DIFFUSION NETWORKS OUTPERFORM CONSENSUS NETWORKS

Sheng-Yuan Tu *and* *Ali H. Sayed*

Department of Electrical Engineering
University of California, Los Angeles

ABSTRACT

Adaptive networks consist of a collection of nodes that interact with each other on a local level and diffuse information across the network to solve estimation and inference tasks in a distributed manner. In this work, we compare the performance of two distributed estimation strategies: diffusion and consensus. Diffusion strategies allow information to diffuse more thoroughly through the network. The analysis in the paper confirms that this property has a favorable effect on the evolution of the network: diffusion networks reach lower mean-square deviation than consensus networks, and their mean-square stability is insensitive to the choice of the combination weights. In contrast, consensus networks can become unstable even if all the individual nodes are mean-square stable; this does not occur for diffusion networks: stability of the individual nodes ensures stability of the diffusion network irrespective of the topology.

Index Terms— Adaptive networks, diffusion strategy, consensus strategy, mean-square performance, combination weights.

1. INTRODUCTION

Adaptive networks consist of a collection of spatially distributed nodes that are linked together through a connection topology and that cooperate with each other through local interactions. Adaptive networks are well-suited to perform decentralized information processing and inference tasks [1–4] and to model complex behavior encountered in biological systems [5, 6].

In this article we compare the performance of two strategies for distributed estimation over networks: diffusion and consensus; the latter has been used extensively in the literature (see, e.g., [7–11]), while the former was introduced more recently in [1–4]. The two strategies differ in an important way: diffusion strategies allow information to diffuse more thoroughly through the network. We will establish analytically that diffusion networks reach lower mean-square deviation than consensus networks, and their mean-square stability is insensitive to the choice of the combination weights. In contrast, consensus networks can become unstable even if all the individual nodes are mean-square stable; this does not occur for diffusion networks: stability of the individual nodes ensures stability of the diffusion network irrespective of the network topology.

2. DISTRIBUTED STRATEGIES

Consider a connected network with N nodes (see Fig. 1). Two nodes are neighbors if they can exchange information. The neighborhood of node k is denoted by $\mathcal{N}_k$. The nodes in the network would like to estimate an unknown $M \times 1$ vector, w°. At every time i, each node k is able to observe realizations $\{d_k(i), u_{k,i}\}$ of a scalar random process $\boldsymbol{d}_k(i)$ and a $1 \times M$ vector random process $\boldsymbol{u}_{k,i}$ with zero mean and covariance matrix $R_{u,k} = \mathbb{E}\boldsymbol{u}_{k,i}^* \boldsymbol{u}_{k,i} > 0$, where $\mathbb{E}$ denotes the expectation operator. All vectors are column vectors with the exception of the regression vector, $\boldsymbol{u}_{k,i}$. The random processes $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ are related to w° via the linear regression model [12]:

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

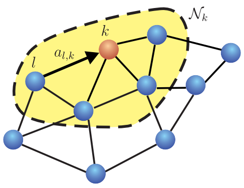

Fig. 1. A connected network showing the neighborhood of node k, denoted by $\mathcal{N}_k$. The weight $a_{l,k}$ scales the data transmitted from node l to node k over the edge linking them.

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^\circ + \boldsymbol{v}_k(i) \tag{1}$$

where $\boldsymbol{v}_k(i)$ is measurement noise with zero mean and variance $\sigma_{v,k}^2$. The noise process $\boldsymbol{v}_k(i)$ is assumed to be temporally white and spatially independent, i.e., $\mathbb{E}\boldsymbol{v}_k^*(i)\boldsymbol{v}_l(j) = \sigma_{v,k}^2 \cdot \delta_{kl} \cdot \delta_{ij}$ in terms of the Kronecker delta function; likewise for the regression process $\boldsymbol{u}_{k,i}$. The $\{\boldsymbol{v}_k(i), \boldsymbol{u}_{l,j}\}$ are also assumed to be independent of each other for all $\{k, l, i, j\}$. Relations of the form (1) can represent well a variety of physical models (e.g., [5, 6, 12]).

