Diffusion Least-Mean Squares Over Adaptive Networks: Formulation and Performance Analysis

Cassio G. Lopes, *Student Member, IEEE*, and Ali H. Sayed, *Fellow, IEEE*

***Abstract*—We formulate and study distributed estimation algorithms based on diffusion protocols to implement cooperation among individual adaptive nodes. The individual nodes are equipped with local learning abilities. They derive local estimates for the parameter of interest and share information with their neighbors only, giving rise to peer-to-peer protocols. The resulting algorithm is distributed, cooperative and able to respond in real time to changes in the environment. It improves performance in terms of transient and steady-state mean-square error, as compared with traditional noncooperative schemes. Closed-form expressions that describe the network performance in terms of mean-square error quantities are derived, presenting a very good match with simulations.**

***Index Terms*—Adaptive networks, consensus, cooperation, diffusion algorithm, distributed processing, distributed estimation.**

I. Introduction

CONSIDER a network of nodes observing temporal data arising from different spatial sources with possibly different statistical profiles. The objective is to enable the nodes to estimate a vector of parameters of interest from the observed data. In a centralized approach, the data or local estimates from all nodes would be conveyed to a central processor where they would be fused and the vector of parameters estimated. Such an approach calls for sufficient communications resources to transmit the data back and forth between the nodes and the central processor, which limits the autonomy of the network, besides adding a critical point of failure in the network due to the presence of a central node. In addition, a centralized solution may limit the ability of the nodes to adapt in real-time to time varying statistical profiles. In other words, tracking performance is degraded.

Alternatively, each node in the network could function as an individual adaptive filter whose aim is to estimate the parameter of interest through local observations [1]–[3]. These individual estimates across the nodes could then be locally fused with their neighboring estimates in the network in order to obtain an estimate that is influenced by the data at the nearby nodes; for instance, by resorting to consensus implementations. However, the several intermediate averaging iterations used to fuse the local estimates are generally performed at a different time scale than the local adaptive processing, and after sufficient convergence has been attained. Such procedures drain substantial energy and communication resources and also tend to limit the tracking ability of the network to respond in real time to statistical changes in the data. In a solution along these lines, the individual filters at the nodes are responding mainly to local temporal data. Their ability to exploit and respond in real-time to the spatial profile of the data across the nodes is decreased.

This paper proposes and studies a cooperation strategy that adopts a peer-to-peer diffusion protocol, in which nodes from the same neighborhood are allowed to communicate with each other at every iteration. At each node, estimates exchanged with neighboring nodes are fused and promptly fed into the local adaptive filter. In other words, information is diffused among neighboring nodes so that the estimate at each node is a function of both its temporal data as well as the spatial data across the neighbors. In doing so, an adaptive network structure is obtained where the structure as a whole is able to respond in real-time to the temporal and spatial variations in the statistical profile of the data [4]–[6]. Different adaptation or learning rules at the nodes, allied with different cooperation protocols, give rise to adaptive networks of various complexities and potential. Our formulation is useful in several problems involving estimation and event detection from multiple nodes collecting space–time data [7]–[13].

The subsequent sections in the paper detail the mathematical model and study the performance of the resulting adaptive network. Explicit expressions are derived for the mean-square performance. Some interesting questions that arise are, Can individual adaptive nodes benefit from cooperation? And how to ally cooperation with adaptation? It should be noted that studying the performance of such networks of adaptive nodes is rather challenging in comparison to the study of the performance of single stand-alone adaptive filters. This is because, in the networked case, the adaptive nodes influence each other. Moreover, the data spatial profile varies across nodes. The contributions of this work are therefore to motivate and propose a diffusion adaptive network, and to study its performance by deriving closed-form expressions for both local (at the node level) and global (at the network level) learning curves and mean-square deviations and errors for Gaussian data and sufficiently small step sizes. In addition, the paper indicates how to add an additional layer of adaptation to the network where nodes are

Manuscript received July 31, 2007; revised November 20, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. William A. Sethares. This material was based on work supported in part by the National Science Foundation under awards ECS-0601266 and ECS-0725441. The work of the C. G. Lopes was also supported by a fellowship from CAPES, Brazil, under award 1168/01-0. A brief version of this work appeared in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing* (ICASSP), Honolulu, HI, April 2007.

The authors are with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA (e-mail: cassio@ee.ucla.edu; sayed@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2008.917383

able to learn which nodes are more or less reliable and to assign different weights to information received from them.

The paper is organized as follows. In Sections II and III, we formulate a global estimation problem and derive a distributed adaptive solution that relies on cooperative diffusion protocols [11]. Sections IV–VI analyze the transient performance of the diffusion protocol, deriving closed form expressions for the network global and local learning behavior. In Section VII, we address the problem of network stability and show the stabilizing effect of the cooperative diffusion protocol. Section VIII deals with the steady-state performance of the adaptive network. We provide closed form expressions for the mean-square deviation (MSD) and the excess mean-square error (EMSE). Finally, in Section IX we propose an adaptive variant of the diffusion protocol inspired by recent work in convex combination of adaptive filters [14]–[17]; Section X closes the work and points directions for future extensions.

II. The Estimation Problem

We would like to estimate an $M\times 1$ unknown vector w^o from measurements collected at N nodes spread over a network (see Fig. 1). Each node k has access to time realizations $\{d_k(i), u_{k,i}\}$, $k = 1,\ldots,N$, of zero-mean random data $\{\boldsymbol{d}_k, \boldsymbol{u}_k\}$, with $d_k(i)$ a scalar measurement and $u_{k,i}$ a $1\times M$ regression row vector; both at time i. The estimation problem can be formulated as follows. We collect the regression and measurement data across all N nodes into two global matrices

$$\boldsymbol{U}_c \triangleq \mathrm{col}\{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_N\} \ (N\times M) \tag{1a}$$

$$\boldsymbol{d} \triangleq \mathrm{col}\{\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_N\} \ (N\times 1) \tag{1b}$$

and then seek the $M\times 1$ vector w that solves

$$\min_{w} \ E\|\boldsymbol{d}-\boldsymbol{U}_c w\|^2 \tag{2}$$

where E is the expectation operator. The optimal solution w^o of (2) satisfies the orthogonality condition [3]

$$E\,\boldsymbol{U}_c^*(\boldsymbol{d}-\boldsymbol{U}_c w^o) = 0 \tag{3}$$

so that w^o is the solution to the *normal equations*

$$R_{du}^c = R_u^c w^o \tag{4}$$

which are defined in terms of the correlation and cross-correlation quantities

$$R_u^c = E\,\boldsymbol{U}_c^*\boldsymbol{U}_c \ (M\times M) \quad \text{and} \quad R_{du}^c = E\,\boldsymbol{U}_c^*\boldsymbol{d} \ (M\times 1). \tag{5}$$

For later reference, we also introduce the block diagonal matrix

$$\boldsymbol{U} \triangleq \mathrm{diag}\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_N\} \qquad (N\times NM) \tag{6}$$

which is related to $\boldsymbol{U}_c$ as

$$\boldsymbol{U}^c = \boldsymbol{U}Q \tag{7}$$

where

$$\boxed{Q = \mathrm{col}\{I_M, I_M, \ldots, I_M\}} \tag{8}$$

is $NM\times M$, with I_M the $M\times M$ identity matrix.

III. Diffusion LMS

Our objective is to develop an adaptive distributed procedure that approximates the solution w^o of (4) and delivers a good estimate of that vector to every node in the network. To design the adaptive estimation protocol we first need to choose a cooperation strategy [4]. In several scenarios, nodes in a network have access only to their peer neighbors; therefore, peer-to-peer protocols are often preferred [7]; they lead to savings in communications and energy resources. We resort to diffusion protocols where every node k in the network continuously combines estimates from its neighborhood. Specifically, at any given time $i-1$, we assume that node k has access to a set of unbiased estimates $\{\psi_k^{(i-1)}\}_{k\in\mathcal{N}_k}$ from its neighborhood $\mathcal{N}_k$, which is defined as the set of all nodes linking to it, including itself. The estimates $\{\psi_k^{(i-1)}\}_{k\in\mathcal{N}_k}$ are generally noisy versions of w^o, say

$$\psi_k^{(i-1)} = w^o - \widetilde{\psi}_k^{(i-1)} \tag{9}$$

for some error $\widetilde{\psi}_k^{(i-1)}$. These local estimates are fused at node k, yielding

$$\phi_k^{(i-1)} = f_k\left(\psi_\ell^{(i-1)}; \ell\in\mathcal{N}_{k,i-1}\right) \tag{10}$$

for some local combiner function f_k. Observe that we are allowing the neighborhood $\mathcal{N}_k$ to be time dependent as well. In this work, we shall employ linear combiners, and replace f_k by some weighted combination, say

$$\phi_k^{(i-1)} = \sum_{\ell\in\mathcal{N}_{k,i-1}} c_{k\ell}\psi_\ell^{(i-1)} \tag{11}$$

for some combination coefficients $\{c_{k\ell}\geq 0\}$ to be determined. The aggregate estimate $\{\phi_k^{(i-1)}\}$ at node k can be interpreted as some weighted least-squares estimate of w^o given the estimates $\{\psi_\ell^{(i-1)}\}$ at the neighbors of node k. Thus, note that if we collect the estimates $\{\psi_\ell^{(i-1)}\}$ into a column vector

$$\psi_{\mathcal{N}_k} \triangleq \mathrm{col}\{\psi_\ell^{(i-1)}\}_{\ell\in\mathcal{N}_{k,i-1}} \tag{12}$$

we may formulate a local weighted least-squares problem of the form

$$\min_{\phi}\|\psi_{\mathcal{N}_k} - Q\phi_k\|_{C_k}^2 \tag{13}$$

where Q is given by (8) and the weighting matrix is

$$C_k = \mathrm{diag}\{c_{k1}I_M, \ldots, c_{k|\mathcal{N}_{k,i-1}|}I_M\}.$$

Then the solution to (13) is easily found to be

$$\phi_k^{(i-1)} = \sum_{\ell\in\mathcal{N}_{k,i-1}} \frac{c_{k\ell}}{\sum_{r\in\mathcal{N}_{k,i-1}} c_{kr}}\psi_\ell. \tag{14}$$

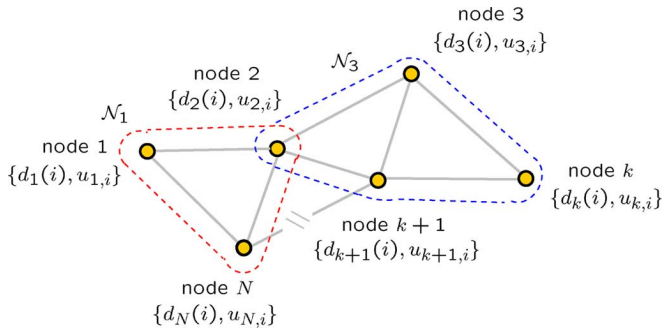

Fig. 1. Distributed network with N nodes.

The coefficients $c_{k\ell}$ give rise to a combination matrix $C = [c_{k\ell}]$ that carries information about the network topology: a nonzero entry $c_{k\ell}$ means that nodes k and ℓ are connected. If we redefine the combining coefficients as

$$c_{k\ell} \leftarrow \frac{c_{k\ell}}{\sum_{r\in\mathcal{N}_{k,i-1}} c_{kr}} \tag{15}$$

then (11) is obtained with

$$\sum_{\ell} c_{k\ell} = 1\,, \quad \ell \in \mathcal{N}_{k,i-1} \ \ \forall k. \tag{16}$$

Because of (16), we shall assume that C is a stochastic matrix hereafter. Note that the aggregation step helps fuse information from nodes across the network (and not just from the neighborhood $\mathcal{N}_k$) into node k. This is because generally every node in $\mathcal{N}_k$ will have a different neighborhood for connected topologies—see Fig. 1.

Once we have an aggregate estimate $\phi_k^{(i-1)}$ for w^o, and in order to foster adaptivity, we subsequently fuse the resulting estimate $\phi_k^{(i-1)}$ into the local adaptive process, so that it can rapidly respond to changes in its neighborhood and update it to $\psi_k^{(i)}$ (see Fig. 2). Analysis and simulation will show that this scheme leads to a robust distributed adaptive system that achieves smaller error levels in steady-state than its noncooperative counterpart (where each node in the network adapts independently of other nodes and of aggregation).

The proposed diffusion strategy can therefore be described in general terms as follows:

$$\phi_k^{(i-1)} = f_k\left(\psi_\ell^{(i-1)}; \ell \in \mathcal{N}_{k,i-1}\right) \tag{17}$$

$$\psi_k^{(i)} = \phi_k^{(i-1)} + \mu_k u_{k,i}^*\left(d_k(i) - u_{k,i}\phi_k^{(i-1)}\right) \tag{18}$$

for local step sizes μ_k. As mentioned before, the combiners may be nonlinear, or even time-variant, to reflect changing topologies or to respond more efficiently to nonstationary conditions. For instance, in Section IX, we propose an adaptive protocol inspired by earlier work on convex combination of filters [14]–[17], which we extend to the multiple data sources case. The neighborhoods $\mathcal{N}_k$ may also be time-variant. The resulting adaptive network is a peer-to-peer estimation framework that is robust to node and link failures and exploits network connectivity.

