SINGLE-LINK DIFFUSION STRATEGIES OVER ADAPTIVE NETWORKS

Xiaochuan Zhao and Ali H. Sayed

Department of Electrical Engineering
University of California, Los Angeles

ABSTRACT

We propose an adaptive diffusion strategy with limited communication overhead by cutting off all links but one for each node in the network. We keep the "best" neighbor that has the smallest estimated variance-product measure and ignore the other neighbors. The combination coefficients for the interacting nodes are calculated via a maximal-ratio-combining rule to minimize the steady-state mean-square-deviation. Simulation results illustrate that, with less communication overhead and less computations, the proposed algorithm performs well and outperforms other related methods with similar overheads.

***Index Terms*—** Diffusion adaptation, adaptive networks, communication constraints, gossip strategy, maximal ratio combing

1. INTRODUCTION

Collaborative learning by agents in a distributed manner is useful in many contexts involving wireless sensor networks, cognitive radios, biological networks, and machine learning. Various strategies for distributed processing have been proposed in the literature including consensus strategies [1,2], incremental strategies [3,4], and diffusion strategies [5, 6]. The latter family of strategies endows networks with real-time adaptation and learning abilities. Diffusion strategies are scalable and robust, and they have been applied successfully to model complex and self-organized behavior encountered in nature [7, 8], and to the solution of general optimization problems [9].

In the traditional diffusion LMS strategy [5,6], each node k consults with every other node l in its neighborhood, $\mathcal{N}_k$. The information exchanged between node k and its neighbors consists of their intermediate estimates [5] and possibly measurement/regression data [6]. In some applications, however, networks cannot afford large communication overhead. For example, in several wireless sensing scenarios, the nodes tend to communicate with each other over bandlimited and power-constrained links. To alleviate the communication burden, reference [10] proposed a diffusion algorithm with limited communication requirements by removing links from the network. At every iteration, the solution required nodes to share estimates of their current mean-square-deviation (MSD) with their neighbors. Each node k then chooses its "best" neighbor based on the MSD information. Gossip and probabilistic strategies were also considered in [11–13] to reduce the communication cost. In these strategies, every node selects randomly a single neighbor at every iteration and processing is performed in coordination with the selected neighbor.

In order to minimize the steady-state network MSD while reducing the communication overhead for diffusion algorithms, we propose an alternative criterion for removing nodes from neighborhoods in the network. We use a new scalar metric to measure the quality of each neighbor. The metric is in the form of a *variance-product* measure, namely, a product of the power of noise and data. It is the combined effect of the power of noise and data that guides the node selection process. We select and keep the "best" neighbor with the lowest value of the variance-product and ignore other neighbors. This process is repeated at every iteration, so that neighbors are continuously adjusted. The combination coefficients are calculated by using the relative variance rule proposed in [14], which has the form of a maximal-ratio-combining (MRC) rule [15]. Simulation results illustrate that, with the same communication overhead and less computations, the proposed algorithm outperforms [10] and the gossip and probabilistic strategies [11, 12] in steady-state.

Notation: We use lowercase letters to denote vectors, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables. We also use $(\cdot)^T$ to denote transposition, $(\cdot)^*$ for conjugate transposition, $(\cdot)^{-1}$ for matrix inverse, $\otimes$ for Kronecker products, $\rho(A)$ for the spectral radius of A, and $\mathrm{vec}(A)$ to denote the column vector constructed by stacking the columns of A on top of each other.

2. ADAPTIVE DIFFUSION STRATEGIES

We consider a connected network consisting of N nodes. Each node k collects scalar measurements $\boldsymbol{d}_k(i)$ and $1 \times M$ regression vectors $\boldsymbol{u}_{k,i}$ over successive time instants $i \geq 0$. The measurements across all nodes are assumed to be related to an unknown vector w^o via a linear regression model [15]:

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i), \qquad k = 1, 2, \ldots, N \tag{1}$$

where $\boldsymbol{v}_k(i)$ denotes measurement noise. The $M \times 1$ vector w^o in (1) denotes the parameter of interest. The network would like to estimate w^o in a distributed manner by seeking the solution that minimizes the global cost function:

