2014 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 21–24, 2014, REIMS, FRANCE

DIFFUSION LMS FOR MULTITASK PROBLEMS WITH OVERLAPPING HYPOTHESIS SUBSPACES

Jie Chen [⋆†] *Cédric Richard* [†] *Alfred O. Hero III* [⋆] *Ali H. Sayed* [‡]

[⋆] University of Michigan, Ann Arbor, USA
[†] Université de Nice Sophia-Antipolis, CNRS, France
[‡] University of California, Los Angeles, USA
{jiechn, hero}@umich.edu cedric.richard@unice.fr sayed@ee.ucla.edu

ABSTRACT

There are many important applications that are multitask-oriented in the sense that there are multiple optimum parameter vectors to be inferred simultaneously by networked agents. In this paper, we formulate an online multitask learning problem where node hypothesis spaces partly overlap. A cooperative algorithm based on diffusion adaptation is derived. Some results on its stability and convergence properties are also provided. Simulations are conducted to illustrate the theoretical results.

Index Terms— Multitask learning, distributed optimization, diffusion strategy, collaborative processing.

1. INTRODUCTION

Distributed adaptive learning strategies over networks enable agents to learn a concept via local information exchange and through continuous adaptation, and continuously adapt to track possible drifts. These strategies offer an attractive alternative to centralized solutions with advantages related to scalability, robustness and decentralization. Many application examples exist in the realm of social, economic and biological networks – see [1, 2] and the references therein. Several distributed strategies for online parameter estimation have been proposed in the literature, including consensus strategies [3–5], incremental strategies [6–9], and diffusion strategies [10–15]. Incremental techniques require the determination of a cyclic path that runs across all nodes, which is generally an NP-hard problem. Besides, they are sensitive to link failures. On the other hand, diffusion strategies have been shown to have superior stability and performance ranges [16] than consensus-based implementations. Accessible overviews of results on diffusion adaptation can be found in [2, 10, 11]. This literature focuses primarily, though not exclusively [17–19], on the case where nodes estimate a single parameter vector collaboratively. We refer to problems of this type as *single-task* problems. However, many problems of interest happen to be *multitask*-oriented in that there are multiple parameter vectors to be inferred simultaneously. Although these parameters are different over the networks, they may have relationships that can be exploited to improve estimation accuracy.

Multitask learning has been studied by the machine learning community in several contexts, including web page categorization [20], web-search ranking [21], disease progression modeling [22], among other areas. This concept is also relevant in the context of estimation over adaptive networks. Initial investigations on multitask problems with diffusion strategies appeared in [17, 23]. One useful way to exploit and model relationships among tasks is to formulate optimization problems with appropriate regularizers. Several regularization schemes have been proposed in the machine learning community, including mean regularization [24], low-rank regularization [25], and clustered regularization [26]. The works [27, 28] considered multitask networks composed of connected clusters of nodes. In each cluster, agents collaboratively estimate a local parameter vector. Co-regularization between neighboring clusters was used to enhance estimation accuracy. An alternative strategy to model relationships between tasks is to assume that the node hypothesis spaces partially overlap [29–31].

We build on this principle to address online distributed estimation problems over multitask networks. Although this work is restricted to the case where the overlap of node hypothesis subspaces is known, we introduce a useful extension of diffusion adaption strategies to multitask problems. We analyse its convergence properties in the mean and mean-square senses. Finally, we provide an illustrative example to verify the theoretical findings.

Notation. Small letters x denote scalars, and boldface small letters $\boldsymbol{x}$ denote column vectors. Boldface capital letters $\boldsymbol{R}$ represent matrices, and the superscript $(\cdot)^\top$ denotes matrix transpose. $\boldsymbol{I}_N$ denotes the $N \times N$ identity matrix. $\mathcal{N}_k$ denotes the neighbors of node k, including k. The operator col$\{\cdot\}$ stacks its vector arguments on the top of each other to generate a connected vector. The operator diag$\{\cdot\}$ formulates a (block) diagonal matrix. Finally, $\otimes$ denotes the Kronecker product, and vec$\{\cdot\}$ stacks the columns of a matrix on top of each other into a vector.

2. MULTITASK LEARNING OVER NETWORKS

Consider a connected network composed of N nodes. The problem is to estimate an $L \times 1$ unknown vector $\boldsymbol{w}_k^\star$ at each node k from collected measurements. Node k has access to local temporal streaming measurement sequences $\{d_{k,n}, \boldsymbol{x}_{k,n}\}$, with $d_{k,n}$ denoting a zero-mean reference signal, and $\boldsymbol{x}_{k,n}$ denoting an $L \times 1$ regression vector with covariance matrix $\boldsymbol{R}_{x,k} = E\{\boldsymbol{x}_{k,n}\boldsymbol{x}_{k,n}^\top\} > 0$. The data at node k are assumed to be related via the linear model:

$$d_{k,n} = \boldsymbol{x}_{k,n}^\top \boldsymbol{w}_k^\star + z_{k,n} \tag{1}$$

The work of C. Richard was partly supported by the Agence Nationale pour la Recherche, France, (ODISSEE project, ANR-13-ASTR- 0030). The work of A. H. Sayed was supported in part by NSF grant CCF-1011918 and ECCS-1407712. The work of A. O. Hero was supported in part by Air Force Research Laboratory Sensors Directorate under grant FA8650-07-D-1220-0006.