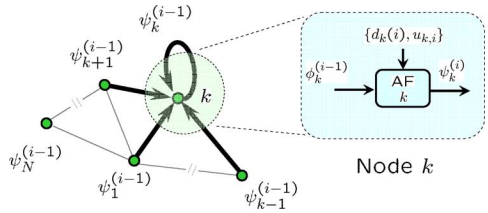

Fig. 2. Network with a diffusion cooperation strategy.

In order to illustrate the technique, we explore a linear combiner model allied with an LMS-type local adaptive rule. The strategy is summarized as follows:

$$\boxed{\begin{aligned} \phi_k^{(i-1)} &= \sum_{\ell\in\mathcal{N}_{k,i-1}} c_{k\ell}\,\psi_\ell^{(i-1)}\,, \quad \phi_k^{(-1)} = 0 && \text{(19a)}\\ \psi_k^{(i)} &= \phi_k^{(i-1)} + \mu_k u_{k,i}^*\left(d_k(i) - u_{k,i}\phi_k^{(i-1)}\right) && \text{(19b)} \end{aligned}}$$

for a set of *local* combiners $c_{k\ell}$ satisfying (16).

IV. Network Global Model

Algorithm (19) embeds the combined effect of several interconnected adaptive filter updates, in addition to the network topology. Hence, performance analysis tends to be challenging. The derivations and analysis in the remainder of the paper will reveal some interesting insights on the role of cooperation and network topology on system performance.

We resort to state-space representations. We introduce the global quantities

$$\boldsymbol{\psi}^i \triangleq \mathrm{col}\{\boldsymbol{\psi}_1^{(i)}, \ldots, \boldsymbol{\psi}_N^{(i)}\}\,, \quad \boldsymbol{\phi}^{i-1} \triangleq \mathrm{col}\{\boldsymbol{\phi}_1^{(i-1)}, \ldots, \boldsymbol{\phi}_N^{(i-1)}\}$$
$$\boldsymbol{U}_i \triangleq \mathrm{diag}\{\boldsymbol{u}_{1,i}, \ldots, \boldsymbol{u}_{N,i}\}\,, \quad \boldsymbol{d}_i = \mathrm{col}\{\boldsymbol{d}_1(i), \ldots, \boldsymbol{d}_N(i)\}$$

in terms of the stochastic quantities whose realizations appear in (19). Let

$$D = \mathrm{diag}\{\mu_1 I_M, \mu_2 I_M, \ldots, \mu_N I_M\} \qquad (NM \times NM) \tag{20}$$

be a diagonal matrix collecting the local step sizes. The measurements are assumed to obey a traditional model of the form [1], [3], [18], [19],

$$\boldsymbol{d}_k(i) = \boldsymbol{u}_{k,i} w^o + \boldsymbol{v}_k(i) \tag{21}$$

where $\boldsymbol{v}_k(i)$ is background noise, assumed independent over time and space and with variance $\sigma_{v,k}^2$. Linear models of the form (21) are able to capture or approximate well many input-output relations for estimation purposes [3], [18]. Using (21), we can write

$$\boxed{\boldsymbol{d}_i = \boldsymbol{U}_i w^{(o)} + \boldsymbol{v}_i} \tag{22}$$

where $w^{(o)} = Q w^o$ and

$$\boldsymbol{v}_i = \mathrm{col}\{\boldsymbol{v}_1(i), \ldots, \boldsymbol{v}_N(i)\}.$$

With these relations, expressions (19) admit the following global representation:

$$\begin{aligned}\boldsymbol{\phi}^{i-1} &= G\boldsymbol{\psi}^{i-1}\\ \boldsymbol{\psi}^{i} &= \boldsymbol{\phi}^{i-1} + D\boldsymbol{U}_i^{*}\big(\boldsymbol{d}_i - \boldsymbol{U}_i\boldsymbol{\phi}^{i-1}\big)\end{aligned} \tag{23}$$

or, in a more compact state-space form:

$$\boxed{\boldsymbol{\psi}^{i} = G\boldsymbol{\psi}^{i-1} + D\boldsymbol{U}_i^{*}\big(\boldsymbol{d}_i - \boldsymbol{U}_i G\boldsymbol{\psi}^{i-1}\big)} \tag{24}$$

where $G = C \otimes I_M$ is the $NM \times NM$ transition matrix and C is the $N \times N$ diffusion combination matrix with entries $[c_{k\ell}]$. Recall that C satisfies $Cq_N = q_N$, where $q_N \triangleq \mathrm{col}\{1, \ldots, 1\}$. Possible choices for the combiner C are the Metropolis, the Laplacian and the nearest neighbor rules [20]–[22]. The Metropolis rule is defined as follows. Let n_k and n_ℓ denote the degree for nodes k and ℓ, i.e., $n_k = |\mathcal{N}_k|$, and choose

$$\begin{cases} c_{k\ell} = \frac{1}{\max(n_k, n_\ell)}, & \text{if } k \neq \ell \text{ are linked}\\ c_{k\ell} = 0, & \text{for } k \text{ and } \ell \text{ not linked}\\ c_{kk} = 1 - \sum\limits_{\ell \in \mathcal{N}_k/k} c_{k\ell}, & \text{for } k = \ell. \end{cases} \tag{25}$$

The Laplacian rule is given by

$$C = I_N - \kappa\mathcal{L} \tag{26}$$

where $\mathcal{L} = \mathcal{D} - A_d$, with $\mathcal{D} = \mathrm{diag}\{n_1, \ldots, n_N\}$, $\kappa = 1/n_{\max}$ and A_d is the $N \times N$ network adjacent matrix formed as

$$[A_d]_{k\ell} = \begin{cases} 1, & \text{if } k \text{ and } \ell \text{ are linked}\\ 0, & \text{otherwise.}\end{cases} \tag{27}$$

By definition, a node is linked to itself, i.e., $[A_d]_{kk} = 1$. For the nearest neighbor rule, the combiner matrix C is defined as

$$\begin{cases} c_{k\ell} = \frac{1}{|\mathcal{N}_k|}, & \ell \in \mathcal{N}_k\\ c_{k\ell} = 0, & \text{otherwise.}\end{cases}$$

V. Mean Transient Analysis

We are interested in studying the transient behavior of cooperative systems governed by equations of the form (24). As is well known, it is rather challenging to study the performance of single stand-alone adaptive filters. Several simplifying assumptions have been traditionally adopted in the literature to gain insight into the performance of such adaptive algorithms. The challenges are compounded in the adaptive network case because we now face a dynamic and interconnected collection of nodes that influence each other's behavior. To proceed with the analysis we shall therefore introduce similar assumptions to what has been used before in the adaptive literature, and rely on them to derive useful performance measures. Simulations will show that the results obtained in this manner match well with real performance for sufficiently small step sizes.

We study initially the mean behavior of the network and show how cooperation has a stabilizing effect on the network. Thus, introduce the global weight error vector

$$\widetilde{\boldsymbol{\psi}}^{i} \triangleq w^{(o)} - \boldsymbol{\psi}^{i}. \tag{28}$$

Noting that $Gw^{(o)} = w^{(o)}$, using the global data model (22) and subtracting $w^{(o)}$ from the left side and $Gw^{(o)}$ from the right side of (24), we get

$$\begin{aligned}\widetilde{\boldsymbol{\psi}}^{i} &= Gw^{(o)} - G\boldsymbol{\psi}^{i-1} - D\boldsymbol{U}_i^{*}\left(\boldsymbol{U}_i w^{(o)} + \boldsymbol{v}_i - \boldsymbol{U}_i G\boldsymbol{\psi}^{i-1}\right)\\ &= G\widetilde{\boldsymbol{\psi}}^{i-1} - D\boldsymbol{U}_i^{*}\left(\boldsymbol{U}_i G\widetilde{\boldsymbol{\psi}}^{i-1} + \boldsymbol{v}_i\right)\end{aligned} \tag{29}$$

or, equivalently

$$\boxed{\widetilde{\boldsymbol{\psi}}^{i} = (I_{NM} - D\boldsymbol{U}_i^{*}\boldsymbol{U}_i)\,G\widetilde{\boldsymbol{\psi}}^{i-1} - D\boldsymbol{U}_i^{*}\boldsymbol{v}_i} \tag{30}$$

Assuming temporal and spatial independence of the regression data $\{\boldsymbol{u}_{k,i}\}$ and taking expectations of both sides of (30) leads to

$$\boxed{E\widetilde{\boldsymbol{\psi}}^{i} = \big(I_{NM} - DR_u\big)G\,E\widetilde{\boldsymbol{\psi}}^{i-1}} \tag{31}$$

where $R_u = \mathrm{diag}\{R_{u,1}, \ldots, R_{u,N}\}$ is block diagonal and $R_{u,k} = E\boldsymbol{u}_{k,i}^{*}\boldsymbol{u}_{k,i}$. Henceforth, for stability in the mean we must have

$$\boxed{\big|\lambda\big(BG\big)\big| < 1} \tag{32}$$

with $B = (I_{NM} - DR_u)$. In other words, the spectrum of $(I_{NM} - DR_u)G$ must be strictly inside the unit disc. In the absence of cooperation (i.e., when the nodes evolve independently of each other and therefore $G = I_{NM}$), the mean error vector would evolve according to

$$E\widetilde{\boldsymbol{\psi}}^{i} = \big(I_{NM} - DR_u\big)\,E\widetilde{\boldsymbol{\psi}}^{i-1}.$$

Thus, we find that in the adaptive network case (31), convergence in the mean will effectively depend on space-time data statistics (represented by B) *and* network topology (represented by G). Using matrix 2-norms[1] we have

$$||BG||_2 \leq ||B||_2 \cdot ||G||_2. \tag{33}$$

Note that due to the block structure of R_u, B is Hermitian; and recall that $G = C \otimes I_M$. Hence, we have

$$\big|\lambda_{\max}\big(BG\big)\big| \leq ||C||_2 \cdot \big|\lambda_{\max}\big(B\big)\big|. \tag{34}$$

That is, the network stability on the mean depends on the local statistics (represented by B) and on the cooperation strategy (represented by C). Whenever a combiner rule is picked so that $||C||_2 \leq 1$, the cooperative scheme will enforce robustness over the noncooperative scheme (in which case $G = I_{NM}$). For combiners that render stochastic and symmetric matrices C, we have that $||C||_2 = 1$. As a result, we conclude that

$$\boxed{|\lambda_{\max}(BG)| \leq |\lambda_{\max}(B)|.} \tag{35}$$

In other words, the spectral radius of BG is generally smaller than the spectral radius of B. Hence, cooperation under the dif-

[1]The 2-norm of a matrix A is defined as the *largest* singular value of A.

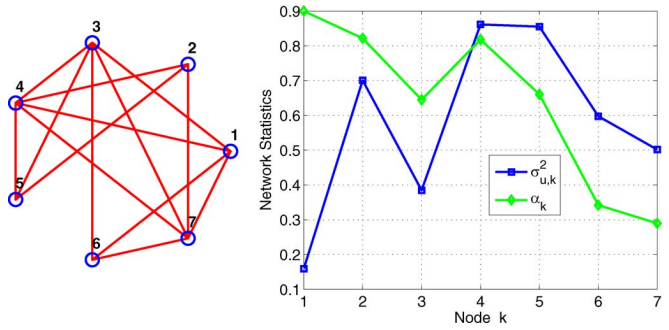

Fig. 3. Network topology (left) and network statistical settings for Example 1.

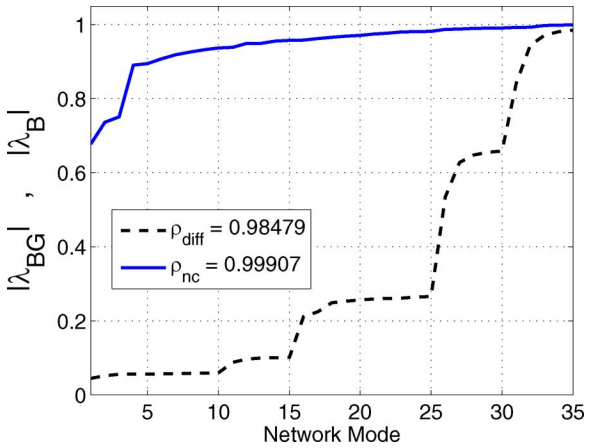

Fig. 4. Example 1: network modes for $N = 7$, $M = 5$, $\mu_k = 0.1$, and $\sigma^2_{v,k} = 10^{-3}$. The symbol ρ denotes the spectral radius (maximum eigenvalue norm) of the coefficient matrices BG (diffusion) and B (no cooperation).

fusion protocol (19) has a *stabilizing* effect on the network. Fig. 3 presents the settings of a typical network (Example 1): the plot on the left depicts the network topology. The plot on the right presents the statistics of the regression data, generated via Gaussian 1–Markov sources with local correlation functions of the form $r(\ell) = \sigma^2_{u,k}\alpha_k^{|\ell|}$, where α_k is the correlation index. Note in Fig. 4 how cooperation decreases the eigenmodes of the mean weight error evolution, as compared with the noncooperative scheme.

