Distributed Inference over Multitask Graphs under Smoothness

Roula Nassif, Stefan Vlaski, Ali H. Sayed
Institute of Electrical Engineering, EPFL, Switzerland

***Abstract*—This paper formulates a multitask optimization problem where agents in the network have individual objectives to meet, or individual parameter vectors to estimate, subject to a smoothness condition over the graph. The smoothness requirement softens the transition in the tasks among adjacent nodes and allows incorporating information about the graph structure into the solution of the inference problem. A diffusion strategy is devised that responds to streaming data and employs stochastic approximations in place of actual gradient vectors, which are generally unavailable. We show, under conditions on the step-size parameter, that the adaptive strategy induces a contraction mapping and leads to small estimation errors on the order of the small step-size. A graph spectral filtering interpretation is provided for the optimization framework.**

***Index Terms*—Multitask inference, diffusion strategy, graph Laplacian regularization, gradient noise, spectral filtering.**

I. Introduction

Distributed inference allows a collection of interconnected agents to perform parameter estimation tasks from streaming data by relying solely on local computations and interactions with immediate neighbors. Most prior literature focuses on single-task problems, where agents with separable objective functions need to agree on a common parameter vector corresponding to the minimizer of an aggregate sum of individual costs [1]–[5]. Many network applications require more complex models and flexible algorithms than single-task implementations since their agents may need to estimate and track multiple objectives simultaneously [6]–[11]. Networks of this kind are referred to as multitask networks. Although agents may generally have distinct though related tasks to perform, they may still be able to capitalize on inductive transfer between them to improve their performance [8]–[10].

In this work, we consider multitask estimation problems where each agent in the network seeks to minimize an individual cost expressed as the expectation of some loss function. The minimizers of the individual costs are assumed to vary smoothly on the topology captured by the graph Laplacian. The smoothness property softens the transition in the tasks among adjacent nodes and allows incorporating information about the graph structure into the solution of the inference problem. We formulate the estimation problem as the minimization of the aggregate sum of individual costs regularized by a term that enforces smoothness. A diffusion strategy is devised that responds to streaming data and employs stochastic approximations in place of actual gradient vectors, which are generally unavailable. By imposing a Gaussian probabilistic prior on the minimizers, we show that for mean-square-error networks, solving the regularized optimization problem leads to finding a maximum a posteriori (MAP) estimate of the unknown parameter vectors. We show, under conditions on the step-size learning parameter μ that the adaptive strategy induces a contraction mapping and that, despite gradient noise, it is able to converge in the mean-square-error sense within $O(\mu)$ from the solution of the regularized problem, for sufficiently small μ. In a second step, we give the graph-based regularized optimization problem an interpretation in terms of graph spectral filtering [12]–[14] and illustrate the influence of the regularization strength on the spectral content of the network output.

The work of A. H. Sayed was supported in part by NSF grants CCF-1524250 and ECCS-1407712. Emails: {roula.nassif, stefan.vlaski, ali.sayed}@epfl.ch

II. Distributed inference under smoothness

A. Problem formulation and adaptive strategy

Consider a connected network (or graph) $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, A\}$, where $\mathcal{N}$ is a set of N nodes, $\mathcal{E}$ is a set of edges connecting nodes with particular relations, and A is a symmetric weighted adjacency matrix. If there is an edge connecting nodes k and ℓ, then $[A]_{k\ell} = a_{k\ell} > 0$ reflects the strength of the relation between k and ℓ; otherwise, $[A]_{k\ell} = 0$. We introduce the graph Laplacian, which is a differential operator defined as $L = D - A$, where the degree matrix D is a diagonal matrix with k-th entry $[D]_{kk} = \sum_{\ell=1}^{N} a_{k\ell}$. Since L is symmetric positive semi-definite, it possesses a complete set of orthonormal eigenvectors. We denote them by $\{v_1, \ldots, v_N\}$. For convenience, we order the set of real, non-negative eigenvalues of L as $0 = \lambda_1 < \lambda_2 \leq \ldots \leq \lambda_N = \lambda_{\max}(L)$, where, since the graph is connected, there is only one zero eigenvalue with corresponding eigenvector $v_1 = \frac{1}{\sqrt{N}}\mathbb{1}_N$ [15]. Thus, the Laplacian can be decomposed as $L = V\Lambda V^{\top}$ where $\Lambda = \text{diag}\{\lambda_1, \ldots, \lambda_N\}$ and $V = [v_1, \ldots, v_N]$.

