Proximal Multitask Learning over Distributed Networks with Jointly Sparse Structure

Danqi Jin, Jie Chen, Jingdong Chen, Cédric Richard

PROXIMAL MULTITASK LEARNING OVER DISTRIBUTED NETWORKS WITH JOINTLY SPARSE STRUCTURE

Danqi Jin ⋆, *Jie Chen* ⋆, *Cédric Richard* †, *Jingdong Chen* ⋆

⋆ Centre of Intelligent Acoustics and Immersive Communications
School of Marine Science and Technology, Northwestern Polytechinical University, China
† Université de la Côte d'Azur, CNRS, France
danqijin@mail.nwpu.edu.cn dr.jie.chen@ieee.org cedric.richard@unice.fr jingdongchen@ieee.org

ABSTRACT

Modeling relations between local optimum parameter vectors in multitask networks has attracted much attention over the last years. This work considers a distributed optimization problem for parameter vectors with a jointly sparse structure among nodes, that is, the parameter vectors share the same support set. By introducing an $\ell_{\infty,1}$-norm penalty at each node, and using a proximal gradient method to minimize the regularized cost, we devise a proximal multitask diffusion LMS algorithm which promotes the joint-sparsity to enhance the estimation performance. Analyses are provided to ensure the stability. Simulation results are presented to highlight the performance.

Index Terms— Distributed optimization, diffusion strategy, proximal operator, joint sparsity, $\ell_{\infty,1}$-norm regularization.

1. INTRODUCTION

Diffusion adaptation has been widely used in multi-agent networks to address estimation problems in a distributed and online manner due to their superior performance and wide stability range [1]. Several diffusion algorithms have been devised under specific settings in the literature [2–6].

According to the relations between the optimal parameter vectors over the entire network, diffusion networks are divided into single-task and multitask networks. In single-task networks, all nodes estimate the same parameter vector. Typical works include [7–9]. In multitask networks, multiple different but related parameter vectors are inferred simultaneously in a cooperative manner, so as to improve the estimation accuracy by exploiting the similarities between tasks. One of the efficient ways to leverage these similarities is to introduce appropriate regularization terms. Related works include [10–13].

Multitask learning considerably enriches the modelling capacity of diffusion networks. Beyond the multitask models appeared in the aforementioned references, there are also applications where the optimal parameter vectors have a jointly sparse structure. Applications include, for instance, distributed dictionary learning and distributed spectrum sensing [14, 15]. Several works have been proposed to address problems with jointly sparse structure over diffusion networks. In [16], the authors propose to use the mixed $\ell_{2,0}$-norm. In [17], the authors devise an algorithm with $\ell_{2,0}$, $\ell_{2,1}$ and reweighted $\ell_{2,1}$ regularizers. However, these works use the subgradient method because the objective functions involve non-differentiable regularization terms, which is unfavorable for an accurate and fast convergence.

Compared to subgradient-based methods, it is known that using proximal operators is a more efficient way to solve optimization problems with non-differentiable regularizers. It often results in iterations with subproblems that admit closed-form solutions or can be solved with simple specialized methods [18]. For distributed estimation over networks, proximal algorithms have been used to estimate sparse parameter vectors in [19]. They are further used to optimize general stochastic costs with non-differentiable regularizers, and non-smooth regularizers in [20] and [21], respectively. Under the multitask assumption, the authors in [11] derive a closed-form solution to the proximal operator of the ℓ_1-distance between parameter vectors. In this paper, we propose a proximal diffusion LMS strategy for multitask networks with jointly sparse structure. In contrast to existing works [16, 17], the optimization problem is formulated with the $\ell_{\infty,1}$-norm regularization on the parameter matrices, and the closed-form solution for the proximal operator under this case is derived. Conditions to ensure the stability in the mean and mean-square sense are also provided. The superiority of the proposed method is validated with numerical experiments.

Notation. Normal font x, boldface small letters $\boldsymbol{x}$ and capital letters $\boldsymbol{X}$ denote scalars, column vectors and matrices, respectively. Symbol $[\,\cdot\,]_m$ denotes the m-th entry of its vector argument. The superscript $(\cdot)^\top$ denotes the transpose operator. The mathematical expectation is denoted by $\mathbb{E}\{\cdot\}$. The Gaussian distribution with mean μ and variance σ^2 is denoted by $\mathcal{N}(\mu,\sigma^2)$. Operator $|\cdot|$ takes the absolute value of its scalar or vector argument. Operator $\max\{\cdot,\cdot\}$ extracts the maximum value of its two arguments. Symbol $\preceq$ denotes a component-wise inequality. The set $\mathcal{N}_k$ denotes the neighbors of node k, including k itself, and $|\mathcal{N}_k|$ denotes its cardinality. The $\mathcal{N}_k^-$ denotes the neighbors of node k, excluding node k. Vector $\mathbb{1}_L$ is the all-one vector of dimension $L \times 1$.