978-1-4799-3694-6/14/$31.00 ©2014 IEEE

where $\boldsymbol{w}_k^\star$ is the unknown parameter vector at node k, and $z_{k,n}$ is a zero-mean i.i.d. noise that is independent of every other signal and has variance $\sigma_{z,k}^2$. Let $J_k(w)$ denote a convex cost function for data fitting associated with node k. We consider the mean-square-error in this paper:

$$J_k(\boldsymbol{w}) = E\{|d_{k,n} - \boldsymbol{w}^\top \boldsymbol{x}_{k,n}|^2\} \tag{2}$$

It can be verified from (1) that each $J_k(\boldsymbol{w})$ is minimized at $\boldsymbol{w}_k^\star$. Depending on whether the minima of all $J_k(\boldsymbol{w})$ are achieved at the same location or not, referred to as tasks, the distributed learning problem can be single-task or multitask oriented [28].

In a single-task network, all nodes have to estimate the same parameter vector $\boldsymbol{w}^\star$. That is, in this case we have that

$$\boldsymbol{w}_k^\star = \boldsymbol{w}^\star, \quad \forall k \in \{1, \dots, N\} \tag{3}$$

Several popular cooperative strategies, such as diffusion [12, 13], were derived for this scenario by seeking the minimizer of the following aggregate cost function:

$$J^{\text{glob}}(\boldsymbol{w}) = \sum_{k=1}^{N} J_k(\boldsymbol{w}) \tag{4}$$

in a distributed manner. In a multitask network, on the other hand, each node needs to determine its specific parameter vector $\boldsymbol{w}_k^\star$. It will be assumed that some similarities or relationships exist among the parameter vectors of neighboring nodes so that cooperation can still be meaningful and useful, namely,

$$\boldsymbol{w}_k^\star \neq \boldsymbol{w}_\ell^\star \quad \text{and} \quad \boldsymbol{w}_k^\star \sim \boldsymbol{w}_\ell^\star \quad \text{if } \ell \in \mathcal{N}_k \tag{5}$$

where the symbol $\sim$ represents a similarity relationship in some sense. Each cost function $J_k(\boldsymbol{w})$ would not be generally minimized at the same point. It was shown in [15] that, in this case, the diffusion solution converges towards a Pareto optimum of the multi-objective optimization problem constructed from the costs in (4). Further results on the convergence behavior of the diffusion strategy under this multitask scenario are presented in [32]. These insights motivated an extension of the diffusion LMS strategy to deal more effectively with multitask problems in [27, 28].

3. SOLUTION MODELS WITH OVERLAPPING HYPOTHESIS SUBSPACES

3.1. Problem formulation

Relationships among optima can be modeled in several ways, and they may help improve the estimation ability of agents. In this paper, we assume that the optimum parameter vector at each node k can be expressed as

$$\boldsymbol{w}_k^\star = \boldsymbol{\Theta}\boldsymbol{u}^\star + \boldsymbol{\epsilon}_k^\star \tag{6}$$

where $\boldsymbol{\Theta}\boldsymbol{u}^\star$ is common to all nodes, $\boldsymbol{\epsilon}_k^\star$ is a node-specific component, and $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M]$ is an $L \times M$ matrix with $M \leq L$. We assume $\boldsymbol{\Theta}$ to be known and full-rank. According to this model, all tasks share a common component that lies in the subspace spanned by the columns of $\boldsymbol{\Theta}$. There are situations where an over-complete matrix $\boldsymbol{\Theta}$ should be used. This would require additional constraints such as sparsity for $\boldsymbol{u}^\star$. We will not consider this case here.

Now replacing (6) into (4) yields the aggregate cost function:

$$J^{\text{glob}}(\boldsymbol{u}, \{\boldsymbol{\epsilon}_k\}_{k=1}^N) = \sum_{k=1}^{N} E\{|d_{k,n} - (\boldsymbol{\Theta}\boldsymbol{u} + \boldsymbol{\epsilon}_k)^\top \boldsymbol{x}_{k,n}|^2\} \tag{7}$$

We expect the estimation of $\boldsymbol{w}_k^\star$ performed by each node to benefit from the cooperative estimation of $\boldsymbol{u}^\star$. It is however not suitable to minimize (7) directly with respect to $\boldsymbol{u}$ and $\{\boldsymbol{\epsilon}_k\}_{k=1}^N$ since the decomposition $\boldsymbol{w}_k = \boldsymbol{\Theta}\boldsymbol{u} + \boldsymbol{\epsilon}_k$ is not unique. Indeed, let $\boldsymbol{s}$ be any vector in the column span of $\boldsymbol{\Theta}$. Then, $\{\boldsymbol{\Theta}\boldsymbol{u}^\star - \boldsymbol{s}, \{\boldsymbol{\epsilon}_k^\star\}_{k=1}^N + \boldsymbol{s}\}$ is also a minimizer of (7). From the point of view of convex analysis, the rank deficiency of the Hessian of (7) results in non-uniqueness of the solution $\{\boldsymbol{u}^\star, \{\boldsymbol{\epsilon}_k^\star\}_{k=1}^N\}$. This ambiguity does not allow us to derive a cooperative strategy based on this decomposition.

