Online Learning and Adaptation over Networks: More Information is Not Necessarily Better

Ali H. Sayed, Sheng-Yuan Tu, and Jianshu Chen

Department of Electrical Engineering
University of California
Los Angeles, CA 90095, USA

***Abstract*—We examine the performance of stochastic-gradient learners over connected networks for global optimization problems involving risk functions that are not necessarily quadratic. We consider two well-studied classes of distributed schemes including consensus strategies and diffusion strategies. We quantify how the mean-square-error and the convergence rate of the network vary with the combination policy and with the fraction of informed agents. Several combination policies are considered including doubly-stochastic rules, the averaging rule, Metropolis rule, and the Hastings rule. It will be seen that the performance of the network does not necessarily improve with a larger proportion of informed agents. A strategy to counter the degradation in performance is presented.**

I. Introduction

We consider connected networks with N agents, as depicted in Fig. 1. The neighborhood of an arbitrary agent k is denoted by $\mathcal{N}_k$ and it consists of all agents that are connected to k by edges. Neighboring agents can share information over the edges linking them. We assign a pair of nonnegative weights $\{a_{k\ell}, a_{\ell k}\}$ to the edges connecting every pair of neighbors k and ℓ. The scalar $a_{\ell k}$ is used by agent k to scale data it receives from agent ℓ and similarly for $a_{k\ell}$. The weights $\{a_{k\ell}, a_{\ell k}\}$ can be different so that the exchange of information between agents k and ℓ need not be symmetric. When at least one a_{kk} is positive for some k, the connected network is said to be a *standard* network. We collect the coefficients $\{a_{\ell k}\}$ into an $N \times N$ matrix $A = [a_{\ell k}]$. We refer to A as the *combination* matrix. The fact that the network is standard implies that A will be a primitive matrix, i.e., there will exist a finite integer power $m > 0$ such that the entries of A^m are all strictly positive [1], [2].

We associate with each agent k an individual real-valued *cost* (or risk or utility) function, denoted by $J_k(w)$. Although unnecessary, we assume in this article that the $M \times 1$ independent variable w is real-valued as well. We also assume that each of the individual costs, $J_k(w)$, is ν_k-strongly-convex and twice-differentiable with respect to w. This condition ensures that the Hessian matrix of $J_k(w)$ is sufficiently bounded away from zero [3], [4], namely,

$$\nabla_w^2 J_k(w) \;\geq\; \nu_k I_M > 0, \quad \text{for all } w \tag{1}$$

where I_M is the $M \times M$ identity matrix. The requirement of strong convexity is not a serious limitation. For instance, it is customary in machine learning problems [5], [6], and in adaptation and estimation problems [7], [8], to incorporate regularization factors into the cost functions; these factors help ensure strong convexity. Studies under more relaxed conditions can be found in [9]–[12].

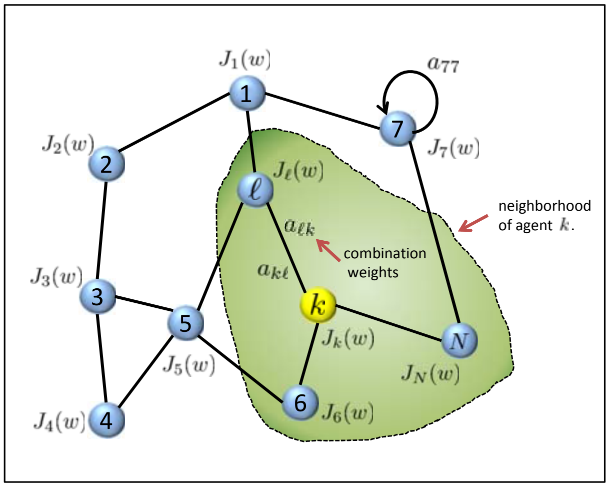

Fig. 1. Neighboring agents can share information over the edge linking them. The neighborhood of agent k is marked by the highlighted area and consists of the set of agents $\mathcal{N}_k = \{6, \ell, k, N\}$.

The unique minimizer for each convex function $J_k(w)$ is denoted by w_k^o. The minimizers $\{w_k^o\}$ across all agents are generally distinct from each other. Nevertheless, there are many useful scenarios where all individual costs $\{J_k(w)\}$ happen to be minimized at the *same* location $w = w^o$ so that

$$\nabla_w J_k(w^o) = 0, \quad k = 1, 2, \ldots, N \tag{2}$$

Examples abound where agents need to work cooperatively to attain a common objective such as tracking a target, locating a food source, or evading a predator, e.g., [13], [14]. This situation is also common in machine learning problems [5], [6], [15]–[17], where data samples are often generated from the same underlying distribution — see Sec. V further ahead. Although the discussion can be extended to the more general case where the $\{w_k^o\}$ are possibly distinct by applying the results of [10]–[12], we will instead focus in this work on the case in which the individual costs are minimized at the same $w = w^o$ so that condition (2) is assumed to hold.

In summary, given a collection of N strongly-convex and differentiable cost functions $\{J_k(w)\}$ whose minimizers coincide, we are interested in determining in a *distributed* manner the unique parameter vector, w^o, of size $M \times 1$, that minimizes the following global cost function:

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

The individual costs $\{J_k(w)\}$ can be distinct across the agents or they can all be identical, i.e., $J_k(w) \equiv J(w)$ $(k = 1, 2, \ldots, N)$. The objective of decentralized processing is to enable the agents to approach the solution w^o by relying solely on in-network (as opposed to centralized) processing.

II. Distributed Strategies

In this article, we examine two classes of distributed strategies for the solution of (3): (a) consensus strategies (see, e.g., [18]–[21] and the references therein), and (b) diffusion strategies (see, e.g., [1], [10], [14], [22], [23] and the references therein). These strategies are based on stochastic-gradient updates and are described by the following expressions.