Let $w_k \in \mathbb{R}^M$ denote some parameter vector at node k and let $\mathcal{w} = \text{col}\{w_1, \ldots, w_N\}$ denote the collection of parameter vectors from across the network. We associate with each agent k a risk function $J_k(w_k) : \mathbb{R}^M \to \mathbb{R}$ assumed to be strongly convex. In most learning and adaptation problems, the risk function is expressed as the expectation of a loss function $Q_k(\cdot)$ and is written as $J_k(w_k) = \mathbb{E}\, Q_k(w_k; \boldsymbol{x})$, where $\boldsymbol{x}$ denotes the random data. The expectation is computed over the distribution of this data (note that, in our notation, we use boldface letters for random quantities and normal letters for

978-1-5386-3512-4/18/$31.00 ©2018 IEEE

deterministic quantities). We denote the unique minimizer of $J_k(w_k)$ by w_k^o. In many situations, there is prior information available about $\mathcal{W}^o = \text{col}\{w_1^o, \ldots, w_N^o\}$. In the current work, the prior belief we want to enforce is that the target signal $\mathcal{W}^o$ is smooth with respect to the underlying weighted graph. References [8]–[10] provide variations for such problems for the special case of mean-square-error costs. Let $\mathcal{L} = L \otimes I_M$ where the symbol $\otimes$ refers to the Kronecker product operation. The smoothness of $\mathcal{W}$ can be measured in terms of a quadratic form of the graph Laplacian [16]:

$$S(\mathcal{W}) = \mathcal{W}^\top \mathcal{L} \mathcal{W} = \frac{1}{2} \sum_{k=1}^{N} \sum_{\ell \in \mathcal{N}_k} a_{k\ell} \|w_k - w_\ell\|^2, \qquad (1)$$

where $\mathcal{N}_k$ is the set of neighbors of k, i.e., the set of nodes connected to agent k by an edge. The smaller $S(\mathcal{W})$ is, the smoother the signal $\mathcal{W}$ on the graph is. Intuitively, given that the weights are non-negative, $S(\mathcal{W})$ shows that $\mathcal{W}$ is considered to be smooth if nodes with a large $a_{k\ell}$ on the edge connecting them have similar weight values $\{w_k, w_\ell\}$. Our objective is to devise a strategy that solves the following regularized problem:

$$\mathcal{W}_\eta^o = \arg\min_{\mathcal{W}} J^{\text{glob}}(\mathcal{W}) = \sum_{k=1}^{N} J_k(w_k) + \frac{\eta}{2} \mathcal{W}^\top \mathcal{L} \mathcal{W}, \qquad (2)$$

in a distributed manner where each agent is interested in estimating the k-th sub-vector of $\mathcal{W}_\eta^o = \text{col}\{w_{1,\eta}^o, \ldots, w_{N,\eta}^o\}$. The tuning parameter $\eta \geq 0$ controls the trade off between the two components of the objective function. We are particularly interested in solving the problem in the stochastic setting when the distribution of the data $\boldsymbol{x}$ is generally unknown. This means that the risks $J_k(w_k)$ and their gradients $\nabla_{w_k} J_k(w_k)$ are unknown. As such, approximate gradient vectors need to be employed. A common construction in the stochastic approximation theory is to employ the following approximation at iteration i:

$$\widehat{\nabla_{w_k} J_k}(w_k) = \nabla_{w_k} Q_k(w_k; \boldsymbol{x}_i), \qquad (3)$$

where $\boldsymbol{x}_i$ represents the data observed at iteration i. The difference between the true gradient and its approximation is called the gradient noise $\boldsymbol{s}_{k,i}(\cdot)$:

$$\boldsymbol{s}_{k,i}(w_k) \triangleq \nabla_{w_k} J_k(w_k) - \widehat{\nabla_{w_k} J_k}(w_k). \qquad (4)$$

Each agent can employ a stochastic gradient descent update to estimate $w_{k,\eta}^o$:

$$\begin{aligned} \boldsymbol{w}_{k,i} =& \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla_{w_k} J_k}(\boldsymbol{w}_{k,i-1}) \\ & - \mu\eta \sum_{\ell \in \mathcal{N}_k} a_{k\ell} (\boldsymbol{w}_{k,i-1} - \boldsymbol{w}_{\ell,i-1}), \end{aligned} \qquad (5)$$

where $\mu > 0$ is a small step-size parameter. In this implementation, each agent k collects from its neighbors the estimates $\boldsymbol{w}_{\ell,i-1}$, and performs a stochastic-gradient descent update on:

$$\bar{J}_{k,i-1}(w_k) \triangleq J_k(w_k) + \frac{\eta}{2} \sum_{\ell \in \mathcal{N}_k} a_{k\ell} \|w_k - \boldsymbol{w}_{\ell,i-1}\|^2. \qquad (6)$$

By introducing an auxiliary variable $\boldsymbol{\psi}_{k,i}$, strategy (5) can be implemented in an incremental manner:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu \widehat{\nabla_{w_k} J_k}(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} = \boldsymbol{\psi}_{k,i} - \mu\eta \sum_{\ell \in \mathcal{N}_k} a_{k\ell} (\boldsymbol{\psi}_{k,i} - \boldsymbol{\psi}_{\ell,i}), \end{cases} \qquad (7)$$

where we replaced $(\boldsymbol{w}_{k,i-1} - \boldsymbol{w}_{\ell,i-1})$ in the second step by the difference $(\boldsymbol{\psi}_{k,i} - \boldsymbol{\psi}_{\ell,i})$ since we expect $\boldsymbol{\psi}_{k,i}$ to be an improved estimate compared to $\boldsymbol{w}_{k,i-1}$.