The objective of the network is to estimate w° in a distributed manner by seeking to minimize the global cost function:

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

In this article we compare the performance of two fully decentralized strategies for estimating w°. One strategy is the consensus strategy, which has been used in various works on distributed optimization and estimation (see, e.g., [8–11]). In the context of estimation theory, this rule (see (5) further ahead) is usually motivated by the desire to exploit the agreement properties of the original elegant consensus strategy proposed in [13] and studied further in [7, 14–16]. The other strategy we consider is the diffusion strategy, which was proposed in [1–3] and is now being applied advantageously to the solution of distributed optimization problems [4, 17, 18], and to the modeling of self-organized and complex behavior encountered in nature [5, 6]. It turns out that consensus and diffusion strategies differ in one fundamental way, which ends up influencing the manner by which the estimation errors evolve over time across the respective networks. The purpose of this article is to highlight these differences analytically and to explain the reason behind the superior performance of diffusion strategies over consensus strategies.

2.1. Consensus Strategy

The consensus strategy appears in the literature in at least two common forms (see, e.g., eqs (7.10) in [8], (1.20) in [9], and (9) in [11]):

$$\begin{aligned} \boldsymbol{w}_{k,i} = \boldsymbol{w}_{k,i-1} - \nu_k \sum_{l \in \mathcal{N}_k \setminus \{k\}} b_{l,k}(\boldsymbol{w}_{k,i-1} - \boldsymbol{w}_{l,i-1}) \\ + \mu_k \boldsymbol{u}_{k,i}^* [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}] \end{aligned} \quad (3)$$

with positive step-sizes $\{\nu_k, \mu_k\}$ for node k and nonnegative weights $\{b_{l,k}\}$. By introducing the coefficients:

$$a_{l,k} = \begin{cases} \nu_k b_{l,k}, & \text{if } l \in \mathcal{N}_k \setminus \{k\} \\ 1 - \sum_{j \in \mathcal{N}_k \setminus \{k\}} \nu_k b_{j,k}, & \text{if } l = k \end{cases} \quad (4)$$

recursion (3) can be rewritten equivalently as:

$$\boxed{\boldsymbol{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{l,k} \boldsymbol{w}_{l,i-1} + \mu_k \boldsymbol{u}_{k,i}^* [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}]} \quad (5)$$

The entry $a_{l,k}$ denotes the weight on the link connecting node l to node k (see Fig. 1). The weights $\{a_{l,k}\}$ form an $N \times N$ matrix A and satisfy

$$a_{l,k} \geq 0, \sum_{l \in \mathcal{N}_k} a_{l,k} = 1 \text{ for all } k, \text{ and } a_{l,k} = 0 \text{ if } l \notin \mathcal{N}_k \quad (6)$$

That is, the matrix A is left-stochastic and its (l, k)th entry is zero when there is no link between nodes l and k.

2.2. Diffusion Strategy

There are several variations of the diffusion strategy. We focus on the ATC strategy [2–4], which has been shown to generally outperform other variants. The ATC diffusion strategy consists of two steps. The first step in (7) involves local adaptation, where node k uses its own data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ to update its weight estimate from $\boldsymbol{w}_{k,i-1}$ to an intermediate value $\boldsymbol{\psi}_{k,i}$. The second step in (7) is a consultation step where the intermediate estimates $\{\boldsymbol{\psi}_{l,i}\}$ from the neighborhood of node k are combined through the weights $\{a_{l,k}\}$ that satisfy (6) to obtain the updated estimate $\boldsymbol{w}_{k,i}$:

$$\boxed{\begin{aligned} \boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}] \\ \boldsymbol{w}_{k,i} &= \sum_{l \in \mathcal{N}_k} a_{l,k} \boldsymbol{\psi}_{l,i} \end{aligned}} \quad (7)$$

We note that algorithms (5) and (7) can be derived and motivated from first principles by following the argument proposed in [3,4] for the distributed optimization of global cost functions of the form (2).