We introduce combination coefficients $\{a_{\ell k}\}$ that are chosen to satisfy the following conditions for each agent $k = 1, 2, \ldots, N$:

$$a_{\ell k} \geq 0, \quad \sum_{\ell=1}^{N} a_{\ell k} = 1, \quad \text{and} \quad a_{\ell k} = 0 \text{ if } \ell \notin \mathcal{N}_k \tag{4}$$

The coefficients $\{a_{\ell k}\}$ are free weighting parameters whose selection influences the performance of the distributed solutions. From (4), the resulting $N \times N$ combination matrix $A = [a_{\ell k}]$ then satisfies

$$A^{\mathsf{T}} \mathbb{1} = \mathbb{1} \tag{5}$$

where the notation $\mathbb{1}$ denotes a column vector with unit entries. That is, the entries on each *column* of A add up to one so that A is a *left-stochastic* matrix.

A. Consensus Strategy

In the consensus strategy, each agent k evaluates estimators for w^o in the following manner:

$$\boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{w}_{\ell,i-1} \;-\; \mu_k \left[\widehat{\nabla_w J}_k(\boldsymbol{w}_{k,i-1})\right]^{\mathsf{T}} \tag{6}$$

where $\mu_k > 0$ is the step-size parameter. We can allow the $\{\mu_k\}$ to be time-dependent as well, such as selecting sequences $\mu_k(i)$ that satisfy the conditions:

$$\sum_{i=0}^{\infty} \mu_k(i) = \infty, \qquad \sum_{i=0}^{\infty} \mu_k^2(i) < \infty \tag{7}$$

Nevertheless, such decaying step-sizes turn-off adaptation and learning as time progresses. For this reason, we focus in this article on the case of *constant* step-sizes in order to endow the distributed solutions with continuous adaptation and learning abilities.

In the consensus implementation (6), the vector $\boldsymbol{w}_{k,i}$ denotes the estimator for w^o that is computed by agent k at time i. Moreover, the term $\widehat{\nabla_w J}_k(\boldsymbol{w}_{k,i-1})$ denotes an approximation for the true gradient vector, $\nabla_w J_k(\cdot)$, evaluated at $\boldsymbol{w}_{k,i-1}$. In our notation, the gradient vector of $J_k(w)$ relative to w is taken to be a row vector (which explains the use of the transposition symbol in (6)). In the consensus update (6), it is seen that at each iteration i, every agent k performs two steps: it combines the estimators from its neighbors using the coefficients $\{a_{\ell k}, \ell \in \mathcal{N}_k\}$ and, subsequently, updates this combination by the approximate gradient vector (*evaluated* at $\boldsymbol{w}_{k,i-1}$).

B. Diffusion Strategy

Diffusion strategies also enable the distributed minimization of (3) and lead to enhanced performance for adaptation and learning over graphs [24]. There are several diffusion variants. It is sufficient for this article to focus on the combine-then-adapt (CTA) and adapt-then-combine (ATC) forms of diffusion. The CTA strategy is described by the following update at each agent k:

$$\left\{\begin{array}{lcll} \boldsymbol{\psi}_{k,i-1} & = & \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{w}_{\ell,i-1} & \text{(CTA diffusion)} \\ \boldsymbol{w}_{k,i} & = & \boldsymbol{\psi}_{k,i-1} - \mu_k \left[\widehat{\nabla_w J}_k\left(\boldsymbol{\psi}_{k,i-1}\right)\right]^{\mathsf{T}} & \end{array}\right. \tag{8}$$

At every iteration i, the update (8) performs two operations. The first step is a combination step where agent k combines the estimators from its neighbors to obtain the intermediate iterate $\boldsymbol{\psi}_{k,i-1}$. The second step is an adaptation step where agent k updates the intermediate estimate by using its approximate gradient vector (*evaluated* at $\boldsymbol{\psi}_{k,i-1}$).

The ATC strategy simply switches the order of the combination and adaptation steps in (8):

$$\left\{\begin{array}{lcll} \boldsymbol{\psi}_{k,i} & = & \boldsymbol{w}_{k,i-1} - \mu_k \left[\widehat{\nabla_w J}_k(\boldsymbol{w}_{k,i-1})\right]^{\mathsf{T}} & \\ \boldsymbol{w}_{k,i} & = & \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i} & \text{(ATC diffusion)} \end{array}\right. \tag{9}$$

We observe that the CTA and ATC diffusion strategies (8) and (9) have fundamentally the same structure. The difference between the two implementations lies in which variable we choose to correspond to the updated estimator $\boldsymbol{w}_{k,i}$. In the ATC case, we choose the result of the *combination* step to be $\boldsymbol{w}_{k,i}$, whereas in the CTA case we choose the result of the *adaptation* step to be $\boldsymbol{w}_{k,i}$.

Observe further that the diffusion strategies (8) and (9) have *exactly the same* computational complexity as the consensus strategy (6) in terms of the number of additions and multiplications required per iteration to update $\boldsymbol{w}_{k,i-1}$ to $\boldsymbol{w}_{k,i}$ at every agent k. However, diffusion strategies differ in an important way from the consensus implementation. For example, the CTA implementation first evaluates an intermediate state variable, $\boldsymbol{\psi}_{k,i-1}$, and then uses it in the subsequent adaptation step. The net effect for CTA diffusion is an update of the form

$$\boldsymbol{w}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{w}_{\ell,i-1} \;-\; \mu_k \left[\widehat{\nabla_w J}_k\left(\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\boldsymbol{w}_{\ell,i-1}\right)\right]^{\mathsf{T}} \tag{10}$$

Observe that the convex combination of the neighborhood estimators appears inside the rightmost error term in (10). In contrast, the consensus algorithm relies solely on $\boldsymbol{w}_{k,i-1}$ to evaluate the error term in (6). This asymmetry in the consensus update is responsible for an anomaly in its behavior [24].

III. Performance of Distributed Strategies

Studying the performance of distributed strategies is more demanding than studying the performance of traditional non-cooperative and centralized schemes. This is because agents in the network influence each other's behavior. Nevertheless, by studying how the variances (or energies) of error vectors evolve over the network, it is possible to derive useful expressions to characterize the mean-square-error performance of the consensus and diffusion strategies for *sufficiently small* step-sizes. These arguments are pursued in some great detail in [1], [10]–[12].