3.2. Cooperative adaptive solution

Problem (7) can be modified so as to guarantee a unique solution. Among other possibilities, we restrict the components $\{\boldsymbol{\epsilon}_k\}_{k=1}^N$ to lie in the complementary subspace of span$\{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M\}$. Imposing this constraint, the problem to be addressed can be formulated as

$$\begin{aligned} &\min_{\boldsymbol{u}, \{\boldsymbol{\epsilon}_k\}_{k=1}^N} \; J^{\text{glob}}\left(\boldsymbol{u}, \{\boldsymbol{\epsilon}_k\}_{k=1}^N\right) \\ &\qquad \text{subject to } \; \boldsymbol{\epsilon}_k \in \text{span}(\boldsymbol{\Theta}_\perp) \qquad \forall k = 1, \dots, N \\ &\qquad \text{with } \; \boldsymbol{\Theta}^\top \boldsymbol{\Theta}_\perp = 0 \end{aligned} \tag{8}$$

where the $L - M$ columns of $\boldsymbol{\Theta}_\perp$ span the complementary subspace of span$\{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M\}$. Since $\boldsymbol{\epsilon}_k$ is an element of span$\{\boldsymbol{\Theta}_\perp\}$, it can be expressed as

$$\boldsymbol{\epsilon}_k = \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k \tag{9}$$

where $\boldsymbol{\xi}_k$ is an $(L - M) \times 1$ vector of coefficients. This representation is useful in several scenarios. First, consider the case where $\boldsymbol{\Theta}$ is partly composed of selected columns of the identity matrix $\boldsymbol{I}_L$. This means that a subset of the entries of $\boldsymbol{w}_k^\star$ are common to all nodes while no further restriction is imposed on the other entries. This case is a direct extension of the single-task scenario. Another example concerns a beamforming problem with a generalized side-lobe canceler (GSC). In that case, the matrix $\boldsymbol{\Theta}$ would act as a blocking matrix to cancel signal components that lie in the constraint space [33].

Now replacing (9) into (8), the optimization problem becomes unconstrained with the following objective function:

$$\begin{aligned} &J^{\text{glob}}(\boldsymbol{u}, \{\boldsymbol{\xi}_k\}_{k=1}^N) \\ &= \sum_{k=1}^{N} E\{|d_{k,n} - (\boldsymbol{\Theta}\,\boldsymbol{u} + \boldsymbol{\Theta}_\perp\,\boldsymbol{\xi}_k)^\top \boldsymbol{x}_{k,n}|^2\} \\ &= \sum_{k=1}^{N} E\{|d_{k,n}|^2\} + \boldsymbol{u}^\top \boldsymbol{\Theta}^\top \sum_{k=1}^{N} \boldsymbol{R}_{x,k} \boldsymbol{\Theta} \boldsymbol{u} - 2 \sum_{k=1}^{N} \boldsymbol{p}_{dx,k}^\top \boldsymbol{\Theta} \boldsymbol{u} \\ &\quad + 2\boldsymbol{u}^\top \boldsymbol{\Theta}^\top \sum_{k=1}^{N} \boldsymbol{R}_{x,k} \boldsymbol{\Theta}_\perp\, \boldsymbol{\xi}_k \\ &\quad + \sum_{k=1}^{N} \boldsymbol{\xi}_k^\top \boldsymbol{\Theta}_\perp^\top \boldsymbol{R}_{x,k}\, \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k - 2 \sum_{k=1}^{N} \boldsymbol{p}_{dx,k}^\top \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k \end{aligned} \tag{10}$$

with $\boldsymbol{R}_{x,k}$ denoting the covariance matrix of the input data $\boldsymbol{x}_{k,n}$, and $\boldsymbol{p}_{dx,k}$ denoting the covariance vector between $\boldsymbol{x}_{k,n}$ and $d_{k,n}$.

Lemma 1 *Under the constraint that components $\{\boldsymbol{\epsilon}_k\}_{k=1}^N$ lie in a subspace orthogonal to span$\{\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_M\}$, problem* (8) *admits a unique solution with respect to $\boldsymbol{u}$ and $\{\boldsymbol{\epsilon}_k\}_{k=1}^N$.* ■

This proposition can be proved via the positive definiteness of the Hessian of (10). This guarantee allows us to derive a distributed strategy for the estimation problem. Focusing on the terms that depend on $\boldsymbol{u}$ in (10) and considering the optimum $\boldsymbol{\xi}_k^\star$, we have the global cost depending on $\boldsymbol{u}$:

$$\begin{aligned} J_u^{\text{glob}}(\boldsymbol{u}) &= \sum_{k=1}^{N} J_{u,k}(\boldsymbol{u}) \\ &= \sum_{k=1}^{N} \Big(\boldsymbol{u}^\top \boldsymbol{\Theta}^\top \boldsymbol{R}_{x,k} \boldsymbol{\Theta}\, \boldsymbol{u} - 2\boldsymbol{p}_{dx,k}^\top \boldsymbol{\Theta} \boldsymbol{u} \\ &\qquad\qquad + 2\boldsymbol{u}^\top \boldsymbol{\Theta}^\top \boldsymbol{R}_{x,k} \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k^\star + g_k(\boldsymbol{\xi}_k^\star)\Big) \end{aligned} \tag{11}$$

where $g_k(\boldsymbol{\xi}_k^\star)$ represents the remaining terms with $\boldsymbol{\xi}_k^\star$ in (10). Since $J_u^{\text{glob}}(\boldsymbol{u})$ has a unique minimizer for all nodes over the network, nodes can adopt a single-task cooperative strategy to enhance estimation accuracy. Without loss of generality and considering the advantage of diffusion adaptation, we shall now derive an algorithm based on diffusion LMS. We introduce a right-stochastic matrix $\boldsymbol{C}$ with nonnegative entries $c_{\ell k}$ such that

$$\sum_{k=1}^{N} c_{\ell k} = 1 \quad \text{and} \quad c_{\ell k} = 0 \text{ if } k \notin \mathcal{N}_\ell \tag{12}$$