B. Theoretical motivation for the optimization framework

In the following, we explain that solving (2) is equivalent to finding a maximum a posteriori (MAP) estimate for $\boldsymbol{\mathcal{W}}$ in the case of mean-square-error (MSE) networks [4], [8] where each agent is subjected to streaming data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ that are assumed to satisfy a linear regression model:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w_k^o + \boldsymbol{v}_k(i), \qquad k = 1, \ldots, N, \qquad (8)$$

for some unknown $M \times 1$ vector w_k^o with $\boldsymbol{v}_k(i)$ a measurement noise. A mean-square-error cost is associated with agent k:

$$J_k(w_k) = \frac{1}{2} \mathbb{E} |\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w_k|^2, \qquad k = 1, \ldots, N. \qquad (9)$$

The processes $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}, \boldsymbol{v}_k(i)\}$ are assumed to represent zero-mean jointly wide-sense stationary random processes satisfying: i) $\mathbb{E} \boldsymbol{u}_{k,i}^\top \boldsymbol{u}_{\ell,j} = R_{u,k} \delta_{k,\ell} \delta_{i,j}$ where $R_{u,k} > 0$ and the Kronecker delta $\delta_{m,n} = 1$ if $m = n$ and zero otherwise; ii) $\mathbb{E} \boldsymbol{v}_k(i) \boldsymbol{v}_\ell(j) = \sigma_{v,k}^2 \delta_{k,\ell} \delta_{i,j}$; iii) the regression and noise processes $\{\boldsymbol{u}_{\ell,j}, \boldsymbol{v}_k(i)\}$ are independent of each other.

Lemma 1. *If the network parameter vector is an intrinsic Gaussian Markov Random field $\boldsymbol{\mathcal{W}} \sim \mathcal{N}(0, \mathcal{L})$, i.e., its density is [17]:*

$$f(\boldsymbol{\mathcal{W}}) = (2\pi)^{-M(N-1)/2} \, (|\mathcal{L}|^*)^{1/2} \, e^{-\frac{1}{2} \boldsymbol{\mathcal{W}}^\top \mathcal{L} \boldsymbol{\mathcal{W}}}, \qquad (10)$$

with $|\cdot|^*$ denoting the pseudo-determinant of a matrix (i.e, the product of all its nonzero eigenvalues), and if the noise process is Gaussian $\boldsymbol{v}_k(i) \sim \mathcal{N}(0, \sigma_{v,k}^2)$ independent over space and time and identically distributed, then problem (2) *is a MAP estimator for $\boldsymbol{\mathcal{W}}$ conditioned on $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$.*

Proof. This is an extension of a well-known result for MAP estimation under single agents. Appendix A provides a proof that establishes the above extension to the multi-agent case in terms of the pseudo-determinant of the Laplacian matrix. □

III. Stochastic performance analysis

Before examining the behavior of algorithm (7), we introduce the following assumptions on the risks $\{J_k(w_k)\}$ and on the gradient noise processes $\{\boldsymbol{s}_{k,i}(\cdot)\}$ defined in (4). As explained in [4], [5], these conditions are satisfied by many objective functions of interest in learning and adaptation such as quadratic and logistic risks. Besides, regularization is a common technique to ensure strong convexity.

Assumption 1. (Strong convexity) *It is assumed that the individual costs $J_k(w_k)$ are each twice differentiable and strongly*

convex such that the Hessian matrix function $H_k(w_k) \triangleq \nabla^2_{w_k} J_k(w_k)$ *is uniformly bounded from below and above:*

$$0 < \lambda_{k,\min} I_M \leq H_k(w_k) \leq \lambda_{k,\max} I_M, \tag{11}$$

where $\lambda_{k,\min} > 0$ *for* $k = 1, \ldots, N$.

Assumption 2. (Gradient noise process) *For each agent* k*, the gradient noise process defined in* (4) *satisfies:*

$$\mathbb{E}[\boldsymbol{s}_{k,i}(\boldsymbol{w}_k)|\boldsymbol{\mathcal{F}}_{i-1}] = 0, \tag{12}$$

$$\mathbb{E}[\|\boldsymbol{s}_{k,i}(\boldsymbol{w}_k)\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \leq \beta_k^2 \|\boldsymbol{w}_k\|^2 + \sigma_{s,k}^2, \tag{13}$$

for some $\beta_k^2 \geq 0$*,* $\sigma_{s,k}^2 \geq 0$*, and where* $\boldsymbol{\mathcal{F}}_{i-1}$ *denotes the filtration generated by the random processes* $\{\boldsymbol{w}_{\ell,j}\}$ *for all* $\ell = 1, \ldots, N$ *and* $j \leq i-1$.