Assume the network is standard so that its combination matrix A is primitive. One important property of such left-stochastic and primitive matrices follows from the Perron-Frobenius Theorem [1], [2], [25], [26], namely, that they have a *single* eigenvalue at one, while all other eigenvalues are strictly inside the unit circle. Thus, let p denote the right-eigenvector of A that is associated with the eigenvalue at one. We normalize the entries of p to add up to one. Then, we also know from the properties of left-stochastic and primitive matrices that the entries of p are strictly positive and less than one. Hence, the eigenvector p so defined satisfies

$$Ap = p, \quad \mathbb{1}^{\mathsf{T}}p = 1, \quad 0 < p_k < 1, \;\; k = 1,2,\ldots,N \tag{11}$$

Let $\widetilde{\boldsymbol{w}}_{k,i}$ denote the weight-error vector at agent k at time i:

$$\widetilde{\boldsymbol{w}}_{k,i} \;\triangleq\; w^o - \boldsymbol{w}_{k,i} \tag{12}$$

Let further $\boldsymbol{s}_{k,i}(\cdot)$ denote the gradient noise random process that is introduced into the distributed algorithms when the true gradient vector, $\nabla_w J_k(\cdot)$, is replaced by the approximation, $\widehat{\nabla_w J}_k(\cdot)$, as is the case with the consensus and diffusion implementations. From expressions (6) and (8)–(9), we conclude that for these implementations, the gradient noise process is defined by either of the following expressions (the first expression applies to the consensus and ATC updates, while the second expression applies to the CTA update):

$$\begin{aligned}\boldsymbol{s}_{k,i}(\boldsymbol{w}_{k,i-1}) &\;\triangleq\; \left(\widehat{\nabla_w J}_k(\boldsymbol{w}_{k,i-1}) - \nabla_w J_k(\boldsymbol{w}_{k,i-1})\right)^{\mathsf{T}}\\ \boldsymbol{s}_{k,i}(\boldsymbol{\psi}_{k,i-1}) &\;\triangleq\; \left(\widehat{\nabla_w J}_k(\boldsymbol{\psi}_{k,i-1}) - \nabla_w J_k(\boldsymbol{\psi}_{k,i-1})\right)^{\mathsf{T}}\end{aligned}$$

We denote the asymptotic covariance matrix of the gradient noise process, when evaluated at $w = w^o$, by

$$R_{s,k} \;\triangleq\; \mathbb{E}\,\boldsymbol{s}_{k,i}(w^o)\boldsymbol{s}_{k,i}^{\mathsf{T}}(w^o), \quad i\to\infty \tag{13}$$

where we are assuming that, asymptotically, the process $\boldsymbol{s}_{k,i}(\cdot)$ is wide-sense stationary with zero mean and bounded variance (in a manner similar to conditions (47)–(48) in [10]). We also introduce the compact notation H_k to refer to the value of the Hessian matrices at the optimal solution, w^o, i.e.,

$$H_k \;\triangleq\; \nabla_w^2\, J_k(w^o), \quad k = 1,2,\ldots,N \tag{14}$$

The $\{H_k\}$ are $M\times M$ positive-definite matrices. One useful way to measure the performance of the distributed solutions is to consider the mean-square-deviation (MSD) of each individual agent k, and the average MSD across the network, which are defined as the following (steady-state) measures:

$$\begin{aligned}\text{MSD}_{\text{dist},k} &\;\triangleq\; \lim_{i\to\infty}\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,i}\|^2 &(15)\\ \text{MSD}_{\text{dist}}^{\text{network}} &\;\triangleq\; \frac{1}{N}\sum_{k=1}^{N}\text{MSD}_{\text{dist},k} &(16)\end{aligned}$$

A second useful measure for the performance of the distributed solutions is to consider the excess-risk (ER) of each individual agent k, and the average ER across the network, which are similarly defined as the following (steady-state) measures:

$$\begin{aligned}\text{ER}_{\text{dist},k} &\;\triangleq\; \lim_{i\to\infty}\mathbb{E}\left\{J_k(\boldsymbol{w}_{k,i-1}) - J_k(w^o)\right\} &(17)\\ \text{ER}_{\text{dist}}^{\text{network}} &\;\triangleq\; \frac{1}{N}\sum_{k=1}^{N}\text{ER}_{\text{dist},k} &(18)\end{aligned}$$

It can be shown that under condition (2), and for sufficiently small step-sizes, the MSD measures are well-approximated by the following trace expression to first-order in the step-size parameters (a variation of this expression holds for the more general case in which the individual costs do not share the same minimizers) [11]:

$$\text{MSD}_{\text{dist},k} \;\approx\; \text{MSD}_{\text{dist}}^{\text{network}} \;\approx \tag{19}$$
$$\frac{1}{2}\cdot\text{Tr}\left[\left(\sum_{k=1}^{N}\mu_k p_k H_k\right)^{-1}\cdot\left(\sum_{k=1}^{N}\mu_k^2 p_k^2 R_{s,k}\right)\right]$$

We observe from (19) that the distributed strategies are able to equalize the MSD levels across all agents; they essentially attain the performance as the overall network. Moreover, the convergence rate of the network MSD towards its steady-state value is governed by the following factor (the smaller the value of $\alpha\in[0,1]$, the faster the convergence):

$$\alpha \approx 1 - 2\lambda_{\min}\left(\sum_{k=1}^{N}\mu_k p_k H_k\right) \tag{20}$$

With regards to the excess-risk performance measures, it can be similarly shown that in the case when the Hessian matrices $\{H_k\}$ are *uniform* across the agents, i.e., when

$$H_k \equiv H, \quad k = 1,2,\ldots,N \tag{21}$$

then, for sufficiently small step-sizes and also under condition (2), it holds that [11]:

$$\mathrm{ER}_{\mathrm{dist},k} \;\approx\; \mathrm{ER}_{\mathrm{dist}}^{\mathrm{network}} \;\approx \tag{22}$$
$$\frac{1}{4}\cdot\left(\sum_{k=1}^{N}\mu_k p_k\right)^{-1}\cdot \mathrm{Tr}\left(\sum_{k=1}^{N}\mu_k^2 p_k^2 R_{s,k}\right)$$

The MSD and ER expressions (19) and (22) are approximations to *first-order* in the step-sizes. Under this condition, the consensus and diffusion strategies lead to essentially similar MSD and ER levels. However, if we take into account higher-order terms of the step-sizes, then it is possible to verify that differences in performance arise between both classes of strategies and that diffusion strategies enhance the performance of the network over consensus strategies — see, e.g., [14], [24].