With each node k, we associate a local cost over the variable $\boldsymbol{u}$:

$$J_{u,k}^{\text{loc}}(\boldsymbol{u}) = \sum_{\ell \in \mathcal{N}_k} c_{\ell k} J_{u,\ell}(\boldsymbol{u}). \tag{13}$$

Since $\boldsymbol{C}$ is right-stochastic, we note that

$$J_u^{\text{glob}}(\boldsymbol{u}) = \sum_{k=1}^{N} J_{u,k}^{\text{loc}}(\boldsymbol{u}). \tag{14}$$

Using instantaneous approximations for second-order statistics, namely, $\boldsymbol{R}_{x,k} \approx \boldsymbol{x}_{k,n}\boldsymbol{x}_{k,n}^\top$, and $\boldsymbol{p}_{dx,k} = d_{k,n}\,\boldsymbol{x}_{k,n}$, following the derivation of the diffusion LMS from [11, 13], and using the instantaneous estimate $\boldsymbol{\xi}_{k,n}$ for approximating the unknown $\boldsymbol{\xi}_k^\star$, we can update the the estimate for the parameter vector $\boldsymbol{u}$ at node k as follows:

$$\begin{aligned} \boldsymbol{u}_{k,n+\frac{1}{2}} &= \boldsymbol{u}_{k,n} + \mu \sum_{\ell \in \mathcal{N}_k} c_{\ell k} \boldsymbol{\Theta}^\top \boldsymbol{x}_{\ell,n} \big[d_{\ell,n} - (\boldsymbol{\Theta}\boldsymbol{u}_{k,n})^\top \boldsymbol{x}_{\ell,n} \\ &\qquad\qquad - (\boldsymbol{\Theta}_\perp \boldsymbol{\xi}_{\ell,n})^\top \boldsymbol{x}_{\ell,n}\big] \\ \boldsymbol{u}_{k,n+1} &= \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{u}_{k,n+\frac{1}{2}} \end{aligned} \tag{15}$$

where $\boldsymbol{u}_{k,n+\frac{1}{2}}$ is the intermediary result provided by the adaptation step. Although $\boldsymbol{\xi}_{k,n}$ is used in place of $\boldsymbol{\xi}_k^\star$, we will show the convergence of the algorithm in the next section. The nonnegative coefficients $a_{\ell k}$ define a left-stochastic matrix $\boldsymbol{A}$ that satisfies the conditions

$$\sum_{\ell=1}^{N} a_{\ell k} = 1 \quad \text{and} \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \tag{16}$$

On the other hand, since the parameter vectors $\boldsymbol{\xi}_k$ are node-specific, if no further constraints are imposed, they can be updated independently of each other via stochastic gradient descent:

$$\begin{aligned} \boldsymbol{\xi}_{k,n+1} &= \boldsymbol{\xi}_{k,n} \\ &\quad + \mu\, \boldsymbol{\Theta}_\perp^\top \boldsymbol{x}_{k,n} \big[d_{k,n} - (\boldsymbol{\Theta}\boldsymbol{u}_{k,n})^\top \boldsymbol{x}_{k,n} - (\boldsymbol{\Theta}_\perp \boldsymbol{\xi}_{k,n})^\top \boldsymbol{x}_{k,n}\big] \end{aligned} \tag{17}$$

At each instant n, node k updates parameter vectors $\boldsymbol{u}_{k,n}$ and $\boldsymbol{\xi}_{k,n}$ using (15) and (17), respectively. The estimate $\boldsymbol{w}_{k,n+1}$ is correspondingly given by

$$\boldsymbol{w}_{k,n+1} = \boldsymbol{\Theta}\, \boldsymbol{u}_{k,n+1} + \boldsymbol{\Theta}_\perp\, \boldsymbol{\xi}_{k,n+1} \tag{18}$$

It is interesting to note from (18) that

$$\boldsymbol{u}_{k,n+1} = (\boldsymbol{\Theta}^\top \boldsymbol{\Theta})^{-1} \boldsymbol{\Theta}^\top \boldsymbol{w}_{k,n+1} \tag{19}$$

$$\boldsymbol{\xi}_{k,n+1} = (\boldsymbol{\Theta}_\perp^\top \boldsymbol{\Theta}_\perp)^{-1} \boldsymbol{\Theta}_\perp^\top \boldsymbol{w}_{k,n+1} \tag{20}$$

This means that update equations (15) and (17) can be expressed in terms of $\boldsymbol{w}_k$, without using the auxiliary variables $\boldsymbol{u}$ and $\{\boldsymbol{\xi}_k\}_{k=1}^N$. This makes the algorithm a direct extension of diffusion LMS with subspace constraints. Specifically, choosing $\boldsymbol{C} = \boldsymbol{I}_N$ in order to avoid raw data exchange and node-specific components, we get the algorithm presented in Algorithm 1.

```
Algorithm 1: ATC diffusion LMS for multitask problems
with hypothesis subspace overlap

Parameters: Preset
        – non-negative step size $\mu$ for all nodes.
        – left-stochastic combination matrix $\boldsymbol{A}$.
        – matrix $\boldsymbol{\Theta}$ with vectors $\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$.
Initialization: Set initial weights $\boldsymbol{w}_{k,0} = \boldsymbol{0}$ for all $k$.
Algorithm: At each instant $n \geq 1$, and for each node $k$,
           updates $\boldsymbol{w}_{k,n}$:

    $\boldsymbol{w}_{k,n+\frac{1}{2}} = \boldsymbol{w}_{k,n} + \mu\, \boldsymbol{S}_{\boldsymbol{\Theta}}\, \boldsymbol{x}_{k,n} \left[d_{k,n} - \boldsymbol{w}_{k,n}^\top \boldsymbol{x}_{k,n}\right]$   (21)
    $\boldsymbol{w}_{k,n+1} = \boldsymbol{P}_{\boldsymbol{\Theta}_\perp}\, \boldsymbol{w}_{k,n+\frac{1}{2}} + \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{P}_{\boldsymbol{\Theta}}\, \boldsymbol{w}_{\ell,n+\frac{1}{2}}$   (22)

where $\boldsymbol{S}_{\boldsymbol{\Theta}} = \boldsymbol{\Theta}\boldsymbol{\Theta}^\top + \boldsymbol{\Theta}_\perp \boldsymbol{\Theta}_\perp^\top$, along with the projection
matrices $\boldsymbol{P}_{\boldsymbol{\Theta}} = \boldsymbol{\Theta}(\boldsymbol{\Theta}^\top\boldsymbol{\Theta})^{-1}\boldsymbol{\Theta}^\top$ and $\boldsymbol{P}_{\boldsymbol{\Theta}_\perp} = \boldsymbol{I}_N - \boldsymbol{P}_{\boldsymbol{\Theta}}$.
```