Naturally, convergence in the mean is only a necessary condition for convergence in the mean-square sense, which will be addressed in the next section.

VI. Mean-Square Transient Analysis

We now proceed to perform a detailed transient analysis of the adaptive network and characterize the evolution of its learning curves [refer to (68)–(69), (72), and (82)], as well as derive expressions for the mean-square-deviation (MSD) and excess mean-square-error (EMSE) [see (73), (74), (89), and (91)]. We also derive conditions and present a design strategy to ensure network mean-square stability [see (68)–(69) and (97)].

A. Weighted Energy and Variance Relations

We start by defining the local output estimation error at node k as

$$\boldsymbol{e}_k(i) = \boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{\phi}_k^{(i-1)} \tag{36}$$

and collect the errors across the network into the global vector $\boldsymbol{e}_i = \{\boldsymbol{e}_1(i), \boldsymbol{e}_2(i), \ldots, \boldsymbol{e}_N(i)\}$, so that

$$\boldsymbol{e}_i = \boldsymbol{d}_i - \boldsymbol{U}_i G\boldsymbol{\psi}^{i-1} = \boldsymbol{U}_i G\widetilde{\boldsymbol{\psi}}^{i-1} + \boldsymbol{v}_i \triangleq \boldsymbol{e}^G_{a,i} + \boldsymbol{v}_i \tag{37}$$

where

$$\boldsymbol{e}^G_{a,i} = \boldsymbol{U}_i G\widetilde{\boldsymbol{\psi}}^{i-1} \tag{38}$$

and recall that

$$\widetilde{\boldsymbol{\psi}}^{i} = G\widetilde{\boldsymbol{\psi}}^{i-1} - D\boldsymbol{U}_i^* e_i. \tag{39}$$

Introduce further the global *a priori* and *a posteriori* weighted estimation errors:

$$\boldsymbol{e}^{D\Sigma G}_{a,i} \triangleq \boldsymbol{U}_i D\Sigma G\widetilde{\boldsymbol{\psi}}^{i-1} \quad \text{and} \quad \boldsymbol{e}^{D\Sigma}_{p,i} \triangleq \boldsymbol{U}_i D\Sigma\widetilde{\boldsymbol{\psi}}^{i} \tag{40}$$

for some arbitrary $NM \times NM$ matrix $\Sigma \geq 0$. The freedom in selecting Σ will enable us later to characterize the MSD and EMSE performance of the network—see (72)–(74) and (88)–(91) further ahead. Substituting (38) into (39), performing weighted energy balance on both sides, and taking expectations gives

$$\begin{aligned} E\|\widetilde{\boldsymbol{\psi}}^{i}\|^2_{\Sigma} = {} & E\|\widetilde{\boldsymbol{\psi}}^{i-1}\|^2_{G^*\Sigma G} - E\big(\boldsymbol{e}^{D\Sigma G}_{a,i}\big)^* \boldsymbol{e}^G_{a,i} - E\big(\boldsymbol{e}^G_{a,i}\big)^* \boldsymbol{e}^{D\Sigma G}_{a,i} \\ & + E\big(\boldsymbol{e}^G_{a,i}\big)^* \boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^* \boldsymbol{e}^G_{a,i} + E\boldsymbol{v}_i^* \boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^* \boldsymbol{v}_i. \end{aligned} \tag{41}$$

Substituting the error definitions (38) and (40) into (41) yields

$$E\|\widetilde{\boldsymbol{\psi}}^{i}\|^2_{\Sigma} = E\|\widetilde{\boldsymbol{\psi}}^{i-1}\|^2_{\boldsymbol{\Sigma}'} + E\boldsymbol{v}_i^* \boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^* \boldsymbol{v}_i \tag{42}$$

$$\begin{aligned} \boldsymbol{\Sigma}' = {} & G^*\Sigma G - G^*\Sigma D\boldsymbol{U}_i^*\boldsymbol{U}_i G - G^*\boldsymbol{U}_i^*\boldsymbol{U}_i D\Sigma G \\ & + G^*\boldsymbol{U}_i^*\boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^*\boldsymbol{U}_i G. \end{aligned} \tag{43}$$

Note that no assumptions are needed to arrive at (42)–(43). However, the weighting matrix $\boldsymbol{\Sigma}'$ is data dependent and, as such, it is a random quantity. This makes the analysis very challenging, so that some assumptions need to be introduced for the sake of mathematical tractability. Once more, we proceed by assuming temporal and spatial independence of the regression data, so that $\boldsymbol{U}_i$ is independent of $\widetilde{\boldsymbol{\psi}}^{i-1}$, as is common in the analysis of traditional adaptive schemes [3], [18]. In this way, the random weighting matrix $\boldsymbol{\Sigma}'$ can be replaced by its mean value (a deterministic matrix $\Sigma' = E\boldsymbol{\Sigma}'$) [3], [23], which reduces (42)–(43) to the following variance relation:

$$E\|\widetilde{\boldsymbol{\psi}}^{i}\|^2_{\Sigma} = E\|\widetilde{\boldsymbol{\psi}}^{i-1}\|^2_{\Sigma'} + E\boldsymbol{v}_i^* \boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^* \boldsymbol{v}_i \tag{44}$$

$$\begin{aligned} \Sigma' = {} & G^*\Sigma G - G^*\Sigma DE\left(\boldsymbol{U}_i^*\boldsymbol{U}_i\right) G \\ & - G^* E\left(\boldsymbol{U}_i^*\boldsymbol{U}_i\right) D\Sigma G \\ & + G^* E\boldsymbol{U}_i^*\boldsymbol{U}_i D\Sigma D\boldsymbol{U}_i^*\boldsymbol{U}_i G. \end{aligned} \tag{45}$$

Although unrealistic in general, the temporal independence assumption is frequently adopted in the adaptive filtering literature, and leads to a good match between theory and simulations for sufficiently small step sizes. Moreover, spatial independence is more likely to hold in the distributed domain. It is important to remark though, that these assumptions do not compromise the spatial–temporal nature of the problem, neither its distributiveness [4].

B. Gaussian Data

In order to evaluate the network mean-square behavior, (44)–(45) require the calculation of certain data moments. In particular, the last term in (45) is difficult to evaluate in closed form for arbitrary data distributions. We illustrate the analysis here for Gaussian data. Thus, assume that the regressors arise from circular Gaussian sources [3]. Introduce the eigendecomposition $R_u = T\Lambda T^*$, where $\Lambda = \mathrm{diag}\{\Lambda_1, \ldots, \Lambda_N\}$, $\Lambda_k > 0$ and diagonal, and define the transformed quantities

$$\begin{aligned} \overline{\boldsymbol{\psi}}^i &= T^*\widetilde{\boldsymbol{\psi}}^i, \qquad \overline{\boldsymbol{U}}_i = \boldsymbol{U}_i T, \qquad \overline{G} = T^*GT \\ \overline{\Sigma} &= T^*\Sigma T, \qquad \overline{\Sigma}' = T^*\Sigma' T, \qquad \overline{D} = T^*DT = D \end{aligned}$$

where $\overline{D} = D$ follows from (20). We then rewrite the variance relation (44)–(45) in the equivalent form in terms of transformed variables

$$E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\Sigma}} = E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{\Sigma}'} + E\boldsymbol{v}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\boldsymbol{v}_i \tag{46}$$

$$\begin{aligned} \overline{\Sigma}' = {} & \overline{G}^*\overline{\Sigma}\overline{G} - \overline{G}^*\overline{\Sigma}DE\left(\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\right)\overline{G} \\ & - \overline{G}^*E\left(\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\right)D\overline{\Sigma}\overline{G} \\ & + \overline{G}^*E\left(\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\right)\overline{G}. \end{aligned} \tag{47}$$

We now exploit the block diagonal structure of several quantities in these relations in order to express them in a more compact manner by using a convenient vector notation [3], [23]. To do so, we first define the block vector operator $\mathrm{bvec}\{\}$, which converts a block matrix Σ into a single column vector σ in *two steps* as follows. Let Σ be an $NM \times NM$ block matrix

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} & \cdots & \Sigma_{1\ell} & \cdots & \Sigma_{1N} \\ \Sigma_{21} & \Sigma_{22} & \cdots & \Sigma_{2\ell} & \cdots & \Sigma_{2N} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \Sigma_{N1} & \Sigma_{N2} & \cdots & \Sigma_{N\ell} & \cdots & \Sigma_{NN} \end{bmatrix} \tag{48}$$

where each block $\Sigma_{k\ell}$ is $M \times M$. First, the block columns

$$\Sigma_\ell = \mathrm{col}\{\Sigma_{1\ell}, \ldots, \Sigma_{N\ell}\}, \quad \ell = 1, \ldots, N$$

are stacked on top of each other, yielding the $N^2M \times M$ matrix

$$\Sigma^c = \begin{bmatrix} \Sigma_1 \\ \Sigma_2 \\ \vdots \\ \Sigma_N \end{bmatrix}. \tag{49}$$

Subsequently, we move along Σ^c and vectorize each individual block $\Sigma_{k\ell}$ via the standard $\mathrm{vec}\{\}$ operator, so that for each stacked block column Σ_ℓ, we get

$$\sigma_\ell = \mathrm{col}\{\sigma_{1\ell}, \ldots, \sigma_{N\ell}\}, \quad \text{with} \qquad \sigma_{k\ell} = \mathrm{vec}\{\Sigma_{k\ell}\} \tag{50}$$

the final vectorized matrix is obtained from

$$\sigma = \mathrm{col}\{\sigma_1, \sigma_2, \ldots, \sigma_N\}. \tag{51}$$

We thus write $\sigma = \mathrm{bvec}\{\Sigma\}$ to denote the conversion of Σ into a single column. We also write $\Sigma = \mathrm{bvec}\{\sigma\}$ to recover the original block matrix form of the column vector σ. We further define the *block Kronecker product* of two block matrices A and B, which is denoted by $A \odot B$. Its $k\ell$-block is defined as

$$[A \odot B]_{k\ell} = \begin{bmatrix} A_{k\ell} \otimes B_{11} & \cdots & A_{k\ell} \otimes B_{1N} \\ \vdots & \ddots & \vdots \\ A_{k\ell} \otimes B_{N1} & \cdots & A_{k\ell} \otimes B_{NN} \end{bmatrix} \tag{52}$$

for $k, \ell = 1, \ldots, N$. The block vector operator (51) and the block Kronecker product (52) are related via [24], [25]

$$\mathrm{bvec}\{A\Sigma B\} = (B \odot A^T)\sigma. \tag{53}$$

We now use these notations to evaluate the required data moments in (46)–(47), namely

$$E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i, \qquad E\boldsymbol{v}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\boldsymbol{v}_i \quad \text{and} \quad E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i. \tag{54}$$

The first moment is trivial and given by $E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i = \Lambda$. By using (53) and after vectorization, the second and third terms on the right-hand side of (47) are given by

$$\begin{aligned} \mathrm{bvec}\{\overline{G}^*\overline{\Sigma}D\Lambda\overline{G}\} &= \left(\overline{G} \odot \overline{G}^{*T}\right)\mathrm{bvec}\{I_{NM}\overline{\Sigma}D\Lambda\} \\ &= \left(\overline{G} \odot \overline{G}^{*T}\right)(\Lambda D \odot I_{NM})\,\overline{\sigma} \end{aligned}$$

and

$$\mathrm{bvec}\{\overline{G}^*\Lambda D\overline{\Sigma}\overline{G}\} = \left(\overline{G} \odot \overline{G}^{*T}\right)(I_{NM} \odot \Lambda D)\,\overline{\sigma}.$$

The second term in (54) can be verified to be

$$E\boldsymbol{v}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\boldsymbol{v}_i = \mathrm{Tr}\{\Lambda_v E\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\} \tag{55}$$

where $\Lambda_v > 0$ is a diagonal matrix given by

$$\Lambda_v = \mathrm{diag}\{\sigma_{v,1}^2, \sigma_{v,2}^2, \ldots, \sigma_{v,N}^2\}.$$

The $k\ell$ entry of $E\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*$ is given by

$$\left[E\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\right]_{k\ell} = \begin{cases} 0, & \text{for } k \neq \ell \\ \mu_k^2\mathrm{Tr}(\Lambda_k\overline{\Sigma}_{kk}) = \mu_k^2\lambda_k^T\overline{\sigma}_{kk}, & \text{for } k = \ell \end{cases}$$

where $\lambda_k = \mathrm{vec}\{\Lambda_k\}$ and $\overline{\sigma}_{k\ell} = \mathrm{vec}\{\overline{\Sigma}_{k\ell}\}$, so that (55) gives

$$\boxed{E\boldsymbol{v}_i^*\overline{\boldsymbol{U}}_i^* D\overline{\Sigma}D\overline{\boldsymbol{U}}_i\boldsymbol{v}_i = b^T\overline{\sigma}} \tag{56}$$

with $b = \mathrm{bvec}\{R_v D^2\Lambda\}$, $R_v = \Lambda_v \odot I_M$ and $\overline{\sigma} = \mathrm{bvec}\{\overline{\Sigma}\}$.