IV. Optimal Combination Rules

We observe from (19) that the MSD performance of the network is dependent on the choice of the combination matrix A. There are several ways by which the matrix A can be selected such as the Laplacian rule, the Metropolis rule, the averaging rule, and the relative-degree rule (see, e.g., [1] and the references therein). For instance, the Metropolis rule selects the coefficients $\{a_{\ell k}\}$ as follows:

$$a_{\ell k} \;=\; \begin{cases} 1/\max\{n_k, n_\ell\}, & \text{if } k \neq \ell \text{ are neighbors} \\ 1-\left(\displaystyle\sum_{m\in\mathcal{N}_k\backslash\{k\}} a_{mk}\right), & k=\ell \\ 0, & \text{otherwise}\end{cases} \tag{23}$$

where n_k denotes the cardinality of $\mathcal{N}_k$ (also called the degree of agent k) and is equal to the number of neighbors that k has:

$$n_k \;\stackrel{\Delta}{=}\; |\mathcal{N}_k| \tag{24}$$

Likewise, the averaging rule selects the $\{a_{\ell k}\}$ as follows:

$$a_{\ell k} = \begin{cases} 1/n_k, & \text{if } k\neq\ell \text{ are neighbors or } k=\ell \\ 0, & \text{otherwise}\end{cases} \tag{25}$$

It is observed that in most available combination rules, the values of the coefficients $\{a_{\ell k}\}$ are defined solely in terms of the degrees of the agents; their constructions ignore the noise profile $\{R_{s,k}\}$ across the agents.

A. Optimizing the MSD Performance

Motivated by this observation, we return to the MSD expression (19) and consider the case in which the Hessian matrices $\{H_k\}$ are uniform across the agents as in (21). This scenario is common in practice as happens, for example, with the study of networks involving the solution of mean-square-error estimation problems (where all individual costs have the same Hessian matrix) or with machine learning applications where all agents minimize the same risk function so that $J_k(w) \equiv J(w)$. We discuss these two important cases in the next section. We also assume that all agents employ the same step-size:

$$\mu_k \;=\; \mu, \quad k=1,2,\ldots,N \tag{26}$$

Under (21) and (26), and using the fact that the entries of p add up to one, the MSD expression (19) reduces to

$$\mathrm{MSD}_{\mathrm{dist},k} \approx \mathrm{MSD}_{\mathrm{dist}}^{\mathrm{network}} \approx \frac{\mu}{2}\cdot \mathrm{Tr}\left(\sum_{k=1}^{N} p_k^2 H^{-1} R_{s,k}\right) \tag{27}$$

We are then motivated to consider the problem of selecting the coefficients $\{a_{\ell k}\}$ optimally by solving:

$$\begin{aligned} A^o \;\stackrel{\Delta}{=}\; \arg\min_{A\in\mathbb{A}}\; \mathrm{Tr}\left(\sum_{k=1}^{N} p_k^2 H^{-1} R_{s,k}\right) \\ \text{subject to } \; Ap=p,\;\; \mathbb{1}^{\mathsf{T}}p=1,\;\; p_k>0 \end{aligned} \tag{28}$$

The symbol $\mathbb{A}$ represents the set of all $N\times N$ primitive left-stochastic matrices whose entries $\{a_{\ell k}\}$ satisfy conditions (4). The alternative problem of minimizing expression (19) over $A\in\mathbb{A}$, without conditions (21) and (26), is more challenging. To solve (28), we introduce the nonnegative scalars

$$\theta_k^2 \;\stackrel{\Delta}{=}\; \mathrm{Tr}(H^{-1}R_{s,k}), \quad k=1,2,\ldots,N \tag{29}$$

These scalars incorporate information about the gradient noise at the various agents through their dependence on the $\{R_{s,k}\}$. Based on the discussions in [27]–[29], an optimal A^o that solves (28) is the following left-stochastic matrix (which we refer to as the Hastings combination rule):

$$a_{\ell k}^o \;=\; \begin{cases} \dfrac{\theta_k^2}{\max\{\, n_k\theta_k^2,\; n_\ell\theta_\ell^2 \,\}}, & \ell\in\mathcal{N}_k\backslash\{k\} \\ 1-\left(\displaystyle\sum_{m\in\mathcal{N}_k\backslash\{k\}} a_{mk}^o\right), & \ell=k\end{cases} \tag{30}$$

The entries of the right-eigenvector, p, that is associated with the eigenvalue at one for the above optimal matrix $A^o=[a_{\ell k}^o]$ are given by:

$$p_k^o \;=\; \frac{1}{\theta_k^2}\cdot\left(\sum_{\ell=1}^{N}\frac{1}{\theta_\ell^2}\right)^{-1}, \quad k=1,2,\ldots,N \tag{31}$$

and the resulting minimum cost from (27) is:

$$\mathrm{MSD}_{\mathrm{dist,A^o}}^{\mathrm{network}} \;\approx\; \frac{\mu}{2}\cdot\left(\sum_{k=1}^{N}\frac{1}{\theta_k^2}\right)^{-1} \tag{32}$$

B. Optimizing the ER Performance

In a similar manner, we consider the minimization of the ER performance level (22) under the same uniformity conditions (21) and (26). In this case, the ER expression (22) reduces to

$$\mathrm{ER}_{\mathrm{dist},k} \approx \mathrm{ER}_{\mathrm{dist}}^{\mathrm{network}} \approx \frac{\mu}{4}\cdot \mathrm{Tr}\left(\sum_{k=1}^{N} p_k^2 R_{s,k}\right) \tag{33}$$

We are then motivated to consider the problem of selecting the coefficients $\{a_{\ell k}\}$ optimally by solving instead:

$$\begin{aligned} A^o \;\stackrel{\Delta}{=}\; \arg\min_{A\in\mathbb{A}}\; \mathrm{Tr}\left(\sum_{k=1}^{N} p_k^2 R_{s,k}\right) \\ \text{subject to } \; Ap=p,\;\; \mathbb{1}^{\mathsf{T}}p=1,\;\; p_k>0 \end{aligned} \tag{34}$$