2. PROBLEM FORMULATION

Consider a connected network consisting of N nodes. Each node k has access to streaming data $\{d_{k,n}, \boldsymbol{u}_{k,n}\}$ at time instant n, where $\boldsymbol{u}_{k,n} \in \mathbb{R}^{L\times 1}$ is the regression vector and $d_{k,n}$ denotes the observed signal. We assume that the data at time instant n are related via the linear model:

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

The work of Jie Chen was supported in part by NSFC grants 61671382 and 61811530283. The work of Jingdong Chen was supported in part by NSFC grants 61761146001 and 61811530283. The work of C. Richard was funded in part by ANR under grant ANR-19-CE48-0002 and by the CoopIntEER program CNRS-NSFC (DIALOG project). Corresponding author: Jie Chen.

where $\boldsymbol{w}_k^\star \in \mathbb{R}^{L\times 1}$ is the unknown system vector to estimate, and $z_{k,n}$ is a zero-mean additive noise. We assume that $z_{k,n}$ is independent of any other signal. Further, we assume that vectors $\boldsymbol{w}_k^\star$ over the entire network are jointly sparse. This means not only each $\boldsymbol{w}_k^\star$ is a sparse vector but, in addition, they all share the same support, namely,

$$\text{supp}(\boldsymbol{w}_1^\star) = \cdots = \text{supp}(\boldsymbol{w}_k^\star) = \cdots = \text{supp}(\boldsymbol{w}_N^\star) \tag{2}$$

where $\text{supp}(\boldsymbol{w}_k^\star) \triangleq \{j : [\boldsymbol{w}_k^\star]_j \neq 0\}$ is the support of $\boldsymbol{w}_k^\star$ [22].

In this work, we focus on distributed processing where only local information exchange is authorized. We thus collect $\boldsymbol{w}_\ell^\star$ over the neighborhood of node k into an $L \times |\mathcal{N}_k|$ matrix, and replace the k-th column $\boldsymbol{w}_k^\star$ by the optimization variable $\boldsymbol{w}_k$. This leads us to the local parameter matrix:

$$\boldsymbol{W}_k \triangleq \big[\boldsymbol{w}_k,\ \boldsymbol{w}_\ell^\star \text{ with } \ell \in \mathcal{N}_k^-\big] \in \mathbb{R}^{L\times|\mathcal{N}_k|}. \tag{3}$$

Without loss of generality, we suppose that the columns of $\boldsymbol{W}_k$ are sorted in increasing order according to the values of k and ℓ. Several mixed-norms have been introduced in the literature to promote the jointly sparse structure of a matrix, including the mixed $\ell_{2,1}$-norm and $\ell_{\infty,1}$-norm. Evaluating the mixed $\ell_{p,1}$-norm of matrix $\boldsymbol{W}_k$ with $p = 2$ or ∞ results in the following two steps:

Step 1: Evaluate the ℓ_p-norm of each row of $\boldsymbol{W}_k$, and stack the results into an $L \times 1$ intermediate vector;

Step 2: Evaluate the ℓ_1-norm of the obtained intermediate vector to promote sparsity.

Though $\ell_{2,1}$-norm can be more efficient in some cases [23], we shall consider the $\ell_{\infty,1}$-norm to promote the joint-sparsity. It is element-wise separable and facilitates the derivation of the proximal operator.

3. PROXIMAL MULTITASK DIFFUSION LMS

Before proceeding, to facilitate the following derivation we also denote $\boldsymbol{W}_k$ by

$$\boldsymbol{W}_k = \big[\bar{\boldsymbol{w}}_{k,1}^\top \quad \cdots \quad \bar{\boldsymbol{w}}_{k,m}^\top \quad \cdots \quad \bar{\boldsymbol{w}}_{k,L}^\top\big]^\top, \tag{4}$$

where $\bar{\boldsymbol{w}}_{k,m}$ is the m-th row of matrix $\boldsymbol{W}_k$. To determine the unknown vectors $\boldsymbol{w}_k^\star$ with jointly sparse structure, we consider the regularized cost at node k:

$$J_k(\boldsymbol{w}_k) = J'_k(\boldsymbol{w}_k) + \lambda_k g(\boldsymbol{w}_k) \tag{5}$$

with $J'_k(\boldsymbol{w}_k) \triangleq \frac{1}{2}\,\mathbb{E}\big\{|d_{k,n} - \boldsymbol{u}_{k,n}^\top \boldsymbol{w}_k|^2\big\}$. The nonnegative parameter λ_k is used to control the regularization strength, $g(\boldsymbol{w}_k) \triangleq \sum_{m=1}^{L} \|\bar{\boldsymbol{w}}_{k,m}\|_\infty$ evaluates the $\ell_{\infty,1}$-norm of $\boldsymbol{W}_k$. At each node k, we then consider the convex optimization problem [24]:

$$\boldsymbol{w}_k^\dagger = \underset{\boldsymbol{w}_k}{\text{argmin}}\, J_k(\boldsymbol{w}_k). \tag{6}$$