If $\boldsymbol{\Theta}_\perp$ is the complementary subspace of $\boldsymbol{\Theta}$, and columns of $\boldsymbol{\Theta}$ and $\boldsymbol{\Theta}_\perp$ are orthonormal, Algorithm 1 can be further simplified using $\boldsymbol{\Theta}\boldsymbol{\Theta}^\top + \boldsymbol{\Theta}_\perp\, \boldsymbol{\Theta}_\perp^\top = \boldsymbol{I}_N$ and $\boldsymbol{P}_{\,\boldsymbol{\Theta}} = \boldsymbol{\Theta}\boldsymbol{\Theta}^\top$. If $\boldsymbol{\Theta} = \boldsymbol{I}_N$, it reduces to the diffusion LMS algorithm (with $\boldsymbol{C} = \boldsymbol{I}_N$) [11].

4. NETWORK PERFORMANCE ANALYSIS

In this section we examine the convergence properties and performance of the adaptive strategy described in Algorithm 1. Detailed proofs or derivations are omitted due to space constraints. In order to perform the analysis, we collect information from across the network into block vectors and matrices. Let us denote by $\boldsymbol{w}_n$ and $\boldsymbol{w}^\star$ the block weight estimate vector and the block optimum weight vector, respectively, both of size $NL \times 1$, that is,

$$\boldsymbol{w}_n = \text{col}\{\boldsymbol{w}_{1,n}, \ldots, \boldsymbol{w}_{N,n}\} \tag{23}$$

$$\boldsymbol{w}^\star = \text{col}\{\boldsymbol{w}_1^\star, \ldots, \boldsymbol{w}_N^\star\} \tag{24}$$

Define the weight error vector $\boldsymbol{v}_n$ as the difference between the instantaneous estimate $\boldsymbol{w}_n$ and the optimum $\boldsymbol{w}^\star$:

$$\boldsymbol{v}_n = \boldsymbol{w}_n - \boldsymbol{w}^\star \tag{25}$$

Let us introduce the following $NL \times NL$ block diagonal matrices:

$$\boldsymbol{H}_x = \text{diag}\,\{\boldsymbol{R}_{x,1},\, \ldots,\, \boldsymbol{R}_{x,N}\} \tag{26}$$

$$\boldsymbol{D}_{\boldsymbol{S}_{\boldsymbol{\Theta}}} = \text{diag}\{\boldsymbol{S}_{\boldsymbol{\Theta}}, \ldots, \boldsymbol{S}_{\boldsymbol{\Theta}}\} \tag{27}$$

$$\boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}}} = \text{diag}\{\boldsymbol{P}_{\boldsymbol{\Theta}}, \ldots, \boldsymbol{P}_{\boldsymbol{\Theta}}\} \tag{28}$$

$$\boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}_\perp}} = \text{diag}\{\boldsymbol{P}_{\boldsymbol{\Theta}_\perp}, \ldots, \boldsymbol{P}_{\boldsymbol{\Theta}_\perp}\} \tag{29}$$

as well as the block matrices and vectors:

$$\boldsymbol{B} = \big(\boldsymbol{\mathcal{A}}^\top \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}}} + \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}_\perp}}\big)(\boldsymbol{I}_{LN} - \mu\, \boldsymbol{D}_{\boldsymbol{S}_{\boldsymbol{\Theta}}} \boldsymbol{H}_x) \tag{30}$$

$$\boldsymbol{r} = (\boldsymbol{\mathcal{A}}^\top - \boldsymbol{I}_{LN})\boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}}} \boldsymbol{w}^\star \tag{31}$$

$$\begin{aligned}\boldsymbol{G} = (\boldsymbol{\mathcal{A}}^\top \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}}} + \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}_\perp}})\, \mathrm{diag}\{\sigma_{z,1}^2 \boldsymbol{R}_{x,1}, \ldots, \sigma_{z,N}^2 \boldsymbol{R}_{x,N}\} \\ \times (\boldsymbol{\mathcal{A}}^\top \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}}} + \boldsymbol{D}_{\boldsymbol{P}_{\boldsymbol{\Theta}_\perp}})^\top\end{aligned} \tag{32}$$

where $\boldsymbol{\mathcal{A}} = \boldsymbol{A} \otimes \boldsymbol{I}_L$. Note that if the optimum $\boldsymbol{w}^\star$ strictly satisfies (6) and the constraints in (8), then $\boldsymbol{r}$ in (31) reduces to 0. We will continue using expression (31) in order to have a more general analysis for the cases where these constraints can be violated. It can be verified that the mean weight error vector $E\{\boldsymbol{v}_n\}$ evolves according to the recursion:

$$E\{\boldsymbol{v}_{n+1}\} = \boldsymbol{B}\, E\{\boldsymbol{v}_n\} - \boldsymbol{r} \tag{33}$$