The fourth-order moment in (54) is challenging, but it can be handled by appealing to the Gaussian factorization theorem [3]. To begin with

$$\begin{aligned} &\mathrm{bvec}\left\{\overline{G}^*E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\overline{G}\right\} \\ &\qquad = \left(\overline{G} \odot \overline{G}^{*T}\right) \cdot \mathrm{bvec}\left\{E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\right\}. \end{aligned}$$

Now both $\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i$ and D are block diagonal, so that

$$E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i D\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i = DE\left(\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\overline{\Sigma}\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\right)D \tag{57}$$

which gives (58), shown at the bottom of the page, where

$$A \triangleq E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\overline{\Sigma}\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i. \tag{59}$$

The $M \times M$ $k\ell$-block of A is given by [3]

$$\begin{aligned} A_{k\ell} &= E\,\overline{\boldsymbol{u}}_{k,i}^*\overline{\boldsymbol{u}}_{k,i}\overline{\Sigma}_{k\ell}\overline{\boldsymbol{u}}_{\ell,i}^*\overline{\boldsymbol{u}}_{\ell,i} \\ &= \begin{cases} \Lambda_k \mathrm{Tr}(\Lambda_k\overline{\Sigma}_{kk}) + \gamma\Lambda_k\overline{\Sigma}_{kk}\Lambda_k, & k=\ell \\ \Lambda_k\overline{\Sigma}_{k\ell},\Lambda_\ell & k\neq\ell \end{cases} \end{aligned} \tag{60}$$

where $\gamma = 1$ for complex data and $\gamma = 2$ for real data. Now express A as

$$A = [A_1\ A_2\ \ldots\ A_\ell\ \ldots\ A_N] \tag{61}$$

where A_ℓ is the ℓ^{th} block column of A:

$$A_\ell = \mathrm{col}\left\{A_{1\ell}, A_{2\ell}, \ldots, A_{k\ell}, \ldots, A_{N\ell}\right\}. \tag{62}$$

It follows that

$$a \triangleq \mathrm{bvec}\{A\} = \mathrm{col}\{a_1, a_2, \ldots, a_\ell, \ldots, a_N\} \tag{63}$$

where

$$a_\ell = \mathrm{col}\{a_{1\ell}, a_{2\ell}, \ldots, a_{k\ell}, \ldots, a_{N\ell}\} \tag{64}$$

and

$$a_{k\ell} = \mathrm{vec}\{A_{k\ell}\} = \begin{cases} \left(\lambda_k\lambda_k^T + \gamma\Lambda_k\otimes\Lambda_k\right)\overline{\sigma}_{kk}, & k=\ell \\ (\Lambda_k\otimes\Lambda_\ell)\overline{\sigma}_{k\ell}, & k\neq\ell. \end{cases}$$

Hence

$$\begin{aligned} a_\ell =& \mathrm{col}\Big\{(\Lambda_1\otimes\Lambda_\ell)\overline{\sigma}_{1\ell}, (\Lambda_1\otimes\Lambda_\ell)\overline{\sigma}_{2\ell}, \ldots, \\ & (\lambda_\ell\lambda_\ell^T + \gamma\Lambda_\ell\otimes\Lambda_\ell)\overline{\sigma}_{1\ell}, \ldots, (\Lambda_N\otimes\Lambda_\ell)\overline{\sigma}_{N\ell}\Big\} \\ \triangleq& \mathcal{A}_\ell\overline{\sigma}_\ell \end{aligned} \tag{65}$$

where

$$\mathcal{A}_\ell = \mathrm{diag}\Big\{\Lambda_1\otimes\Lambda_\ell, \ldots, \lambda_\ell\lambda_\ell^T + \gamma\Lambda_\ell\otimes\Lambda_\ell, \ldots, \Lambda_N\otimes\Lambda_\ell\Big\}$$

and $\overline{\sigma}_\ell = \mathrm{col}\{\overline{\sigma}_{1\ell}, \overline{\sigma}_{2\ell}, \ldots, \overline{\sigma}_{N\ell}\}$. We thus find that

$$\boxed{\mathrm{bvec}\{A\} = \mathrm{col}\Big\{\mathcal{A}_1\overline{\sigma}_1, \mathcal{A}_2\overline{\sigma}_2, \ldots, \mathcal{A}_N\overline{\sigma}_N\Big\} = \mathcal{A}\overline{\sigma}} \tag{66}$$

where

$$\mathcal{A} = \mathrm{diag}\{\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_N\} \quad \text{and} \quad \overline{\sigma} = \mathrm{bvec}\{\overline{\Sigma}\}. \tag{67}$$

In summary, grouping the results and substituting into (46), we conclude that the mean-square behavior of the adaptive network is described by the following recursion for Gaussian sources:

$$\boxed{E\|\overline{\boldsymbol{\psi}}^i\|_{\overline{\sigma}}^2 = E\|\overline{\boldsymbol{\psi}}^{i-1}\|_{\overline{F}\overline{\sigma}}^2 + b^T\overline{\sigma}} \tag{68}$$

$$\begin{aligned} \overline{F} = \left(\overline{G}\odot\overline{G}^{*T}\right)\Big[& I_{N^2M^2} - (I_{NM}\odot\Lambda D) \\ & -(\Lambda D\odot I_{NM}) + (D\odot D)\mathcal{A}\Big]\overline{\sigma}. \end{aligned} \tag{69}$$

In the above we are using the compact notation $\|x\|_\sigma^2$ to refer to $\|x\|_\Sigma^2$, with the weighting matrix Σ replaced by its vector representation $\sigma = \mathrm{bvec}\{\Sigma\}$. Iteration (68)–(69) captures the essence of the global dynamic behavior of the adaptive network. We now illustrate how it can be used to extract useful information about the learning, convergence, and stability behavior of the network.

C. Learning Behavior

Iterating (68)–(69), we get

$$\begin{aligned} E\|\overline{\boldsymbol{\psi}}^i\|_{\overline{\sigma}}^2 &= E\|\overline{\boldsymbol{\psi}}^{i-1}\|_{\overline{F}\overline{\sigma}}^2 + b^T\overline{\sigma} \\ E\|\overline{\boldsymbol{\psi}}^{i-1}\|_{\overline{F}\overline{\sigma}}^2 &= E\|\overline{\boldsymbol{\psi}}^{i-2}\|_{\overline{F}^2\overline{\sigma}}^2 + b^T\overline{F}\overline{\sigma} \\ E\|\overline{\boldsymbol{\psi}}^{i-2}\|_{\overline{F}^2\overline{\sigma}}^2 &= E\|\overline{\boldsymbol{\psi}}^{i-3}\|_{\overline{F}^3\overline{\sigma}}^2 + b^T\overline{F}^2\overline{\sigma} \\ &\vdots \\ E\|\overline{\boldsymbol{\psi}}^0\|_{\overline{F}^i\overline{\sigma}}^2 &= \|\overline{w}^{(o)}\|_{\overline{F}^{i+1}\overline{\sigma}}^2 + b^T\overline{F}^i\overline{\sigma} \end{aligned} \tag{70}$$

where $\overline{w}^{(o)} = T^*w^{(o)}$ and the last equality in (70) follows from the fact that the local adaptive filters are initialized with zeros. Relations (70) lead to the result

$$E\|\overline{\boldsymbol{\psi}}^i\|_{\overline{\sigma}}^2 = E\|\overline{w}^{(o)}\|_{\overline{F}^{i+1}\overline{\sigma}}^2 + b^T\left(\sum_{k=0}^{i}\overline{F}^k\right)\overline{\sigma} \tag{71}$$

which in turn motivates the following useful recursion:

$$\boxed{E\|\overline{\boldsymbol{\psi}}^i\|_{\overline{\sigma}}^2 = E\|\overline{\boldsymbol{\psi}}^{i-1}\|_{\overline{\sigma}}^2 + b^T\overline{F}^i\overline{\sigma} - \|\overline{w}^{(o)}\|_{\overline{F}^i(I-\overline{F})\overline{\sigma}}^2.} \tag{72}$$

This recursion describes the evolution of the variance of the transformed weight error vector $\overline{\boldsymbol{\psi}}^i$. By iterating this recursion, we are able to obtain the global learning curve, $E\|\overline{\boldsymbol{\psi}}^i\|_{\overline{\sigma}}^2$, for the adaptive network. We define the global mean-square deviation as the average of the global quantity $E\|\overline{\boldsymbol{\psi}}^i\|^2$, i.e., $\eta(i) = (1/N)E\|\overline{\boldsymbol{\psi}}^i\|^2$. Therefore, choosing $\overline{\sigma} = (1/N)\mathrm{bvec}\{I_{NM}\} \triangleq q_\eta$ in (72) leads to

$$\boxed{\eta(i) = \eta(i-1) + b^T\overline{F}^i q_\eta - \|\overline{w}^{(o)}\|_{\overline{F}^i(I-\overline{F})q_\eta}^2} \tag{73}$$

$$\boxed{\mathrm{bvec}\{\overline{G}^*E\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_iD\overline{\Sigma}D\overline{\boldsymbol{U}}_i^*\overline{\boldsymbol{U}}_i\overline{G}\} = \left(\overline{G}\odot\overline{G}^{*T}\right)(D\odot D)\,\mathrm{bvec}\{A\}} \tag{58}$$

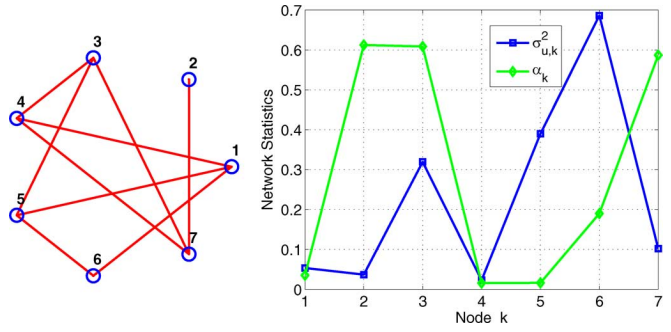

Fig. 5. Network topology and statistics for Example 2.

with initial condition $\eta(-1) = \|\overline{w}^{(o)}\|^2$. In a similar vein, choosing $\overline{\sigma} = (1/N)\text{bvec}\{\Lambda\} \triangleq \lambda_\zeta$ leads to the global learning curve for the excess mean-square error $\zeta(i) = (1/N)E\|\overline{\psi}^i\|_\Lambda = (1/N)E\|\boldsymbol{e}_{a,i}\|^2$, where $\boldsymbol{e}_{a,i} = \boldsymbol{U}_i\widetilde{\boldsymbol{\psi}}^{i-1}$, so that

$$\boxed{\zeta(i) = \zeta(i-1) + b^T\overline{F}^i\lambda_\zeta - \|\overline{w}^{(o)}\|^2_{\overline{F}^i(I-\overline{F})\lambda_\zeta}} \tag{74}$$

with initial condition $\zeta(-1) = \|\overline{w}^{(o)}\|^2_\Lambda$.

Fig. 5 presents the network settings of Example 2: an example of an adaptive network operating with the diffusion protocol (19), with the respective performance depicted in Figs. 6 and 7. Note that an expressive improvement in terms of speed of convergence and steady-state performance is achieved over the noncooperative scheme.

The result (72) can also be rewritten in the form of a linear first order state-space model. To see this, we use (70) to write

$$E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\sigma}} = E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}\overline{\sigma}} + b^T\overline{\sigma} \tag{75}$$

$$E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}\overline{\sigma}} = E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}^2\overline{\sigma}} + b^T\overline{F}\overline{\sigma} \tag{76}$$

$$E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}^2\overline{\sigma}} = E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}^3\overline{\sigma}} + b^T\overline{F}^2\overline{\sigma} \tag{77}$$

$$\vdots$$

$$E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}^{N^2M^2-1}\overline{\sigma}} = E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}^{N^2M^2}\overline{\sigma}} + b^T\overline{F}^{N^2M^2-1}\overline{\sigma}. \tag{78}$$

These relations express the variances of the transformed weight error vector in terms of increasing powers of $\overline{F}$. The recursion can be halted, leading to a closed form solution that describes the global network transient behavior. To do so, we appeal to the Cayley-Hamilton theorem. It states that any matrix satisfies its characteristic polynomial $p(x) = 0$. That is

$$\overline{F}^{N^2M^2} + p_{N^2M^2-1}\overline{F}^{N^2M^2-1} + \cdots + p_2\overline{F}^2 + p_1\overline{F} + p_0 = 0 \tag{79}$$

in terms of the coefficients $\{p_\ell\}$ of the characteristic polynomial of $\overline{F}$. It follows that

$$\overline{F}^{N^2M^2} = -p_{N^2M^2-1}\overline{F}^{N^2M^2-1} - \cdots - p_2\overline{F}^2 - p_1\overline{F} - p_0. \tag{80}$$

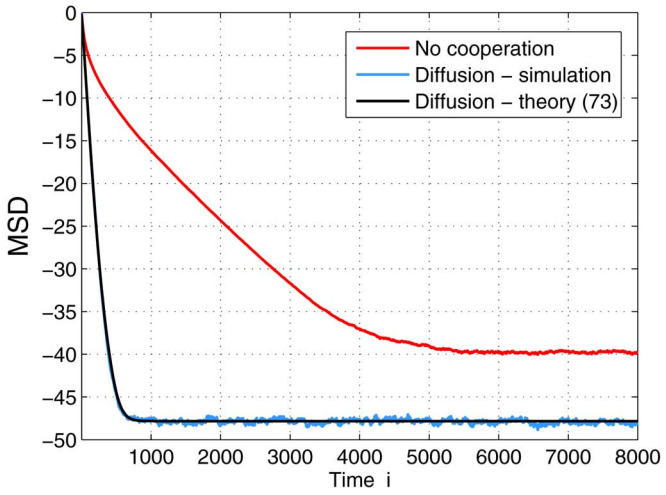

Fig. 6. Example 2: Global mean-square deviation (MSD) curve. This curve was obtained by averaging $E\|\widetilde{\boldsymbol{\psi}}_k^{(i-1)}\|^2$ across all nodes and over several experiments.