Within the context of online learning, such optimization problem is usually solved via subgradient-based methods. In this paper, we propose to devise a proximal algorithm since it is more stable than subgradient iterations [19, 25]. Proximal gradient methods generate a sequence of estimates by the following iterations [18]:

$$\boldsymbol{w}_{k,n+1} = \text{prox}_{\mu_k\lambda_k g}\big(\boldsymbol{w}_{k,n} - \mu_k \nabla J'_k(\boldsymbol{w}_{k,n})\big), \tag{7}$$

where μ_k is a positive small step-size, and the proximal operator is defined by

$$\text{prox}_{\lambda g}(\boldsymbol{v}) \triangleq \underset{\boldsymbol{w}_k}{\text{argmin}}\Big(g(\boldsymbol{w}_k) + \frac{1}{2\lambda}\|\boldsymbol{w}_k - \boldsymbol{v}\|_2^2\Big). \tag{8}$$

By introducing the intermediate quantity $\boldsymbol{\psi}_{k,n+1}$, calculating the gradient of $J'_k(\boldsymbol{w}_k)$ at $\boldsymbol{w}_{k,n}$ and using instantaneous approximation for unknown statistical quantities, we obtain from (7) the proximal multitask diffusion LMS algorithm for jointly sparse networks reported in Algorithm 1. Different from regular diffusion algorithms where the intermediate estimates are fused by weighted average, in this algorithm, estimates from neighboring nodes are fused via the proximal operator of $g(\boldsymbol{w}_{k,n})$.

Algorithm 1: Proximal multitask diffusion LMS

Initialize $\boldsymbol{w}_{k,0}$ for all $k = 1, 2, \cdots, N$, and repeat:

$$\begin{cases} \boldsymbol{\psi}_{k,n+1} = \boldsymbol{w}_{k,n} + \mu_k \boldsymbol{u}_{k,n}\big(d_{k,n} - \boldsymbol{u}_{k,n}^\top \boldsymbol{w}_{k,n}\big) \\ \boldsymbol{w}_{k,n+1} = \text{prox}_{\mu_k\lambda_k g}\big(\boldsymbol{\psi}_{k,n+1}\big) \end{cases} \tag{9}$$

4. PROXIMAL OPERATOR EVALUATION

To perform **Algorithm 1**, we need to derive a closed-form expression for the following proximal operator:

$$\begin{aligned} \boldsymbol{w}_{k,n+1} &= \text{prox}_{\mu_k\lambda_k g}(\boldsymbol{\psi}_{k,n+1}) \\ &= \underset{\boldsymbol{w}_k}{\text{argmin}}\,\big(g(\boldsymbol{w}_k) + \frac{1}{2\mu_k\lambda_k}\|\boldsymbol{w}_k - \boldsymbol{\psi}_{k,n+1}\|_2^2\big). \end{aligned} \tag{10}$$

As $g(\boldsymbol{w}_k)$ is separable over its all entries, its proximal operator can be evaluated in an element-wise manner as [18]:

$$[\text{prox}_{\mu_k\lambda_k g}(\boldsymbol{\psi}_{k,n+1})]_m = \text{prox}_{\mu_k\lambda_k g_m}([\boldsymbol{\psi}_{k,n+1}]_m) \tag{11}$$

with $g_m([\boldsymbol{w}_k]_m) \triangleq \|\bar{\boldsymbol{w}}_{k,m}\|_\infty$, $[\boldsymbol{w}_k]_m$ is the m-th entry of $\boldsymbol{w}_k$, and $\bar{\boldsymbol{w}}_{k,m}$ is the m-th row of matrix $\boldsymbol{W}_k$ in (3). This leads us to:

$$\begin{aligned} [\boldsymbol{w}_{k,n+1}]_m = \underset{[\boldsymbol{w}_k]_m}{\text{argmin}}\Big(&\max\{|[\boldsymbol{w}_k]_m|, |[\boldsymbol{w}_\ell^\star]_m| \text{ with } \ell \in \mathcal{N}_k^-\} \\ &+ \frac{1}{2\mu_k\lambda_k}\big([\boldsymbol{w}_k]_m - [\boldsymbol{\psi}_{k,n+1}]_m\big)^2\Big). \end{aligned} \tag{12}$$

For ease of presentation, we shall denote $[\boldsymbol{w}_{k,n+1}]_m$ by $\hat{w}$ as long as there is no ambiguity, and denote the maximal value of $|[\boldsymbol{w}_\ell^\star]_m|$ for $\ell \in \mathcal{N}_k^-$ as $[\boldsymbol{w}_k^o]_m$. According to the relation between $|[\boldsymbol{w}_k]_m|$ and $[\boldsymbol{w}_k^o]_m$, we further split the problem into the following two cases:

- Case 1: $|[\boldsymbol{w}_k]_m| < [\boldsymbol{w}_k^o]_m$. In this case, (12) becomes:

$$\hat{w} = \underset{\substack{[\boldsymbol{w}_k]_m \\ |[\boldsymbol{w}_k]_m| < [\boldsymbol{w}_k^o]_m}}{\text{argmin}}\ [\boldsymbol{w}_k^o]_m + \frac{1}{2\mu_k\lambda_k}\big([\boldsymbol{w}_k]_m - [\boldsymbol{\psi}_{k,n+1}]_m\big)^2. \tag{13}$$

  The solution is directly given by:

$$\hat{w} = \begin{cases} [\boldsymbol{\psi}_{k,n+1}]_m, & \text{if } |[\boldsymbol{\psi}_{k,n+1}]_m| < [\boldsymbol{w}_k^o]_m \\ [\boldsymbol{w}_k^o]_m, & \text{if } [\boldsymbol{\psi}_{k,n+1}]_m \geq [\boldsymbol{w}_k^o]_m \\ -[\boldsymbol{w}_k^o]_m, & \text{if } [\boldsymbol{\psi}_{k,n+1}]_m \leq -[\boldsymbol{w}_k^o]_m. \end{cases} \tag{14}$$

- Case 2: $|[\boldsymbol{w}_k]_m| \geq [\boldsymbol{w}_k^o]_m$. Equation (12) becomes:

$$\hat{w} = \underset{\substack{[\boldsymbol{w}_k]_m \\ |[\boldsymbol{w}_k]_m| \geq [\boldsymbol{w}_k^o]_m}}{\text{argmin}}\ \Big(|[\boldsymbol{w}_k]_m| + \frac{1}{2\mu_k\lambda_k}\big([\boldsymbol{w}_k]_m - [\boldsymbol{\psi}_{k,n+1}]_m\big)^2\Big) \tag{15}$$

We shall first discard the constraint $|[\boldsymbol{w}_k]_m| \geq [\boldsymbol{w}_k^o]_m$, and denote by $\hat{w}^o$ the solution of the unconstrained problem. As the cost function in (15) is convex on the real domain, this constraint will be taken into account in the course of the calculation. Consider first:

$$\hat{w}^o = \underset{[\boldsymbol{w}_k]_m}{\operatorname{argmin}} \Big(|[\boldsymbol{w}_k]_m| + \frac{1}{2\mu_k\lambda_k} \big([\boldsymbol{w}_k]_m - [\boldsymbol{\psi}_{k,n+1}]_m \big)^2 \Big) \tag{16}$$

the solution is given by the soft thresholding operator defined as [26]:

$$\hat{w}^o = S_{\mu_k\lambda_k}\big([\boldsymbol{\psi}_{k,n+1}]_m\big) = \tag{17}$$
$$\begin{cases} [\boldsymbol{\psi}_{k,n+1}]_m + \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m < -\mu_k\lambda_k \\ [\boldsymbol{\psi}_{k,n+1}]_m - \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m > \mu_k\lambda_k \\ 0 \qquad\qquad \text{otherwise.} \end{cases}$$

If $[\boldsymbol{w}_k^o]_m = 0$, problem (15) becomes unconstrained and we have:

$$\hat{w} = \hat{w}^o \tag{18}$$

Otherwise, since problem (15) is convex, considering constraint $|[\boldsymbol{w}_k]_m| \geq [\boldsymbol{w}_k^o]_m > 0$ with (17) leads to:

$$\hat{w} = \tag{19}$$
$$\begin{cases} [\boldsymbol{\psi}_{k,n+1}]_m + \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m \leq -[\boldsymbol{w}_k^o]_m - \mu_k\lambda_k \\ \qquad -[\boldsymbol{w}_k^o]_m, \text{ if } -[\boldsymbol{w}_k^o]_m - \mu_k\lambda_k < [\boldsymbol{\psi}_{k,n+1}]_m < 0 \\ -[\boldsymbol{w}_k^o]_m \text{ or } [\boldsymbol{w}_k^o]_m, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m = 0 \\ \qquad [\boldsymbol{w}_k^o]_m, \text{ if } 0 < [\boldsymbol{\psi}_{k,n+1}]_m < [\boldsymbol{w}_k^o]_m + \mu_k\lambda_k \\ [\boldsymbol{\psi}_{k,n+1}]_m - \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m \geq [\boldsymbol{w}_k^o]_m + \mu_k\lambda_k \end{cases}$$

To evaluate the proximal operator (12), several issues have to be addressed.

1. One of the main issues is that we first need to know which of (14), (17) or (19) has to be applied as the proximal operator of (12). We now consider the following two cases: $[\boldsymbol{w}_k^o]_m = 0$ and $[\boldsymbol{w}_k^o]_m > 0$.