We note from (33) that Algorithm 1 asymptotically converges in the mean sense for step-sizes μ that ensure $\rho(\boldsymbol{B}) < 1$, where $\rho(\cdot)$ denotes the spectral radius of its matrix argument. In particular, if the optimal vectors $\boldsymbol{w}^\star$ satisfy (6) subject to constraints (8), then Algorithm 1 becomes unbiased with respect to $\boldsymbol{w}^\star$, that is,

$$\lim_{n\to\infty} E\{\boldsymbol{v}_n\} = 0. \tag{34}$$

Otherwise, the bias is given by

$$\lim_{n\to\infty} E\{\boldsymbol{v}_n\} = (\boldsymbol{B} - \boldsymbol{I}_{LN})^{-1} \boldsymbol{r}. \tag{35}$$

However, even in this latter case, cooperation among nodes may still be beneficial as long as the contrast between the optimal vectors $\boldsymbol{w}_k^\star$ in span$\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$ is small. Such situation is discussed in [23] for diffusion LMS when operating in a multitask environment.

Let us assume that the step-size μ is sufficiently small so that higher-order powers of μ can be neglected, and let

$$\boldsymbol{K} = \boldsymbol{B}^\top \otimes \boldsymbol{B}^\top. \tag{36}$$

It can be verified that the squared norm of $\boldsymbol{v}_n$ weighted by any positive-definite matrix $\boldsymbol{\Sigma}$ (represented in vector form as $\boldsymbol{\sigma} = \mathrm{vec}\{\boldsymbol{\Sigma}\}$) evolves approximately according to

$$E\{\|\boldsymbol{v}_{n+1}\|_{\boldsymbol{\sigma}}^2\} = E\{\|\boldsymbol{v}_n\|_{\boldsymbol{K\sigma}}^2\} + \boldsymbol{s}_n^\top \boldsymbol{\sigma} \tag{37}$$

where

$$\boldsymbol{s}_n = \mathrm{vec}\Big(\boldsymbol{G}^\top + \boldsymbol{r}\boldsymbol{r}^\top - 2\,\boldsymbol{r}\,(\boldsymbol{B}\, E\{\boldsymbol{v}_n\})^\top\Big) \tag{38}$$

We note from (37) and (38) that the algorithm is mean-square stable when $\boldsymbol{K}$ is stable. In this case, we define the network mean-square-deviation (MSD) learning curve as

$$\zeta_n = \frac{1}{N} E\{\|\boldsymbol{v}(n)\|\}^2. \tag{39}$$

It can be verified that ζ_n evolves according to the following recursion for $n \geq 0$:

$$\zeta_{n+1} = \zeta_n + \frac{1}{N}\big((\boldsymbol{\gamma}_n + \boldsymbol{s}_n)^\top \mathrm{vec}(\boldsymbol{I}_{LN}) - \|\boldsymbol{v}_0\|^2_{(\boldsymbol{I}_{(LN)^2} - \boldsymbol{K})\boldsymbol{K}^n \boldsymbol{\sigma}}\big) \tag{40}$$

$$\boldsymbol{\gamma}_{n+1} = \boldsymbol{K}^\top \boldsymbol{\gamma}_n + (\boldsymbol{K} - \boldsymbol{I}_{(LN)^2})^\top \boldsymbol{s}_n \tag{41}$$

with the initial conditions $\zeta_0 = \frac{1}{N}\|\boldsymbol{v}_0\|^2$ and $\boldsymbol{\gamma}_0 = \boldsymbol{0}$. Once convergence is achieved, then the steady-state MSD, namely, the limiting value of ζ_n as $n \to \infty$, is given by

$$\zeta_\infty = \frac{1}{N}\, \boldsymbol{s}_\infty^\top\, (\boldsymbol{I}_{(LN)^2} - \boldsymbol{K})^{-1} \mathrm{vec}(\boldsymbol{I}_{LN}) \tag{42}$$

with $\boldsymbol{s}_\infty$ determined from (38) and $E\{\boldsymbol{v}_\infty\}$ from (35) .

5. SIMULATIONS

In this section we provide an example to show how the algorithm converges, and to illustrate theoretical models. We consider a network consisting of 12 nodes with connections shown in Fig. 1(a). Inputs $\boldsymbol{x}(n)$ were zero-mean 4×1 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 are depicted in Fig. 1(b). Matrix $\boldsymbol{\Theta}$ was chosen as

$$\boldsymbol{\Theta} = \begin{pmatrix} 0.3162 & 0.5573 & -0.6325 \\ -0.6325 & -0.4352 & -0.3162 \\ 0.6325 & -0.4352 & 0.3162 \\ -0.3162 & 0.5573 & 0.6325 \end{pmatrix} \tag{43}$$

Note that its columns are orthonormal. The complementary subspace is spanned by:

$$\boldsymbol{\Theta}_\perp = (0.4352 \quad 0.5573 \quad 0.5573 \quad 0.4352)^\top \tag{44}$$

The coefficient vector $\boldsymbol{u}^\star$ was set to

$$\boldsymbol{u}^\star = (0.6 \ -0.4 \quad 0.3)^\top \tag{45}$$

Let us now consider two cases:

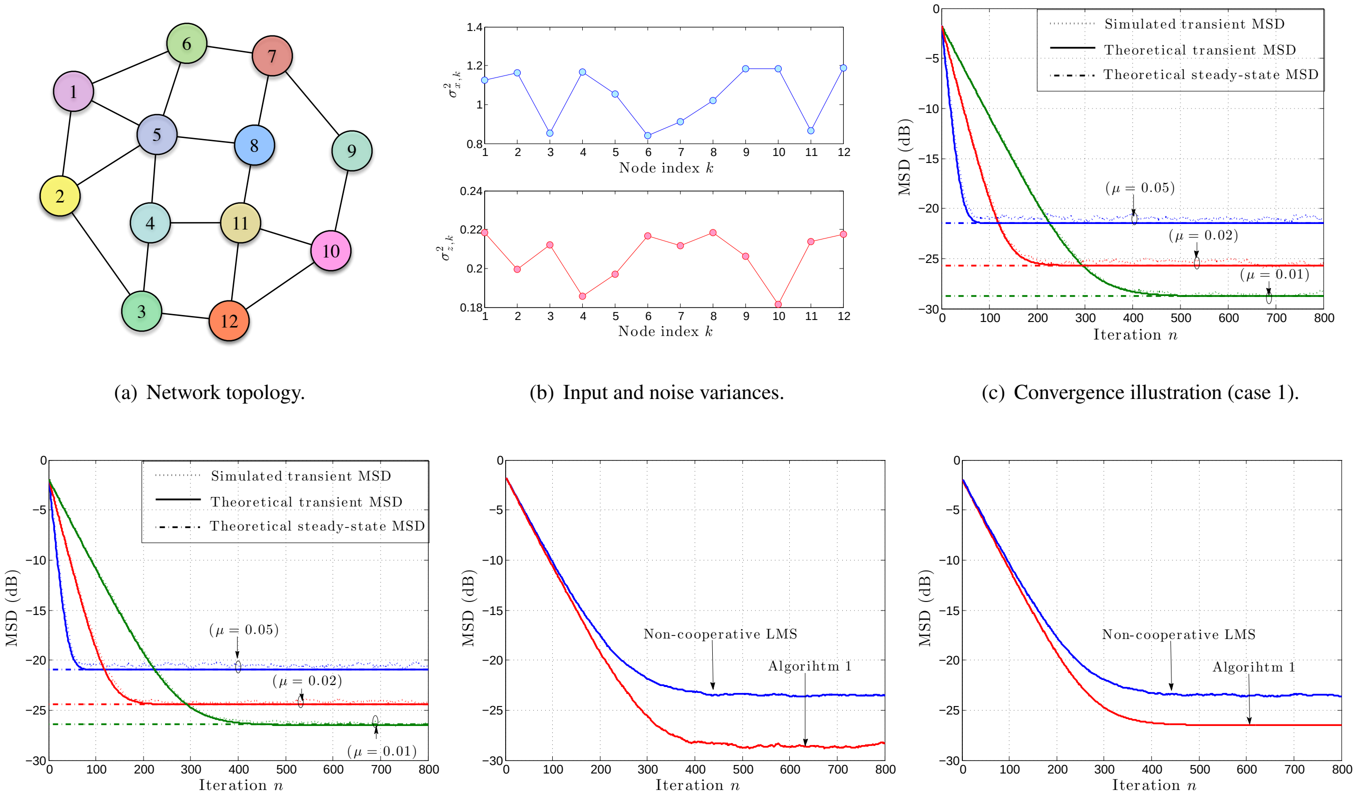
Case 1: Assume that the optimal parameter vectors $\boldsymbol{w}_k^\star$ follow model (6) with $\boldsymbol{\epsilon}_k^\star = \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k^\star$ and $\boldsymbol{\xi}_k^\star$ is a realization of a zero-mean Gaussian distribution with standard deviation 0.2. The step-sizes of the algorithm were set to $\mu = 0.01$, 0.02, and 0.05, respectively. According to Corollary 1, the algorithm is unbiased in this case. The MSD learning curves are illustrated in Figs. 1(c). The simulated curves were obtained by averaging over 100 Monte-Carlo runs. The theoretical transient behavior and theoretical steady-state MSD were calculated via Theorem 3 and Corollary 2, respectively. The simulation results agree with the theoretical results, and illustrate the trade-off between the step-size and steady-state MSD as in usual adaptive strategies.

Case 2: Assume that the node-specific components $\boldsymbol{\epsilon}_k^\star$ in (6) do not strictly lie in the complementary subspace of $\boldsymbol{\Theta}$. In this experiment, we set

$$\boldsymbol{\epsilon}_k^\star = \boldsymbol{\Theta}\boldsymbol{\nu}_k^\star + \boldsymbol{\Theta}_\perp \boldsymbol{\xi}_k^\star \tag{46}$$

with $\boldsymbol{\nu}_k^\star$ corresponding to realizations of a zero-mean Gaussian distribution with standard deviation 0.02, and $\boldsymbol{\xi}_k^\star$ determined as in Case 1. Case 2 is thus a non-ideal situation where components $\boldsymbol{\Theta}(\boldsymbol{u}^\star + \boldsymbol{\nu}_k^\star)$ that lie in span$\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$ are not the same for all nodes. The simulated and theoretical MSD learning curves with various step-sizes are illustrated in Fig. 1(d).