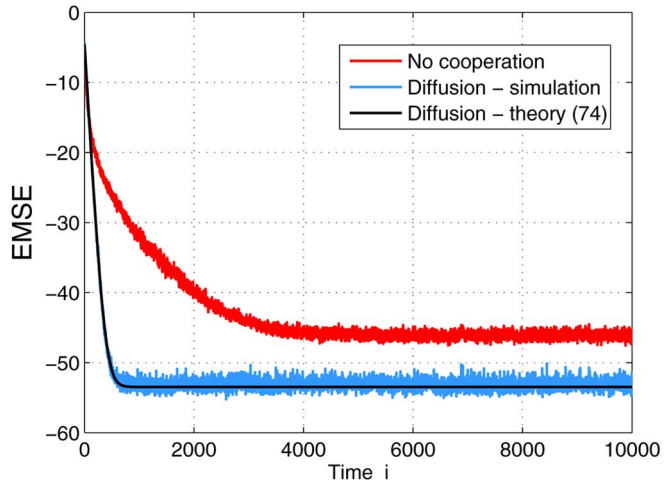

Fig. 7. Example 2: Global excess mean-square error (EMSE) curve. This curve was obtained by averaging $E|\boldsymbol{e}_{a,k}(i)|^2$, where $\boldsymbol{e}_{a,k}(i) = \boldsymbol{u}_{k,i}\widetilde{\boldsymbol{\psi}}_k^{(i-1)}$, across all nodes and over several experiments.

Substituting (80) into (78) gives

$$\begin{aligned} E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}^{N^2M^2-1}\overline{\sigma}} = &-p_{N^2M^2-1}E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}^{N^2M^2-1}\overline{\sigma}} \\ &- \cdots - p_2E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}^2\overline{\sigma}} - p_1E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\overline{F}\overline{\sigma}} \\ &-p_0E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\sigma}} + b^T\overline{F}^{N^2M^2-1}\overline{\sigma}. \end{aligned} \tag{81}$$

We can now group (75)–(77) with (81) into the state-space form

$$\boxed{X^i = \mathcal{G}X^{i-1} + \mathcal{Y}} \tag{82}$$

where the state vector is defined by

$$X^i = \text{col}\{E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\sigma}}, E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}\overline{\sigma}}, \ldots, E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{F}^{N^2M^2-1}\overline{\sigma}}\} \tag{83}$$

and the feedback matrix $\mathcal{G}$ is given by

$$\mathcal{G} = \begin{bmatrix} 0 & 1 & 0 & & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & & 0 \\ \vdots & & \vdots & & \\ 0 & 0 & 0 & & 1 \\ -p_0 & -p_1 & -p_2 & \cdots & -p_{N^2M^2-1} \end{bmatrix}. \quad (84)$$

Moreover

$$\mathcal{Y} = \mathrm{col}\{b^T\overline{\sigma}, b^T\overline{F}\overline{\sigma}, \ldots, b^T\overline{F}^{N^2M^2-1}\overline{\sigma}\}. \quad (85)$$

The linear state-space model (82) describes the global network mean-square behavior in a fundamental way, stating that the adaptive network indeed behaves as a global adaptive entity on its own right. The network mean-square learning behavior may be retrieved by observing the leading entry of the network state vector X.

D. Local Node Performance

The global quantity $E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\sigma}}$ aggregates the contributions of the individual nodes' errors. As a result, by filtering out the other nodes' components we may retrieve the local node component from the global error $E\|\overline{\boldsymbol{\psi}}^i\|^2_{\overline{\sigma}}$. We do so by exploiting the degree of freedom in selecting the weighting matrix $\overline{\Sigma} = \mathrm{bvec}\{\overline{\sigma}\}$, and define the following spatial filtering matrices whose purpose is to extract the local quantities from the global expressions:

$$J_{q,k} = \mathrm{diag}\{\mathbf{0}_{(k-1)M}, I_M, \mathbf{0}_{(N-k)M}\} \quad \text{(MSD)} \quad (86a)$$
$$J_{\lambda,k} = \mathrm{diag}\{\mathbf{0}_{(k-1)M}, \Lambda_k, \mathbf{0}_{(N-k)M}\} \quad \text{(EMSE)} \quad (86b)$$

where Λ_k is the diagonal matrix with the eigenvalues corresponding to node k and $\mathbf{0}_L$ is a block of $L \times L$ zeros. For convenience of notation, let us further define the vectors

$$j_{q,k} = \mathrm{bvec}\{J_{q,k}\} \quad \text{and} \quad j_{\lambda,k} = \mathrm{bvec}\{J_{\lambda,k}\}. \quad (87)$$

Selecting $\overline{\sigma}$ as the filtering vectors (87) in the global learning recursion (72) yields the local MSD at node k [compare with expression (73)]:

$$\boxed{\overline{f}_{q,i} = \overline{F}^i(I_{N^2M^2} - \overline{F})j_{q,k}} \quad (88)$$

$$\boxed{\eta_k(i) = \eta_k(i-1) + b^T\overline{F}^i j_{\lambda,k} - \|\overline{w}^{(o)}\|^2_{\mathrm{bvec}\{\overline{f}_{q,i}\}}} \quad (89)$$

with initial condition $\eta_k(-1) = \|\overline{w}^{(o)}\|^2_{j_{q,k}}$. Similarly, for the EMSE at node k we have [compare with expression (74)]:

$$\boxed{\overline{f}_{\lambda,i} = \overline{F}^i(I_{N^2M^2} - \overline{F})j_{\lambda,k}} \quad (90)$$

$$\boxed{\zeta_k(i) = \zeta_k(i-1) + b^T\overline{F}^i j_{\lambda,k} - \|\overline{w}^{(o)}\|^2_{\mathrm{bvec}\{\overline{f}_{\lambda,i}\}}} \quad (91)$$

with initial condition $\zeta_k(-1) = \|\overline{w}^{(o)}\|^2_{j_{\lambda,k}}$.

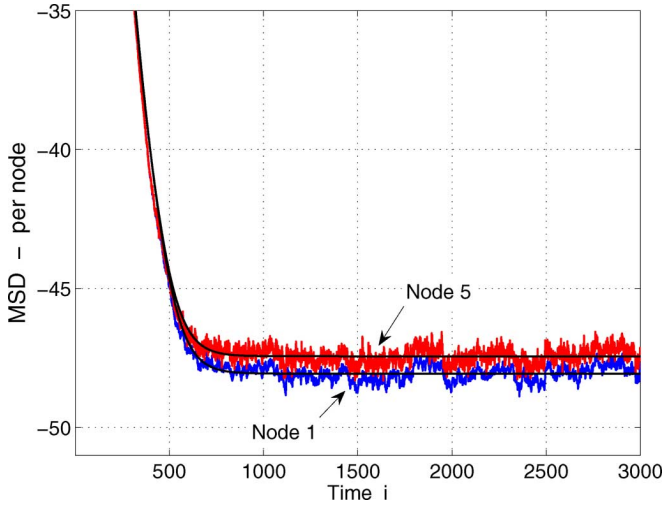

Fig. 8. Local MSD evolution at nodes 1 and 5 for Example 2, simulation and theory (89). We depicted the curves corresponding to the nodes whose performance deviated the most from each other, the other nodes presented practically the same performance.

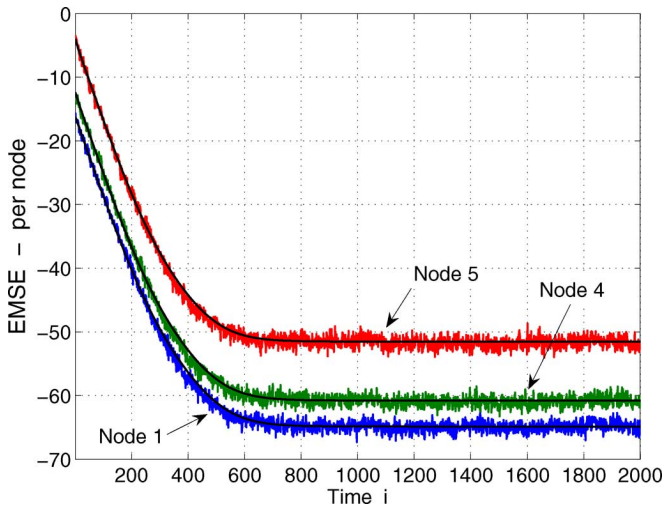

Fig. 9. Local EMSE evolution at nodes 1, 4 and 5 for Example 2, simulation and theory (91).

The plots in Figs. 8 and 9 illustrate the local transient performance for a few nodes in the same network as that of Fig. 5 (Example 2). Note the good matching between theory and simulations. Moreover, a very interesting effect may be noted in those pictures: despite the diversity of network statistics and relatively low connectivity of the topology considered, an equalization effect takes place at the node level: all the nodes learn practically at the same rate and experience an equal learning behavior, in both transient and steady-state (see, particularly, Fig. 8). In other words, the analytical model and simulations suggest that no consensus iterations are necessary to drive all the agents in the network to a reasonable agreement [4].

We now summarize the main results of the mean-square performance of the adaptive network in the following theorem

Theorem 1: Consider an adaptive network operating under the diffusion protocol (19) with space-time data $\{\boldsymbol{d}_k(i), \boldsymbol{u}_{k,i}\}$ satisfying (21). Assume further that the regressors $\boldsymbol{u}_{k,i}$ are circularly Gaussian and independent over time and space. The network mean-square deviation and excess mean-square error evolve as follows:

$$\eta(i) = \eta(i-1) + b^T \overline{F}^i q_\eta - \|\overline{w}^{(o)}\|^2_{\overline{F}^i(I-\overline{F})q_\eta} \quad \text{(MSD)}$$

$$\zeta(i) = \zeta(i-1) + b^T \overline{F}^i \lambda_\zeta - \|\overline{w}^{(o)}\|^2_{\overline{F}^i(I-\overline{F})\lambda_\zeta} \quad \text{(EMSE)}$$

with initial conditions $\eta(-1) = \|\overline{w}^{(o)}\|^2$ and $\zeta(-1) = \|\overline{w}^{(o)}\|^2_\Lambda$, respectively. Matrix $\overline{F}$ is given by (69), $q_\eta = (1/N)\text{bvec}\{I_{NM}\}$, $\lambda_\zeta = (1/N)\text{bvec}\{\Lambda\}$ and Λ collects the regressors eigenmodes and it is obtained from $R_u = T\Lambda T^*$. Similarly, the local mean-square performance at any node k evolves as

$$\overline{f}_{q,i} = \overline{F}^i(I_{N^2M^2} - \overline{F})j_{q,k}\,, \quad \eta_k(-1) = \|\overline{w}^{(o)}\|^2_{J_{q,k}}$$

$$\eta_k(i) = \eta_k(i-1) + b^T \overline{F}^i j_{\lambda,k} - \|\overline{w}^{(o)}\|^2_{\text{bvec}\{\overline{f}_{q,i}\}} \quad \text{(MSD)}$$

and

$$\overline{f}_{\lambda,i} = \overline{F}^i(I_{N^2M^2} - \overline{F})j_{\lambda,k}\,, \quad \zeta_k(-1) = \|\overline{w}^{(o)}\|^2_{J_{\lambda,k}}$$

$$\zeta_k(i) = \zeta_k(i-1) + b^T \overline{F}^i j_{\lambda,k} - \|\overline{w}^{(o)}\|^2_{\text{bvec}\{\overline{f}_{\lambda,i}\}} \quad \text{(EMSE)}$$

with $j_{q,k} = \text{bvec}\{J_{q,k}\}$, $j_{\lambda,k} = \text{bvec}\{J_{\lambda,k}\}$, and filtering matrices $J_{q,k}$ and $J_{\lambda,k}$ given by (86).