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

This minimization problem can be solved in an adaptive and distributed manner by the adapt-then-combine (ATC) diffusion strategy of [5]:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* \left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} \boldsymbol{w}_{k,i-1}\right] \\ \boldsymbol{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{lk} \boldsymbol{\psi}_{l,i} \end{cases} \tag{3}$$

where the $\{a_{lk}\}$ are the nonnegative entries of a combination matrix A; a_{lk} is zero whenever node l is not connected to node k, i.e., $l \notin \mathcal{N}_k$. The $N \times N$ matrix A is required to be left-stochastic, i.e.,

This work was supported in part by NSF grants CCF-0942936 and CCF-1011918. Email: {xzhao, sayed}@ee.ucla.edu.

$A^T \mathbb{1}_N = \mathbb{1}_N$, where $\mathbb{1}_N$ denotes the $N \times 1$ vector whose entries are all equal to one. The combination coefficients $\{a_{lk}\}$ can remain static [6] or be adjusted over time [14, 16]. The mean-square performance and convergence properties of the ATC diffusion strategy (3) were studied in detail in [5, 6]. In the sequel, we review the network MSD performance for the ATC algorithm (3) and use the results to motivate a procedure for selecting the most effective neighbor, i.e., the neighbor having the lowest variance-product metric.

3. SINGLE-LINK STRATEGY

Introduce the error vectors:

$$\widetilde{\boldsymbol{w}}_{k,i} \triangleq w^o - \boldsymbol{w}_{k,i}, \qquad k = 1, \ldots, N \tag{4}$$

and substitute the data model (1) into the ATC algorithm (3). Then, we can establish that the error vector across the network evolves according to the following difference equation:

$$\boxed{\widetilde{\boldsymbol{w}}_i = \mathcal{A}^T \left(I_{NM} - \mathcal{M}\boldsymbol{\mathcal{R}}_i\right) \widetilde{\boldsymbol{w}}_{i-1} - \mathcal{A}^T \mathcal{M} \boldsymbol{s}_i} \tag{5}$$

where we are introducing the following block vector and matrix quantities:

$$\widetilde{\boldsymbol{w}}_i \triangleq \mathrm{col}\{\widetilde{\boldsymbol{w}}_{1,i}, \widetilde{\boldsymbol{w}}_{2,i} \ldots, \widetilde{\boldsymbol{w}}_{N,i}\} \tag{6}$$

$$\mathcal{M} \triangleq \mathrm{diag}\{\mu_1 I_M, \mu_2 I_M, \ldots, \mu_N I_M\} \tag{7}$$

$$\boldsymbol{\mathcal{R}}_i \triangleq \mathrm{diag}\left\{\boldsymbol{u}_{1,i}^* \boldsymbol{u}_{1,i}, \boldsymbol{u}_{2,i}^* \boldsymbol{u}_{2,i}, \ldots, \boldsymbol{u}_{N,i}^* \boldsymbol{u}_{N,i}\right\} \tag{8}$$

$$\boldsymbol{s}_i \triangleq \mathrm{col}\{\boldsymbol{u}_{1,i}^* \boldsymbol{v}_1(i), \boldsymbol{u}_{2,i}^* \boldsymbol{v}_2(i), \ldots, \boldsymbol{u}_{N,i}^* \boldsymbol{v}_N(i)\} \tag{9}$$

$$\mathcal{A} \triangleq A \otimes I_M \tag{10}$$

and where I_n denotes the $n \times n$ identity matrix. We define the network MSD as

$$\overline{\mathrm{MSD}} \triangleq \lim_{i \to \infty} \frac{1}{N} \left(\sum_{k=1}^{N} \mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i}\|^2\right) \tag{11}$$

We introduce the following assumptions on the statistical properties of the measurement data and noise signals.