To examine the convergence properties of (7), we extend the energy analysis framework of [3] to handle multitask distributed optimization. We first show that algorithm (7), in the absence of gradient noise, converges and has a unique fixed-point. Then, we analyze the distance between this point and the vectors $w_{k,\eta}^o$ and $\boldsymbol{w}_{k,i}$ in the mean-square-sense.

A. Existence and uniqueness of fixed-point

Let us introduce the network block vector $\boldsymbol{\mathcal{W}}_i = \text{col}\{\boldsymbol{w}_{1,i}, \ldots, \boldsymbol{w}_{N,i}\}$. At each iteration, we can view (7) as a mapping from $\boldsymbol{\mathcal{W}}_{i-1}$ to $\boldsymbol{\mathcal{W}}_i$:

$$\boldsymbol{\mathcal{W}}_i = (I_{MN} - \mu\eta\mathcal{L}) \left(\boldsymbol{\mathcal{W}}_{i-1} - \mu \,\text{col}\left\{ \widehat{\nabla_{w_k} J_k}(\boldsymbol{w}_{k,i-1}) \right\}_{k=1}^N \right) \tag{14}$$

Without gradient noise, this relation reduces to:

$$\mathcal{W}_i = (I_{MN} - \mu\eta\mathcal{L}) \left(\mathcal{W}_{i-1} - \mu \,\text{col}\left\{ \nabla_{w_k} J_k(w_{k,i-1}) \right\}_{k=1}^N \right). \tag{15}$$

Lemma 2. (Contractive mapping) *The deterministic mapping defined in* (15) *is Lipchitz continuous with constant* $\gamma = \max_{1\leq k\leq N}\{\gamma_k\}$ *where* $\gamma_k \triangleq \max\{|1-\mu\lambda_{k,\min}|, |1-\mu\lambda_{k,\max}|\}$*. This mapping is contractive when* μ *and* η *satisfy:*

$$0 \leq \mu\eta \leq \frac{2}{\lambda_{\max}(L)}, \quad \text{and} \quad 0 < \mu < \min_{1\leq k\leq N} \left\{ \frac{2}{\lambda_{k,\max}} \right\}. \tag{16}$$

Proof. Proof omitted due to space limitations. □

It then follows from Banach's fixed point theorem [18, pp. 299–303] that this mapping converges to a unique fixed point $\mathcal{W}_\infty$ at an exponential rate given by γ.

B. Fixed point analysis

Now we analyze how far this fixed point $\mathcal{W}_\infty$ is from the desired solution $\mathcal{W}_\eta^o$. Since $\mathcal{W}_\infty$ is the fixed point for the strategy (7) in the absence of gradient noise, we have at convergence:

$$\mathcal{W}_\infty = (I_{MN} - \mu\eta\mathcal{L}) \left(\mathcal{W}_\infty - \mu \,\text{col}\left\{ \nabla_{w_k} J_k(w_{k,\infty}) \right\}_{k=1}^N \right). \tag{17}$$

Let $\widetilde{w}_{k,\infty} \triangleq w_{k,\eta}^o - w_{k,\infty}$ and $\widetilde{\mathcal{W}}_\infty \triangleq \mathcal{W}_\eta^o - \mathcal{W}_\infty$. Using the mean-value theorem [19, pp. 24], we can write:

$$\nabla_{w_k} J_k(w_{k,\infty}) = \nabla_{w_k} J_k(w_{k,\eta}^o) - H_{k,\infty} \widetilde{w}_{k,\infty}, \tag{18}$$

where

$$H_{k,\infty} = \int_0^1 \nabla^2_{w_k} J_k(w_{k,\eta}^o - t\widetilde{w}_{k,\infty}) dt.$$

Subtracting the vector $(I_{MN} - \mu\eta\mathcal{L})\mathcal{W}_\eta^o$ from both sides of recursion (17) and using relation (18), we obtain:

$$\begin{aligned} \widetilde{\mathcal{W}}_\infty =& (I_{MN} - \mu\eta\mathcal{L})(I_{MN} - \mu\mathcal{H}_\infty)\widetilde{\mathcal{W}}_\infty + \mu\eta\mathcal{L}\mathcal{W}_\eta^o \\ &+ \mu(I_{MN} - \mu\eta\mathcal{L})\text{col}\{\nabla_{w_k} J_k(w_{k,\eta}^o)\}_{k=1}^N, \end{aligned} \tag{19}$$

where $\mathcal{H}_\infty \triangleq \text{diag}\{H_{1,\infty}, \ldots, H_{N,\infty}\}$. From the optimality condition of (2), we have:

$$\text{col}\{\nabla_{w_k} J_k(w_{k,\eta}^o)\}_{k=1}^N = -\eta\mathcal{L}\mathcal{W}_\eta^o, \tag{20}$$

and recursion (19) can be written alternatively as:

$$\widetilde{\mathcal{W}}_\infty = (I_{MN} - \mu\eta\mathcal{L})(I_{MN} - \mu\mathcal{H}_\infty)\widetilde{\mathcal{W}}_\infty + \mu^2\eta^2\mathcal{L}^2\mathcal{W}_\eta^o. \tag{21}$$

From (21), $\widetilde{\mathcal{W}}_\infty$ is zero when $\eta = 0$ and when $w_k^o = w_\ell^o \;\forall\; k, \ell$ since in the latter case $\mathcal{W}_\eta^o = \mathcal{W}^o$ and $\mathcal{L}^2\mathcal{W}_\eta^o = 0$.