E. To Cooperate or Not to Cooperate?

In this section, we compare the diffusion cooperative scheme with the two noncooperative strategies mentioned in the introduction, which run independent adaptive filters at each node:

$$\psi^{(i)}_{nc,k} = \psi^{(i-1)}_{nc,k} + \mu_k u^*_{k,i}\Big(d_k(i) - u_{k,i}\psi^{(i-1)}_{nc,k}\Big). \quad (92)$$

In the first case, the local estimates are fused at a central node and a global estimate $\phi^{(i-1)}_{glb}$ is generated, which is sent back to the nodes. In the second case, each node fuses its local estimate and the estimates received from the neighborhood into $\phi^{(i-1)}_{nc,k}$, following the same rule as (19a). The noncooperative fusion rules are

$$\phi^{(i-1)}_{glb} = \frac{1}{N}\sum_{k=1}^{N} \psi^{(i-1)}_{nc,k} \quad \text{(global)} \quad (93a)$$

$$\phi^{(i-1)}_{nc,k} = \sum_{\ell \in \mathcal{N}_{k,i-1}} c_{k\ell}\psi^{(i-1)}_{nc,\ell} \quad \text{(local)}. \quad (93b)$$

Note that in the noncooperative schemes the adaptive process does not take advantage of the fusion step. For comparison purposes, the mean-square deviation is calculated as

$$\eta_{glb}(i) = E\big\|w^o - \boldsymbol{\phi}^{(i-1)}_{glb}\big\|^2 \quad \text{(global)} \quad (94a)$$

$$\eta^{nc}_k(i) = E\big\|w^o - \boldsymbol{\phi}^{(i-1)}_{nc,k}\big\|^2 \quad \text{(local)}. \quad (94b)$$

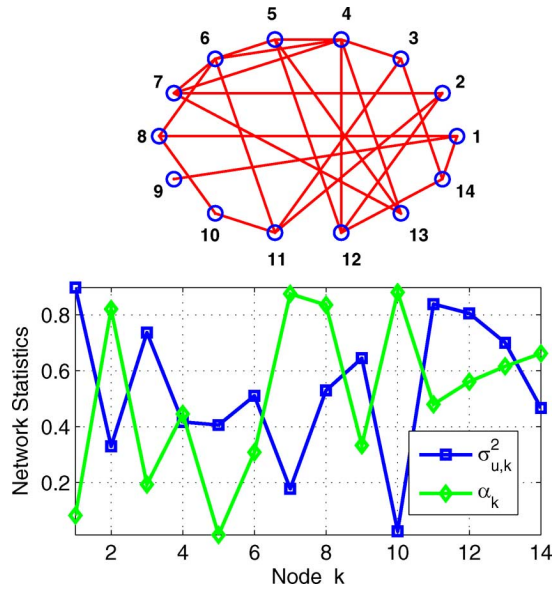

Fig. 10. Network topology and data statistics for Example 3.

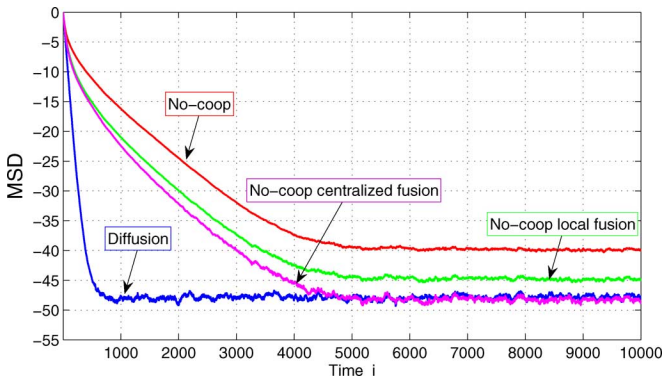

Fig. 11. MSD comparison for Example 3. The step size employed was $\mu = 0.042$ and the background noise power was $\sigma^2_v = 10^{-3}$.

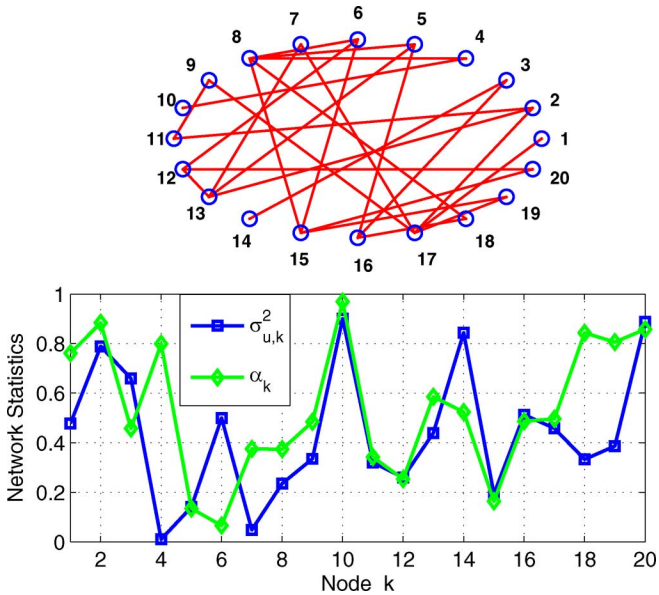

Fig. 12. Network topology and data statistics for Example 4.

Examples 3 and 4 are depicted in Figs. 10–13 and present comparisons in terms of MSD evolution for the diffusion protocol (19) and the noncooperative schemes (92)–(93). In Ex-

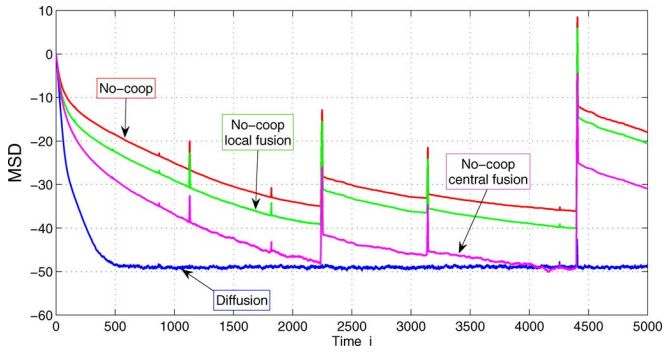

Fig. 13. MSD comparison for Example 4. Note the robustness of the diffusion algorithm (19). The step size employed was $\mu = 0.042$ and the background noise power was $\sigma_v^2 = 10^{-3}$.

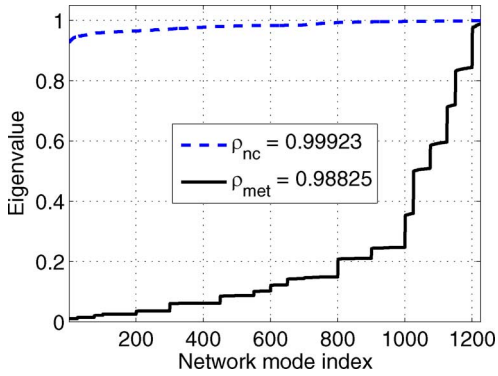

Fig. 14. The N^2M^2 modes of the coefficient matrix $\overline{F}$ for Example 2. $\rho_{\rm met}$ stands for the spectral radius of the diffusion metropolis protocol. $\rho_{\rm nc}$ is the spectral radius of the noncooperative counterpart.

ample 3, depicted in Figs. 10 and 11, the diffusion scheme outperformed both noncooperative schemes; moreover, it achieved practically the same steady-state performance as the centralized fusion scheme, which could be also implemented approximately by running a consensus algorithm. Example 4 is captured in Fig. 13, which shows the robustness and superior performance of our cooperative scheme. Fig. 12 plots the corresponding network settings.

Compared to the local noncooperative fusion technique (93b), our scheme has the same computational and communication complexity, but it is more robust and superior in terms of estimation performance. The global fusion scheme (93a) implies the existence of a fusion center, which for general networks may generate considerable communication overheads and drains valuable energy resources to be implemented. It is fundamentally not distributed and outperformed by the diffusion cooperative protocol (19).

VII. Mean-Square Stability

Let us rewrite $\overline{F}$ in (69) as

$$\overline{F} = \left(\overline{G} \odot \overline{G}^{*T}\right) S \tag{95}$$

in terms of the *symmetric* matrix

$$S \triangleq \Big[I_{N^2M^2} - (I_{NM} \odot \Lambda D) - (\Lambda D \odot I_{NM}) + (D \odot D)\mathcal{A}\Big]. \tag{96}$$

For stability in the mean-square sense, we must ensure through the selection of μ_k and the cooperation protocol C (i.e., G) that all eigenvalues of $\overline{F}$ satisfy

$$\boxed{|\lambda\left(\overline{F}\right)| < 1} \tag{97}$$

so that stability in the mean *and* mean-square senses require that the $\{\mu_k\}$ must satisfy (32) and (97). We will present here a simple procedure that is *sufficient* to ensure global stability. Resorting once more to matrix 2-norms allows us to write

$$\|\overline{F}\|_2 = \left\|\left(\overline{G} \odot \overline{G}^{*T}\right) S\right\|_2 \leq \left\|\overline{G} \odot \overline{G}^{*T}\right\|_2 \cdot \|S\|_2 \tag{98}$$

which can be expressed in terms of the standard Kronecker product [26]

$$\begin{aligned}\left\|\left(\overline{G} \odot \overline{G}^{*T}\right) S\right\|_2 &= \left\|P^T\left(\overline{G} \otimes \overline{G}^{*T}\right) P \cdot S\right\|_2 \\ &\leq \left\|\overline{G} \otimes \overline{G}^{*T}\right\|_2 \cdot \|S\|_2 \\ &= \left\|\overline{G}\right\|_2 \cdot \left\|\overline{G}^{*T}\right\|_2 \cdot \|S\|_2 \end{aligned} \tag{99}$$

for some (orthogonal) permutation matrix P. The transformation T is unitary, thus we have $\|\overline{G}\|_2 = \|G\|_2$. In addition, recall that S is symmetric and $G = C \otimes I_M$, hence

$$\boxed{\left|\lambda_{\max}\Big(\left(\overline{G} \odot \overline{G}^{*T}\right) \cdot S\Big)\right| \leq \|C\|_2^2 \cdot \left|\lambda_{\max}(S)\right|.} \tag{100}$$

Equation (100) reveals a fundamental property of the adaptive network: stability of the overall system is governed by the design of the individual adaptive nodes (represented by S) and by the chosen cooperation protocol (represented by C). For combiners that render stochastic and symmetric matrices C, as the Metropolis and the Laplacian rules, we conclude from (100) that

$$\boxed{\left|\lambda_{\max}\left(\left(\overline{G} \odot \overline{G}^{*T}\right) \cdot S\right)\right| \leq \left|\lambda_{\max}(S)\right|.} \tag{101}$$

That is, the spectral radius of $\left(\overline{G} \odot \overline{G}^{*T}\right) S$, which represents the cooperative system, is generally smaller than the spectral radius of S, which may be interpreted as the noncooperative scheme, implemented by individual adaptive filters. Choosing a cooperation protocol that ensures $\|C\|_2^2 \leq 1$ is sufficient to enforce stability on the network level, because of (100). Variable step size strategies $\mu_k(i)$ may also be employed by following similar guidelines. We thus conclude that cooperation under the diffusion protocol (19) has a *stabilizing* effect on the network *also* in the mean-square sense.

Fig. 14 depicts the eigenmodes for the example whose settings and curves were depicted in Figs. 5–7. Note how cooperation radically decreases the learning modes of the global system, confirming what was observed in Figs. 6 and 7.

VIII. Steady-State Performance

A. Global Network Performance

We now examine how the network stabilizes after the cooperative learning process reaches steady-state. The global steady-state quantities MSD and EMSE are defined as

$$\eta = \frac{1}{N} E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2 \qquad \text{(MSD)} \tag{102a}$$

$$\zeta = \frac{1}{N} E\|\overline{\boldsymbol{\psi}}^{i-1}\|^2_{\Lambda} \qquad \text{(EMSE)} \tag{102b}$$

as $i \to \infty$. In steady-state, (68) leads to

$$E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_{(I-\overline{F})\overline{\sigma}} = b^T \overline{\sigma} \tag{103}$$

so that to calculate the MSD and the EMSE we need to evaluate the weighted norms $E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_q$ and $E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_{\lambda}$, where $q = \text{bvec}\{I_{NM}\}$ and $\lambda = \text{bvec}\{\Lambda\}$. We are free to select $\overline{\sigma}$ in (103). Thus, consider two possibilities for $\overline{\sigma}$ defined by

$$\left(I - \overline{F}\right)\overline{\sigma}_{\eta} = q \quad \text{and} \quad \left(I - \overline{F}\right)\overline{\sigma}_{\zeta} = \lambda. \tag{104}$$

They lead to

$$\boxed{\eta = \tfrac{1}{N} b^T \left(I - \overline{F}\right)^{-1} q \qquad \text{(MSD)}} \tag{105a}$$

$$\boxed{\zeta = \tfrac{1}{N} b^T \left(I - \overline{F}\right)^{-1} \lambda \qquad \text{(EMSE)}} \tag{105b}$$

which describe the global network performance in steady-state.