Assumption 1 (Statistical properties)**.**

1. *The regression data $\boldsymbol{u}_{k,i}$ are temporally and spatially independent and identically distributed (i.i.d.) random variables with zero mean and covariance matrix $R_{u,k} \triangleq \mathbb{E}\boldsymbol{u}_{k,i}^* \boldsymbol{u}_{k,i} > 0$.*
2. *The noise signals $\boldsymbol{v}_k(i)$ are temporally and spatially i.i.d. random variables with zero mean and variances $\sigma_{v,k}^2$.*
3. *The regression data $\boldsymbol{u}_{k,i}$ and noise signals $\boldsymbol{v}_l(j)$ are mutually-independent for all i and j, k and l.* ■

Based on Assumption 1, it can be verfied that the following weighted variance relation holds [5, 6]:

$$\boxed{\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|_{\Sigma}^2 = \mathbb{E}\|\widetilde{\boldsymbol{w}}_{i-1}\|_{\Sigma'}^2 + \mathrm{Tr}\left(\mathcal{Y}\Sigma\right)} \tag{12}$$

where Σ is a positive semi-definite matrix that we are free to choose, and

$$\Sigma' \triangleq \mathcal{B}^* \Sigma \mathcal{B} + O(\mathcal{M}^2) \tag{13}$$

$$\mathcal{B} \triangleq \mathcal{A}^T \left(I_{NM} - \mathcal{M}\mathcal{R}_u\right) \tag{14}$$

$$\mathcal{Y} \triangleq \mathcal{A}^T \mathcal{M} \mathcal{S} \mathcal{M} \mathcal{A} \tag{15}$$

$$\mathcal{S} \triangleq \mathbb{E}\boldsymbol{s}_i \boldsymbol{s}_i^* = \mathrm{diag}\left\{\sigma_{v,1}^2 R_{u,1}, \ldots, \sigma_{v,N}^2 R_{u,N}\right\} \tag{16}$$

$$\mathcal{R}_u \triangleq \mathbb{E}\boldsymbol{\mathcal{R}}_i = \mathrm{diag}\left\{R_{u,1}, \ldots, R_{u,N}\right\} \tag{17}$$

It is shown in [16] that $\mathcal{B}$ is stable for step-sizes that satisfy:

$$\mu_k < \frac{2}{\rho(R_{u,k})} \tag{18}$$

Thus, we further introduce a small step-size assumption.

Assumption 2 (Small step-sizes)**.** *The step-sizes are sufficiently small, i.e., $\mu_k \ll 1$.* ■

Then, it can be deduced from (12) that the steady-state network MSD for the ATC algorithm (3) is given by

$$\overline{\mathrm{MSD}} \approx \frac{1}{N} [\mathrm{vec}(\mathcal{Y})]^* (I_{N^2M^2} - \mathcal{F})^{-1} \mathrm{vec}(I_{NM}) \tag{19}$$

where

$$\mathcal{F} \triangleq \mathcal{B}^T \otimes \mathcal{B}^* \tag{20}$$

Since $\mathcal{B}$ is stable under (18) and Assumption 2, we can rewrite the network MSD in (19) as

$$\overline{\mathrm{MSD}} = \frac{1}{N} \sum_{j=0}^{\infty} \mathrm{Tr}(\mathcal{B}^j \mathcal{Y} \mathcal{B}^{*j}) \tag{21}$$

where both $\mathcal{Y}$ and $\mathcal{B}$ depend on $\mathcal{A}$. Minimizing (21) over left-stochastic matrices A is generally non-trivial. We pursue an approximate solution that optimizes an upper bound on the network MSD. Let $\|X\|_*$ denote the nuclear norm of a matrix $X \in \mathbb{C}^{m \times n}$, which is defined as [17]:

$$\|X\|_* \triangleq \sum_{k=1}^{\min\{m,n\}} \sigma_k \tag{22}$$

where $\{\sigma_k\}$ are the singular values of X. It can be verified that

$$\|X\|_* = \|X^*\|_* \tag{23}$$

Moreover, for any Hermitian and positive semi-definite matrix X,

$$\|X\|_* = \mathrm{Tr}(X) \tag{24}$$

Let further $\|X\|_{b,\infty}$ denote the block maximum norm of a matrix X; it is induced by the block maximum norm of vectors [16]:

$$\|x\|_{b,\infty} \triangleq \max_{1 \le k \le N} \|x_k\|_2 \tag{25}$$

where x consists of N blocks of size $M \times 1$ each:

$$x = \mathrm{col}\{x_1, x_2, \ldots, x_N\}, \qquad x_k \in \mathbb{C}^{M \times 1} \tag{26}$$

and $\|\cdot\|_2$ denotes the Euclidean norm of its vector argument. Then,

$$\|X\|_{b,\infty} \triangleq \max_{x \neq 0} \frac{\|Xx\|_{b,\infty}}{\|x\|_{b,\infty}} \tag{27}$$

Now noting that the term $\mathcal{B}^j \mathcal{Y} \mathcal{B}^{*j}$ is Hermitian and positive semi-definite and using (24), we get

$$\begin{aligned}
\mathrm{Tr}(\mathcal{B}^j \mathcal{Y} \mathcal{B}^{*j}) &= \|\mathcal{B}^j \mathcal{Y} \mathcal{B}^{*j}\|_* \\
&\le \|\mathcal{B}^j\|_* \cdot \|\mathcal{Y}\|_* \cdot \|\mathcal{B}^{*j}\|_* \\
&\overset{(25)}{=} \|\mathcal{B}^j\|_*^2 \cdot \mathrm{Tr}(\mathcal{Y}) \\
&\overset{(a)}{\le} c^2 \|\mathcal{B}^j\|_{b,\infty}^2 \cdot \mathrm{Tr}(\mathcal{Y}) \\
&\le c^2 \|\mathcal{B}\|_{b,\infty}^{2j} \cdot \mathrm{Tr}(\mathcal{Y}) \\
&\le c^2 \left(\|\mathcal{A}^T\|_{b,\infty} \cdot \|I_{NM} - \mathcal{M}\mathcal{R}_u\|_{b,\infty}\right)^{2j} \cdot \mathrm{Tr}(\mathcal{Y}) \\
&\overset{(b)}{=} c^2 \cdot \rho(I_{NM} - \mathcal{M}\mathcal{R}_u)^{2j} \cdot \mathrm{Tr}(\mathcal{Y})
\end{aligned} \tag{28}$$

where (a) c is some positive scalar such that $\|X\|_* \leq c\|X\|_{b,\infty}$ because $\|X\|_*$ and $\|X\|_{b,\infty}$ are submultiplicative norms and all such norms are equivalent [18], and (b) we used the fact that

$$\|\mathcal{A}^T\|_{b,\infty} = 1, \quad \|I_{NM} - \mathcal{M}\mathcal{R}_u\|_{b,\infty} = \rho(I_{NM} - \mathcal{M}\mathcal{R}_u) \quad (29)$$

In this way we can upper bound the network MSD by

$$\begin{aligned}\overline{\text{MSD}} &\leq \frac{1}{N}\sum_{j=0}^{\infty} c^2 \cdot \rho(I_{NM} - \mathcal{M}\mathcal{R}_u)^{2j} \cdot \text{Tr}(\mathcal{Y}) \\ &= \frac{c^2}{N} \cdot \frac{\text{Tr}(\mathcal{A}^T\mathcal{M}\mathcal{S}\mathcal{M}\mathcal{A})}{1-\rho(I_{NM} - \mathcal{M}\mathcal{R}_u)^2} \end{aligned} \quad (30)$$

where the combination matrix $\mathcal{A}$ appears only in the numerator. This result suggests one approach to to selecting A by minimizing the upper bound that appears in (30) over A:

$$\begin{aligned} \underset{A}{\text{minimize}} \quad & \text{Tr}(\mathcal{A}^T\mathcal{M}\mathcal{S}\mathcal{M}\mathcal{A}) \\ \text{subject to} \quad & A^T\mathbb{1}_N = \mathbb{1}_N, \;\; a_{lk} \geq 0 \\ & a_{lk} = 0, \;\; \text{if } l \notin \mathcal{N}_k \end{aligned} \quad (31)$$