To solve (34), we now define the nonnegative scalars θ_k^2 as:

$$\theta_k^2 \overset{\Delta}{=} \text{Tr}(R_{s,k}), \quad k = 1, 2, \ldots, N \tag{35}$$

to arrive at the same expressions (30) and (31), while the minimum cost from (33) is found to be:

$$\text{ER}_{\text{dist},\text{A}^o}^{\text{network}} \approx \frac{\mu}{4} \cdot \left(\sum_{k=1}^{N} \frac{1}{\theta_k^2} \right)^{-1} \tag{36}$$

V. Role of Informed Agents

We now examine the situation in which only a *fraction* of the agents in the network are informed. Informed agents are defined as those that are able to evaluate the gradient vector approximations $\{\widehat{\nabla_w J_k}(\cdot)\}$ continuously and perform the two tasks of combination and adaptation. On the other hand, uninformed agents only participate in the combination tasks. The presentation below extends the result of [30] to cost functions $J_k(w)$ that are not necessarily quadratic in w by relying on arguments that exploit the useful MSD and ER expressions (19) and (22). It will be seen that when the set of informed agents is enlarged, the convergence rate of the network becomes faster albeit at the expense of some possible deterioration in MSD or ER performance — see Fig. 2.

We model uninformed agents by setting their step-sizes to zero, i.e., we set $\mu_k = \mu$ for informed agents and $\mu_k = 0$ for uninformed agents. Although uninformed agents do not perform adaptation, they still contribute to the diffusion of information through the network. We assume a uniform step-size for the informed agents in order not to bias the subsequent comparisons by having agents with more powerful learning abilities than other agents. Let the symbol $\mathcal{N}_I$ represent the set of indices corresponding to the informed agents in the network; we denote its size by $N_I = |\mathcal{N}_I|$. That is, we assume that the network has N_I informed agents, while the remaining agents are uninformed. We further assume the network has at least one informed agent so that $N_I \geq 1$. It can be verified that, as long as the informed agents employ step-sizes μ that are sufficiently small, then the variances $\mathbb{E}\,\|\widetilde{\boldsymbol{w}}_{k,i}\|^2$ will continue to converge for *all* agents (both informed and uninformed).

Now, substituting $\mu_k = \mu$ for $k \in \mathcal{N}_I$ and $\mu_k = 0$ otherwise, into expressions (19)–(20) we find that the convergence rate and the MSD performance of the network with N_I informed agents are captured by the following relations:

$$\alpha \approx 1 - 2\mu \cdot \lambda_{\min} \left\{ \sum_{k \in \mathcal{N}_I} p_k H_k \right\} \tag{37}$$

and

$$\text{MSD}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{2} \cdot \text{Tr} \left[\left(\sum_{k \in \mathcal{N}_I} p_k H_k \right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k^2 R_{s,k} \right) \right] \tag{38}$$

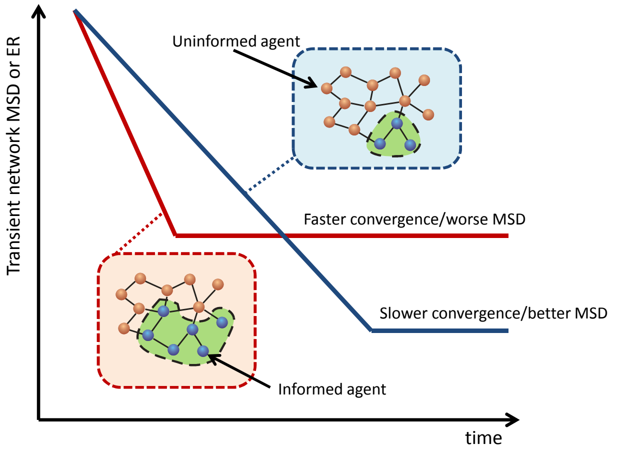
Transient network MSD or ER
Uninformed agent
Faster convergence/worse MSD
Slower convergence/better MSD
Informed agent
time

Fig. 2. Enlarging the set of informed agents improves convergence rate but does not necessarily improve the MSD or ER performance.

Likewise, under the uniform assumption (21), the ER expression (22) reduces to

$$\text{ER}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{4} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k \right)^{-1} \cdot \text{Tr} \left(\sum_{k \in \mathcal{N}_I} p_k^2 R_{s,k} \right) \tag{39}$$

Expressions (37)–(39) are in terms of the entries $\{p_k\}$ of the eigenvector p defined by (11). Since the entries of p are positive for primitive left-stochastic matrices A, it is clear from (37) that if the set of informed agents is enlarged from $\mathcal{N}_I$ to $\mathcal{N}_I' \supset \mathcal{N}_I$, then the convergence rate of the network improves. However, from (38), the network MSD and ER levels given by (38)–(39) may decrease, remain unchanged, or increase depending on the values of $\{H_k, R_{s,k}\}$. The examples discussed in the next two subsections illustrate these possibilities.

A. Mean-Square-Error Estimation

We consider the case studied in [30] and re-examine it in light of the performance expressions (37)–(38). We consider a standard network where each agent k collects streaming data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ that are assumed to satisfy a linear regression model with additive measurement noise of the form:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i), \quad i \geq 0 \tag{40}$$

for some unknown $M \times 1$ vector w^o and where the $\{\boldsymbol{u}_{k,i}\}$ are row vectors. A mean-square-error cost function is associated with each agent, namely,

$$J_k(w) = \mathbb{E}\,|\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w|^2, \quad k = 1, 2, \ldots, N \tag{41}$$

The processes $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}, \boldsymbol{v}_k(i)\}$ that appear in (40) are assumed to represent zero-mean jointly wide-sense stationary random processes that satisfy the following three conditions:

(a) The regression data $\{\boldsymbol{u}_{k,i}\}$ are temporally white and independent over space with uniform covariance matrix so that

$$\mathbb{E}\,\boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{u}_{\ell,j} \;\overset{\Delta}{=}\; R_u \cdot \delta_{k,\ell} \cdot \delta_{i,j} \tag{42}$$

where the symbol $\delta_{m,n}$ denotes the Kronecker delta sequence. The cross-correlation vector for the processes $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ is denoted by

$$r_{du} \;\overset{\Delta}{=}\; \mathbb{E}\,\boldsymbol{d}_k(i)\boldsymbol{u}_{k,i}^{\mathsf{T}} \tag{43}$$

(b) The noise process $\{\boldsymbol{v}_k(i)\}$ is temporally white and independent over space so that

$$\mathbb{E}\,\boldsymbol{v}_k(i)\boldsymbol{v}_\ell(j) \;\overset{\Delta}{=}\; \sigma_{v,k}^2 \cdot \delta_{k,\ell} \cdot \delta_{i,j} \tag{44}$$

(c) The regression and noise processes $\{\boldsymbol{u}_{\ell,j}, \boldsymbol{v}_k(i)\}$ are independent of each other for all k, ℓ, i, j.