- Case A: $[\boldsymbol{w}_k^o]_m = 0$. Since condition $|[\boldsymbol{w}_k]_m| < [\boldsymbol{w}_k^o]_m$ of Case 1 cannot hold, we only consider Case 2. The proximal operator is given by (17) directly.

Let us now consider the second case:

- Case B: $[\boldsymbol{w}_k^o]_m > 0$. Proximal operators (14) and (19) hold simultaneously. We shall choose the solution that minimizes the cost (12). We arrive at the following expression:

$$\hat{w} = \tag{20}$$
$$\begin{cases} [\boldsymbol{\psi}_{k,n+1}]_m + \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m \leq -[\boldsymbol{w}_k^o]_m - \mu_k\lambda_k \\ \qquad -[\boldsymbol{w}_k^o]_m, \text{ if } -[\boldsymbol{w}_k^o]_m - \mu_k\lambda_k < [\boldsymbol{\psi}_{k,n+1}]_m \leq -[\boldsymbol{w}_k^o]_m \\ \qquad [\boldsymbol{\psi}_{k,n+1}]_m, \text{ if } \big|[\boldsymbol{\psi}_{k,n+1}]_m\big| < [\boldsymbol{w}_k^o]_m \\ \qquad [\boldsymbol{w}_k^o]_m, \text{ if } [\boldsymbol{w}_k^o]_m \leq [\boldsymbol{\psi}_{k,n+1}]_m < [\boldsymbol{w}_k^o]_m + \mu_k\lambda_k \\ [\boldsymbol{\psi}_{k,n+1}]_m - \mu_k\lambda_k, \text{ if } [\boldsymbol{\psi}_{k,n+1}]_m \geq [\boldsymbol{w}_k^o]_m + \mu_k\lambda_k \end{cases}$$

2. Another issue is that $\hat{w}$ cannot be evaluated with (17) and (20) since $[\boldsymbol{w}_k^o]_m$ is unknown. To bypass this problem, we follow a strategy adopted in the literature [27] by using $\boldsymbol{\psi}_{\ell,n+1}$ as an approximation of $\boldsymbol{w}_\ell^\star$. An approximation of $[\boldsymbol{w}_k^o]_m$ is then given by $\max_{\ell\in\mathcal{N}_k^-}\{\big|[\boldsymbol{\psi}_{\ell,n+1}]_m\big|\}$.

3. Examining expressions in Case A and Case B, we notice that only the proximal operator (17) in Case A has the capability to drive $[\boldsymbol{w}_k]_m$ to zero and promote sparsity. Condition $[\boldsymbol{w}_k^o]_m = 0$ has to be satisfied to trigger Case A, otherwise Case B is considered. Within the context of online learning with stochastic gradient descent algorithms, due to the existence of gradient noise, the estimates of $[\boldsymbol{w}_k^o]_m$ for zero-valued entries often fluctuate around zero rather than being exact null, so that condition $[\boldsymbol{w}_k^o]_m = 0$ of Case A is seldom satisfied. To promote the sparsity of the estimates, we introduce a small positive threshold value τ instead of zero to make a distinguish between zero-valued and nonzero-valued entries. As a consequence, we arrive at conditions $[\boldsymbol{w}_k^o]_m \leq \tau$ to trigger Case A and $[\boldsymbol{w}_k^o]_m > \tau$ to select Case B. The value of τ needs to be fine-tuned to ensure the performance.

We summarize the proximal operator of $\ell_{\infty,1}$-norm in Algorithm 2.

Algorithm 2: Proximal operator of $\ell_{\infty,1}$-norm

1 **Initialization:** Choose threshold value $\tau > 0$.
2 **Proximal operator:** At each instant $n \geq 0$, for each node k, utilize $\boldsymbol{\psi}_{k,n+1}$ to evaluate $\boldsymbol{w}_{k,n+1}$ in an elementwise manner:
   1. Calculate $[\boldsymbol{w}_k^o]_m$ as the maximal value of $\big|[\boldsymbol{\psi}_{\ell,n+1}]_m\big|$ for all $\ell \in \mathcal{N}_k^-$;
   2. If $[\boldsymbol{w}_k^o]_m \leq \tau$, then calculate $[\boldsymbol{w}_{k,n+1}]_m$ as $\hat{w}^o$ via (17);
   3. If $[\boldsymbol{w}_k^o]_m > \tau$, then calculate $[\boldsymbol{w}_{k,n+1}]_m$ as $\hat{w}$ via (20).