Theorem 1. *Under condition* (16) *and for small* μ*, the steady-state bias* $\widetilde{\mathcal{W}}_\infty = \mathcal{W}_\eta^o - \mathcal{W}_\infty$ *of the mapping* (15) *satisfies:*

$$\|\mathcal{W}_\eta^o - \mathcal{W}_\infty\| \leq \frac{O(\mu\eta^2)}{(O(1) + O(\eta))^2}. \tag{22}$$

Proof. Proof omitted due to space limitations. □

C. Evolution of the stochastic recursion

We now examine how close the stochastic algorithm (7) approaches $\mathcal{W}_\eta^o$. First, we introduce the mean-square perturbation vector at time i relative to w_∞:

$$\text{MSP}_i \triangleq \text{col}\left\{\mathbb{E}\|w_{k,\infty} - \boldsymbol{w}_{k,i}\|^2\right\}_{k=1}^N. \tag{23}$$

Theorem 2. *By choosing* $\mu\eta$ *such that* $I - \mu\eta L$ *has positive diagonal entries, i.e.,* $\mu\eta \leq \min_{1\leq k\leq N}\left\{(\sum_{\ell=1}^N [A]_{k\ell})^{-1}\right\}$*, the* MSP *at time* i *can be recursively bounded as:*

$$\text{MSP}_i \preceq (I_N - \mu\eta L)\, G\, \text{MSP}_{i-1} + \mu^2 d, \tag{24}$$

where G *is a diagonal matrix with elements* $\gamma_k^2 + 3\mu^2\beta_k^2$ *and* $d = O(1) + O(\mu^2\eta^4)(O(1) + O(\eta))^{-4}$*. A sufficient condition for stability of the above recursion is:*

$$0 < \mu < \min_{1\leq k\leq N} \left\{ \frac{2\lambda_{k,\min}}{\lambda_{k,\min}^2 + 3\beta_k^2}, \frac{2\lambda_{k,\max}}{\lambda_{k,\max}^2 + 3\beta_k^2} \right\}. \tag{25}$$

It follows that

$$\|\limsup_{i\to\infty} \text{MSP}_i\|_\infty = O(\mu), \tag{26}$$

and the steady-state $\text{MSD} \triangleq \limsup_{i\to\infty} \frac{1}{N}\mathbb{E}\|\mathcal{W}_\eta^o - \boldsymbol{\mathcal{W}}_i\|^2$ *is:*

$$\text{MSD} = O(\mu) + \frac{O(\mu^2\eta^4)}{(O(1) + O(\eta))^4} = O(\mu). \tag{27}$$

Proof. Proof omitted due to space limitations. □

IV. Graph filter interpretation

In the following, we show that for MSE networks with uniform Hessian matrices, i.e., $\nabla^2_{w_k} J_k(w_k) = R_{u,k} = R_u$ $\forall k$, the solution $\mathcal{W}^o_\eta$ of problem (2) can be interpreted as the output of a smooth graph filter applied to the graph signal $\mathcal{W}^o$.

Before proceeding, we briefly review the notion of graph frequencies, Graph Fourier transform, and graph filtering [12]–[14]. Consider the connected graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}, A\}$ equipped with a Laplacian matrix L, which can be decomposed as $L = V\Lambda V^\top$. A graph signal supported on the set $\mathcal{N}$ is defined as a vector $x \in \mathbb{R}^N$ whose k-th component $x_k \in \mathbb{R}$ represents the value of the signal at the k-th node. By analogy to the classical Fourier analysis, the eigenvectors of the Laplacian are used to define a graph Fourier basis V and the eigenvalues are considered as the graph frequencies [12, pp. 86–88]. The Graph Fourier Transform (GFT) transforms a graph signal x into the graph frequency domain according to $\overline{x} = V^\top x$ where $\{\overline{x}_1, \ldots, \overline{x}_N\}$ are called the spectrum of x. The inverse GFT is given by $x = V\overline{x}$ which reconstructs the signal from its spectrum. A graph filter Φ is an operator that acts upon a graph signal x by amplifying or attenuating its spectrum as: $\Phi x = \sum_{m=1}^N \Phi(\lambda_m)\overline{x}_m v_m$. The frequency response of the filter $\Phi(\lambda)$ controls how much Φ amplifies the signal spectrum. Low frequencies correspond to small eigenvalues, and low-pass or smooth filters correspond to decaying functions $\Phi(\lambda)$.