Now, observe that the minimizers of the individual cost functions $\{J_k(w)\}$ defined above occur at the same location since

$$w_k^o \;=\; R_u^{-1} r_{du}, \quad k = 1, 2, \ldots, N \tag{45}$$

Moreover, if we multiply both sides of the assumed linear model (40) by $\boldsymbol{u}_{k,i}^{\mathsf{T}}$ from the left, and take expectations, we find that the unknown w^o satisfies the linear equations:

$$r_{du} \;=\; R_u w^o \;+\; 0 \tag{46}$$

We therefore conclude that the desired w^o is given by the same expression as the local minimizers w_k^o in (45), for any $k = 1, 2, \ldots, N$. This conclusion means that the current problem setting corresponds to a situation in which all individual costs $\{J_k(w)\}$ attain their minima at the same location, $w = w^o$, (so that condition (2) is satisfied). Furthermore, the Hessian matrices are all equal and given by

$$\nabla_w^2 J_k(w) \;=\; 2R_u \equiv H \tag{47}$$

To describe the structure of the distributed solutions for determining w^o, we first note that the true gradient vector of each $J_k(w)$ is given by

$$\nabla_w J_k(w) \;=\; 2\left(R_u w - r_{du}\right)^{\mathsf{T}} \tag{48}$$

However, the data moments $\{R_u, r_{du}\}$ are generally unknown beforehand to the agents. In that case, the true gradient vectors need to be approximated. One simple and effective construction is to rely on stochastic-gradient approximations. In this construction, each agent k uses its data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ to compute instantaneous approximations for the unavailable moments as follows:

$$r_{du} \;\approx\; \boldsymbol{d}_k(i)\boldsymbol{u}_{k,i}^{\mathsf{T}}, \qquad R_u \;\approx\; \boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{u}_{k,i} \tag{49}$$

Doing so, the resulting consensus strategy (6) will be given by

$$\boldsymbol{w}_{k,i} \;=\; \sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{w}_{\ell,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}}[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}] \tag{50}$$

while the CTA and ATC diffusion strategies (8)–(9) will be given by

$$\begin{cases} \boldsymbol{\psi}_{k,i-1} &=\; \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{w}_{\ell,i-1}, \qquad \text{(CTA diffusion)} \\ \boldsymbol{w}_{k,i} &=\; \boldsymbol{\psi}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}}\left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{\psi}_{k,i-1}\right] \end{cases} \tag{51}$$

and

$$\begin{cases} \boldsymbol{\psi}_{k,i} &=\; \boldsymbol{w}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}}\left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}\right] \\ \boldsymbol{w}_{k,i} &=\; \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k}\, \boldsymbol{\psi}_{\ell,i}, \qquad \text{(ATC diffusion)} \end{cases} \tag{52}$$

It is sufficient to continue the discussion by considering the ATC update (a similar discussion applies to consensus and CTA). It is straightforward to verify that the gradient noise vector for ATC is given by:

$$\boldsymbol{s}_{k,i}(\boldsymbol{w}_{k,i-1}) \;=\; 2\left(R_u - \boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{u}_{k,i}\right) \cdot \widetilde{\boldsymbol{w}}_{k,i-1} \;-\; 2\boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{v}_k(i) \tag{53}$$

Consequently,

$$\boldsymbol{s}_{k,i}(w^o) \;=\; -2\boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{v}_k(i) \tag{54}$$

and

$$\begin{aligned} R_{s,k} \;&\overset{\Delta}{=}\; \mathbb{E}\,\boldsymbol{s}_{k,i}(w^o)\boldsymbol{s}_{k,i}^{\mathsf{T}}(w^o) \\ &=\; 4\sigma_{v,k}^2 R_u \end{aligned} \tag{55}$$

Now assuming uniform step-sizes, i.e., $\mu_k = \mu$ over the set of informed agents $k \in \mathcal{N}_I$ and $\mu_k = 0$ otherwise, we conclude from expressions (37) and (38) that

$$\alpha \;\approx\; 1 - 4\mu \cdot \lambda_{\min}(R_u) \cdot \left(\sum_{k \in \mathcal{N}_I} p_k\right) \tag{56}$$

$$\text{MSD}_{\text{dist}}^{\text{network}} \;\approx\; \mu M \cdot \left(\sum_{k \in \mathcal{N}_I} p_k\right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k^2 \sigma_{v,k}^2\right) \tag{57}$$

It is again clear that if the set of informed agents is enlarged from $\mathcal{N}_I$ to $\mathcal{N}_I' \supset \mathcal{N}_I$, then the convergence rate improves (i.e., faster convergence with α becoming smaller). However, from (57), the network MSD may decrease, remain unchanged, or increase depending on the values of the noise variances $\{\sigma_{v,k}^2\}$ at the new informed agents. We illustrate this behavior by considering two selections for the combination matrix A.

Assume first that A is chosen to be a doubly-stochastic matrix (such as the Metropolis rule (23)), i.e., it satisfies

$$A\mathbb{1} = \mathbb{1}, \quad A^{\mathsf{T}}\mathbb{1} = \mathbb{1} \tag{58}$$

so that the entries on each of its columns and on each of its rows add up to one. Then, $p_k = 1/N$ and the above expressions reduce to:

$$\alpha \;\approx\; 1 - 4\mu \cdot \left(\frac{N_I}{N}\right) \cdot \lambda_{\min}(R_u) \tag{59}$$

$$\text{MSD}_{\text{dist}}^{\text{network}} \;\approx\; \mu M \cdot \frac{1}{N} \cdot \left(\frac{1}{N_I}\sum_{k \in \mathcal{N}_I} \sigma_{v,k}^2\right) \tag{60}$$

It is seen that if we add a new informed agent of index $k' \notin \mathcal{N}_I$, then the convergence rate improves but the MSD performance of the network will get worse if

$$\sigma_{v,k'}^2 \;>\; \frac{1}{N_I} \cdot \sum_{k \in \mathcal{N}_I} \sigma_{v,k}^2 \tag{61}$$

That is, the MSD performance gets worse if the incoming noise power at the newly added agent is worse than the average noise power at the existing informed agents.