5. CONVERGENCE ANALYSIS

The proximal operator of $\ell_{\infty,1}$-norm can be expressed as:

$$\operatorname{prox}_{\mu_k\lambda_k g}\big(\boldsymbol{\psi}_{k,n+1}\big) = \boldsymbol{\psi}_{k,n+1} - \boldsymbol{\gamma}_{k,n+1}, \tag{21}$$

where $\boldsymbol{\gamma}_{k,n+1}$ is a vector of dimension $L\times 1$. The explicit expression of $\boldsymbol{\gamma}_{k,n+1}$ is omitted here, which can be derived from (17) and (20) in an element-wise manner. Define

$$\tilde{\boldsymbol{w}}_{k,n+1} \triangleq \boldsymbol{w}_{k,n+1} - \boldsymbol{w}_k^\star. \tag{22}$$

By collecting $\boldsymbol{w}_k^\star$, $\boldsymbol{w}_{k,n+1}$, $\tilde{\boldsymbol{w}}_{k,n+1}$ and $\boldsymbol{\gamma}_{k,n+1}$ over the entire network into block column vectors, we obtain quantities $\boldsymbol{w}^\star$, $\boldsymbol{w}_{n+1}$, $\tilde{\boldsymbol{w}}_{n+1}$ and $\boldsymbol{\gamma}_{n+1}$, respectively. To facilitate theoretical analysis, we introduce the following assumptions:

A1 (*Independent Regressors*): The regression vector $\boldsymbol{u}_{k,n}$, generated from a zero-mean random process, is temporally stationary, white (over n) and spatially independent (over k) with covariance matrix $\boldsymbol{R}_{u,k} = \mathbb{E}\{\boldsymbol{u}_{k,n}\boldsymbol{u}_{k,n}^\top\} > 0$.

A2 (*Small step-sizes*): The step-sizes μ_k of the network are small enough, so that terms depending of higher-order powers of the step-sizes can be ignored.

5.1. Mean behavior analysis

Subtracting $\boldsymbol{w}_k^\star$ from (9), using signal model (1) and block notations, and taking expectation under assumption **A1**, we arrive at the mean behavior given by:

$$\mathbb{E}\{\tilde{\boldsymbol{w}}_{n+1}\} = \boldsymbol{B}\mathbb{E}\{\tilde{\boldsymbol{w}}_n\} - \mathbb{E}\{\boldsymbol{\gamma}_{n+1}\}, \tag{23}$$

where

$$\boldsymbol{B} \triangleq \boldsymbol{I} - \boldsymbol{U}\boldsymbol{M} \tag{24}$$

$$\boldsymbol{M} \triangleq \operatorname{diag}\big\{\boldsymbol{R}_{u,1}, \boldsymbol{R}_{u,2}, \cdots, \boldsymbol{R}_{u,N}\big\} \tag{25}$$

$$\mathbb{E}\{\boldsymbol{\gamma}_{n+1}\} = \operatorname{col}\big\{\mathbb{E}\{\boldsymbol{\gamma}_{1,n+1}\}, \cdots, \mathbb{E}\{\boldsymbol{\gamma}_{N,n+1}\}\big\}. \tag{26}$$

We point out that vector $\mathbb{E}\{\boldsymbol{\gamma}_{k,n+1}\}$ is absolutely bounded with $|\mathbb{E}\{\boldsymbol{\gamma}_{k,n+1}\}| \preceq \mu_k\lambda_k \mathbb{1}_L$ at all time instant n. This can be derived from the explicit expression of $\mathbb{E}\{\boldsymbol{\gamma}_{k,n+1}\}$. We provide the following condition on the step-size to ensure the mean stability, without proof due to the limited space.

Theorem 1. *(Mean stability)* Assume data model (1) and assumption **A1** hold. Then for any initial conditions, the distributed networks with proximal multitask diffusion LMS algorithm (9) is stable in the mean, if the step-sizes μ_k of the network satisfies:

$$0 < \mu_k < \frac{2}{\lambda_{\max}\{\boldsymbol{R}_{u,k}\}}, \quad k = 1, \cdots, N, \tag{27}$$

where $\lambda_{\max}\{\cdot\}$ denotes the maximal eigenvalue of its matrix argument.

5.2. Mean-square behavior analysis

Under assumptions **A1**, **A2**, and ignoring terms containing higher-order powers of the step-size, then for any semi-positive definite matrix $\boldsymbol{\Sigma}$ of compatible dimension, the weighted mean-square behavior of $\tilde{\boldsymbol{w}}_{n+1}$ is given by:

$$\begin{aligned}\mathbb{E}\{\|\tilde{\boldsymbol{w}}_{n+1}\|_{\boldsymbol{\sigma}}^2\} = &\mathbb{E}\{\|\tilde{\boldsymbol{w}}_n\|_{\boldsymbol{F}\boldsymbol{\sigma}}^2\} + [\mathrm{vec}\{\boldsymbol{H}\}]^\top \boldsymbol{\sigma} + \mathbb{E}\{\|\boldsymbol{\gamma}_{n+1}\|_{\boldsymbol{\sigma}}^2\} \\ &- 2\mathbb{E}\{\tilde{\boldsymbol{w}}_n^\top \boldsymbol{B}^\top \boldsymbol{\Sigma}\boldsymbol{\gamma}_{n+1}\},\end{aligned} \tag{28}$$