Let us consider the MSE network presented in section II-B where we assume $R_{u,k} = R_u$ $\forall k$. Since we are dealing with vectors $w_k \in \mathbb{R}^M$ instead of scalars $x_k \in \mathbb{R}$, the graph transformation $\overline{x} = V^\top x$ becomes $\overline{\mathcal{W}} = (V^\top \otimes I_M)\mathcal{W}$.

Lemma 3. *For* MSE *networks with uniform covariance matrices, it holds that the m-th subvector corresponding to the m-th eigenvalue (or graph frequency) of $\overline{\mathcal{W}}^o_\eta \triangleq (V^\top \otimes I_M)\mathcal{W}^o_\eta$ is given by:*

$$\left[\overline{\mathcal{W}}^o_\eta\right]_m = (R_u + \eta\lambda_m I_M)^{-1} R_u \left[\overline{\mathcal{W}}^o\right]_m, \quad m = 1, \ldots, N. \tag{28}$$

where $\overline{\mathcal{W}}^o \triangleq (V^\top \otimes I_M)\mathcal{W}^o$. Moreover,

$$\left\| \left[\overline{\mathcal{W}}^o_\eta\right]_m \right\|_2 \leq \frac{1}{1 + \eta \frac{\lambda_m}{\lambda_{\max}(R_u)}} \left\| \left[\overline{\mathcal{W}}^o\right]_m \right\|_2, \quad m = 1, \ldots, N. \tag{29}$$

where the equality holds for $m = 1$.

Proof. Proof omitted due to space limitations. □

If $\eta = 0$, we are in the case of an all-pass graph filter since the frequency content of the output signal $\mathcal{W}^o_\eta$ is the same as the frequency content of the input signal $\mathcal{W}^o$.

For $\eta > 0$, we are in the case of a low-pass graph filter since the norm of the m-th frequency content of the output signal $\mathcal{W}^o_\eta$, namely, $\| \left[\overline{\mathcal{W}}^o_\eta\right]_m \|_2$, is less than or equal to the norm of the m-th frequency content of the input signal $\mathcal{W}^o$, namely, $\| \left[\overline{\mathcal{W}}^o\right]_m \|_2$. For fixed η, as m increases, the ratio in (29) decreases. This validates the low-pass filter interpretation. The regularization parameter η controls the sharpness of the low-pass filter. For sufficiently large η, $[\overline{\mathcal{W}}^o_\eta]_m$ will be equal to $[\overline{\mathcal{W}}^o]_1$ if $m = 1$ and approaching zero otherwise. In this case,

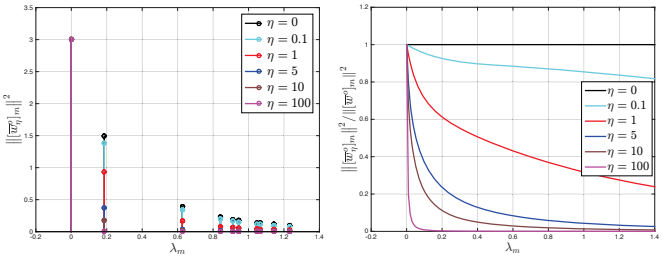

Fig. 1. MSE networks ($R_{u,k} = R_u$ $\forall k$). *(Left)* Graph frequency content of $\mathcal{W}^o_\eta$. *(Right)* The ratio $\|[\overline{\mathcal{W}}^o_\eta]_m\|^2 / \|[\overline{\mathcal{W}}^o]_m\|^2$ for $\lambda_m \in [0, 1.4]$ from (28).

$\mathcal{W}^o_\eta = (V \otimes I_M)\overline{\mathcal{W}}^o_\eta = (v_1 \otimes I_M)[\overline{\mathcal{W}}^o]_1 = \mathbb{1}_N \otimes \left(\frac{1}{N}\sum_{k=1}^N w^o_k\right)$ and all nodes converge to $\frac{1}{N}\sum_{k=1}^N w^o_k$, the minimizer of $\sum_{k=1}^N J_k(w_k)$ subject to $w_k = w_\ell$ $\forall k, \ell$.