2.3. Comparing Consensus and Diffusion Strategies

In order to highlight the difference in the dynamics of the algorithms (5) and (7), we combine the adaptation step into the consultation step and rewrite (7) equivalently as (compare with (5)):

$$\boldsymbol{w}_{k,i} = \sum_{l \in \mathcal{N}_k} a_{l,k} \boldsymbol{w}_{l,i-1} + \sum_{l \in \mathcal{N}_k} \mu_l a_{l,k} \boldsymbol{u}_{l,i}^* [\boldsymbol{d}_l(i) - \boldsymbol{u}_{l,i}\boldsymbol{w}_{l,i-1}] \quad (8)$$

Note that the first terms on the right hand side of (5) and (8) are the same. For the second terms, only the variables $\boldsymbol{w}_{k,i-1}$, $\boldsymbol{d}_k(i)$, and $\boldsymbol{u}_{k,i}$ appear in the consensus strategy (5), while the diffusion strategy (8) incorporates the influence of the estimates $\{\boldsymbol{w}_{l,i-1}\}$ and the data $\{\boldsymbol{d}_l(i), \boldsymbol{u}_{l,i}\}$ from the neighborhood of node k into the update of $\boldsymbol{w}_{k,i}$. This fact has important implications on the evolution of the weight-error vectors across the network in the diffusion case. Note that the diffusion strategy is able to incorporate additional information into its processing steps *without* being more complex than the consensus strategy. These two strategies have *exactly* the *same* computational complexity and require sharing the same amount of data, as can be ascertained by comparing the actual implementations (5) and (7). Note that the diffusion strategy (7) first generates an intermediate state $\boldsymbol{\psi}_{k,i}$, which is subsequently used in the final update. This important ordering of the calculations has a critical influence on the performance of the algorithms, as we now reveal.

Table 1. Variables for diffusion and consensus implementations.

	Diffusion (7)	**Consensus** (5)
$\boldsymbol{\mathcal{B}}_i$	$\mathcal{A}^T(I_{NM} - \mathcal{M}\boldsymbol{\mathcal{R}}_i)$	$\mathcal{A}^T - \mathcal{M}\boldsymbol{\mathcal{R}}_i$
$\mathcal{B} \triangleq \mathbb{E}\boldsymbol{\mathcal{B}}_i$	$\mathcal{A}^T(I_{NM} - \mathcal{M}\mathcal{R})$	$\mathcal{A}^T - \mathcal{M}\mathcal{R}$
$\boldsymbol{y}_i$	$\mathcal{A}^T\mathcal{M}\boldsymbol{s}_i$	$\mathcal{M}\boldsymbol{s}_i$
$\mathcal{Y} \triangleq \mathbb{E}\boldsymbol{y}_i\boldsymbol{y}_i^*$	$\mathcal{A}^T\mathcal{M}\mathcal{S}\mathcal{M}\mathcal{A}$	$\mathcal{M}\mathcal{S}\mathcal{M}$

3. PERFORMANCE ANALYSIS

3.1. Recursions for the Error Vectors

Let $\tilde{\boldsymbol{w}}_{k,i} \triangleq w^\circ - \boldsymbol{w}_{k,i}$ denote the error vector for node k. We collect all error vectors and step-sizes across the network for $k = 1, 2, \ldots, N$ into the block vector $\tilde{\boldsymbol{w}}_i = \text{col}\{\tilde{\boldsymbol{w}}_{k,i}\}$ and the diagonal matrix $\mathcal{M} = \text{diag}\{\mu_k I_M\}$. We also introduce the extended combination matrix: $\mathcal{A} = A \otimes I_M$, where $\otimes$ denotes the Kronecker product. From (5) or (7) along with model (1), some algebra similar to [3] can show that the recursions for $\tilde{\boldsymbol{w}}_i$ for the consensus and diffusion strategies are special cases of a general recursion of the form:

$$\boxed{\tilde{\boldsymbol{w}}_i = \boldsymbol{\mathcal{B}}_i \tilde{\boldsymbol{w}}_{i-1} - \boldsymbol{y}_i} \quad (9)$$

where $\boldsymbol{\mathcal{B}}_i$ and $\boldsymbol{y}_i$ are shown in Table 1 with $\boldsymbol{\mathcal{R}}_i \triangleq \text{diag}\{\boldsymbol{u}_{k,i}^*\boldsymbol{u}_{k,i}\}$ and $\boldsymbol{s}_i \triangleq \text{col}\{\boldsymbol{u}_{k,i}^*\boldsymbol{v}_{k,i}\}$.

3.2. Mean Stability

Since the regressors $\{\boldsymbol{u}_{k,i}\}$ are temporally white and spatially independent, then $\boldsymbol{\mathcal{B}}_i$ in Table 1 is independent of $\tilde{\boldsymbol{w}}_{i-1}$. Taking expectation of both sides of (9), we find that the mean of $\tilde{\boldsymbol{w}}_i$ evolves in time according to the recursion:

$$\boxed{\mathbb{E}\tilde{\boldsymbol{w}}_i = \mathcal{B} \cdot \mathbb{E}\tilde{\boldsymbol{w}}_{i-1}} \quad (10)$$

where $\mathcal{B} = \mathbb{E}\boldsymbol{\mathcal{B}}_i$ is also shown in Table 1 with $\mathcal{R} \triangleq \mathbb{E}\boldsymbol{\mathcal{R}}_i = \text{diag}\{R_{u,k}\}$. The condition to ensure mean stability of the network is therefore to select step-sizes that ensure

$$\boxed{\rho(\mathcal{B}) < 1} \quad (11)$$

where $\rho(\cdot)$ denotes the spectral radius of its matrix argument. We observe from the $\mathcal{B}$ in Table 1 that the mean stability of the consensus strategy is sensitive to the choice of the combination matrix A. This is not the case for the diffusion strategy because it can be verified that, for any left-stochastic matrix A, the matrix $\mathcal{B} = \mathcal{A}^T(I_{NM} - \mathcal{M}\mathcal{R})$ is stable whenever $(I_{NM} - \mathcal{M}\mathcal{R})$ is stable. Therefore, we can select the step-sizes to satisfy $\mu_k < 2/\rho(R_{u,k})$ for the diffusion strategy and ensure its mean stability *regardless* of the matrix A.

Note further that if each node were to run an LMS filter individually to estimate w°, then the filter would take the form

$$\boldsymbol{w}_{k,i} = \boldsymbol{w}_{k,i-1} + \mu_k \boldsymbol{u}_{k,i}^* [\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}] \tag{12}$$

Each of these individual filters is mean stable if its step-size satisfies the same condition $\mu_k < 2/\rho(R_{u,k})$ [12]. Therefore, we conclude that if the individual nodes are mean stable, then the diffusion network will also be mean stable. The same conclusion is not necessarily true for consensus networks: all nodes can be stable in the mean and yet the consensus network can become unstable. Figure 2 illustrates this situation for two cases. In one case, both nodes are mean stable, and yet the consensus network becomes unstable. In the second case, one of the nodes is unstable in which case the consensus strategy is always unstable while diffusion can stabilize the network.

3.3. Mean-Square Stability

Let $\Sigma \geq 0$ denote an arbitrary Hermitian matrix. Some algebra will establish the following variance relation for small step-sizes [3]:

$$\boxed{\mathbb{E}\|\tilde{\boldsymbol{w}}_i\|_\Sigma^2 \approx \mathbb{E}\|\tilde{\boldsymbol{w}}_{i-1}\|_{\mathcal{B}^*\Sigma\mathcal{B}}^2 + \text{Tr}(\Sigma\mathcal{Y})} \tag{13}$$

where $\mathcal{Y} = \mathbb{E}\boldsymbol{y}_i\boldsymbol{y}_i^*$ appears in Table 1 and $\mathcal{S} \triangleq \mathbb{E}\boldsymbol{s}_i\boldsymbol{s}_i^* = \text{diag}\{\sigma_{v,k}^2 R_{u,k}\}$. Then, the mean-square stability of the network is guaranteed for step-sizes that are sufficiently small and satisfy (11). Again, we find that the mean-square stability of the consensus network is sensitive to the choice of A.

