It achieved significantly enhanced performance. The algorithm setting involving $M = 6$ groups outperformed the setting with only $M = 4$ groups, by using finer groups at the cost of higher computational complexity. Finally, we considered the optimal grouping used to generate the data. This grouping led to the lowest MSD. In Fig. 3, we present the combination weight matrices $\boldsymbol{A}_1$, $\boldsymbol{A}_2$ and $\boldsymbol{A}_3$. Cooperations can be observed for $\mathcal{G}_1$ and $\mathcal{G}_3$, but not for $\mathcal{G}_2$. This figure illustrates the efficiency of the proposed algorithm.

5.2. Nonstationary environment

We also considered a nonstationary setting with the network depicted in Fig. 1. Between time instants $n = 0$ to $n = 500$, the same setting as in the stationary case studied before was used. From time instant $n = 501$, the sub-vectors indexed by $\mathcal{G}_1$ and $\mathcal{G}_2$ were modified to be a common vector from $\mathcal{U}(0, 1)$ across nodes 2, 6, 7, 8. From time instant $n = 1001$, the optimum parameter vectors for nodes 2, 4, 5 were set independently by sampling them from the uniform distribution $\mathcal{U}(-1, 1)$. Simulation results for the non-cooperative LMS, and the group diffusion LMS with $M = 4$ and $M = 6$, are provided in Fig. 2(c). The performance gain and tracking ability of the proposed grouping strategy can be observed.

6. CONCLUSION AND PERSPECTIVES

In this paper, we introduced grouping into diffusion adaptation to take advantage of structural similarity among parameter vectors to estimate. Simulation results illustrated the effectiveness of the grouping strategy and of the information fusion rule. The work presented here assumed predefined group structures. In the future, we will investigate adaptive grouping and other flexible partitioning techniques.

7. REFERENCES

[1] A. H. Sayed, S.-Y Tu, J. Chen, X. Zhao, and Z. Towfic, "Diffusion strategies for adaptation and learning over networks," *IEEE Sig. Process. Mag.*, vol. 30, no. 3, pp. 155–171, May 2013.

[2] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Libraray in Signal Processing*, R. Chellapa and S. Theodoridis, Eds., pp. 322–454. Elsevier, 2013.

[3] A. H. Sayed, "Adaptive networks," *Proc. of the IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.

[4] 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.

[5] A. H. Sayed, "Adaptation, learning, and optimization over networks," in *Foundations and Trends in Machine Learning*, vol. 7, pp. 311–801. NOW Publishers, Boston-Delft, Jul. 2014.

[6] D. P. Bertsekas, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optimiz.*, vol. 7, no. 4, pp. 913–926, Nov. 1997.

[7] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE J. of Sel. Topics Areas Commun.*, vol. 23, no. 4, pp. 798–808, Apr. 2005.

[8] D. Blatt, A. O. Hero, and H. Gauchman, "A convergent incremental gradient method with constant step size," *SIAM J. Optimiz.*, vol. 18, no. 1, pp. 29–51, Feb. 2007.

[9] 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.

[10] 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.

[11] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks: Link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 355–369, Jan. 2009.

[12] K. Srivastava and A. Nedic, "Distributed asynchronous constrained stochastic optimization," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 772–790, Aug. 2011.

[13] J. Chen, C. Richard, and A. H. Sayed, "Diffusion LMS over multitask networks," *IEEE Trans. Signal Process.*, vol. 63, no. 11, pp. 2733–2748, Jun. 2015.

[14] X. Zhao and A. H. Sayed, "Distributed clustering and learning over networks," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3285–3300, Jul. 2015.

[15] J. Chen, C. Richard, and A. H. Sayed, "Adaptive clustering for multitask diffusion networks," in *Proc. 23th European Signal Process. Conf. (EUSIPCO)*, Nice, France, Sep. 2015.

[16] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129–4144, Aug. 2014.

[17] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Performance analysis of multitask diffusion adaptation over asynchronous networks," in *Proc. 48th Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2014, pp. 788 – 792.

[18] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Multitask diffusion LMS with sparsity-based regularization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Brisbane, Australia, May 2015, pp. 3516–3520.

[19] J. Chen, C. Richard, A. O. Hero, and A. H. Sayed, "Diffusion LMS for multitask problems with overlapping hypothesis subspaces," in *Proc. IEEE Int. Workshop on Machine Learn. for Signal Process.*, Reims, France, Sept. 2014, pp. 1–6.

[20] N. Bogdanović, J. Plata-Chaves, and K. Berberidis, "Distributed incremental-based LMS for node-specific parameter estimation over adaptive networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Vancouver, Canada, May 2013, pp. 5425–5429.

[21] N. Bogdanović, J. Plata-Chaves, and K. Berberidis, "Distributed diffusion-based LMS for node-specific parameter estimation over adaptive networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Florence, Italy, May 2014, pp. 7223–7227.

[22] S.-K. Ting, *Adaptive Techniques for Mitigating Circuit Imperfections in High Performance A/D Converters*, Ph.D. thesis, Electrical Engineering Department, UCLA, Mar. 2014.

[23] X. Zhao and A. H. Sayed, "Clustering via diffusion adaptation over networks," in *Proc. Int. Workshop Cognitive Inf. Process. (CIP)*, Parador de Baiona, Spain, May 2012, pp. 1–6.