V. Simulation results

A. MSE networks

To illustrate the low-pass filter interpretation, we consider a circular d-regular network of $N = 20$ nodes and $M = 3$, generated by taking a circle graph and connecting each node to its $d = 3$ neighbors to each side on the circle. We set $a_{k\ell} = 1/7$ if $\ell \in \mathcal{N}_k$ and 0 otherwise. In this case, the Laplacian matrix has 10 distinct eigenvalues. We generate $\mathcal{W}^o = \text{col}\{w^o_1, \ldots, w^o_N\}$ directly in the spectral domain according to $\overline{w}^o_m = \text{col}\{e^{-\tau_j \lambda_m}\}_{j=1}^3$ where $\tau_j = j$. The matrix R_u is diagonal with entries generated from the uniform distribution $\mathcal{U}(0.5, 1.5)$. We illustrate in Fig. 1 (left) the squared ℓ_2-norm of $\left[\overline{\mathcal{W}}^o_\eta\right]_m$ for different values of η. In order to visualize the frequency response of the graph filter, we plot in Fig. 1 (right) the ratio $\frac{\|[\overline{\mathcal{W}}^o_\eta]_m\|^2}{\|[\overline{\mathcal{W}}^o]_m\|^2}$ from (28) for $\lambda_m \in [0, 1.4]$.

B. Pattern classification application

Let $\boldsymbol{\gamma}_k = \pm 1$ denote a class binary random variable and $\boldsymbol{h}_k \in \mathbb{R}^M$ denote the corresponding feature vector. During the training phase, at each instant i, agent k receives $\{\boldsymbol{\gamma}_k(i), \boldsymbol{h}_{k,i}\}$. The feature vector $\boldsymbol{h}_{k,i} \in \mathbb{R}^2$ is generated according to

$$\boldsymbol{h}_{k,i} = \boldsymbol{\gamma}_k(i) \cdot r \cdot \text{col}\{\cos(\theta_k), \sin(\theta_k)\} + \boldsymbol{v}_{k,i},$$

where $\boldsymbol{v}_{k,i}$ is drawn from $\mathcal{N}(0, \sigma^2_{v,k} I_2)$ and where $\boldsymbol{\gamma}_k(i)$ is Bernoulli distributed with $p(\boldsymbol{\gamma}_k(i) = +1) = 0.5$. We set $r = \sqrt{2}$ and $\theta_k = \frac{\pi}{6} + \frac{k-1}{N-1} \cdot \frac{7\pi}{6}$. Using logistic regression [4], [20], [21], we are interested in finding a decision rule, parameterized by w^o_k, such that $\widehat{\boldsymbol{\gamma}}_k(i) = \text{sign}(\boldsymbol{h}^\top_{k,i} w^o_k)$ and

$$w^o_k \triangleq \arg\min_{w_k} \mathbb{E} \ln\left(1 + e^{-\boldsymbol{\gamma}_k(i)\boldsymbol{h}^\top_{k,i} w_k}\right) + \rho\|w_k\|^2. \tag{30}$$

We consider a network of 50 nodes where node k is connected to nodes $k-1$ and $k+1$ if $k \neq \{1, 50\}$, node 1 is connected to node 2, and node 50 is connected to node 49. The weight over a link is set to $1/3$. We set $\mu = 10^{-3}$ and $\rho = 0.025$. The noise variances $\sigma^2_{v,k}$ are generated from $\mathcal{U}(0, 2)$. We run strategy (7) for different values of η as shown in Fig. 2 (left) ($\eta = 0$ corresponds to the non-cooperative scenario). At each iteration, classification accuracy is evaluated on a separate

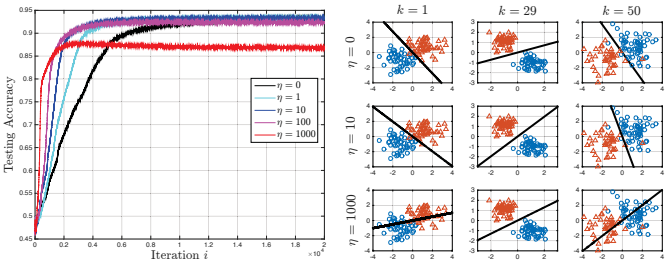

Fig. 2. Pattern classification. *(Left)* Classification accuracy. *(Right)* One realization (after convergence) of the classifier. Blue and red circles correspond to feature vectors of 100 test samples at each agent k. Rows: first ($\eta = 0$), second ($\eta = 10$), and third ($\eta = 10^3$). Columns: first ($k = 1$, $\sigma^2_{v,1} = 0.93$), second ($k = 29$, $\sigma^2_{v,29} = 0.25$), and third ($k = 50$, $\sigma^2_{v,50} = 1.3$).

testing set. As η increases, the convergence rate improves. However, a large value of η yields a deterioration in the accuracy since in this case all agents converge approximately to the same classifier as illustrated in Fig. 2 (right).