3.4. Mean-Square Performance

The network mean-square deviation (MSD) is a measure that assesses how well the network estimates the weight vector, w°:

$$\text{MSD} \triangleq \lim_{i\to\infty} \frac{1}{N}\sum_{k=1}^{N} \mathbb{E}\|\tilde{\boldsymbol{w}}_{k,i}\|^2 \tag{14}$$

where $\|\cdot\|$ denotes the Euclidean norm for vectors. Iterating (13), we can obtain a series expression for the network MSD [3]:

$$\boxed{\text{MSD} = \frac{1}{N}\sum_{j=0}^{\infty} \text{Tr}[\mathcal{B}^j\mathcal{Y}(\mathcal{B}^*)^j]} \tag{15}$$

We use this expression next to show that diffusion networks achieve lower MSDs than consensus networks.

4. COMPARING DIFFUSION AND CONSENSUS

We assume in this section that all nodes use the same step-size, $\mu_k = \mu$, and observe data arising from the same statistical distribution, $R_{u,k} = R_u$.

4.1. Stability and Convergence Rate

Since stability and convergence rate depend only on the spectral radius of $\mathcal{B}$, we examine more closely the eigen-structure of $\mathcal{B}$. Let r_k and s_k $(k = 1, \ldots, N)$ denote the right and left eigenvectors of A^T corresponding to the eigenvalue $\lambda_k(A)$. Note that $\rho(A) = 1$ for left-stochastic matrices. Let z_m $(m = 1, \ldots, M)$ denote the orthonormal eigenvectors ($z_n^* z_m = \delta_{mn}$) of the covariance matrix R_u associated with the eigenvalues $\lambda_m(R_u) > 0$. The following result characterizes the eigen-structure of the matrix $\mathcal{B}$ in terms of the eigen-structures of $\{A^T, R_u\}$.

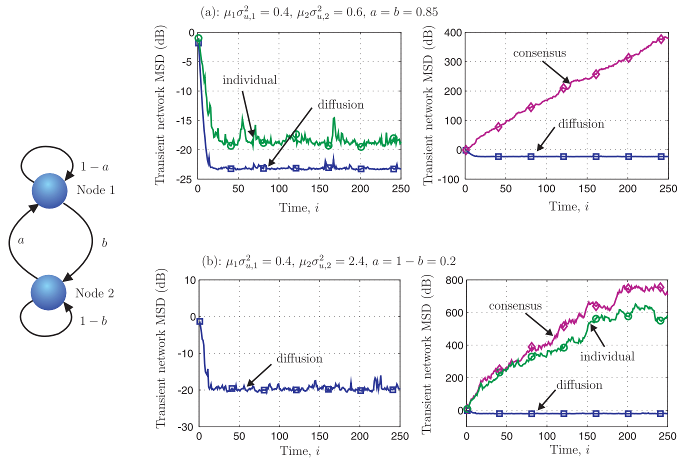

Fig. 2. Transient network MSD over time with $N = 2$: (a) $\mu_1\sigma_{u,1}^2 = 0.4$, $\mu_2\sigma_{u,2}^2 = 0.6$, and $a = b = 0.85$. As seen in the right plot, the consensus strategy is unstable even though the individual nodes are stable; (b) $\mu_1\sigma_{u,1}^2 = 0.4$, $\mu_2\sigma_{u,2}^2 = 2.4$, and $a = 1 - b = 0.2$. As seen in the right plot, the diffusion strategy is stable even when the non-cooperative and consensus strategies are unstable.