Problem (31) can be decoupled into N separate minimization problems of the form:

$$\boxed{\begin{aligned} \underset{\{a_{lk};\, l\in\mathcal{N}_k\}}{\text{minimize}} \quad & \sum_{l\in\mathcal{N}_k} a_{lk}^2\mu_l^2\sigma_{v,l}^2\text{Tr}(R_{u,l}) \\ \text{subject to} \quad & \sum_{l\in\mathcal{N}_k} a_{lk} = 1, \quad a_{lk} \geq 0, \\ & a_{lk} = 0 \quad \text{if } \; l \notin \mathcal{N}_k \end{aligned}} \quad (32)$$

The optimal solution of (32) is given by

$$\boxed{a_{lk} \triangleq \frac{\gamma_l^{-2}}{\sum_{l\in\mathcal{N}_k}\gamma_l^{-2}}} \quad (33)$$

where the variance-product metric γ_k^2 is defined as

$$\boxed{\gamma_k^2 \triangleq \mu_k^2\sigma_{v,k}^2\text{Tr}(R_{u,k})} \quad (34)$$

Observe that, for each node k, the measure γ_k^2 is a scaled product of the noise variance and the regression variance at that node. Thus, nodes with large noise variance but small data variance can lead to a small variance-product. Likewise, nodes with small noise variance but large data variance can still lead to a small variance-product. It is the combined effect of the noise and data variances that determines how small or how large γ_k^2 is. Observe further from (34) that the value of the variance-product for each node k depends solely on the noise and data statistics at that same node. In other words, the value of γ_k^2 is a purely local value. Still, node k does not generally know its noise variance, $\sigma_{v,k}^2$, and its data variance, $\text{Tr}(R_{u,k})$, to be able to compute its variance-product from the definition (34); an alternative computational path is needed, as we explain below. But first note that the relative variance rule (33) is such that the weighting coefficient to neighbor l is inversely proportional to the metric γ_l^2; the smaller the metric is, the larger the weight. Therefore, if some nodes are to be removed from the neighborhood of node k (to reduce the communication overhead), then it is natural to remove nodes with larger variance-products (or, equivalently, to maintain nodes with smaller variance-products). To implement this rule with limited communications, each node k needs to evaluate its own metric γ_k^2 based on the data that are available to it.

To do so, we first note that, at each iteration i, every node k in the network can approximate its variance-product measure by means of the following calculation:

$$\hat{\boldsymbol{\gamma}}_k^2(i) \approx \|\boldsymbol{w}_{k,i-1} - \boldsymbol{\psi}_{k,i}\|^2 \quad (35)$$

The motivation behind this approximation is the fact from (1), (3), and (4) that:

$$\begin{aligned}\mathbb{E}\hat{\boldsymbol{\gamma}}_k^2(i) &\approx \mathbb{E}\,\|\mu_k\boldsymbol{u}_{k,i}^*(\boldsymbol{u}_{k,i}\widetilde{\boldsymbol{w}}_{k,i-1} + \boldsymbol{v}_k(i))\|^2 \\ &= \mu_k^2\left(\mathbb{E}\,\|\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i}\widetilde{\boldsymbol{w}}_{k,i-1}\|^2 + \mathbb{E}\,\|\boldsymbol{u}_{k,i}^*\boldsymbol{v}_k(i)\|^2\right) \\ &= \mu_k^2\left[\mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i-1}\|^2_{(\mathbb{E}\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i}\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i})} + \sigma_{v,k}^2\text{Tr}(R_{u,k})\right] \\ &\approx \mu_k^2\sigma_{v,k}^2\text{Tr}(R_{u,k}), \quad \text{as } i \to \infty \end{aligned} \quad (36)$$

because, through the variance relation (12), the term

$$\begin{aligned}&\lim_{i\to\infty}\mathbb{E}\|\widetilde{\boldsymbol{w}}_{k,i-1}\|^2_{\mathbb{E}(\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i}\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i})} \\ &\approx [\text{vec}(\mathcal{Y})]^*(I_{N^2M^2} - \mathcal{F})^{-1}\text{vec}[\mathcal{J}_k\mathbb{E}(\boldsymbol{\mathcal{R}}_i^*\boldsymbol{\mathcal{R}}_i)\mathcal{J}_k] \\ &\sim O(\mathcal{M}) \end{aligned} \quad (37)$$