APPENDIX A
PROOF OF LEMMA 1

Let

$$\boldsymbol{d}(i) \triangleq \text{col}\{\boldsymbol{d}_1(i), \ldots, \boldsymbol{d}_N(i)\}, \text{ and } \boldsymbol{U}_i \triangleq \text{diag}\{\boldsymbol{u}_{1,i}, \ldots, \boldsymbol{u}_{N,i}\}.$$

From (8), we have:

$$\bar{d} = \mathbb{E}[\boldsymbol{d}(i)|\boldsymbol{U}_i, \boldsymbol{\mathcal{W}}] = \boldsymbol{U}_i \boldsymbol{\mathcal{W}}, \tag{31}$$

$$\mathbb{E}[(\boldsymbol{d}(i) - \bar{d})(\boldsymbol{d}(i) - \bar{d})^\top | \boldsymbol{U}_i, \boldsymbol{\mathcal{W}}] = R_v, \tag{32}$$

where $R_v \triangleq \text{diag}\{\sigma^2_{v,1}, \ldots, \sigma^2_{v,N}\}$. Thus, we can write:

$$f(\boldsymbol{d}(i)|\boldsymbol{U}_i, \boldsymbol{\mathcal{W}}) = \frac{e^{-\frac{1}{2}(\boldsymbol{d}(i) - \boldsymbol{U}_i\boldsymbol{\mathcal{W}})^\top R_v^{-1}(\boldsymbol{d}(i) - \boldsymbol{U}_i\boldsymbol{\mathcal{W}})}}{\sqrt{(2\pi)^N |R_v|}}. \tag{33}$$

Applying Bayes rule

$$f(\boldsymbol{\mathcal{W}}|\boldsymbol{d}(i), \boldsymbol{U}_i) = \frac{f(\boldsymbol{d}(i)|\boldsymbol{U}_i, \boldsymbol{\mathcal{W}}) f(\boldsymbol{\mathcal{W}}|\boldsymbol{U}_i)}{f(\boldsymbol{d}(i)|\boldsymbol{U}_i)},$$

the MAP estimator is given by:

$$\begin{aligned}
\boldsymbol{\mathcal{W}}_{\text{MAP}} &= \arg\max_{\boldsymbol{\mathcal{W}}} f(\boldsymbol{\mathcal{W}}|\boldsymbol{d}(i), \boldsymbol{U}_i) \\
&= \arg\max_{\boldsymbol{\mathcal{W}}} \log(f(\boldsymbol{\mathcal{W}}|\boldsymbol{d}(i), \boldsymbol{U}_i)) \\
&= \arg\min_{\boldsymbol{\mathcal{W}}} - \log(f(\boldsymbol{d}(i)|\boldsymbol{U}_i, \boldsymbol{\mathcal{W}})) - \log(f(\boldsymbol{\mathcal{W}}|\boldsymbol{U}_i)) \\
&= \arg\min_{\boldsymbol{\mathcal{W}}} \frac{1}{2} \sum_{k=1}^{N} \frac{1}{\sigma^2_{v,k}} |\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}^\top \boldsymbol{w}_k|^2 + \frac{1}{2} \boldsymbol{\mathcal{W}}^\top \mathcal{L} \boldsymbol{\mathcal{W}}.
\end{aligned} \tag{34}$$

When $\sigma^2_{v,k} = \sigma^2_v \ \forall k$, the optimal choice of η in (2) would be σ^2_v.

REFERENCES

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

[2] S. S. Ram, A. Nedić, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *J. Optim. Theory Appl.*, vol. 147, no. 3, pp. 516–545, 2010.

[3] 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, 2013.

[4] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4-5, pp. 311–801, 2014.

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

[6] C. Eksin and A. Ribeiro, "Distributed network optimization with heuristic rational agents," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5396–5411, Oct. 2012.

[7] V. Kekatos and G. B. Giannakis, "Distributed robust power system state estimation," *IEEE Trans. Signal Process.*, vol. 28, no. 2, pp. 1617–1626, 2013.

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

[9] R. Nassif, C. Richard, A. Ferrari, and A. H. Sayed, "Multitask diffusion adaptation over asynchronous networks," *IEEE Trans. Signal Process.*, vol. 64, no. 11, pp. 2835–2850, 2016.

[10] 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, 2016.

[11] J. Plata-Chaves, A. Bertrand, and M. Moonen, "Incremental multiple error filtered-X LMS for node-specific active noise control over wireless acoustic sensor networks," in *Proc. IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM)*, Brazil, Jul. 2016, pp. 1–5.

[12] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[13] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1644–1656, Apr. 2013.

[14] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3042–3054, Jun. 2014.

[15] F. R. K. Chung, *Spectral Graph Theory*, American Mathematical Society, 1997.

[16] D. Zhou and B. Schölkopf, "A regularization framework for learning from graph data," in *ICML Workshop on Statistical Relational Learning and Its Connections to Other Fields*, Banff, Canada, 2004, vol. 15, pp. 67–68.

[17] H. Rue and L. Held, *Gaussian Markov Random Fields: Theory and Applications*, Chapman and Hall/CRC, 2005.

[18] E. Kreyszig, *Introductory Functional Analysis with Applications*, John Wiley & Sons, 1989.

[19] B. T. Polyak, *Introduction to Optimization*, Optimization Software, New York, 1987.

[20] D. W. Hosmer and S. Lemeshow, *Applied Logistic Regression*, Wiley, NJ, 2nd edition, 2000.

[21] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 4th edition, 2008.