where we use the notations $\mathbb{E}\{\|\tilde{\boldsymbol{w}}_{n+1}\|_{\boldsymbol{\sigma}}^2\}$ and $\mathbb{E}\{\|\tilde{\boldsymbol{w}}_{n+1}\|_{\boldsymbol{\Sigma}}^2\}$ interchangeably with $\|\boldsymbol{x}\|_{\boldsymbol{\Sigma}}^2 \triangleq \boldsymbol{x}^\top \boldsymbol{\Sigma}\boldsymbol{x}$, and we have defined the following quantities:

$$\boldsymbol{\sigma} \triangleq \mathrm{vec}\{\boldsymbol{\Sigma}\} \tag{29}$$

$$\boldsymbol{F} \triangleq \boldsymbol{B}^\top \otimes \boldsymbol{B}^\top \tag{30}$$

$$\boldsymbol{H} \triangleq \boldsymbol{U}\mathrm{diag}\{\sigma_{z,1}^2 \boldsymbol{R}_{u,1}, \cdots, \sigma_{z,N}^2 \boldsymbol{R}_{u,N}\}\boldsymbol{U}^\top. \tag{31}$$

It is noted that (30) holds only for sufficiently small step-sizes. We provide the following condition on the step-size to ensure the mean-square stability, without proof due to the limited space.

Theorem 2. *(Mean-square stability)* Assume data model (1) and assumptions **A1**, **A2** hold. Further assume that approximation (30) is reasonable for sufficiently small step-sizes. Then for any initial conditions, the distributed networks with proximal multitask diffusion LMS algorithm (9) is stable in the mean-square sense, if the step-sizes μ_k of the network are sufficiently small and satisfy (27).

6. SIMULATION RESULTS

In this section, we present simulation results to validate the effectiveness of the proposed algorithm. All curves were obtained by averaging over 100 Monte-Carlo runs.

We considered a nonstationary jointly sparse system identification scenario with $\boldsymbol{w}_k^\star$ varying over time. The evolution of $\boldsymbol{w}_k^\star$ was divided into three stationary stages and two transient stages. During stationary stages, sparse vectors $\boldsymbol{w}_k^\star$ were set to sparsity degree of $1/30$, $5/30$ and $3/30$, respectively. Each nonzero element of $\boldsymbol{w}_k^\star$ was generated independently from standard Gaussian distribution. The transient stages were designed by using linear interpolation over 500 time instants. The regressors $\boldsymbol{u}_{k,n}$ were generated from a zero-mean Gaussian distribution with covariance matrices $\boldsymbol{R}_{u,k} = \sigma_{u,k}^2 \boldsymbol{I}_{30}$ for white inputs, and with $\boldsymbol{R}_{u,k} = \sigma_{u,k}^2 \boldsymbol{R}^\dagger$ for colored inputs, where $\boldsymbol{R}^\dagger$ is an 30×30 Hermite matrix with eigenvalue spread $\frac{\lambda_{\max}\{\boldsymbol{R}^\dagger\}}{\lambda_{\min}\{\boldsymbol{R}^\dagger\}} = 31$, and $\lambda_{\min}\{\cdot\}$ denotes the minimal eigenvalue of its matrix argument. For comparison purpose, non-cooperative diffusion LMS algorithm, sparse diffusion LMS [6] with zero-attracting (ZA) regularizer and reweighted zero-attracting (RZA) regularizer, multitask diffusion LMS with adaptive combiner [27] and jointly sparse multitask diffusion LMS [17] with $\ell_{2,1}$-regularization were taken into consideration. We adopted a uniform step-size 0.01 for all algorithms. For proximal multitask LMS with $\ell_{\infty,1}$-regularization, we set λ_k to 0.08 and τ to 0.1. For the other algorithms, we adjusted the parameters to reach the best performance.

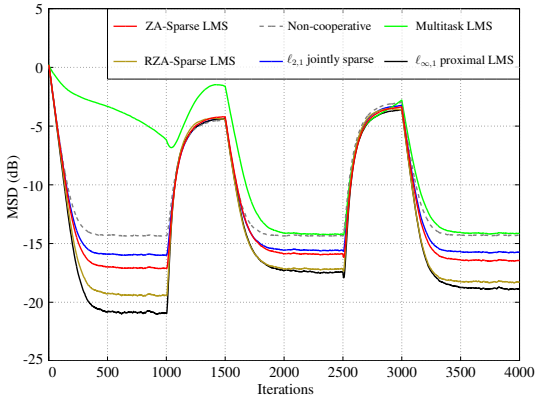

Fig. 1. Simulation results with white inputs.