Lemma 1. *The matrices $\mathcal{B}$ from Table 1 for the diffusion and consensus strategies have the same eigenvector pairs $\{r_{k,m}^b, s_{k,m}^b\}$:*

$$r_{k,m}^b = r_k \otimes z_m, \quad s_{k,m}^b = s_k \otimes z_m \tag{16}$$

The corresponding eigenvalues, $\lambda_{k,m}(\mathcal{B})$, are given by:

$$\lambda_{k,m}(\mathcal{B}) = \begin{cases} \lambda_k(A)(1 - \mu\lambda_m(R_u)) & \textit{(diffusion)} \\ \lambda_k(A) - \mu\lambda_m(R_u) & \textit{(consensus)} \end{cases} \tag{17}$$

Proof. The results follow by noting that we can write $\mathcal{B}_{\text{diff}} = A^T \otimes (I_M - \mu R_u)$ and $\mathcal{B}_{\text{cons}} = A^T \otimes I_M - I_N \otimes \mu R_u$. $\square$

Theorem 1. *The diffusion strategy is more stable and converges faster than the consensus strategy, i.e.,*

$$\rho(\mathcal{B}_{\textit{diff}}) \leq \rho(\mathcal{B}_{\textit{cons}}) \tag{18}$$

Proof. The result follows from (17). $\square$

4.2. Network MSD

The network MSD in (15) depends on $\mathcal{B}$ in a nontrivial manner. Nevertheless, under certain conditions, we can simplify the MSD expression by using the eigen-structure of $\mathcal{B}$.

Lemma 2. *If the matrix A is diagonalizable and the right eigenvectors $\{r_k\}$ of A satisfy*

$$r_l^* r_k \ll \|r_k\|^2 \tag{19}$$

for all k and l, then the MSD from (15) can be approximated by:

$$\text{MSD} \approx \sum_{k=1}^{N}\sum_{m=1}^{M} \frac{\|r_k\| \cdot s_{k,m}^{b*}\mathcal{Y}s_{k,m}^b}{N \cdot (1 - |\lambda_{k,m}(\mathcal{B})|^2)} \tag{20}$$

Table 2. Combination rules with $a_{l,k} = 0$ if $l \notin \mathcal{N}_k$.

Name	Rule
Relative-variance [19]	$a_{l,k} = \sigma_{v,l}^{-2} / \sum_{j \in \mathcal{N}_k} \sigma_{v,j}^{-2}$
Uniform	$a_{l,k} = 1/n_k$
Metropolis [16]	$a_{l,k} = \begin{cases} 1 - \sum_{j \neq k} a_{k,j}, & l = k \\ 1/\max\{n_k, n_l\}, & l \in \mathcal{N}_k \setminus \{k\} \end{cases}$

Proof. The result follows by substituting eigen-decomposition of $\mathcal{B}$ from Lemma 1 into the MSD expression in (15). □

Note that any symmetric A is diagonalizable and satisfies condition (19) since $r_l^* r_k = \delta_{kl}$. We then obtain the following conclusion.

Theorem 2. *Under the conditions of Lemma 2, the diffusion strategy achieves lower network MSD than the consensus strategy, i.e.,*

$$\text{MSD}_{\textit{diff}} \leq \text{MSD}_{\textit{cons}} \tag{21}$$

Proof. The argument uses the expression for $\mathcal{Y}$ from Table 1, the results of Lemma 1, and (20) to establish (21). □

5. SIMULATION RESULTS

We consider a network with 20 nodes and random topology. The regression covariance matrix R_u is diagonal of size $M = 10$ with entries randomly generated from $[2, 4]$, and the noise variances $\{\sigma_{v,k}^2\}$ are randomly generated over $[-30, -10]$ dB. The step-size μ is set to $\mu = 0.05$. The transient network MSD over time is shown on the left hand side of Fig. 3 with three possible combination rules: relative-variance [19], uniform, and Metropolis [16] (see Table 2). We observe that the diffusion strategy outperforms the consensus strategy in all three cases. We further show the steady-state MSD at each node on the right hand side of Fig. 3. We observe that the diffusion strategy has lower MSD at each node compared to the consensus strategy. These observations are in agreement with the results predicted by the theoretical analysis.