is negligible under Assumption 2, where $\mathcal{J}_k$ is a block diagonal matrix whose kth block is the identity matrix, i.e.,

$$\mathcal{J}_k \triangleq \text{diag}\{0,\ldots,0,I_M,0,\ldots,0\} \quad (38)$$

Relation (36) indicates that on average, the estimated quantity given by (35) approaches the desired variance-product measure (34). We can implement the averaging step by means of a smoothing operation as follows:

$$\boldsymbol{\gamma}_k^2(i) = (1-\nu)\boldsymbol{\gamma}_k^2(i-1) + \nu\hat{\boldsymbol{\gamma}}_k^2(i) \quad (39)$$

where ν is a forgetting factor satisfying $0 < \nu < 1$ and is close to one.

In view of the above discussion, each node k collects at every iteration i, the smoothed variance-products $\{\gamma_l^2(i)\}$ from its neighbors. It then selects the node with the smallest variance-product and requests for its intermediate estimate, $\psi_{l,i}$; more generally, if desired, each node k can decide to maintain a *subset* of neighbors by ordering the variance-products of its neighbors and selecting those nodes with smaller metrics.

4. SINGLE-LINK DIFFUSION ALGORITHM

ATC Single-Link Diffusion Algorithm

Initialize $w_{k,-1} = 0$ and $\gamma_k^2(-1) = 0$ for $k = 1, 2, \ldots, N$.
for $i \geq 0$ **do**

$$\begin{aligned} e_k(i) &= d_k(i) - u_{k,i}w_{k,i-1} \\ \psi_{k,i} &= w_{k,i-1} + \mu_k u_{k,i}^* e_k(i) \\ \gamma_k^2(i) &= (1-\nu)\gamma_k^2(i-1) + \nu\|\mu_k u_{k,i}^* e_k(i)\|^2 \\ m &= \underset{l\in\mathcal{N}_k\setminus\{k\}}{\arg\min}\,\gamma_l^2(i) \\ a_{kk} &= \frac{\gamma_k^{-2}(i)}{\gamma_k^{-2}(i) + \gamma_m^{-2}(i)} \\ w_{k,i} &= a_{kk}\psi_{k,i} + (1-a_{kk})\psi_{m,i} \end{aligned}$$

end for

Remarks:

1. The value of the forgetting factor ν is close to one, say, 0.95.

2. The communication overhead of the proposed algorithm is less than the traditional diffusion algorithms [5, 6]. Every node k only receives the scalars $\{\gamma_l^2(i)\}$ from its neighbors and the intermediate estimate $\phi_{m,i}$, which is an $M\times 1$ vector.

3. The combination coefficients are calculated according to the relative variance rule (33) by using the quantity (39).

5. SIMULATION RESULTS

We consider the topology of Fig. 1a with $N = 20$ nodes. The unknown parameter w^o of length $M = 3$ is randomly generated. The regression data are i.i.d. circular complex Gaussian with zero mean and covariance matrices $\{R_{u,k}\}$ that are randomly generated; their traces are shown in the upper part of Fig. 1b. The noise signals are also i.i.d. zero-mean circular complex Gaussian, whose variances, $\sigma_{v,k}^2$, are randomly generated and shown in the lower part of Fig. 1b. The step-size $\mu = 0.02$ is uniform across the network. We compare the proposed single-link ATC algorithm with other algorithms, including the traditional ATC algorithm with the relative variance combination rule (33) using all neighbors, the randomized gossip algorithm [11, 12], the algorithm proposed in [10], and non-cooperative stand-alone LMS filters. We plot the average network MSD curves in Fig. 1c by averaging over 100 experiments. From Fig. 1c we see that, all five algorithms have the similar mean-square convergence rates during the transient phase. Moreover, as expected, the traditional ATC algorithm employing the relative variance combination rule with full neighbors attains the lowest MSD among all algorithms; the proposed algorithm reaches the second lowest MSD; the algorithm from [10] is about 4 dB worse than the proposed algorithm; the randomized gossip algorithm is about 1 dB worse than the algorithm from [10]; and the non-cooperative LMS algorithm performs the worst.