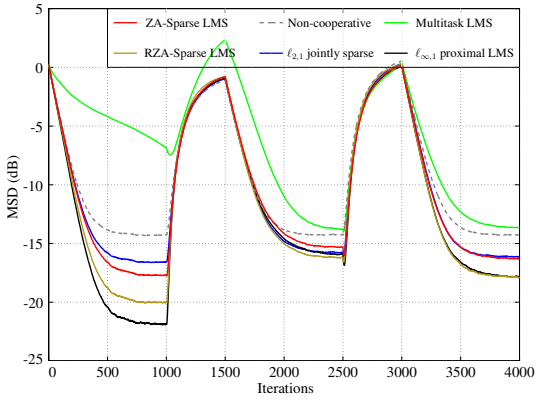

Fig. 2. Simulation results with colored inputs.

The results are illustrated in Fig. 1 for white inputs. We observe that multitask LMS with adaptive combiner is the worst one due to inappropriate cooperation between nodes. Since jointly sparse systems can be regarded as special cases of general sparse systems, sparse diffusion LMS with ZA regularizer and RZA regularizer work better than non-cooperative LMS. Similarly, two jointly sparse multitask LMS algorithms have better performance than the non-cooperative LMS. The proposed proximal multitask LMS with $\ell_{\infty,1}$-regularization has the best performance among all competing algorithms, especially when systems are more sparse. Similar conclusions can be achieved from Fig. 2 for colored inputs.

7. CONCLUSIONS

Many practical problems of interest happen to have the jointly-sparse structure. In this paper, by evaluating the proximal operator of $\ell_{\infty,1}$-norm, we obtained a proximal multitask diffusion LMS algorithm for networks with jointly-sparse structure. We derived conditions to ensure the stabilities in the mean and mean-square sense. Simulation results illustrated the effectiveness of the proposed algorithm.

8. REFERENCES

[1] S.-Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive netowrks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, Dec. 2012.

[2] 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.

[3] F. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, May 2008.

[4] L. Li and J. Chambers, "Distributed adaptive estimation based on the APA algorithm over diffusion networks with changing topology," in *Proc. IEEE SSP*, 2009, pp. 757–760.

[5] 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.

[6] P. Di Lorenzo and A. H. Sayed, "Sparse distributed learning based on diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 61, no. 6, pp. 1419–1433, Mar. 2013.

[7] A. H. Sayed, "Diffusion adaptation over networks," in *Academic Press Libraray in Signal Processing*, R. Chellapa and S. Theodoridis, Eds., vol. 3, pp. 322–454. Elsevier, 2014.

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

[9] A. H. Sayed, *Adaptation, Learning, and Optimization over Networks*, vol. 7, Now Publishers Inc., Hanover, MA, USA, Jul. 2014.

[10] 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.

[11] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Proximal multitask learning over networks with sparsity-inducing coregularization," *IEEE Trans. Signal Process.*, vol. 64, no. 23, pp. 6329–6344, Dec. 2016.

[12] 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, 2017.

[13] 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.

[14] P. D. Lorenzo, S. Barbarossa, and A. H. Sayed, "Distributed spectrum estimation for small cell networks based on sparse diffusion adaptation," *IEEE Signal Process. Letters*, vol. 20, no. 12, pp. 1261–1265, Dec. 2013.

[15] J. Liang, M. Zhang, X. Zeng, and G. Yu, "Distributed dictionary learning for sparse representation in sensor networks," *IEEE Trans. Image Process.*, vol. 23, no. 6, pp. 2528–2541, Jun. 2014.

[16] Y. Gu and M. Wang, "Learning distributed jointly sparse systems by collaborative LMS," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2014, pp. 7228–7232.

[17] C. Li, S. Huang, Y. Liu, and Z. Zhang, "Distributed jointly sparse multitask learning over networks," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 151–164, Jan. 2018.

[18] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[19] W. M. Wee and I. Yamada, "A proximal splitting approach to regularized distributed adaptive estimation in diffusion networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2013, pp. 5420–5424.

[20] S. Vlaski and A. H. Sayed, "Proximal diffusion for stochastic costs with non-differentiable regularizers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2015, pp. 3352–3356.

[21] S. Vlaski, L. Vandenberghe, and A. H. Sayed, "Diffusion stochastic optimization with non-smooth regularizers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2016, pp. 4149–4153.

[22] J. Huang and T. Zhang, "The benefit of group sparsity," *The Annals of Statistics*, vol. 38, no. 4, pp. 1978–2004, Aug. 2010.

[23] S. N. Negahban and M. J. Wainwright, "Simultaneous support recovery in high dimensions: Benefits and perils of block ℓ_1/ℓ_∞-regularization," *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3841–3863, Jun. 2011.

[24] J. A. Tropp, "Algorithms for simultaneous sparse approximation. Part II: Convex relaxation," *Signal Process.*, vol. 86, no. 3, pp. 589 – 602, 2006.

[25] D. P. Bertsekas, "Incremental proximal methods for large scale convex optimization," *Math. Program.*, vol. 129, no. 2, pp. 163, Jun. 2011.

[26] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[27] 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.