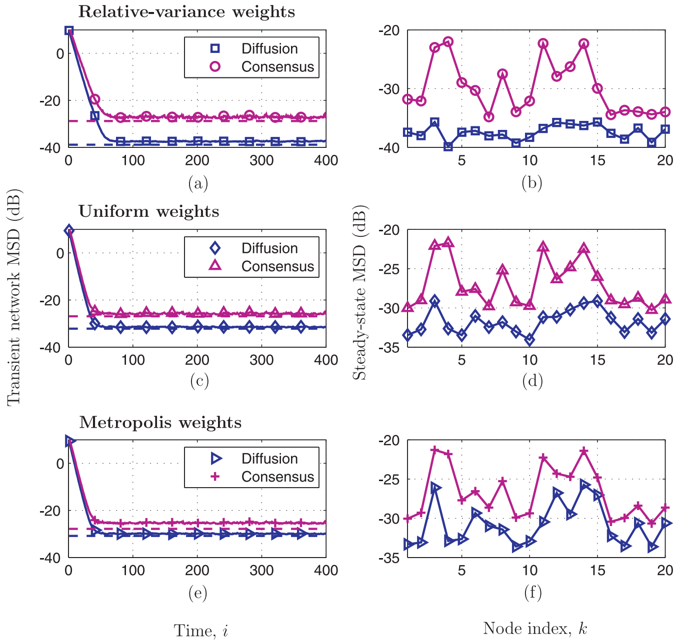

Fig. 3. Transient network MSD over time (left) and steady-state MSD at nodes (right) for (a)-(b) the relative-variance, (c)-(d) uniform, and (e)-(f) Metropolis rules. The dashed lines in in the plots on the left indicate the theoretical value (15).

6. REFERENCES

[1] C. G. Lopes and A. H. Sayed, "Distributed processing over adaptive networks," *Proc. Adaptive Sensor Array Processing Workshop*, pp. 1–5, MIT Lincoln Laboratory, MA, Jun. 2006.

[2] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. on Signal Processing*, vol. 56, no. 7, pp. 3122–3136, Jul. 2008.

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

[4] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," to apear in *IEEE Trans. on Signal Processing*, 2012. Also available on arXiv:1111.0034, Oct. 2011.

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

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

[7] J. N. Tsitsiklis, J. N. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimzation algorithms," *IEEE Trans. on Autom. Control*, vol. 31, no. 9, pp. 803–812, Sep. 1986.

[8] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Singapore, 1997.

[9] A. Nedic and A. Ozdaglar, "Cooperative distributed multi-agent optimization," in the book *Convex Optimization in Signal Processing and Communications,* Y. Eldar and D. Palomar *(Eds.),* Cambridge University Press, pp. 340–386, 2009.

[10] 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, Nov. 2010.

[11] S. Kar and J. M. F. Moura, "Convergence rate analysis of distributed gossip (linear parameter) estimation: Fundamental limits and tradeoffs," *IEEE J. Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 674–690, Aug. 2011.

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

[13] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical Association*, vol. 69, pp. 118–121, 1974.

[14] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. on Automatic Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

[15] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Automatic Control*, vol. 49, pp. 1520–1533, Sep. 2004.

[16] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," *Proc. Int. Symp. Information Processing Sensor Networks*, pp. 63–70, Los Angeles, CA, Apr. 2005.

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

[18] K. Srivastava and A. Nedic, "Distributed asynchronous constrained stochastic optimization," *IEEE J. Selected Topics on Signal Processing*, vol. 5, no. 4, pp. 772–790, Aug. 2011.

[19] S. Y. Tu and A. H. Sayed, "Optimal combination rules for adaptation and learning over networks," *Proc. IEEE CAMSAP*, pp. 317–320, San Juan, Puerto Rico, Dec. 2011.