Finally, we compared the proposed cooperative algorithm with non-cooperative LMS in the two cases described above. The MSD learning curves are shown in Figs. 1(e) and 1(f). It clearly appears that cooperation between nodes is beneficial. It can also be noticed that the violation of the orthogonality assumption of $\boldsymbol{\Theta}\boldsymbol{u}^\star$ and $\boldsymbol{\epsilon}_k^\star$ leads to a degradation in the performance of Algorithm 1. However, in this case, it still outperforms the non-cooperative strategy since the deviation caused by $\boldsymbol{\nu}_k^\star$ is small. It can be expected that Algorithm 1 should fail to perform well if $\boldsymbol{\nu}_k$ becomes significant. Autonomous clustering algorithms that adjust combination coefficients $a_{\ell k}$ can remedy this problem [32].



(a) Network topology.

(b) Input and noise variances.

(c) Convergence illustration (case 1).

(d) Convergence illustration (case 2).

(e) Performance Comparison (case 1).

(f) Performance Comparison (case 2).

Fig. 1. Network configuration and simulation result illustration.

6. CONCLUSION AND PERSPECTIVES

In this paper we formulated an online multitask learning problem with the assumption that optimums to be estimated consist of an offset component shared by all agents and a node-specific component in an orthogonal subspace. An algorithm that extends the single-task diffusion LMS algorithm was derived and its convergence properties were analyzed. Further work will include extensions of this multitask problem to other structural constraints, and applications to relevant scenarios.

7. REFERENCES

[1] A. H. Sayed, S. Barbarossa, and S. Theodoridis, Eds., *Special issue on Adaptation and Learning over Complex Networks, IEEE Signal Processing Magazine*, vol. 30, no. 3, May 2013.

[2] A. H. Sayed, “Adaptive networks,” *Proc. of the IEEE*, vol. 102, no. 4, pp. 460–497, April 2014.

[3] 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.

[4] 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.

[5] 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.

[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. 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.

[11] A. H. Sayed, “Diffusion adaptation over networks,” in *Academic Press Library in Signal Processing*, R. Chellapa and S. Theodoridis, Eds., pp. 322–454. Elsevier, 2014. Also available as arXiv:1205.4220 [cs.MA], May 2012.

[12] 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.

[13] 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.

[14] 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.

[15] J. Chen and A. H. Sayed, "Distributed Pareto optimization via diffusion strategies," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 205–220, Apr. 2013.

[16] 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.

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

[18] S.-Y. Tu and A. H. Sayed, "Adaptive decision making over complex networks," in *Proc. Asilomar Conf. on Signals Systems and Computers*, Pacific Grove, CA, Nov. 2012, pp. 525–530.

[19] 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. (ICASSP)*, Vancouver, Canada, May 2013, pp. 5425–5429.

[20] J. Chen, L. Tang, J. Liu, and J. Ye, "A convex formulation for leaning shared structures from muliple tasks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Montreal, Canada, Jun. 2009, pp. 137–144.

[21] O. Chapelle, P. Shivaswmy, K. Q. Vadrevu, S. Weinberger, Y. Zhang, and B. Tseng, "Multi-task learning for boosting with application to web search ranking," in *Proc. Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, Washington DC, USA, Jul. 2010, pp. 1189–1198.

[22] J. Zhou, L. Yuan, J. Liu, and J. Ye, "A multi-task learning formulation for predicting disease progression," in *Proc. Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*, San Diego, CA, USA, Aug. 2011, pp. 814–822.

[23] J. Chen and C. Richard, "Performance analysis of diffusion LMS in multitask networks," in *Proc. IEEE Int. Workshop on Comp. Adv. in Multi-Sensor Adaptive Process. (CAMSAP)*, Saint Martin, France, Dec. 2013, pp. 15–28.

[24] T. Evgeniou and M. Pontil, "Regularized multi-task learning," in *Proc. Int. Conf. Knowledge Discovery and Data Mining (SIGKDD)*, Seattle, WA, USA, Aug. 2004, pp. 109–117.

[25] J. Chen, J. Liu, and J. Ye, "Learning incoherent sparse and low-rank patterns from multiple tasks," *ACM Trans. Knowl. Discov. Data*, vol. 5, no. 4, pp. 1–44, Feb. 2012.

[26] L. Jacob, F Bach, and J. P. Vert, "Clustered multitask learning: A convex formulation," *Adv. Neural Inf. Process. Syst.*, vol. 21, pp. 745–752, 2008.

[27] J. Chen, C. Richard, and A. H. Sayed, "Diffusion LMS for clustered multitask networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Florence, Italy, May 2014, pp. 5524–5528.

[28] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," to appear in *IEEE Transactions on Signal Processing*, 2014. Also available as arXiv:1311.4894 [cs.MA], Nov. 2013.

[29] J. Baxter, "A model of inductive bias learning," *J. Artif. Intell. Res.*, vol. 12, no. 1, pp. 149–198, Feb. 2000.

[30] R. K. Anto and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *J. Mach. Learn. Res.*, vol. 6, pp. 1817–1853, Nov. 2005.

[31] J. Zhou, J. Chen, and J. Ye, "Clustered multi-task learning via alternating structure optimization," *Adv. Neural Inf. Process. Syst.*, vol. 25, pp. 702–710, 2011.

[32] J. Chen, C. Richard, and A. H. Sayed, "Diffusion LMS over multitask networks," *submitted for publication*, 2014. Also available as arXiv: 1404.6813 [CS.SY], Apr. 2014.

[33] L. J. Griffiths and C. W. Jim, "An alternative approach to linear constrained adaptive beamforming," *IEEE Trans. Antenn. Propag.*, vol. 30, no. 1, pp. 27–34, Jan. 1982.