B. Local Node Performance

Steady-state performance at the node level may also be retrieved from the global expressions by exploiting again the degree of freedom in selecting $\overline{\sigma}$ in (103). To begin with, note that the local mean-square performance of node k is defined as

$$\eta_k = E\|\overline{\boldsymbol{\psi}}_k^{\infty}\|^2 \quad \text{and} \quad \zeta_k = E\|\overline{\boldsymbol{\psi}}_k^{\infty}\|^2_{\lambda_k} \tag{106}$$

in terms of the local stationary vectors $\overline{\boldsymbol{\psi}}_k^{\infty}$ and where $\lambda_k = \text{vec}\{\Lambda_k\}$. Now, inspecting the global steady-state quantities (102) and considering the block diagonal structure of Λ we have

$$\eta = \frac{1}{N} E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2 = \frac{1}{N}\sum_{k=1}^{N} E\|\overline{\boldsymbol{\psi}}_k\|^2 \tag{107a}$$

$$\zeta = \frac{1}{N} E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_{\lambda} = \frac{1}{N}\sum_{k=1}^{N} E\|\overline{\boldsymbol{\psi}}_k\|^2_{\lambda_k}. \tag{107b}$$

Thus the global mean-square performance is the average of the individual node contributions. We want to retrieve the individual node component from the global summation. Once more, we resort to the filtering matrices (86) and rewrite the local quantities (106) in terms of the global quantities and the filtering matrices:

$$\eta_k = E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_{J_{q,k}}$$

$$\zeta_k = E\|\overline{\boldsymbol{\psi}}^{\infty}\|^2_{J_{\lambda,k}}.$$

Thus we select the $\overline{\sigma}$ in (103) as the solution to the linear systems of equations

$$\left(I - \overline{F}\right)\overline{\sigma}_{\eta} = \text{bvec}\{J_{q,k}\} \quad \text{(MSD)} \tag{108a}$$

$$\left(I - \overline{F}\right)\overline{\sigma}_{\zeta} = \text{bvec}\{J_{\lambda,k}\} \quad \text{(EMSE)} \tag{108b}$$

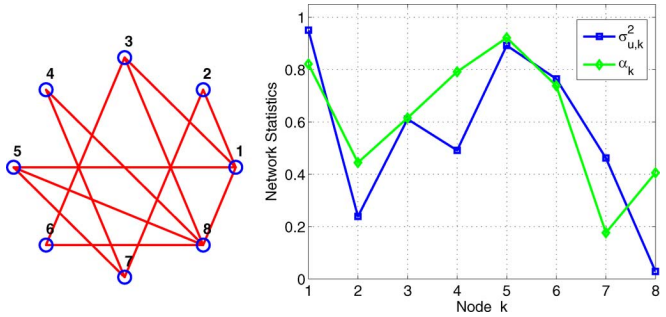

Fig. 15. Network topology and statistics for Example 5.

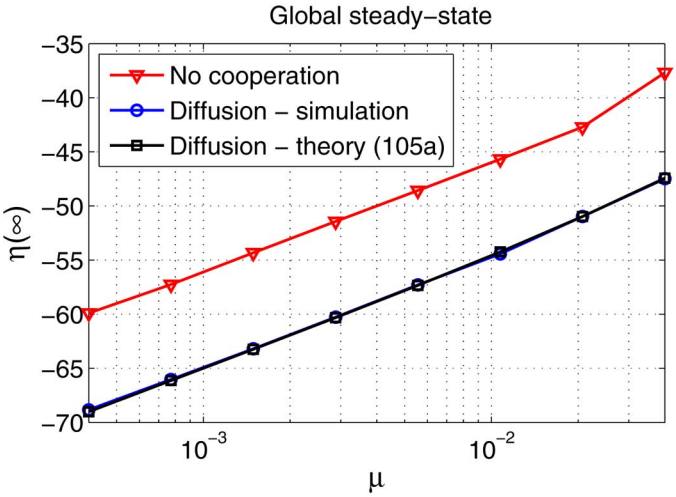

Fig. 16. Global MSD performances for Example 5.

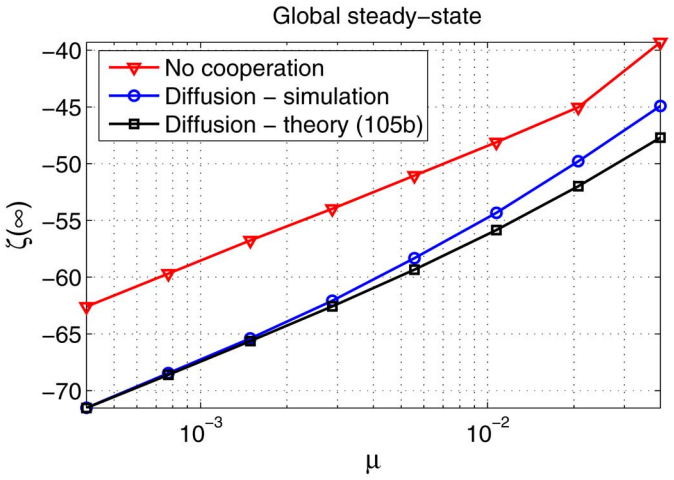

Fig. 17. Global EMSE performances for Example 5.

so that

$$\boxed{\eta_k = b^T \left(I - \overline{F}\right)^{-1} \text{bvec}\{J_{q,k}\} \quad \text{(MSD)}} \tag{109a}$$

$$\boxed{\zeta_k = b^T \left(I - \overline{F}\right)^{-1} \text{bvec}\{J_{\lambda,k}\} \quad \text{(EMSE)}} \tag{109b}$$

which describe the steady-state performance attained at node k.

Figs. 15–19 depict Example 5, which is an example of an adaptive network operating in steady-state. The network profile is given in Fig. 15. Figs. 16 and 17 present the network

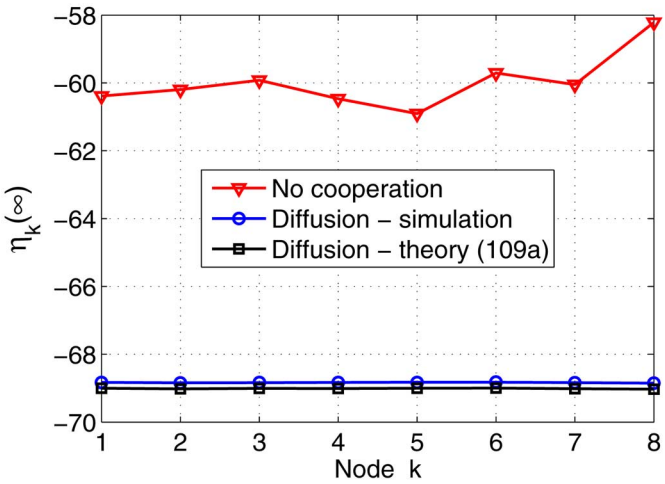

Fig. 18. Local MSD performance for Example 5.

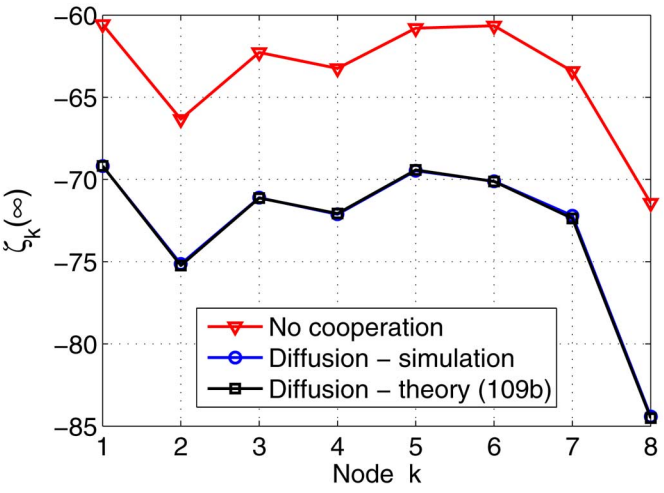

Fig. 19. Local EMSE performance for Example 5.

global performance, and Figs. 18 and 19 show the network performance at the individual nodes. One may observe the close match between simulation and theory, as well as the improvement in performance over the noncooperative case.

IX. Adaptive Diffusion

We have assumed so far static combination of nearby estimates through the use of constant combination coefficients $c_{k\ell}$. However, in some instances, a particular node may be performing better estimation than its neighbors. As a consequence, a "blind" aggregation that assigns equal weights to every node in the neighborhood may not be the best policy. An alternative to this approach is to let the network adjust these weights as well. In this way, we end up adding another layer of adaptation to the network.

There are many ways to design the combiner function f_k in (17) and make it adaptive. We will illustrate the idea by resorting to the adaptive convex combination studied in [14], [27].

The strategy dynamically generates the aggregate estimate $\phi_k^{(i-1)}$ as follows (see Fig. 20):

$$\phi_k^{(i-1)} = g_k \psi_k^{(i-1)} + (1-g_k)\,\widehat{\psi}_k^{(i-1)} \tag{110}$$

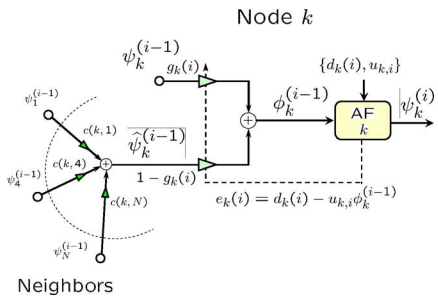

Fig. 20. Adaptive diffusion strategy.

where $g_k \in [0,1]$ is a local dynamic combiner and $\hat{\psi}_k^{(i-1)}$ is obtained via [compare with (11)]

$$\widehat{\psi}_k^{(i-1)} = \sum_{\ell \in \mathcal{N}_k/k} c_{k\ell}\psi_\ell^{(i-1)}. \tag{111}$$

The resulting $\phi_k^{(i-1)}$ is then presented to the local adaptive node

$$e_k(i) = d_k(i) - u_{k,i}\phi_k^{(i-1)} \tag{112}$$

$$\psi_k^{(i)} = \phi_k^{(i-1)} + \mu_k\, u_{k,i}^* e_k(i). \tag{113}$$

The combiner $g_k(i)$ is defined as a real function of a time-varying parameter δ_k in order to enforce convexity. We select a sigmoidal function relating δ_k to g_k due to its good performance [27], [28] yielding

$$g_k(i) = \frac{1}{1+\left|\exp\left(-\frac{\delta_k(i-1)}{2}\right)\right|^2}. \tag{114}$$

At each node, $g_k(i)$ is adapted indirectly via δ_k in order to minimize the local mean-square error, say as

$$\delta_k(i) = \delta_k(i-1) - \mu_\delta\left[\nabla_\delta |e_k|^2\right]^*. \tag{115}$$

Adopting the gradient-descent rule (115) and using (110)–(113) lead to the following adaptation rule for δ_k:

$$h_{k,i} = \left(u_{k,i}\psi_k^{(i-1)} - u_{k,i}\widehat{\psi}_k^{(i-1)}\right)^* e_k(i) g_k(i)\big(1-g_k(i)\big)$$

$$\delta_k(i) = \delta_k(i-1) + \mu_\delta \frac{1}{\|u_{k,i}\|^\beta} h_{k,i}$$

where β is a nonnegative exponent introduced to control convergence. In summary, the adaptive diffusion protocol is described by the following equations:

$$\begin{aligned}
&g_k(i) = \frac{1}{1+\left|\exp\left(-\frac{\delta_k(i-1)}{2}\right)\right|^2}\\
&\widehat{\psi}_k^{(i-1)} = \sum_{\ell \in \mathcal{N}_k/k} c_{k\ell}\psi_\ell^{(i-1)}\\
&\phi_k^{(i-1)} = g_k(i)\psi_k^{(i-1)} + \big(1-g_k(i)\big)\widehat{\psi}_k^{(i-1)}\\
&e_k(i) = d_k(i) - u_{k,i}\phi_k^{(i-1)}\\
&h_{k,i} = \left(u_{k,i}\psi_k^{(i-1)} - u_{k,i}\widehat{\psi}_k^{(i-1)}\right)^* e_k(i) g_k(i)\big(1-g_k(i)\big)\\
&\delta_k(i) = \delta_k(i-1) + \mu_\delta \frac{1}{\|u_{k,i}\|^\beta} h_{k,i}\\
&\psi_k^{(i)} = \phi_k^{(i-1)} + \mu_k\, u_{k,i}^* e_k(i).
\end{aligned} \tag{116}$$

The operation of the adaptive diffusion scheme (116) is illustrated in Example 6, which is captured in Figs. 21–24. For this example, $\sigma_v^2 = 10^{-3}$, $\mu = 0.007$, and $\mu_\delta = \{150,200\}$

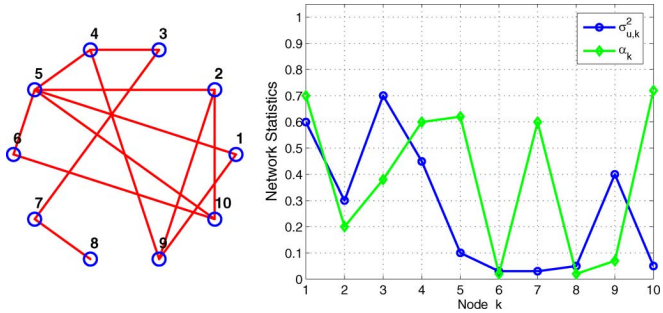

Fig. 21. Network topology and statistical profile for Example 6.