Let us consider next the case in which the combination weights $\{a_{\ell k}\}$ are selected according to the averaging rule (25). It can be verified that the right-eigenvector p corresponding to the eigenvalue at one will be given by:

$$p \;=\; \left(\sum_{k=1}^{N} n_k\right)^{-1} \cdot \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_N \end{bmatrix} \tag{62}$$

so that expressions (56) and (57) reduce to

$$\alpha \;\approx\; 1 - 4\mu \cdot \lambda_{\min}(R_u) \cdot \left(\frac{\sum_{k \in \mathcal{N}_I} n_k}{\sum_{k=1}^{N} n_k}\right) \tag{63}$$

and

$$\text{MSD}_{\text{dist}}^{\text{network}} \;\approx \tag{64}$$

$$\mu M \cdot \left(\frac{1}{\sum_{k=1}^{N} n_k}\right) \cdot \left(\frac{1}{\sum_{k \in \mathcal{N}_I} n_k}\right) \cdot \left(\sum_{k \in \mathcal{N}_I} n_k^2 \sigma_{v,k}^2\right)$$

It is seen now that if we add a new informed agent of index $k' \notin \mathcal{N}_I$, then the convergence rate improves but the MSD performance of the network will get worse if

$$n_{k'} \sigma_{v,k'}^2 \;>\; \left(\sum_{k \in \mathcal{N}_I} n_k\right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} n_k^2 \sigma_{v,k}^2\right) \tag{65}$$

The condition in this case depends on both the noise powers and the degrees of connectivity of the agents.

B. Online Learning

The second situation we consider is one that deals with a collection of N learners, where each learner k receives a streaming sequence of vector data samples $\{\boldsymbol{x}_{k,i}, i \geq 0\}$ that arise from some fixed probability distribution $\mathcal{X}$:

$$\boldsymbol{x}_{k,i} \;\sim\; \mathcal{X}, \quad k = 1, 2, \ldots, N \tag{66}$$

The goal is to learn the vector w^o that optimizes a $\nu-$strongly-convex risk function $J(w)$:

$$w^o \;\overset{\Delta}{=}\; \arg\min_{w} \; J(w) \tag{67}$$

where $J(w)$ is the average of some loss measure $Q(\cdot,\cdot)$ [31], [32], say,

$$J(w) \;\overset{\Delta}{=}\; \mathbb{E}\, Q(w, \boldsymbol{x}_{k,i}) \tag{68}$$

Each agent k optimizes (67) by running any of the distributed algorithms introduced before (consensus or diffusion). For example, the ATC diffusion strategy would take the following form:

$$\boldsymbol{\psi}_{k,i} \;=\; \boldsymbol{w}_{k,i-1} \;-\; \mu \cdot [\nabla_w Q(\boldsymbol{w}_{k,i-1}, \boldsymbol{x}_{k,i})]^{\mathsf{T}} \tag{69}$$

$$\boldsymbol{w}_{k,i} \;=\; \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{k,i} \tag{70}$$

where the gradient of the loss function is used as an approximation for the true gradient of the risk function. The gradient noise vector is then given by

$$\boldsymbol{s}_{k,i}(\boldsymbol{w}_{k,i-1}) \;=\; [\nabla_w Q(\boldsymbol{w}_{k,i-1}, \boldsymbol{x}_{k,i}) \;-\; \nabla_w J(\boldsymbol{w}_{k,i-1})]^{\mathsf{T}} \tag{71}$$

so that

$$\boldsymbol{s}_{k,i}(w^o) \;=\; [\nabla_w Q(w^o, \boldsymbol{x}_{k,i})]^{\mathsf{T}} \tag{72}$$

Since the data $\{\boldsymbol{x}_{k,i}\}$ are sampled independently by the agents from the same distribution $\mathcal{X}$, it is reasonable to assume that the covariance matrices of $\{\boldsymbol{s}_{k,i}(w^o)\}$ are uniform across the agents so that

$$R_{s,k} \;\overset{\Delta}{=}\; \mathbb{E}\, [\nabla_w Q(w^o, \boldsymbol{x}_{k,i})]^{\mathsf{T}} \, [\nabla_w Q(w^o, \boldsymbol{x}_{k,i})] \;\equiv R_s \tag{73}$$

Therefore, the current network setting corresponds to a situation in which all agents are minimizing the *same* cost function $J(w)$ and where the Hessian matrices at $w = w^o$ are uniform and given by

$$H \;\overset{\Delta}{=}\; \nabla_w^2 J(w^o) \tag{74}$$

The MSD and ER performance levels for the distributed solution, using N_I informed agents with step-sizes $\mu_k = \mu$, can be deduced from (19) and (22) as

$$\text{MSD}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{2} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k\right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k^2\right) \cdot \text{Tr}(H^{-1} R_s) \tag{75}$$

and

$$\text{ER}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{4} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k\right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k^2\right) \cdot \text{Tr}\left(R_s\right) \tag{76}$$

In particular, it is seen that if we add a new informed agent of index $k' \notin \mathcal{N}_I$, then the MSD or ER performance levels will get worse if

$$p_{k'} \;>\; \left(\sum_{k \in \mathcal{N}_I} p_k\right)^{-1} \cdot \left(\sum_{k \in \mathcal{N}_I} p_k^2\right) \tag{77}$$

This condition is in terms of the entries $\{p_k\}$, which are determined by the combination policy, A. We again consider two choices for the combination matrices.