6. CONCLUSION

In this work we proposed a diffusion algorithm to reduce the communication overhead by removing some links from the network. By choosing the "best" neighbor that has the smallest variance-product measure, the proposed algorithm can achieve lower MSD performance than the algorithms in [10–12]. For band- or power-limited applications such as wireless sensor networks, the proposed algorithm can be used to deliver a scalable and robust distributed solution with good performance and limited communication overhead.

7. REFERENCES

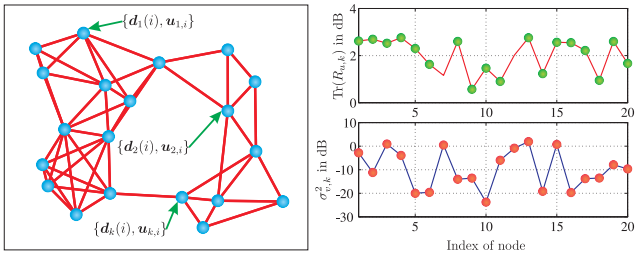
[1] M. H. DeGroot, "Reaching a consensus," *J. Am. Statist. Assoc.*, vol. 69, no. 345, pp. 118–121, 1974.

[2] S. Kar and J. M. F. Moura, "Sensor networks with random links: Topology design for distributed consensus," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3315–3326, July 2008.

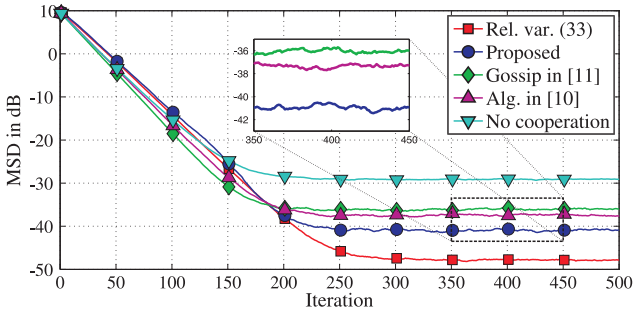
[3] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. 31, no. 9, pp. 803–812, Sept. 1986.

[4] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 48, no. 8, pp. 223–229, Aug. 2007.

[5] 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.



(a) Network topology. (b) Variance profiles.



(c) Network MSD curves.

Fig. 1. Simulation profiles and results for network MSD.

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

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

[8] F. Cattivelli and A. H. Sayed, "Modeling bird flight formations using diffusion adaptation," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2038–2051, May 2011.

[9] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *arXiv:1111.0034v1 [math.OC]*, Oct. 2011.

[10] O. L. Rortveit, J. H. Husoy, and A. H. Sayed, "Diffusion LMS with communication constraints," in *Proc. Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Nov. 2010, pp. 1645–1649.

[11] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, June 2006.

[12] C. G. Lopes and A. H. Sayed, "Diffusion adaptive netowrks with changing topology," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Las Vegas, NV, Mar. / Apr. 2008, pp. 3285–3288.

[13] N. Takahashi and I. Yamada, "Link probability control for probabilistic diffusion least-mean squares over resource-constrained networks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Dallas, TX, Mar. 2010, pp. 3518–3521.

[14] S-Y. Tu and A. H. Sayed, "Optimal combination rules for adaptation and learning over netowrks," in *Proc. IEEE Int. Workshop Comput. Advances Multi-Sensor Adapt. Process. (CAMSAP)*, San Juan, Puerto Rico, Dec. 2011.

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

[16] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, Sept. 2010.

[17] A. J. Laub, *Matrix Analysis for Scientists and Engineers*, SIAM, PA, 2005.

[18] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge Univ. Press, Cambridge, UK, 1985.