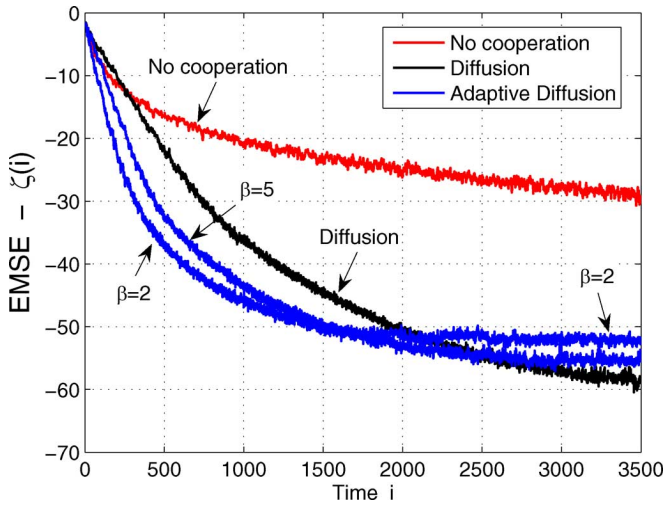

Fig. 22. Example 6: Transient EMSE curve.

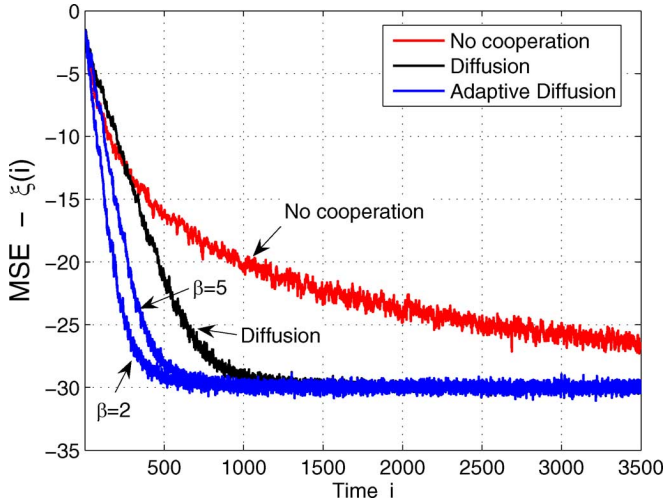

Fig. 23. Example 6: transient MSE curve.

and $\beta = \{2{,}5\}$ respectively. Note in Fig. 22 that adaptive diffusion is faster than the standard diffusion protocol, but with slightly larger error (7 dB @ −60 dB) due to the extra adaptive layer (gradient noise). However this effect can be balanced by designing β, although convergence gets slightly slower. Fig. 24 shows the adaptive weights g_k for a few nodes. A weight close to one means that the corresponding node is performing better than its neighborhood's average estimates, e.g., nodes 1 and 3. Likewise, nodes 5 and 10 were assigned small weights, meaning they are performing worse than the aggregated neighbors' estimates. Note how nodes with higher SNR, e.g., nodes 1, 3, and 4,

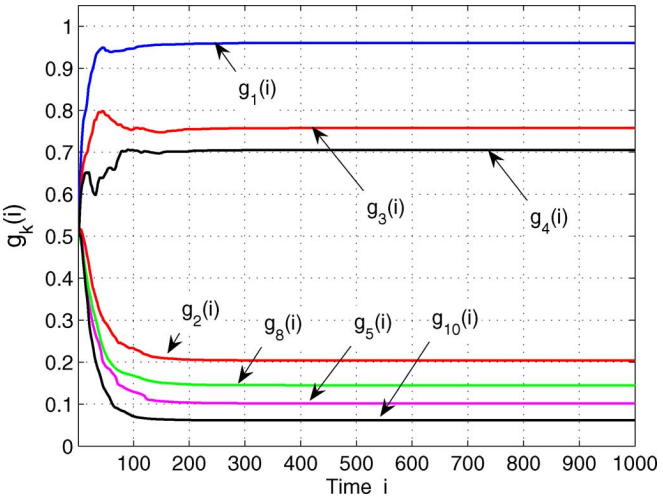

Fig. 24. Adaptive combiners for Example 6.

were assigned larger weights and nodes with lower SNR, e.g., nodes 2, 5, 8, and 10 were assigned smaller weights.

An important remark arises from the equalization effect produced by the diffusion protocol: it may limit the impact of the adaptive layer because all the nodes may be performing nearly the same. In the framework proposed here, it is mostly effective to combat pathological cases, or node/link failures. However, other arrangements that further enhance the adaptive layer performance are possible and are currently under study.

X. Concluding Remarks

One of the main results of this work is to show that cooperation improves performance from the estimation point of view, not only in terms of saving computation and communication resources. Particularly, cooperation has a stabilizing effect on the network. One can design the individual filters using local information *only* in order to achieve (local) stability and implement diffusion protocols to improve global performance.

Closed-form expressions for global and local mean and mean-square performance have been derived, matching very well the simulations that have been carried out. As a natural evolution of the (static) diffusion protocol, an adaptive implementation has been proposed, which in many cases improves the network learning process when there is an imbalance among the nodes in terms of learning rates.

In this work, we studied the LMS implementation operating with Gaussian signals. Other strategies can be studied using the formulation presented here, such as the distributed normalized LMS (dNLMS), the distributed affine projection algorithms (dAPA) and distributed RLS implementations [6]. Analysis for these algorithms operating in networks with changing topologies and observing non-Gaussian data is available and it will be treated in future publications.

References

[1] N. Kalouptsidis and S. Theodoridis, *Adaptive System Identification and Signal Processing Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[2] H. Sakai, J. M. Yang, and T. Oka, "Exact convergence analysis of adaptive filter algorithms without the persistently exciting condition," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 2077–2083, May 2007.

[3] A. H. Sayed, *Fundamentals of Adaptive Filtering*. New York: Wiley, 2003.
[4] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans, Signal Process,*, vol. 55, no. 8, pp. 4064–4077, Aug. 2007.
[5] A. H. Sayed and C. G. Lopes, "Adaptive processing over distributed networks," *IEICE Trans. Fund. Electron., Commun. Comput. Sci.*, vol. E90-A, no. 8, pp. 1504–1510, 2007.
[6] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, May 2008.
[7] D. Estrin, G. Pottie, and M. Srivastava, "Instrumenting the world with wireless sensor networks," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Salt Lake City, UT, May 2001, pp. 2033–2036.
[8] L. A. Rossi, B. Krishnamachari, and C.-C. J. Kuo, "Distributed parameter estimation for monitoring diffusion phenomena using physical models," in *Proc. 1st IEEE Conf. Sensor Ad Hoc Communications Networks*, Santa Clara, CA, Oct. 2004, pp. 460–469.
[9] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayeed, "Detection, classification, and tracking of targets," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 17–29, Mar. 2002.
[10] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.
[11] C. G. Lopes and A. H. Sayed, "Diffusion least-mean squares over adaptive networks," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Honolulu, HI, Apr. 2007, vol. 3, pp. 917–920.
[12] A. H. Sayed and C. G. Lopes, "Distributed recursive least-squares strategies over adaptive networks," in *Proc. 40th Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Oct. 2006, pp. 233–237.
[13] D. Spanos, R. Olfati-Saber, and R. Murray, "Distributed sensor fusion using dynamic consensus," presented at the 16th IFAC World Congr., Prague, Czech Republic, Jul. 2005.
[14] J. Arenas-Garcia, A. R. Figueiras-Vidal, and A. H. Sayed, "Mean-square performance of a convex combination of two adaptive filters," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 1078–1090, Mar. 2006.
[15] Y. Zhang and J. A. Chambers, "Convex combination of adaptive filters for a variable tap-length LMS algorithm," *IEEE Signal Process. Lett.*, vol. 13, no. 10, pp. 628–631, Oct. 2006.
[16] M. Silva and V. H. Nascimento, "Convex combination of adaptive filters with different tracking abilities," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Honolulu, HI, Apr. 2007, vol. 3, pp. 925–928.
[17] D. Mandic, P. Vayanos, C. Boukis, B. Jelfs, S. L. Goh, T. Gautama, and T. Rutkowski, "Collaborative adaptive learning using hybrid filters," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, Honolulu, HI, Apr. 2007, vol. 3, pp. 921–924.
[18] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 2001.
[19] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
[20] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, Sep. 2004.
[21] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, pp. 1520–1533, Sep. 2004.
[22] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.
[23] T. Y. A. Naffouri and A. H. Sayed, "Transient analysis of data-normalized adaptive filters," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 639–652, Mar. 2003.
[24] D. S. Tracy and R. P. Singh, "A new matrix product and its applications in partitioned matrix differentiation," *Statistica Neerlandica*, vol. 26, no. NR. 4, pp. 143–157, 1972.
[25] R. H. Koning, H. Neudecker, and T. Wansbeek, "Block Kronecker products and the vecb operator," Economics Dept., Institute of Economics Research, Univ. of Groningen, Groningen, The Netherlands, Research Memo. No. 351, 1990.
[26] Y. Wei and F. Zhang, "Equivalence of a matrix product to the Kronecker product," *Hadronic J. Supp.*, vol. 15, no. 3, pp. 327–331, 2000.
[27] C. G. Lopes, E. Satorius, and A. H. Sayed, "Adaptive carrier tracking for direct-to-earth mars communications," in *Proc. 40th Asilomar Conf. Signals, Systems, Computers*, Pacific Grove, CA, Oct. 2006, pp. 1042–1046.
[28] C. G. Lopes, E. H. Satorius, P. Eastbrook, and A. H. Sayed, "Efficient adaptive carrier tracking for Mars to Earth communications during entry, descent, and landing," in *Proc. IEEE Workshop on Statistical Signal Processing (SSP)*, Madison, WI, Aug. 2007, pp. 517–521.

Cassio G. Lopes (S'06) received the B.S. degree and the M.S. degree in electrical engineering from the Federal University of Santa Catarina, Brazil, in 1997 and 1999, respectively, and the M.S. degree and the Ph.D. degree in electrical engineering from the University of California, Los Angeles (UCLA) in 2004 and 2008, respectively.

From 1999 to 2002, he worked with hardware design for biomedical signals acquisition and transmission over the Internet. He also taught undergraduate-level courses in computer science and electrical engineering. From 2005 to 2007, he worked in cooperation with the NASA Jet Propulsion Laboratory to develop frequency tracking algorithms to support direct-to-Earth Mars communications during entry, descent, and landing. Recently, he has joined the Instituto Tecnologico de Aeronautica (ITA), Sao Jose dos Campos, Brazil, as a Postdoctoral Researcher, where he is developing distributed inertial navigation systems for aerospace applications, and techniques for impact zone estimation of missiles and sounding rockets. His current research interests are theory and methods for adaptive and statistical signal processing, distributed adaptive estimation, frequency tracking algorithms for space communications and data fusion for inertial navigation systems.

Ali H. Sayed (F'01) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1992.

He is Professor and Chairman of electrical engineering at the University of California, Los Angeles (UCLA). He is also the Principal Investigator of the UCLA Adaptive Systems Laboratory (www.ee.ucla.edu/asl). He has more than 270 journal and conference publications, is the author of *Fundamentals of Adaptive Filtering* (New York: Wiley, 2003), is the coauthor of the research monograph *Indefinite Quadratic Estimation and Control* (Philadelphia, PA: SIAM, 1999) and of the graduate-level textbook *Linear Estimation* (Englewood Cliffs, NJ: Prentice-Hall, 2000). He is also coeditor of the volume *Fast Reliable Algorithms for Matrices with Structure* (Philadelphia, PA: SIAM, 1999). He has contributed several articles to engineering and mathematical encyclopedias and handbooks and has served on the program committees of several international meetings. His research interests span several areas, including adaptive and statistical signal processing, filtering and estimation theories, signal processing for communications, interplays between signal processing and control methodologies, system theory, and fast algorithms for large-scale problems.

Dr. Sayed is the recipient of the 1996 IEEE Donald G. Fink Award. He received the Best Paper Award from the IEEE Signal Processing Society in 2002, the Kuwait Prize in Basic Science in 2003, the Frederick E. Terman Award in 2005, a Young Author Best Paper Award in 2005 from the IEEE Signal Processing Society, and two Best Student Paper awards at international meetings. He is also a member of the technical committees on Signal Processing Theory and Methods (SPTM) and on Signal Processing for Communications (SPCOM), both of the IEEE Signal Processing Society. He has served as Editor-in-Chief of the IEEE TRANSACTIONS ON SIGNAL PROCESSING (2003–2005) and is now serving as Editor-in-Chief of the *EURASIP Journal on Applied Signal Processing*. He is serving as General Chairman of ICASSP 2008 and sits on the Board of Governors of the IEEE Signal Processing Society.