Assume first that A is doubly-stochastic (such as the Metropolis rule (23)) so that $p_k = 1/N$ and condition (77) cannot be satisfied. In other words, the addition of informed agents cannot degrade the network performance. Indeed, in this case, it can be readily seen that the MSD and ER expressions

(75)–(76) reduce to

$$\text{MSD}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{2}\cdot\frac{1}{N}\cdot\text{Tr}(H^{-1}R_s) \tag{78}$$

$$\text{ER}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{4}\cdot\frac{1}{N}\cdot\text{Tr}\left(R_s\right) \tag{79}$$

Both of these expressions are independent of N_I.

Let us consider next the case in which the combination weights $\{a_{\ell k}\}$ are selected according to the averaging rule (25). Using (62), condition (77) would then indicate that the network MSD or ER levels will degrade if the degree of the newly added informed agent satisfies:

$$n_{k'} > \left(\sum_{k\in\mathcal{N}_I} n_k\right)^{-1}\cdot\left(\sum_{k\in\mathcal{N}_I} n_k^2\right) \tag{80}$$

C. Controlling Degradation in Performance

The previous arguments indicate that the MSD or ER performance of networks need not improve with the addition of informed agents. The deterioration in network performance can be controlled through proper selection of the combination weights, for example, when the matrix A is selected as the Hastings rule (30). Using expression (31) in (37) and (38) we find that the convergence rate and the MSD level of a network with N_I informed agents (using uniform step-sizes, $\mu_k = \mu$, and with uniform Hessian matrices, $H_k = H$) are now given by

$$\alpha \approx 1 - 2\mu\cdot\lambda_{\min}(H)\cdot\left(\sum_{k\in\mathcal{N}_I}\frac{1}{\theta_k^2}\right)\cdot\left(\sum_{k=1}^{N}\frac{1}{\theta_k^2}\right)^{-1} \tag{81}$$

$$\text{MSD}_{\text{dist}}^{\text{network}} \approx \frac{\mu}{2}\cdot\left(\sum_{k=1}^{N}\frac{1}{\theta_k^2}\right)^{-1} \tag{82}$$

Observe that the network MSD level is now independent of N_I, while the convergence rate continues to decrease (i.e., becomes faster) as the set of informed agents is enlarged (since the expression for α depends on N_I). This result highlights the importance of selecting the combination weights [30]. For example, under the conditions discussed in Sec. V-B for the online learning problem, the Hastings rule (30) reduces to the doubly-stochastic Metropolis rule (23), which explains why the MSD and ER results (78)–(79) are independent of N_I.

Acknowledgement. This work was supported in part by NSF grant CCF-1011918.

References

[1] A. H. Sayed, "Diffusion adaptation over networks," in *E-Reference Signal Processing*, R. Chellapa and S. Theodoridis, Eds., Elsevier, 2013. [Also available online at http://arxiv.org/abs/1205.4220 as manuscript arXiv:1205.4220v1 [cs.MA], May 2012.]

[2] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 2003.

[3] B. Poljak, *Introduction to Optimization*, Optimization Software, NY, 1987.

[4] D. Bertsekas, *Convex Analysis and Optimization*, Athena Scientific, 2003.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2007.

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

[7] A. H. Sayed, *Adaptive Filters*, Wiley, NJ, 2008.

[8] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*, Prentice Hall, NJ, 2000.

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

[10] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Processing*, vol. 60, no. 8, pp. 4289–4305, August 2012.

[11] J. Chen and A. H. Sayed, "On the limiting behavior of distributed optimization strategies," *Proc. 50th Annual Allerton Conference on Communication, Control, and Computing*, pp. 1–8, Allerton, IL, October 2012.

[12] J. Chen and A. H. Sayed, "Distributed Pareto optimization via diffusion strategies," *IEEE J. Selected Topics on Signal Processing*, vol. 7, April 2013. [Also available online as manuscript arXiv:1208.2503 [cs.MA], August 2012.]

[13] S.-Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE J. Sel. Topics. Signal Process.*, vol. 5, no. 4, pp. 649–664, Aug. 2011.

[14] A. H. Sayed, S-Y. Tu, J. Chen, X. Zhao, and Z. Towfic, "Diffusion strategies for adaptation and learning over networks," *IEEE Signal Processing Magazine*, vol. 30, May 2013.

[15] O. Dekel, R. Gilad-Bachrach, O. Shamir, and L. Xiao, "Optimal distributed online prediction," *Proc. International Conference on Machine Learning (ICML)*, Bellevue, WA, pp. 713–720, Jun. 2011.

[16] A. Agarwal and J. Duchi, "Distributed delayed stochastic optimization," *Proc. Neural Information Processing Systems (NIPS)*, Granada, Spain, pp. 873–881, Dec. 2011.

[17] J. B. Predd, S. B. Kulkarni, and H. V. Poor, "Distributed learning in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 56–69, Jul. 2006.

[18] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, Sep. 2004.

[19] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, November 2010.

[20] 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.

[21] A. Nedic and A. Ozdaglar, "Cooperative distributed multi-agent optimization," in *Convex Optimization in Signal Processing and Communications*, Y. Eldar and D. Palomar (Eds.), Cambridge University Press, pp. 340-386, 2010.

[22] 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, July 2008.

[23] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, March 2010.

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

[25] S. U. Pillai, T. Suel, and S. Cha, "The Perron–Frobenius theorem: Some of its applications," *IEEE Signal Process. Mag.*, vol. 22, no. 2, pp. 62–75, Mar. 2005.

[26] A. Berman and R. J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, SIAM, PA, 1994.

[27] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, Apr. 1970.

[28] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing Markov chain on a graph," *SIAM Review*, vol. 46, no. 4, pp. 667–689, Dec. 2004.

[29] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over LMS adaptive networks," *IEEE Trans. Signal Processing*, vol. 60, no. 10, pp. 5107–5124, October 2012.

[30] S-Y. Tu and A. H. Sayed, "On the influence of informed agents on learning and adaptation over networks," to appear in *IEEE Trans. Signal Processing*, vol. 61, 2013. Also available on http://arxiv.org/abs/1203.1524 as arXiv:1203.1524v1 [cs.IT].

[31] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, NY, 2000.

[32] Z. Towfic, J. Chen, and A. H. Sayed, "On the generalization ability of distributed online learners," *Proc. IEEE Workshop on Machine Learning for Signal Processing* (MLSP), Santander, Spain, pp. 1–6, Sep. 2012.